# LAB 7: Spectrogram Analysis - The Short Time Fourier Transform

## Table of Contents




**AUTHOR**: Himanshu Sharma **Roll Number**: 1610110149 **Email ID**: hs583@snu.edu.in **Instructor**: Prof. Vijay K. Chakka


# Governing Equation

$$X(k,m) = \sum_{n=0}^{N-1} x(mM+n)W_N^{kn}$$

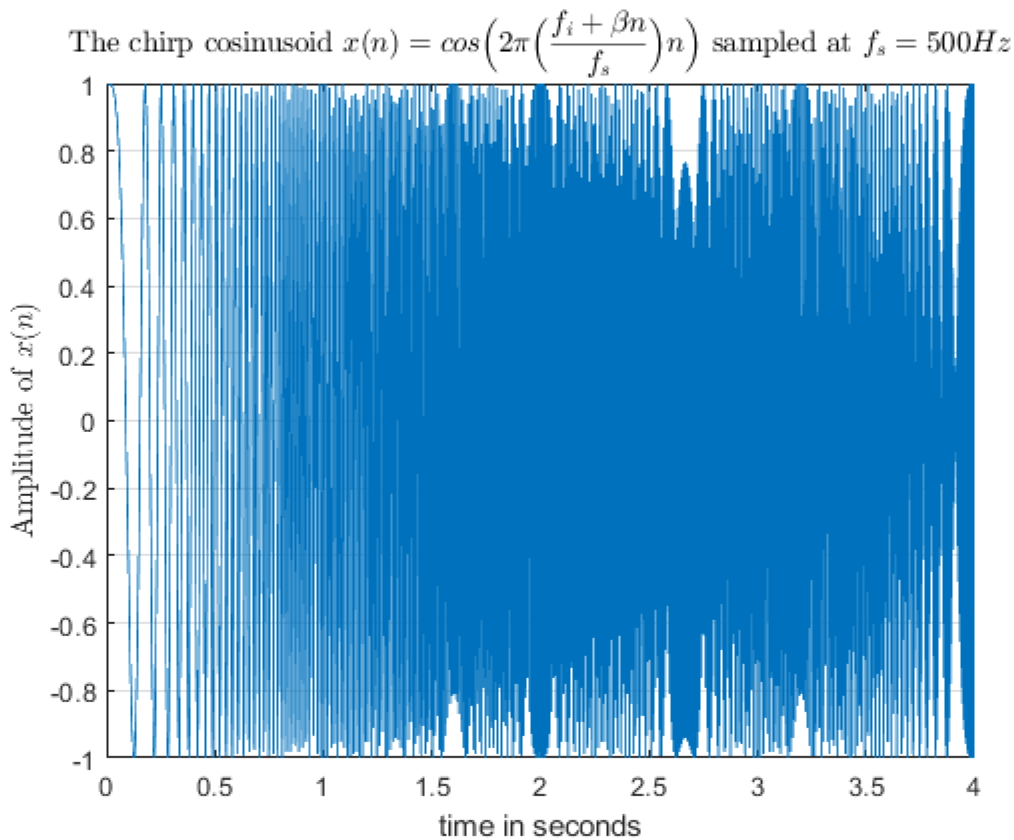
# Main Namespace

```
function main()
```

# Outputs

clear the terminal.

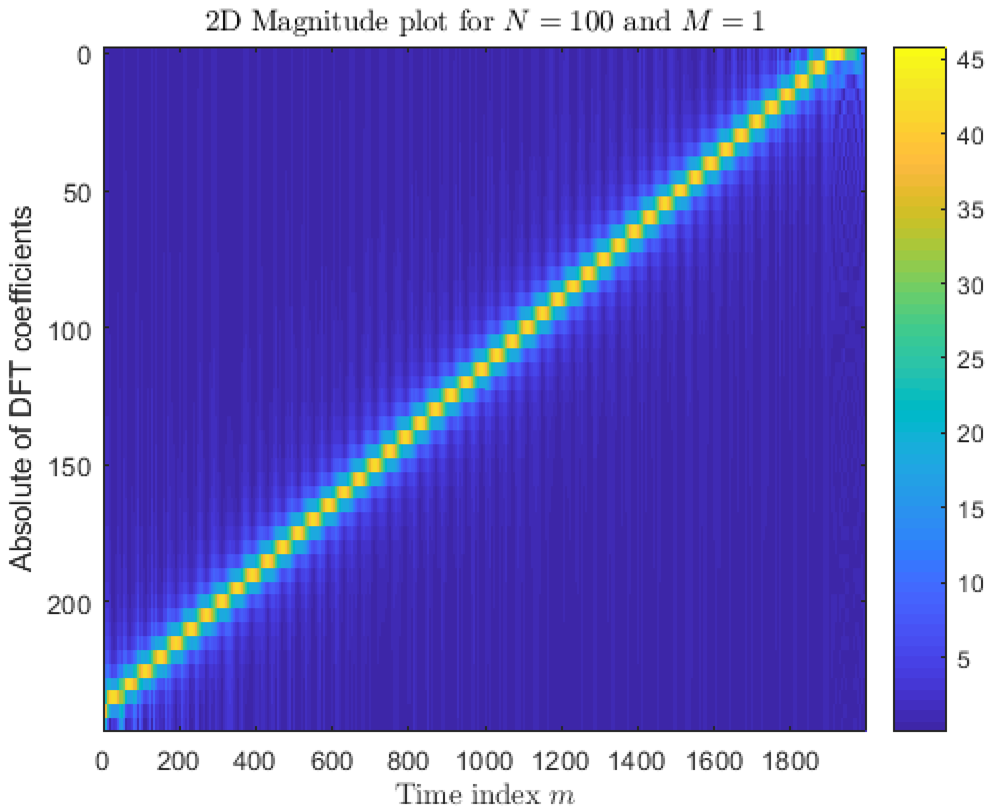
```
clc;
clear;
output_chirp = question1();
```

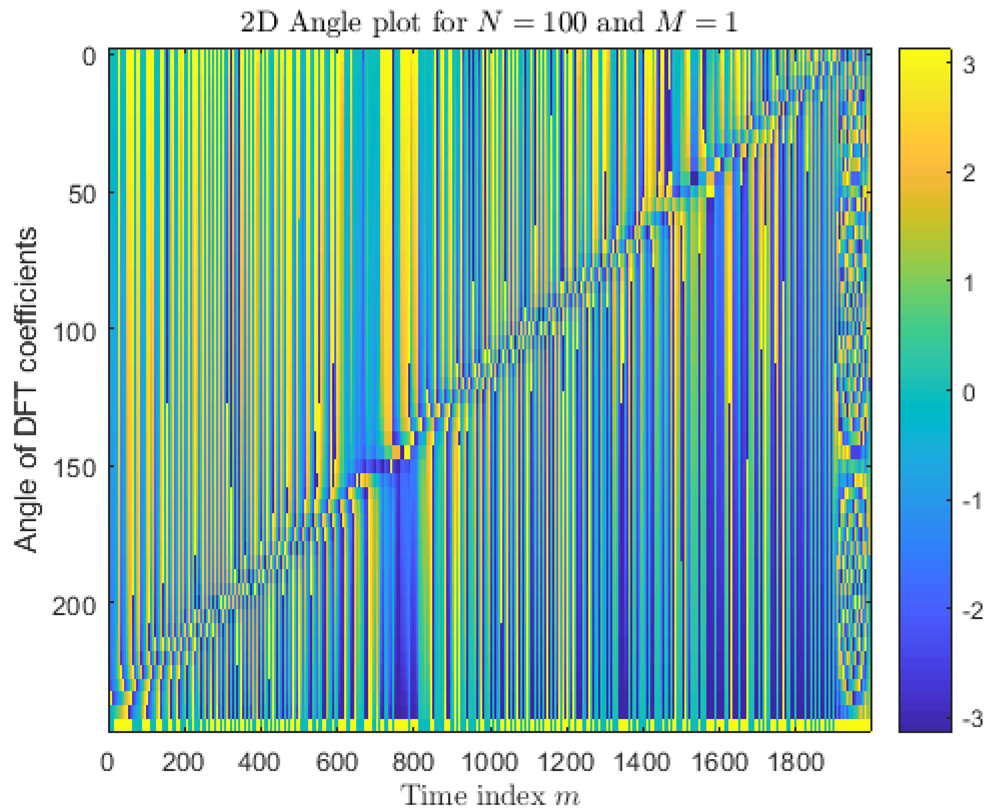


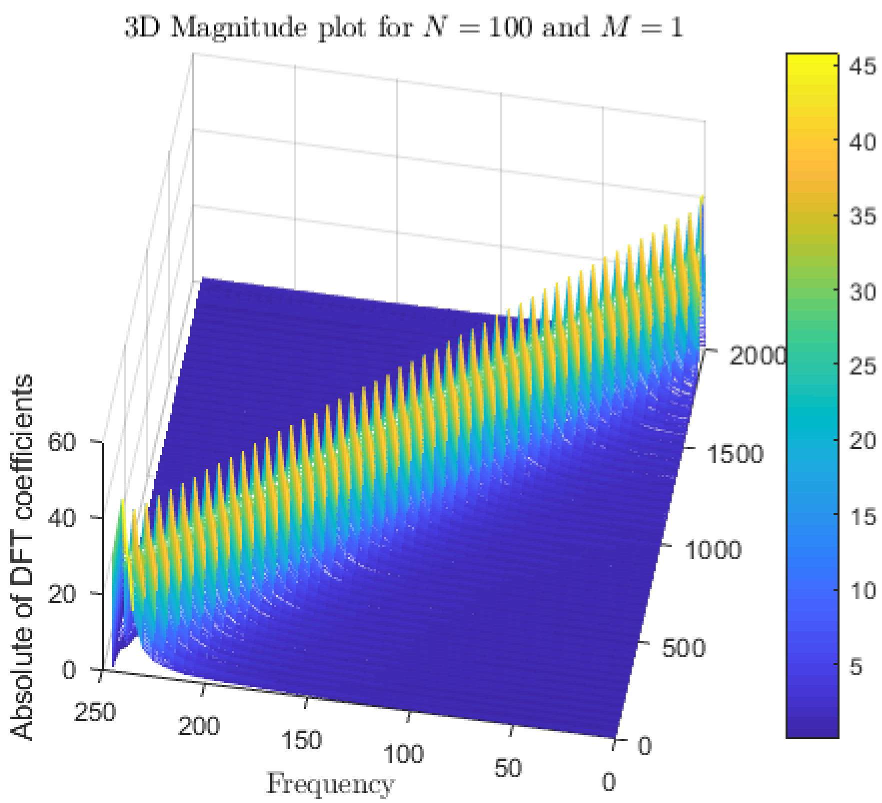

# Plot of 2D and 3D graph for N=100 and M=1

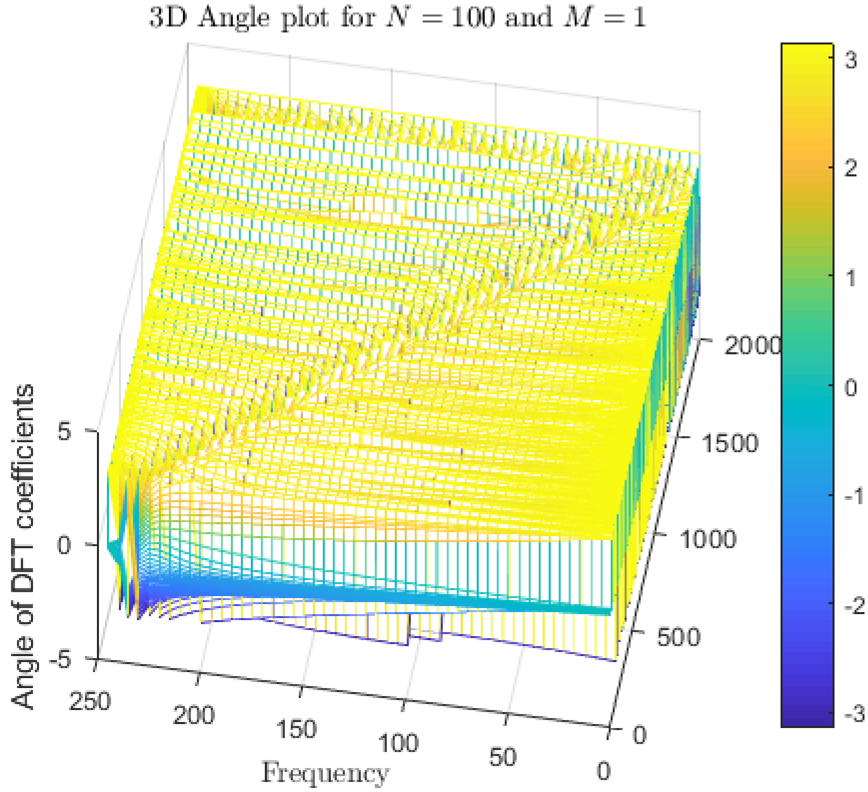
```
% run the block_it function on the above output.
mtx_N100_M1 = block_it(output_chirp, 100, 1);
dftN100M1 = colDFT(100, 1, mtx_N100_M1);
% since DFT is symmetric around a unit circle, either take
% first 50 or last 50 to plot the graph.
figure;
imagesc([0:1999], [49*5:-5:0],abs(dftN100M1(1:50, :)));
xlabel('Time index $m$', 'interpreter', 'latex');
ylabel('Absolute of DFT coefficients');
title('2D Magnitude plot for $N=100$ and
$M=1$', 'interpreter', 'latex');
colorbar;
figure;
imagesc([0:1999], [49*5:-5:0],angle(dftN100M1(1:50, :)));
xlabel('Time index $m$', 'interpreter', 'latex');
ylabel('Angle of DFT coefficients');
title('2D Angle plot for $N=100$ and
$M=1$', 'interpreter', 'latex');
colorbar;

figure;
mesh([0:1999], [49*5:-5:0], abs(dftN100M1(1:50, :)));
colorbar;
```

```
    view(-80, 60);
    xlabel('Time index $m$', 'interpreter', 'latex');
    ylabel('Frequency', 'interpreter', 'latex');
    zlabel('Absolute of DFT coefficients');
    title('3D Magnitude plot for $N=100$ and
$M=1$', 'interpreter', 'latex');
    figure;
    mesh([0:1999], [49*5:-5:0], angle(dftN100M1(1:50, :)));
    colorbar;
    view(-80, 60);
    xlabel('Time index $m$', 'interpreter', 'latex');
    ylabel('Frequency', 'interpreter', 'latex');
    zlabel('Angle of DFT coefficients');
    title('3D Angle plot for $N=100$ and
$M=1$', 'interpreter', 'latex');
```




2D Angle plot for $N = 100$ and $M = 1$
Angle of DFT coefficients
0
50
100
150
200
0
200
400
600
800
1000
1200
1400
1600
1800
Time index $m$
3
2
1
0
-1
-2
-3

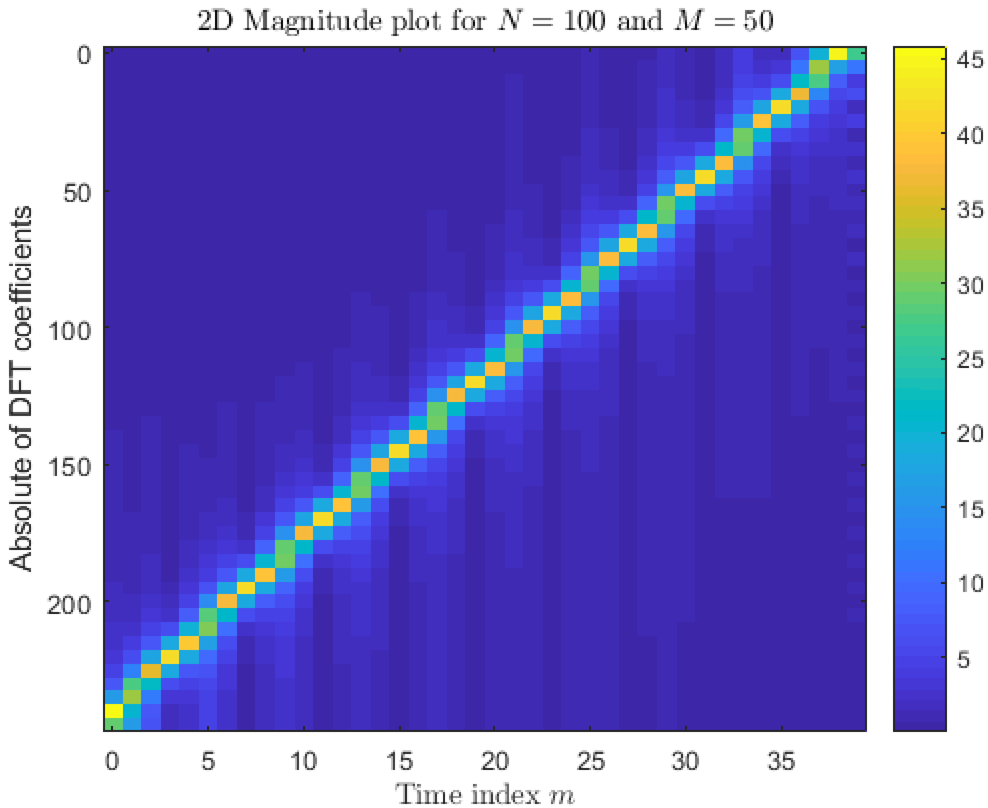

3D Magnitude plot for $N = 100$ and $M = 1$
Absolute of DFT coefficients
60
40
20
0
250
200
150
100
50
0
Frequency
0
500
1000
1500
2000
45
40
35
30
25
20
15
10
5

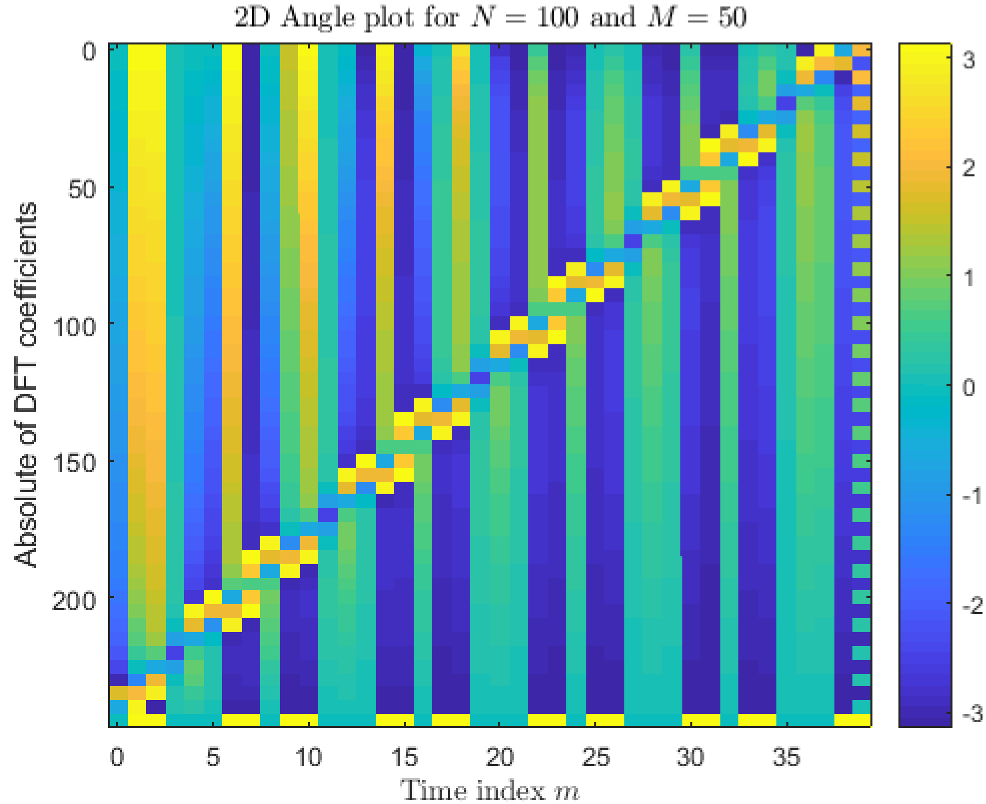


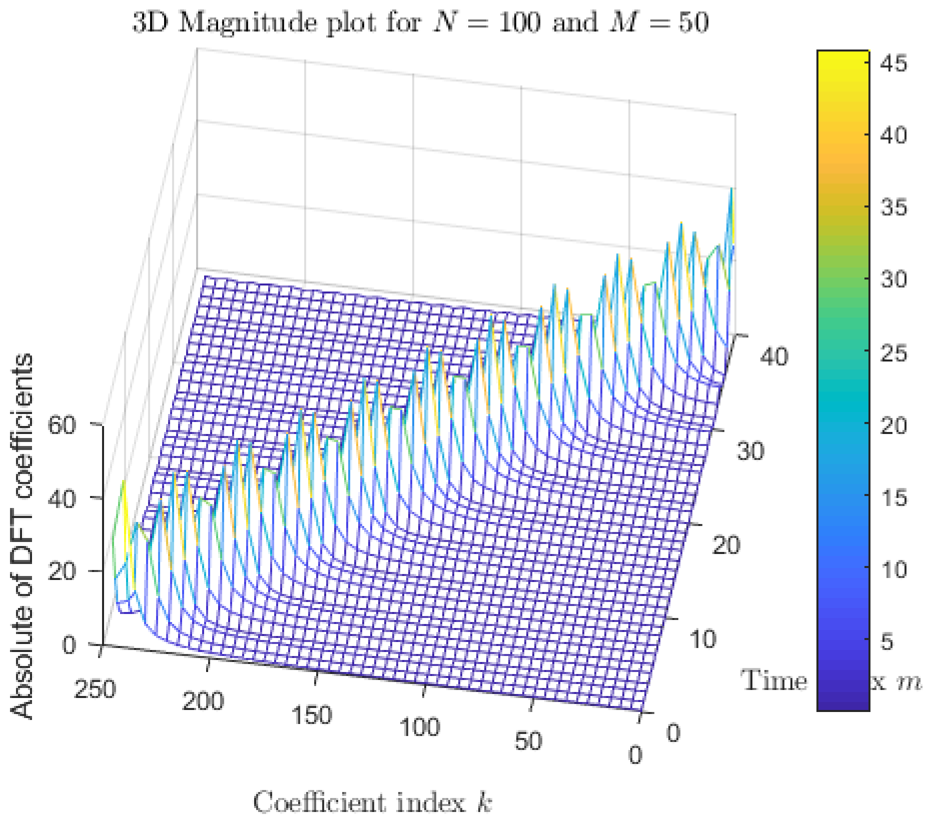

# Plot of 2D and 3D graph for N=100 and M=50

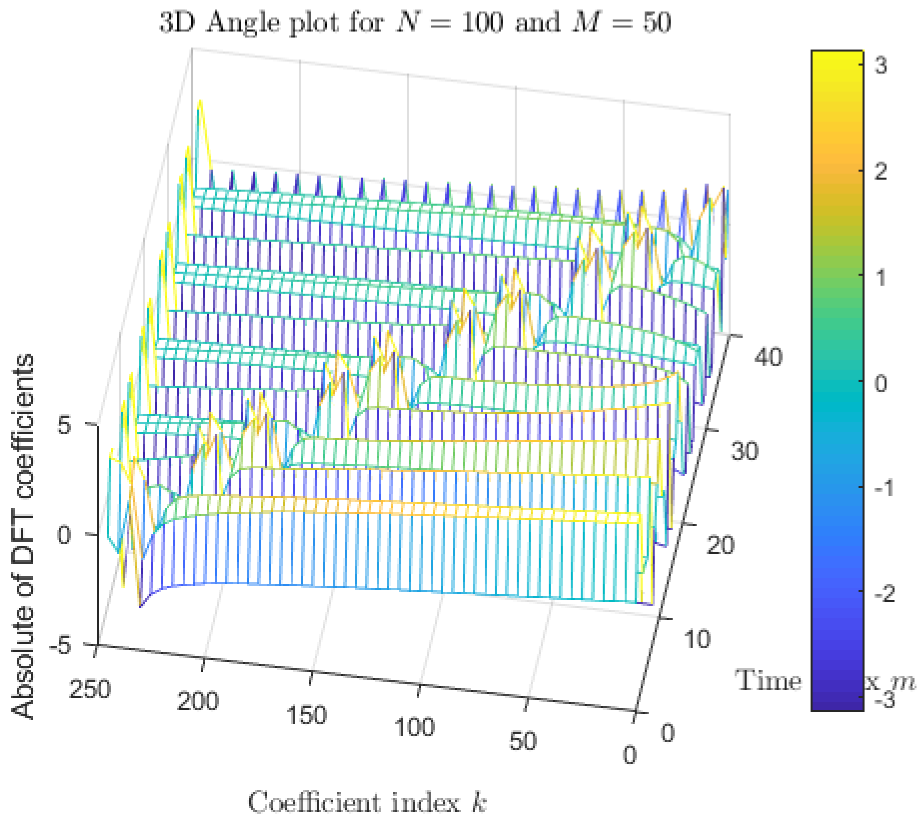
```
% run the block_it function on the above output.
mtx_N100_M50 = block_it(output_chirp, 100, 50);
dftN100M50 = colDFT(100, 1, mtx_N100_M50);
% since DFT is symmetric around a unit circle, either take
% first 50 or last 50 to plot the graph.
figure;
imagesc([0:39], [49*5:-5:0], abs(dftN100M50(1:50, :)));
xlabel('Time index $m$', 'interpreter', 'latex');
ylabel('Absolute of DFT coefficients');
title('2D Magnitude plot for $N=100$ and
$M=50$', 'interpreter', 'latex');
colorbar;
figure;
imagesc([0:39], [49*5:-5:0], angle(dftN100M50(1:50, :)));
xlabel('Time index $m$', 'interpreter', 'latex');
ylabel('Absolute of DFT coefficients');
title('2D Angle plot for $N=100$ and
$M=50$', 'interpreter', 'latex');
colorbar;
figure;
mesh([0:39], [49*5:-5:0], abs(dftN100M50(1:50, :)));
colorbar;
view(-80, 60);
```

```
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Magnitude plot for $N=100$ and
$M=50$', 'interpreter', 'latex');
   figure;
   mesh([0:39], [49*5:-5:0], angle(dftN100M50(1:50, :)));
   colorbar;
   view(-80, 60);
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Angle plot for $N=100$ and
$M=50$', 'interpreter', 'latex');
```




2D Angle plot for $N = 100$ and $M = 50$
Absolute of DFT coefficients
Time index $m$
0
50
100
150
200
0
5
10
15
20
25
30
35
3
2
1
0
-1
-2
-3

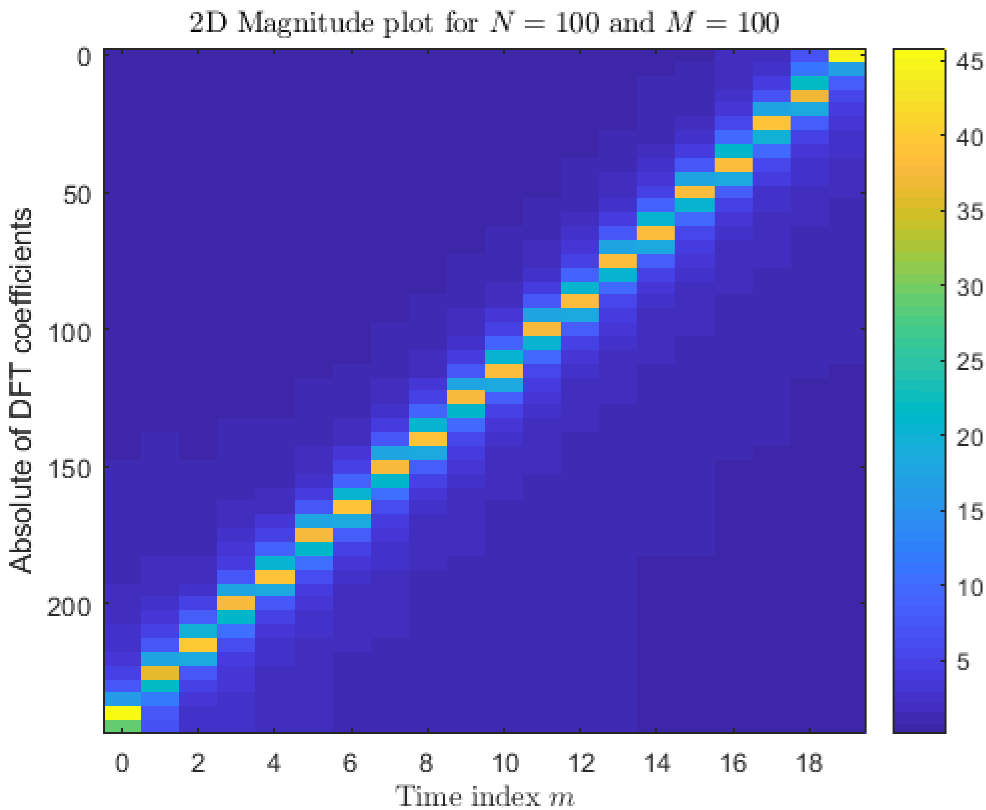

3D Magnitude plot for $N = 100$ and $M = 50$
Absolute of DFT coefficients
Coefficient index $k$
Time index $m$
0
20
40
60
250
200
150
100
50
0
0
10
20
30
40
45
40
35
30
25
20
15
10
5

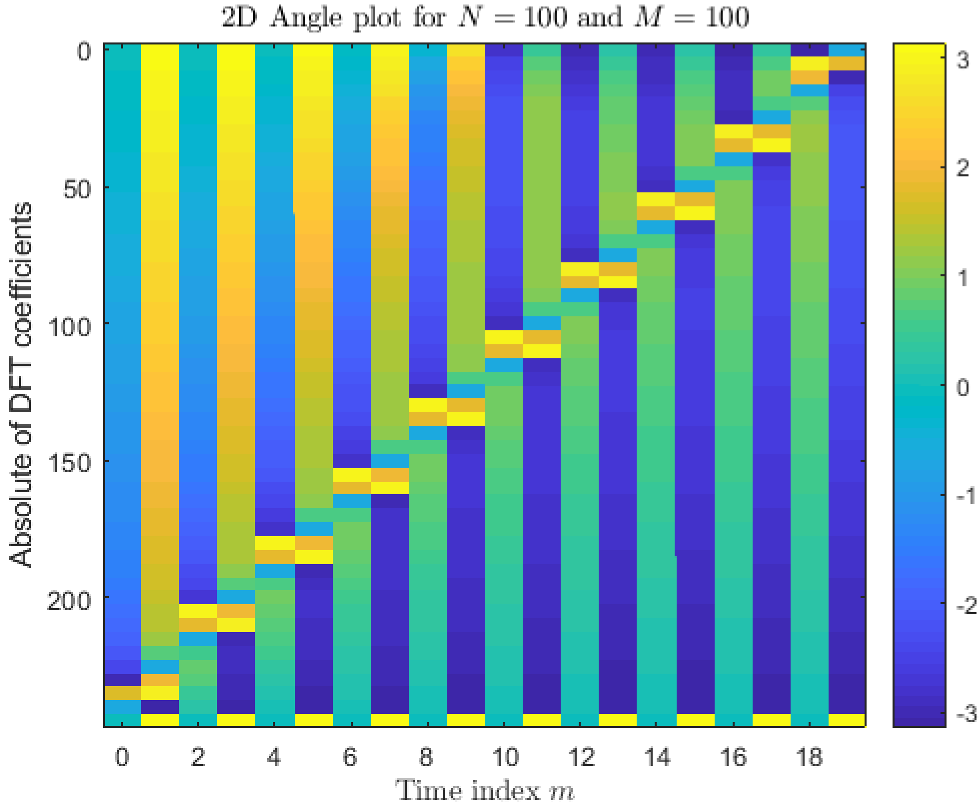


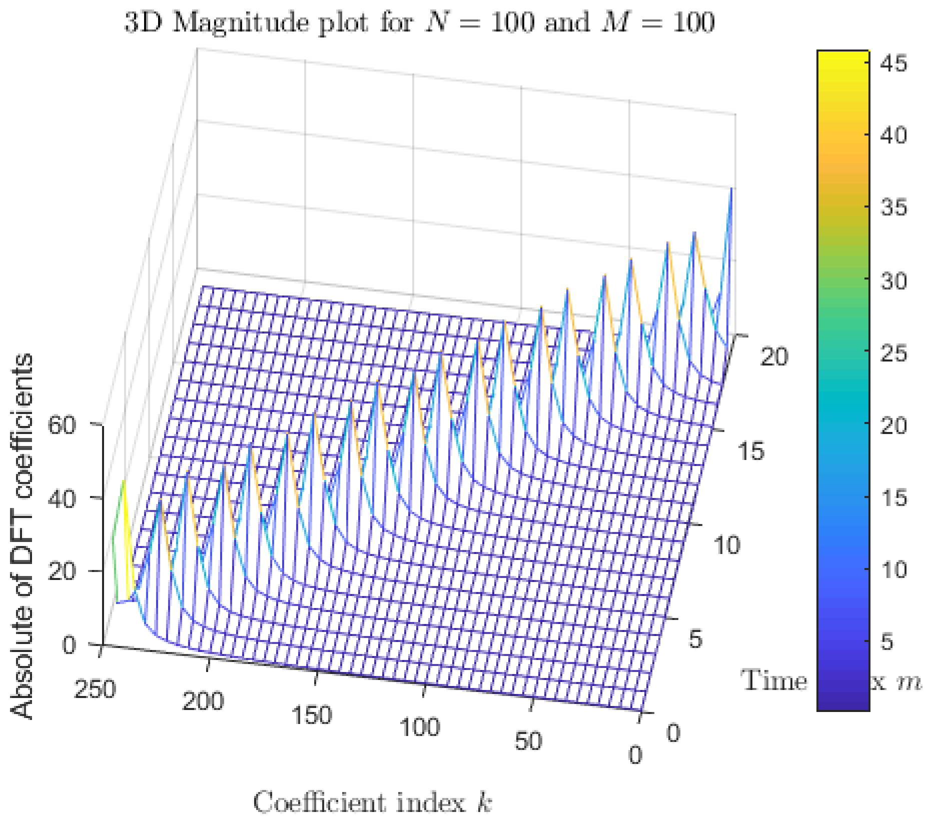

# Plot of 2D and 3D graph for N=100 and M=100

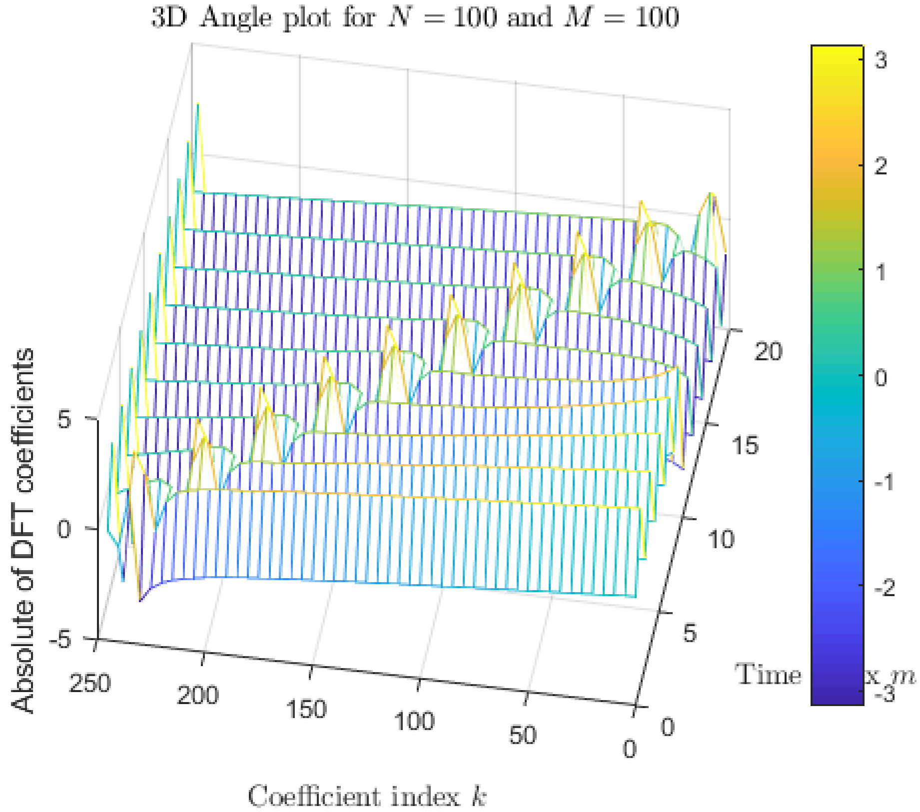
```
% run the block_it function on the above output.
mtx_N100_M100 = block_it(output_chirp, 100, 100);
dftN100M100 = colDFT(100, 1, mtx_N100_M100);
% since DFT is symmetric around a unit circle, either take
% first 50 or last 50 to plot the graph.
figure;
imagesc([0:19], [49*5:-5:0], abs(dftN100M100(1:50, :)));
xlabel('Time index $m$', 'interpreter', 'latex');
ylabel('Absolute of DFT coefficients');
title('2D Magnitude plot for $N=100$ and
$M=100$', 'interpreter', 'latex');
colorbar;
figure;
imagesc([0:19], [49*5:-5:0], angle(dftN100M100(1:50, :)));
xlabel('Time index $m$', 'interpreter', 'latex');
ylabel('Absolute of DFT coefficients');
title('2D Angle plot for $N=100$ and
$M=100$', 'interpreter', 'latex');
colorbar;
figure;
mesh([0:19], [49*5:-5:0], abs(dftN100M100(1:50, :)));
colorbar;
view(-80, 60);
```

```
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Magnitude plot for $N=100$ and
$M=100$', 'interpreter', 'latex');
   figure;
   mesh([0:19], [49*5:-5:0], angle(dftN100M100(1:50, :)));
   colorbar;
   view(-80, 60);
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Angle plot for $N=100$ and
$M=100$', 'interpreter', 'latex');
```




2D Angle plot for $N = 100$ and $M = 100$
Absolute of DFT coefficients
Time index $m$
0
50
100
150
200
0
2
4
6
8
10
12
14
16
18
3
2
1
0
-1
-2
-3

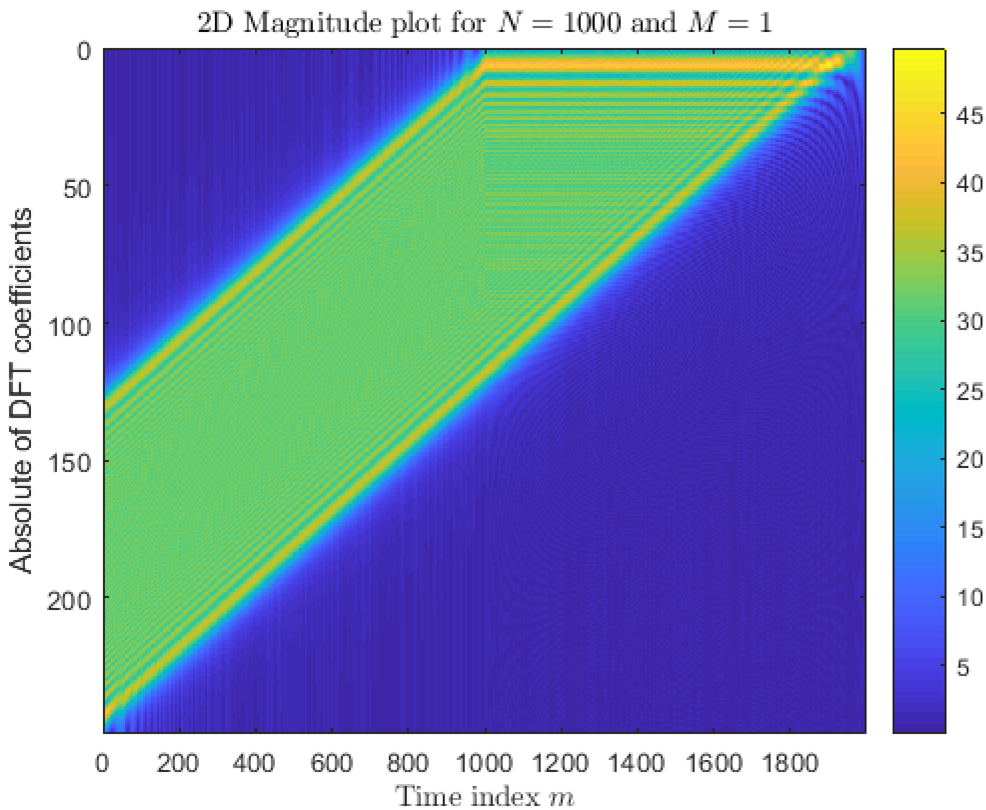

3D Magnitude plot for $N = 100$ and $M = 100$
Absolute of DFT coefficients
Coefficient index $k$
Time index $m$
0
20
40
60
250
200
150
100
50
0
5
10
15
20
45
40
35
30
25
20
15
10
5

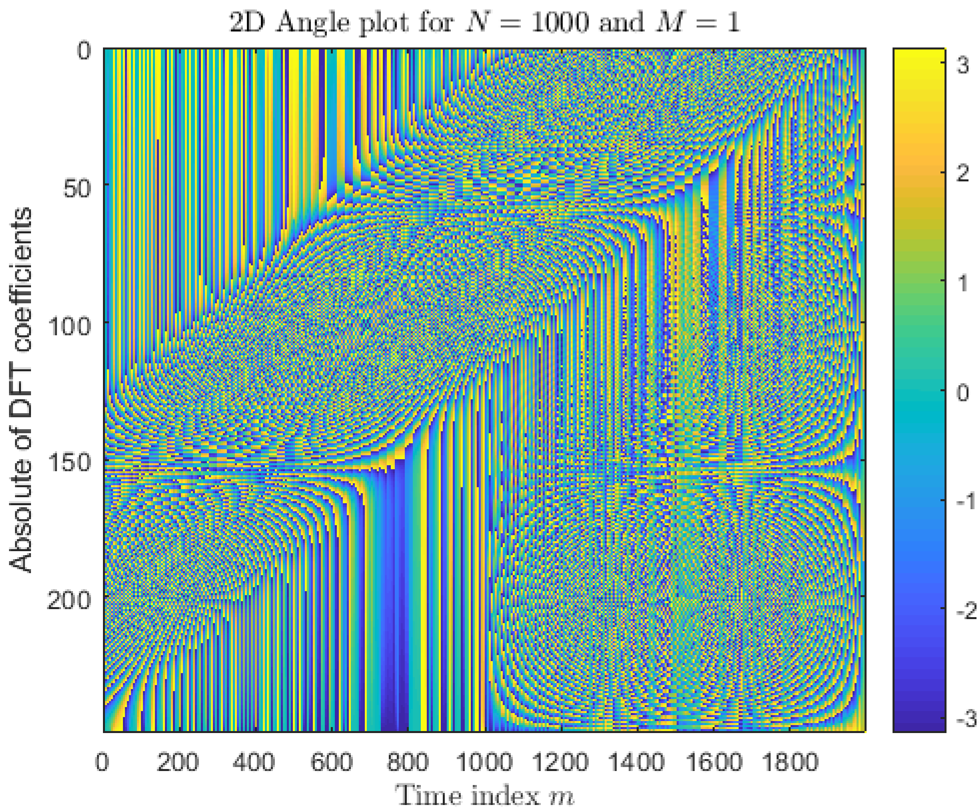


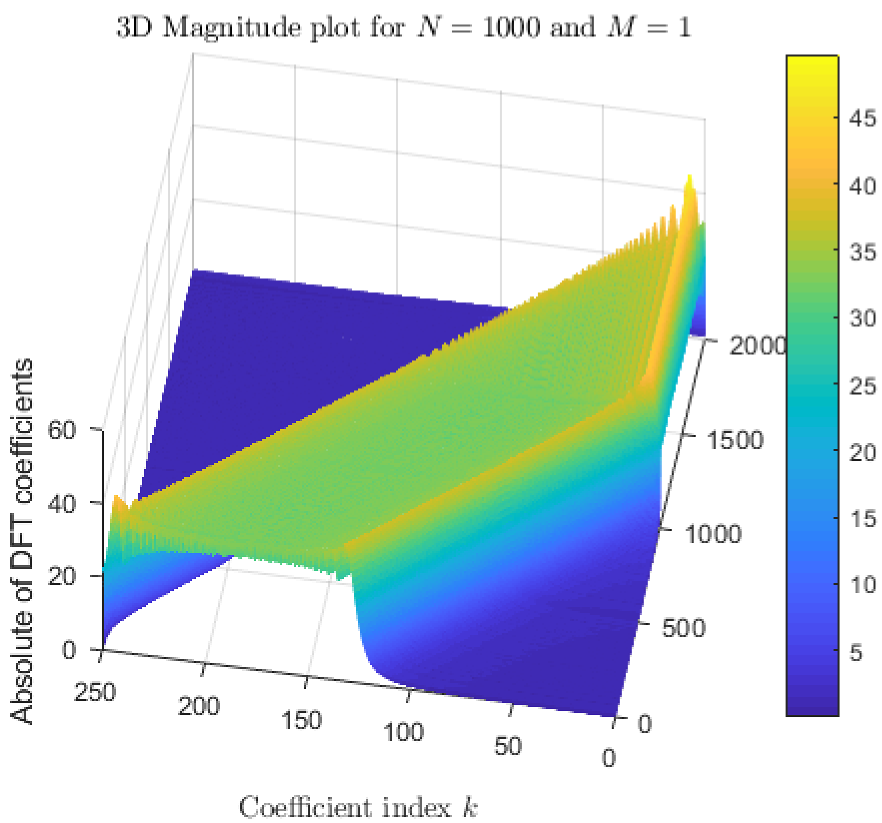

# Plot of 2D and 3D graph for N=1000 and M=1

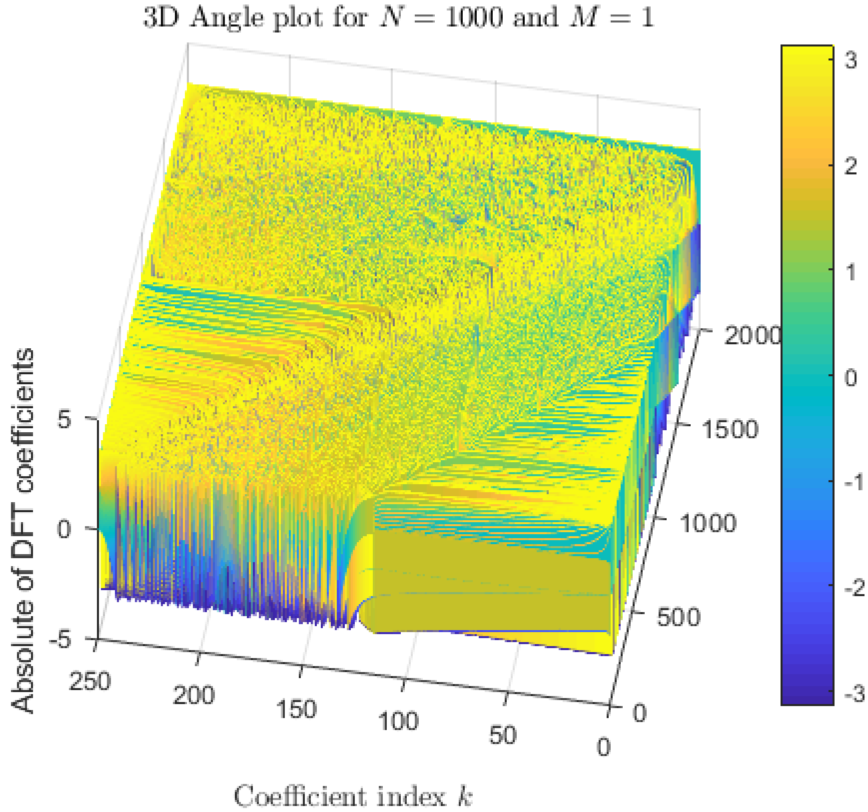
```
% run the block_it function on the above output.
mtx_N1000_M1 = block_it(output_chirp, 1000, 1);
dftN1000M1 = colDFT(1000, 1, mtx_N1000_M1);
% since DFT is symmetric around a unit circle, either take
% first 50 or last 50 to plot the graph.
figure;
imagesc([0:1999], [499*0.5:-0.5:0], abs(dftN1000M1(1:500, :)));
xlabel('Time index $m$', 'interpreter', 'latex');
ylabel('Absolute of DFT coefficients');
title('2D Magnitude plot for $N=1000$ and
$M=1$', 'interpreter', 'latex');
colorbar;
figure;
imagesc([0:1999], [499*0.5:-0.5:0], angle(dftN1000M1(1:500, :)));
xlabel('Time index $m$', 'interpreter', 'latex');
ylabel('Absolute of DFT coefficients');
title('2D Angle plot for $N=1000$ and
$M=1$', 'interpreter', 'latex');
colorbar;
figure;
mesh([0:1999], [499*0.5:-0.5:0], abs(dftN1000M1(1:500, :)));
colorbar;
view(-80, 60);
```

```
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Magnitude plot for $N=1000$ and
$M=1$', 'interpreter', 'latex');
   figure;
   mesh([0:1999], [499*0.5:-0.5:0], angle(dftN1000M1(1:500, :)));
   colorbar;
   view(-80, 60);
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Angle plot for $N=1000$ and
$M=1$', 'interpreter', 'latex');
```




2D Angle plot for $N = 1000$ and $M = 1$
Absolute of DFT coefficients
Time index $m$
0
50
100
150
200
0
200
400
600
800
1000
1200
1400
1600
1800
3
2
1
0
-1
-2
-3


3D Magnitude plot for $N = 1000$ and $M = 1$
Absolute of DFT coefficients
Coefficient index $k$
0
20
40
60
250
200
150
100
50
0
0
500
1000
1500
2000
45
40
35
30
25
20
15
10
5




# inputSignal.mat

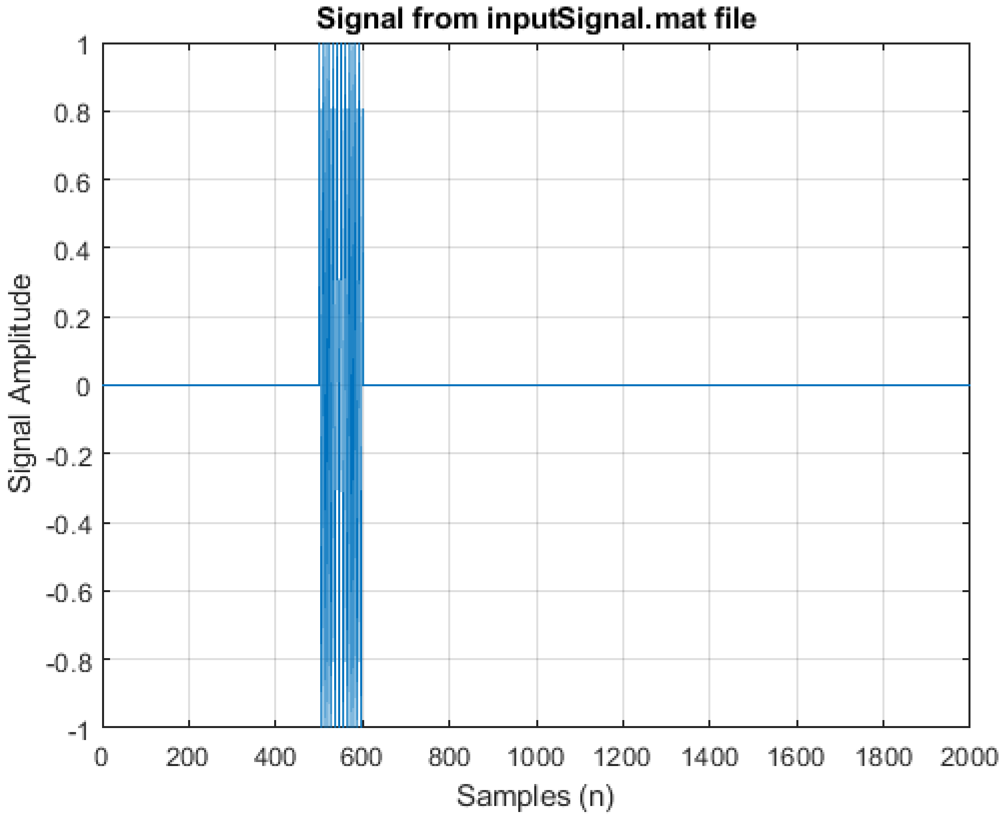
first step is to load the signal.

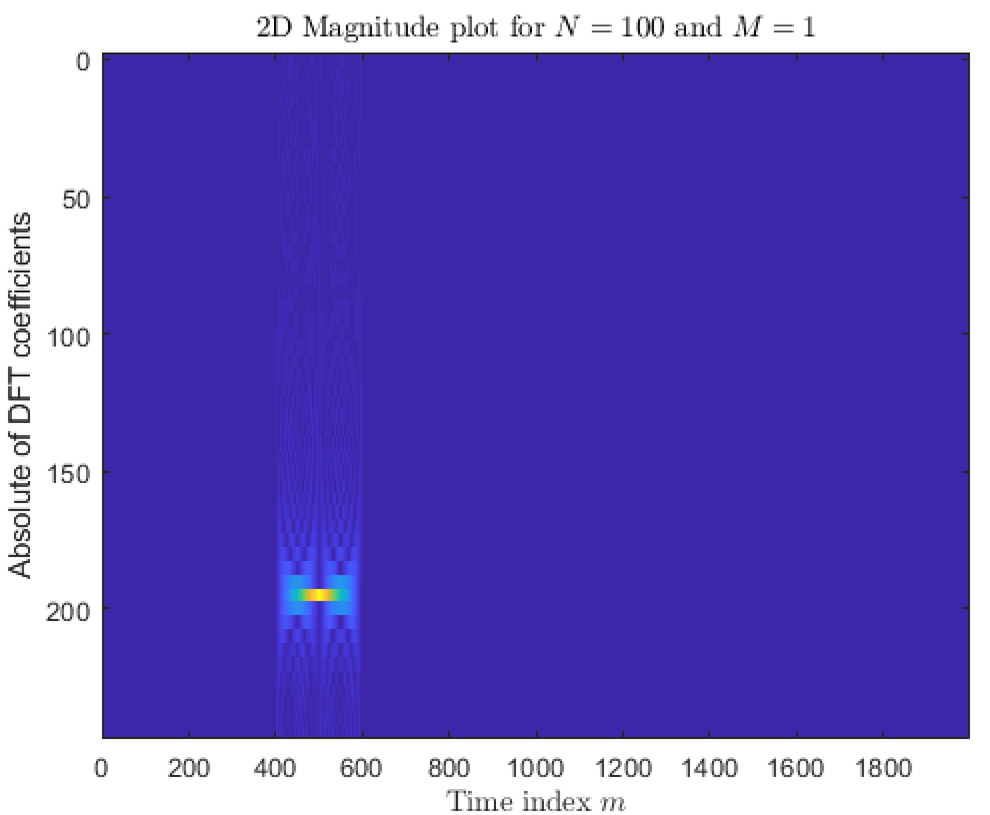
```
signal = load('inputSignal.mat', '-mat');
signal = signal.inputSignal;

% plot the signal as well.
figure;
plot(signal);
grid on;
title('Signal from inputSignal.mat file');
xlabel('Samples (n)')
ylabel('Signal Amplitude');

% run the block_it function on the above signal for N = 100 and M
= 1.
sig_N100_M1 = block_it(signal, 100, 1);
sigdftN100M1 = colDFT(100, 1, sig_N100_M1);
% since DFT is symmetric around a unit circle, either take
% first 50 or last 50 to plot the graph.
figure;
imagesc([0:1999], [49*5:-5:0], abs(sigdftN100M1(1:50, :)));
xlabel('Time index $m$', 'interpreter', 'latex');
ylabel('Absolute of DFT coefficients');
```

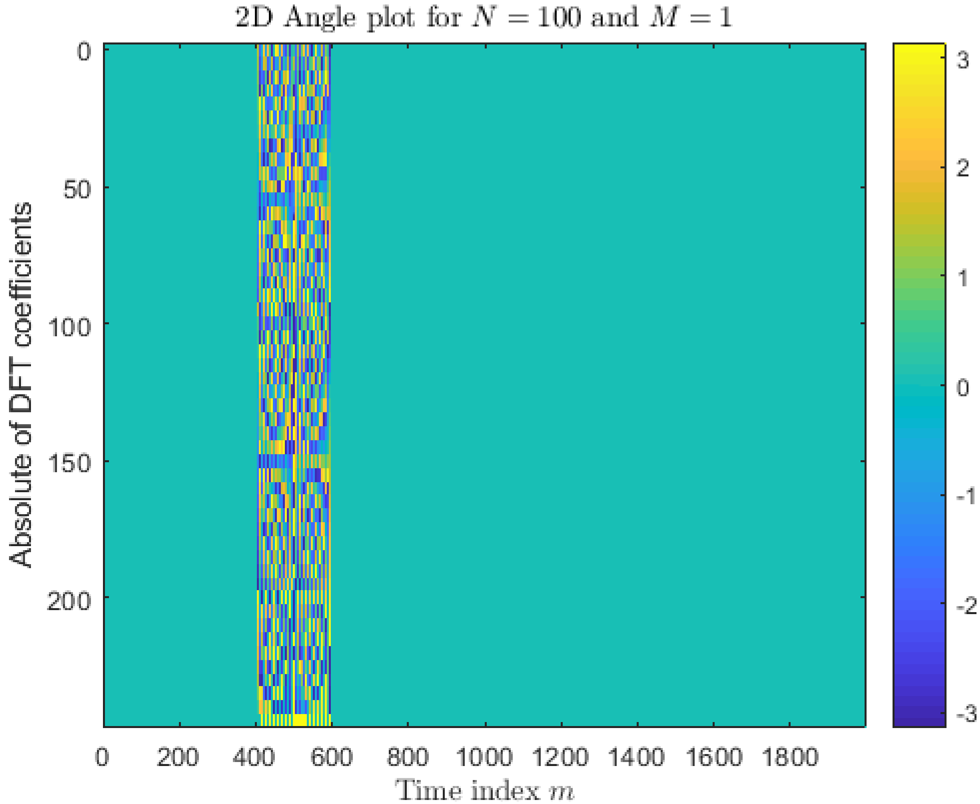
```
   title('2D Magnitude plot for $N=100$ and
$M=1$', 'interpreter', 'latex');
   figure;
   imagesc([0:1999], [49*5:-5:0], angle(sigdftN100M1(1:50, :)));
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Absolute of DFT coefficients');
   title('2D Angle plot for $N=100$ and
$M=1$', 'interpreter', 'latex');
   colorbar;
   figure;
   mesh([0:1999], [49*5:-5:0], abs(sigdftN100M1(1:50, :)));
   colorbar;
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Magnitude plot for $N=100$ and
$M=1$', 'interpreter', 'latex');
   figure;
   mesh([0:1999], [49*5:-5:0], angle(sigdftN100M1(1:50, :)));
   colorbar;
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Angle plot for $N=100$ and
$M=1$', 'interpreter', 'latex');

   % run the block_it function on the above signal for N = 100 and M
= 50.
   sig_N100_M50 = block_it(signal, 100, 50);
   sigdftN100M50 = colDFT(100, 50, sig_N100_M50);
   % since DFT is symmetric around a unit circle, either take
   % first 50 or last 50 to plot the graph.
   figure;
   imagesc([0:39], [49*5:-5:0], abs(sigdftN100M50(1:50, :)));
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Absolute of DFT coefficients');
   title('2D Magnitude plot for $N=100$ and
$M=50$', 'interpreter', 'latex');
   colorbar;
   figure;
   imagesc([0:39], [49*5:-5:0], angle(sigdftN100M50(1:50, :)));
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Absolute of DFT coefficients');
   title('2D Angle plot for $N=100$ and
$M=50$', 'interpreter', 'latex');
   colorbar;
   figure;
   mesh([0:39], [49*5:-5:0], abs(sigdftN100M50(1:50, :)));
   colorbar;
   view(-80, 60);
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
```

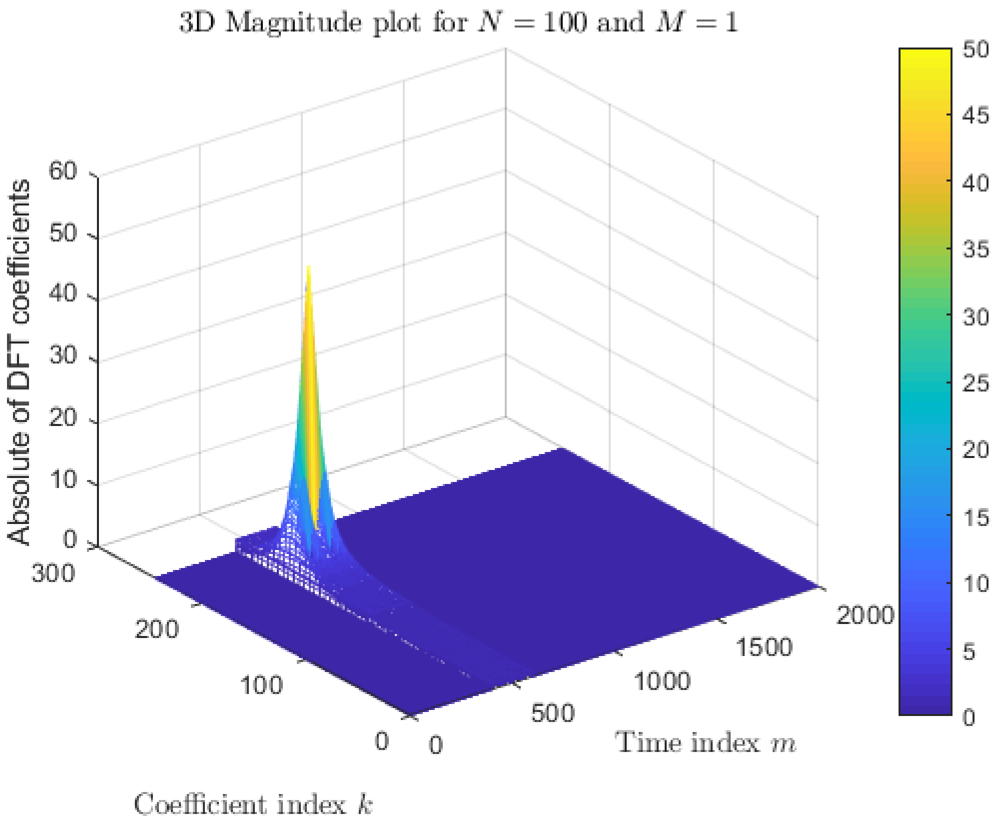
```
   title('3D Magnitude plot for $N=100$ and
$M=50$', 'interpreter', 'latex');
   figure;
   mesh([0:39], [49*5:-5:0], angle(sigdftN100M50(1:50, :)));
   colorbar;
   view(-80, 60);
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Angle plot for $N=100$ and
$M=50$', 'interpreter', 'latex');

   % run the block_it function on the above signal for N = 100 and M
= 100.
   sig_N100_M100 = block_it(signal, 100, 100);
   sigdftN100M100 = colDFT(100, 100, sig_N100_M100);
   % since DFT is symmetric around a unit circle, either take
   % first 50 or last 50 to plot the graph.
   figure;
   imagesc([0:19], [49*5:-5:0], abs(sigdftN100M100(1:50, :)));
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Absolute of DFT coefficients');
   title('2D Magnitude plot for $N=100$ and
$M=100$', 'interpreter', 'latex');
   colorbar;
   figure;
   imagesc([0:19], [49*5:-5:0], angle(sigdftN100M100(1:50, :)));
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Absolute of DFT coefficients');
   title('2D Angle plot for $N=100$ and
$M=100$', 'interpreter', 'latex');
   colorbar;
   figure;
   mesh([0:19], [49*5:-5:0], abs(sigdftN100M100(1:50, :)));
   colorbar;
   view(-80, 60);
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Magnitude plot for $N=100$ and
$M=100$', 'interpreter', 'latex');
   figure;
   mesh([0:19], [49*5:-5:0], angle(sigdftN100M100(1:50, :)));
   colorbar;
   view(-80, 60);
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Angle plot for $N=100$ and
$M=100$', 'interpreter', 'latex');

   % run the block_it function on the above signal for N = 1000 and M
= 1.
   sig_N1000_M1 = block_it(signal, 1000, 1);
```

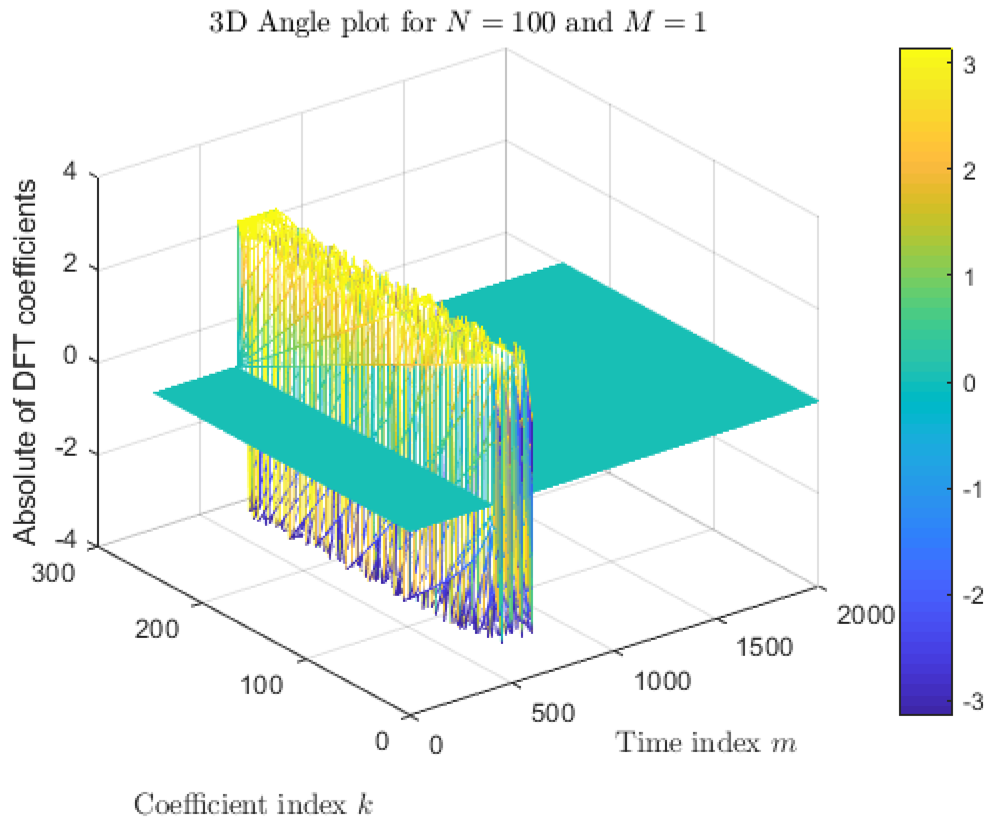
```
   sigdftN1000M1 = colDFT(1000, 1, sig_N1000_M1);
   % since DFT is symmetric around a unit circle, either take
   % first 50 or last 50 to plot the graph.
   figure;
   imagesc([0:1999], [499*0.5:-0.5:0], abs(sigdftN1000M1(1:500, :)));
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Absolute of DFT coefficients');
   title('2D Magnitude plot for $N=1000$ and
$M=1$', 'interpreter', 'latex');
   colorbar;
   figure;
   imagesc([0:1999], [499*0.5:-0.5:0],
angle(sigdftN1000M1(1:500, :)));
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Absolute of DFT coefficients');
   title('2D Angle plot for $N=1000$ and
$M=1$', 'interpreter', 'latex');
   colorbar;
   figure;
   mesh([0:1999], [499*0.5:-0.5:0], abs(sigdftN1000M1(1:500, :)));
   view(-80, 60);
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Magnitude plot for $N=1000$ and
$M=1$', 'interpreter', 'latex');
   figure;
   mesh([0:1999], [499*0.5:-0.5:0], angle(sigdftN1000M1(1:500, :)));
   view(-80, 60);
   xlabel('Time index $m$', 'interpreter', 'latex');
   ylabel('Coefficient index $k$', 'interpreter', 'latex');
   zlabel('Absolute of DFT coefficients');
   title('3D Angle plot for $N=1000$ and
$M=1$', 'interpreter', 'latex');
```

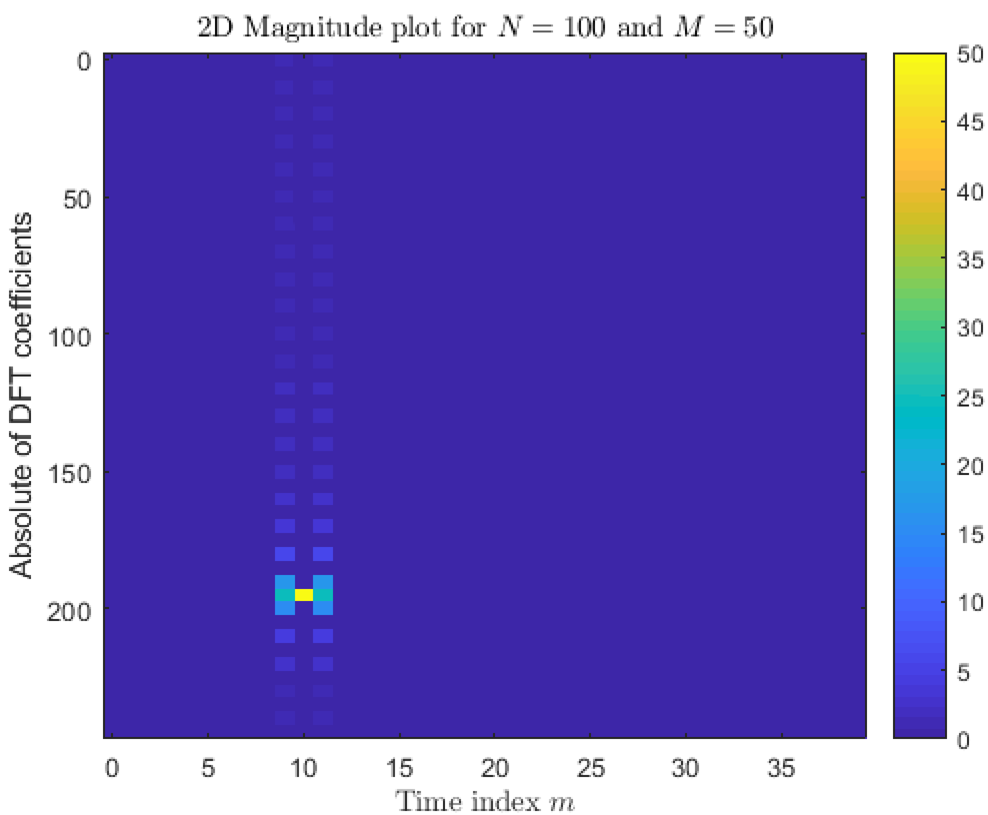

Signal from inputSignal.mat file
Signal Amplitude
Samples (n)
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
200
400
600
800
1000
1200
1400
1600
1800
2000

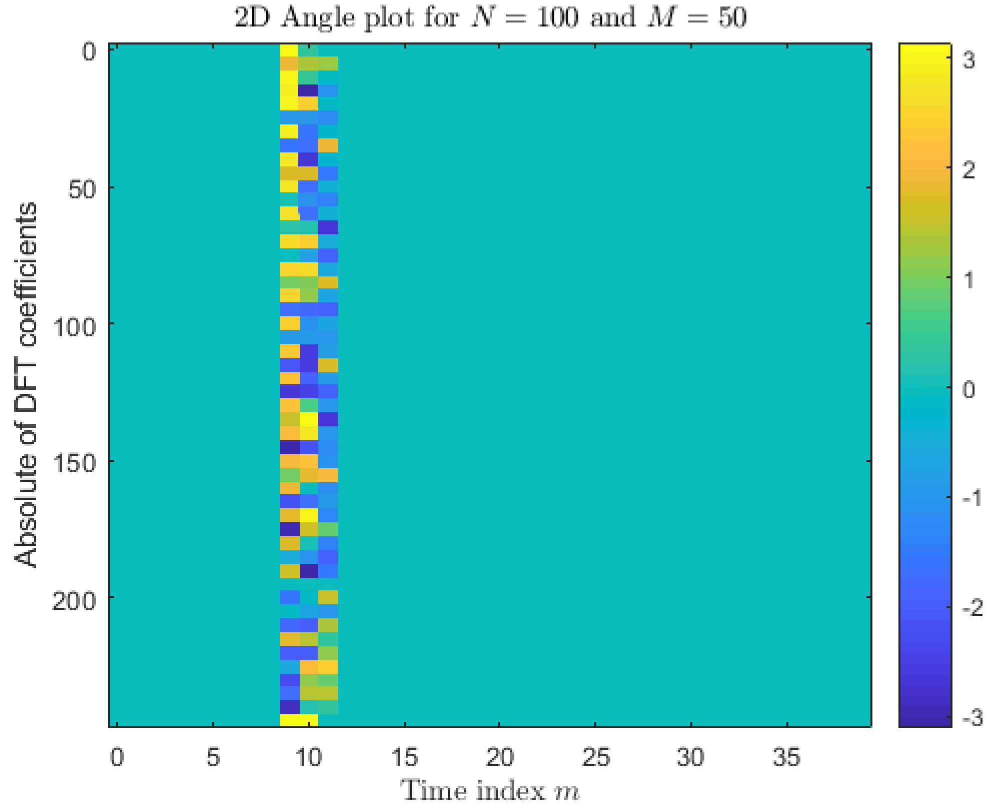

2D Magnitude plot for $N = 100$ and $M = 1$
Absolute of DFT coefficients
Time index $m$
0
50
100
150
200
0
200
400
600
800
1000
1200
1400
1600
1800

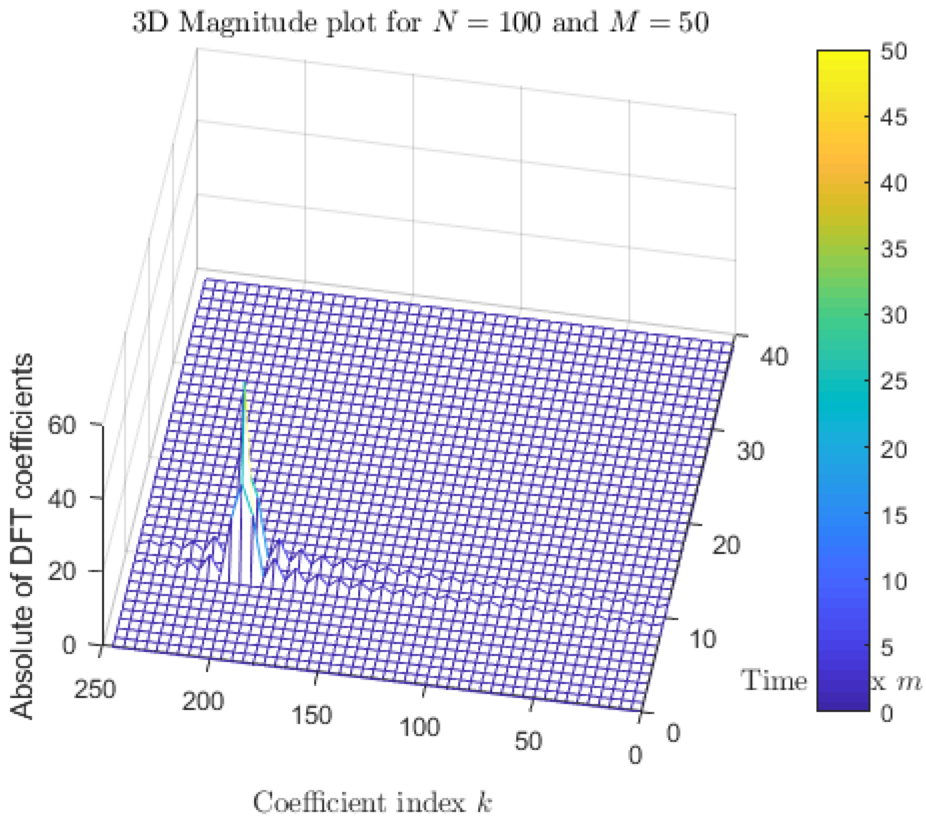

2D Angle plot for $N = 100$ and $M = 1$
Absolute of DFT coefficients
0
50
100
150
200
0
200
400
600
800
1000
1200
1400
1600
1800
Time index $m$
3
2
1
0
-1
-2
-3

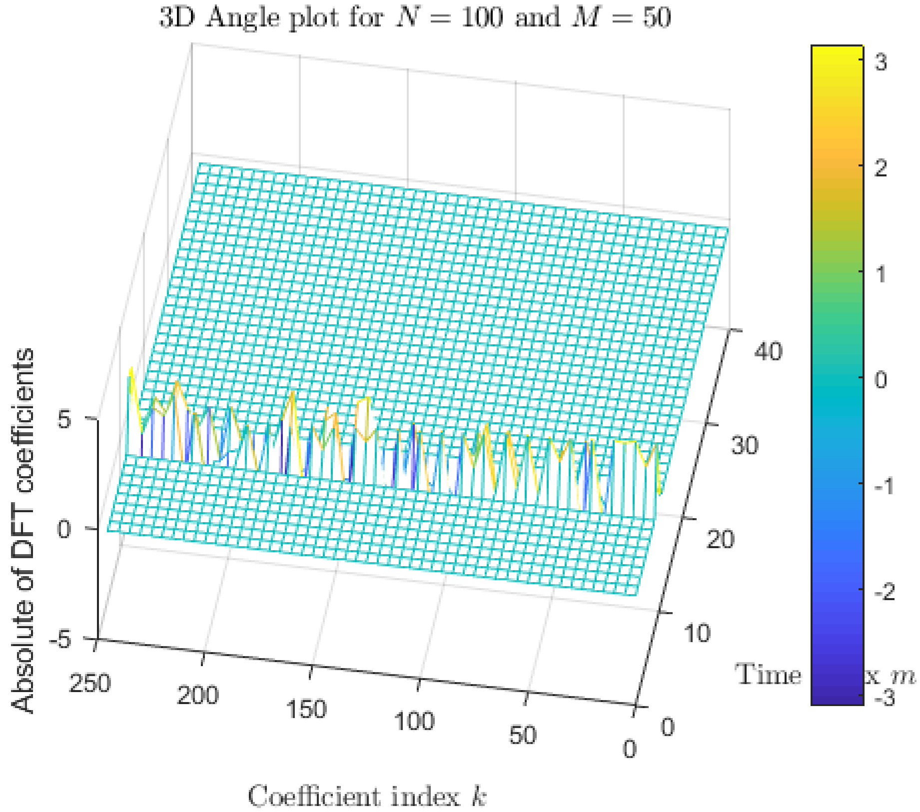

3D Magnitude plot for $N = 100$ and $M = 1$
Absolute of DFT coefficients
60
50
40
30
20
10
0
300
200
100
0
0
500
1000
1500
2000
Time index $m$
Coefficient index $k$
50
45
40
35
30
25
20
15
10
5
0

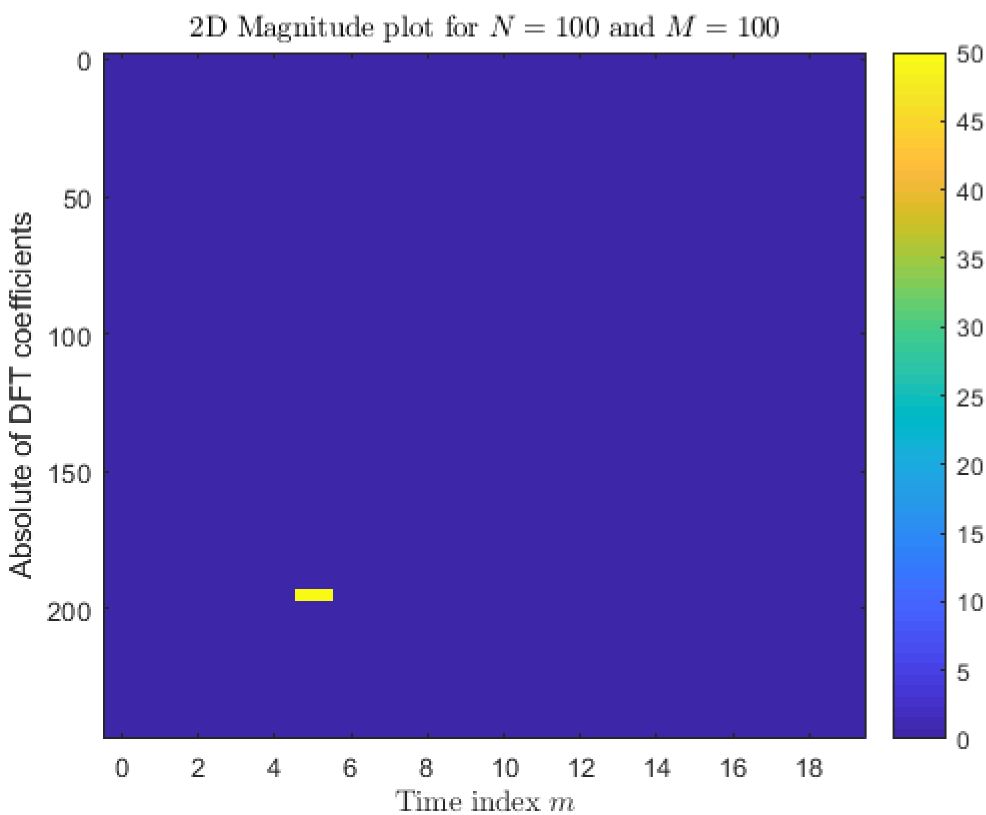

3D Angle plot for $N = 100$ and $M = 1$
Absolute of DFT coefficients
4
2
0
-2
-4
300
200
100
0
0
500
1000
1500
2000
Time index $m$
Coefficient index $k$
3
2
1
0
-1
-2
-3

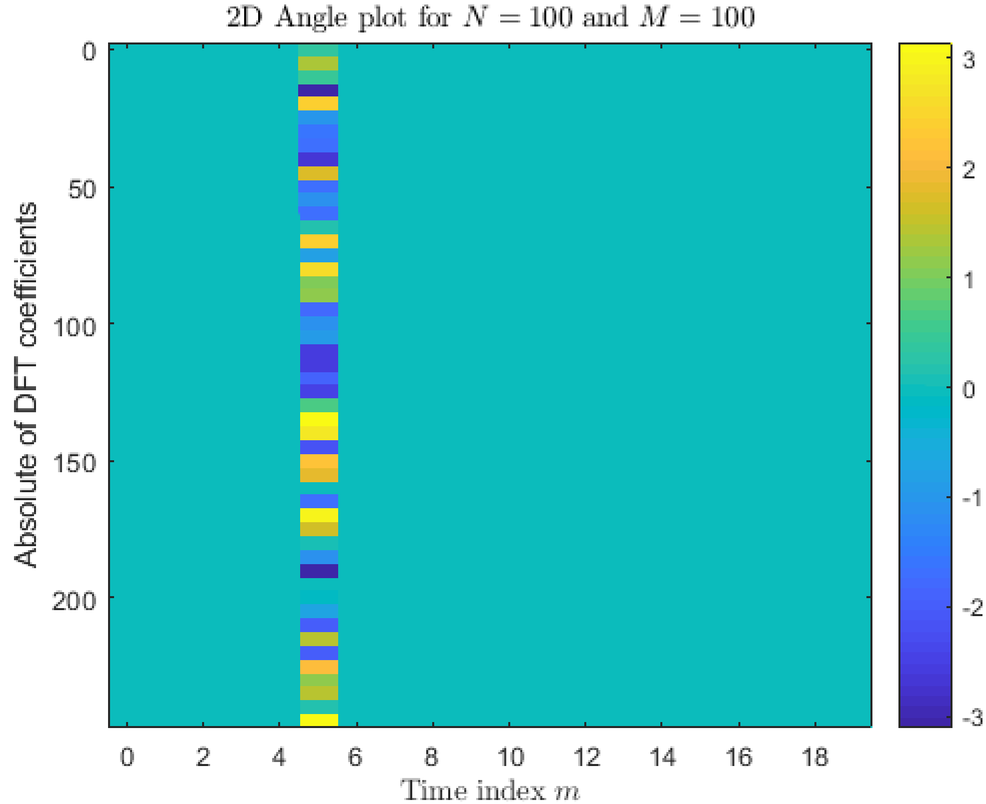

2D Magnitude plot for $N = 100$ and $M = 50$
Absolute of DFT coefficients
0
50
100
150
200
0
5
10
15
20
25
30
35
Time index $m$
50
45
40
35
30
25
20
15
10
5
0

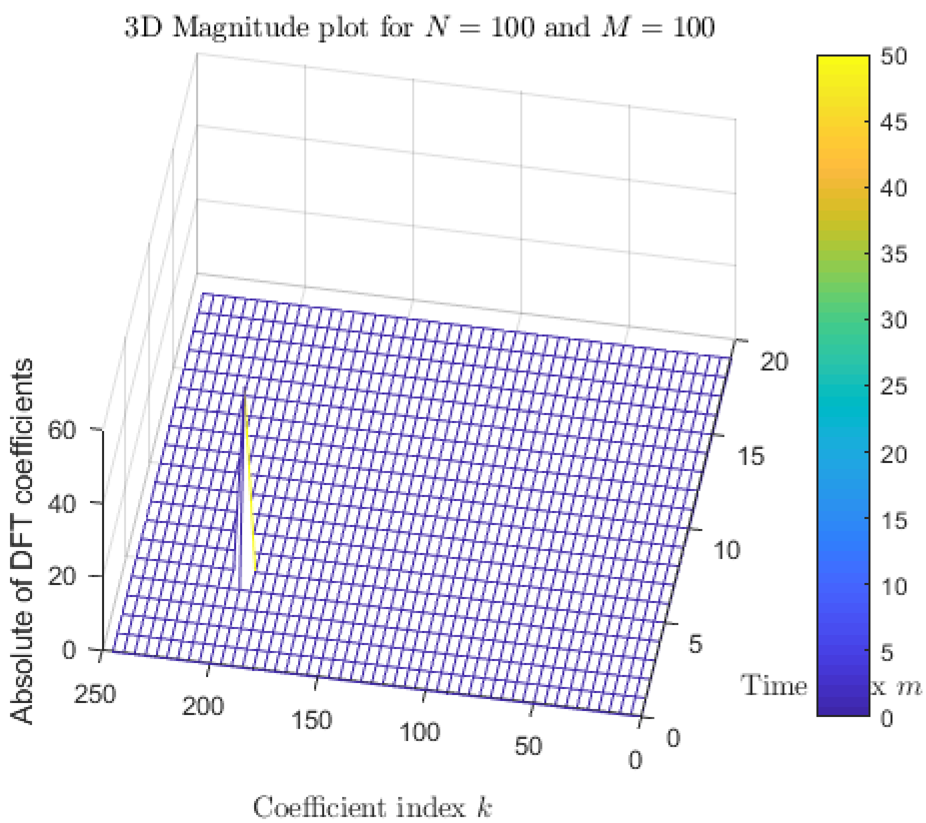

2D Angle plot for $N = 100$ and $M = 50$
Absolute of DFT coefficients
Time index $m$
0
50
100
150
200
0
5
10
15
20
25
30
35
3
2
1
0
-1
-2
-3

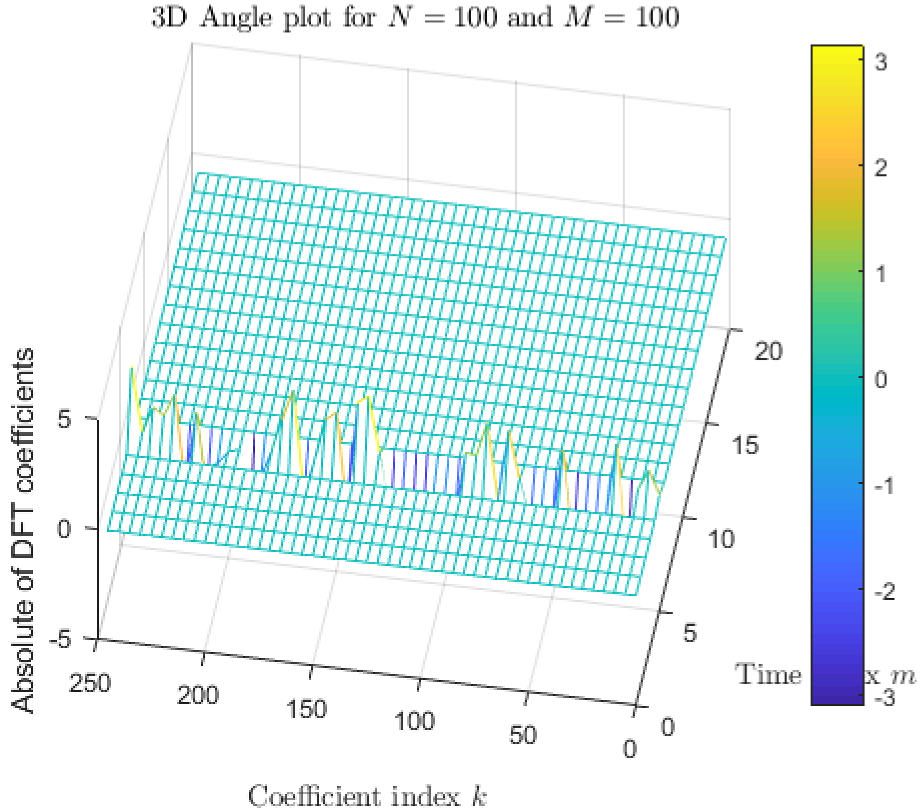

3D Magnitude plot for $N = 100$ and $M = 50$
Absolute of DFT coefficients
Coefficient index $k$
Time index $m$
0
20
40
60
250
200
150
100
50
0
0
10
20
30
40
50
45
40
35
30
25
20
15
10
5
0

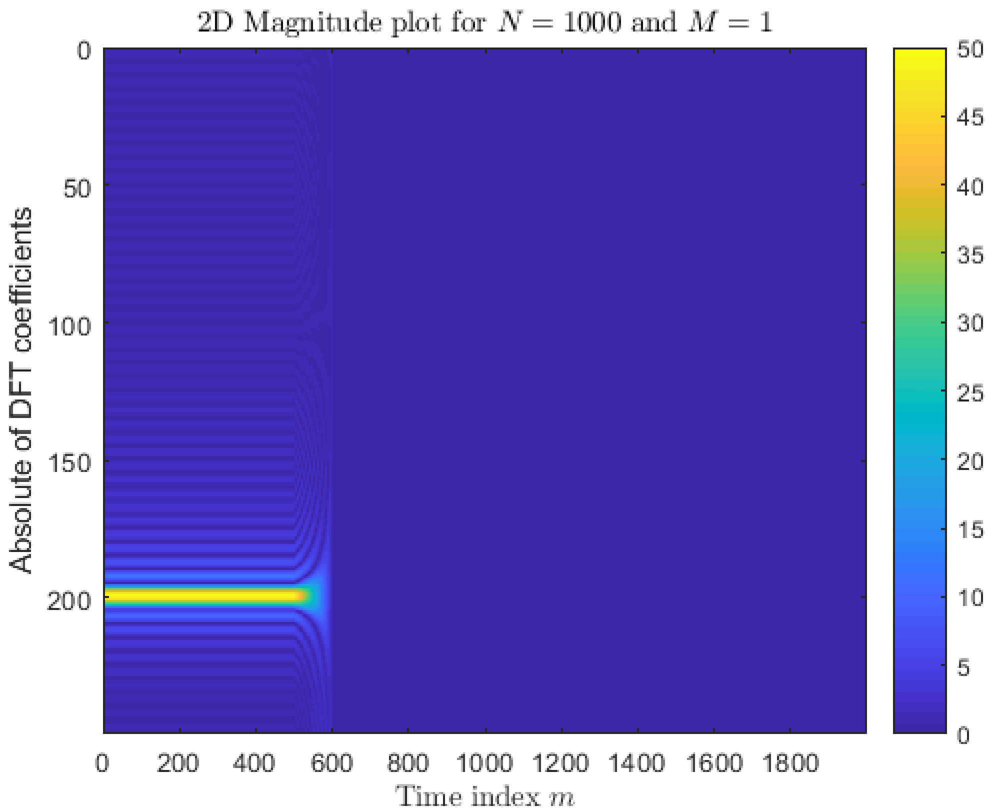

3D Angle plot for $N = 100$ and $M = 50$
Absolute of DFT coefficients
Coefficient index $k$
Time index $m$
5
0
-5
250
200
150
100
50
0
10
20
30
40
3
2
1
0
-1
-2
-3

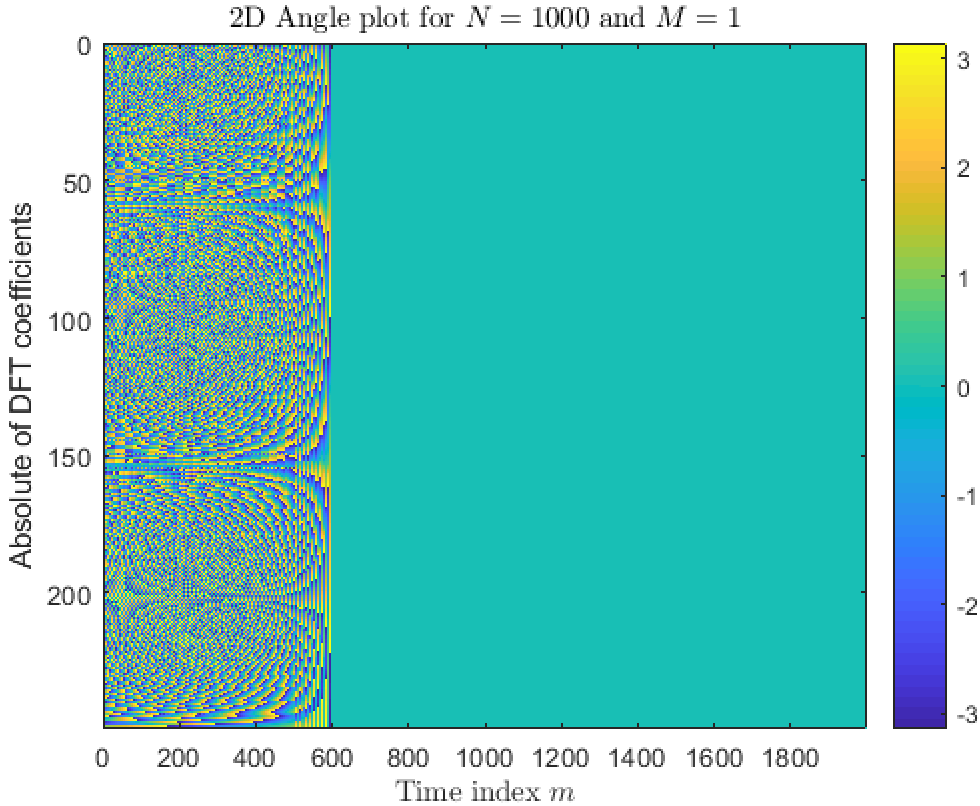

2D Magnitude plot for $N = 100$ and $M = 100$
Absolute of DFT coefficients
Time index $m$
0
50
100
150
200
0
2
4
6
8
10
12
14
16
18
50
45
40
35
30
25
20
15
10
5
0

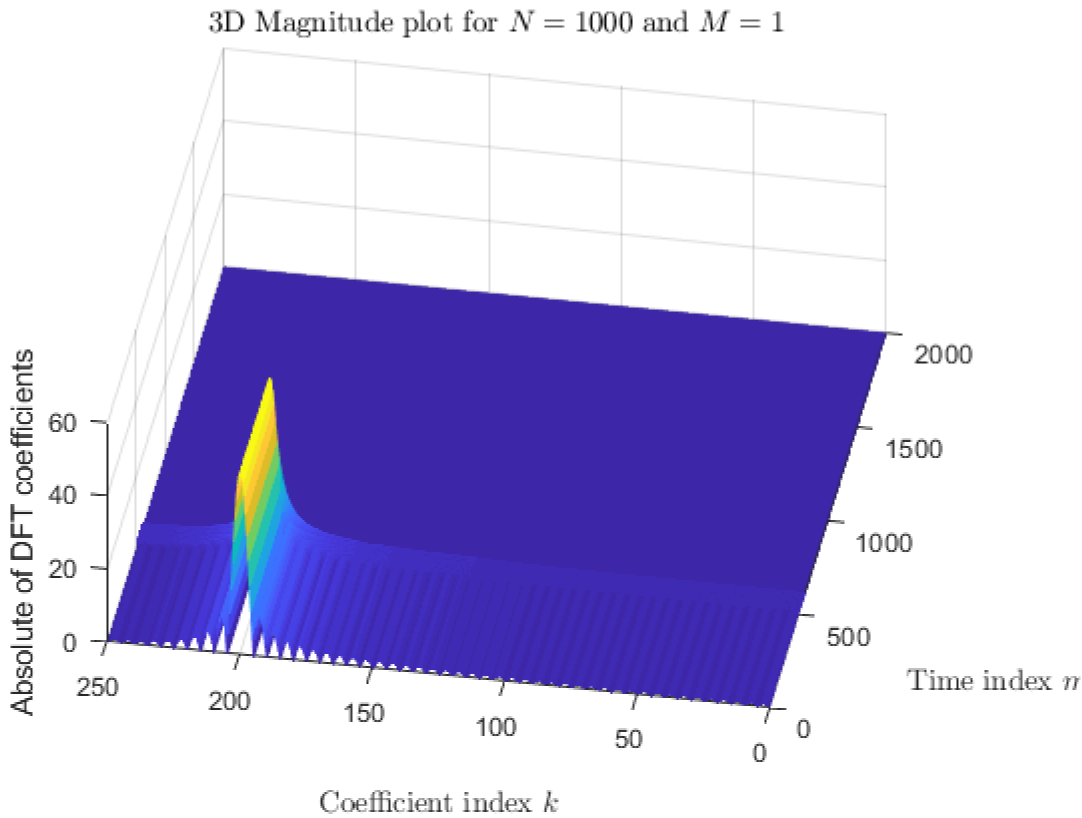

2D Angle plot for $N = 100$ and $M = 100$
Absolute of DFT coefficients
Time index $m$
0
50
100
150
200
0
2
4
6
8
10
12
14
16
18
3
2
1
0
-1
-2
-3

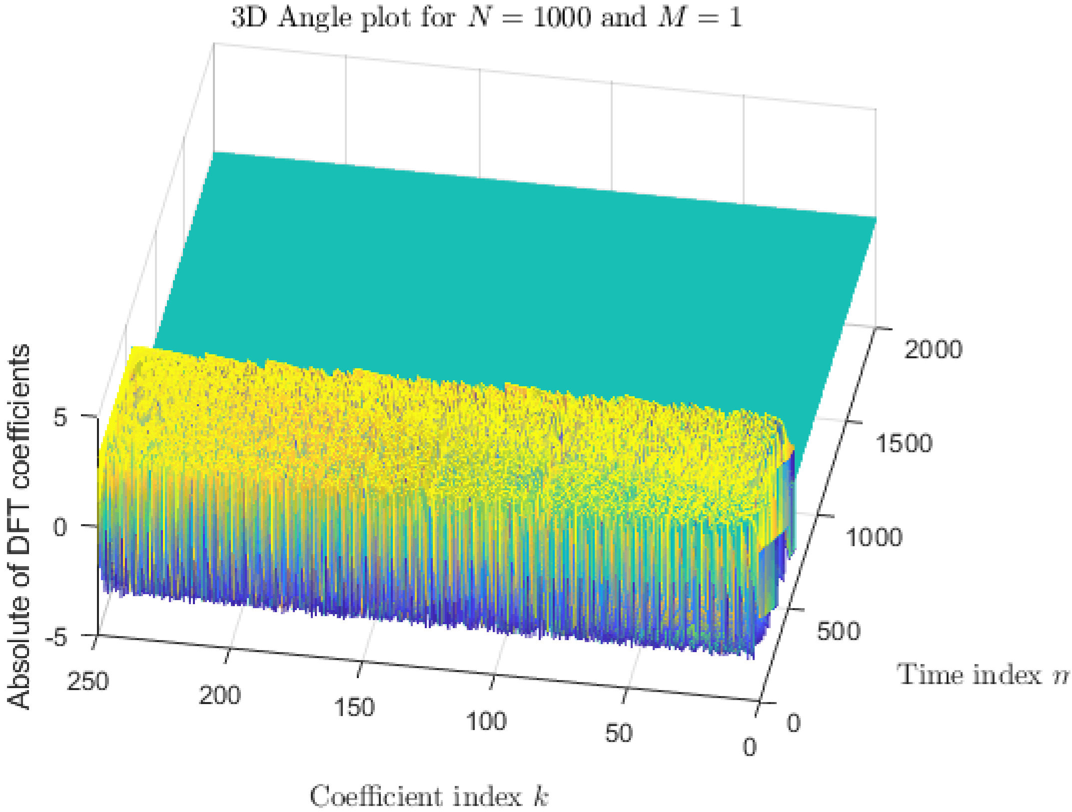

3D Magnitude plot for $N = 100$ and $M = 100$
Absolute of DFT coefficients
Coefficient index $k$
Time index $m$
0
20
40
60
250
200
150
100
50
0
5
10
15
20
50
45
40
35
30
25
20
15
10
5


3D Angle plot for $N = 100$ and $M = 100$
Absolute of DFT coefficients
Coefficient index $k$
Time index $m$
250
200
150
100
50
0
5
0
-5
5
10
15
20
3
2
1
0
-1
-2
-3


2D Magnitude plot for $N = 1000$ and $M = 1$
Absolute of DFT coefficients
Time index $m$
0
50
100
150
200
0
200
400
600
800
1000
1200
1400
1600
1800
50
45
40
35
30
25
20
15
10
5
0


2D Angle plot for $N = 1000$ and $M = 1$
0
50
100
150
200
Absolute of DFT coefficients
0
200
400
600
800
1000
1200
1400
1600
1800
Time index $m$
3
2
1
0
-1
-2
-3


3D Magnitude plot for $N = 1000$ and $M = 1$
60
40
20
0
Absolute of DFT coefficients
250
200
150
100
50
0
Coefficient index $k$
2000
1500
1000
500
0
Time index $m$




# Function for Columnwise DFT.

define a function which will calculate the DFTs for each column and store it as another matrix.

```
function dftOut = colDFT(N, M, matr)
   % find the N pt DFT matrix (using the user defined DFT matrix
generator).
   D = UDFT(N);
   % find the number of columns in the matrix.
   numcols = size(matr, 2);
   % initialize a blank dftOut matrix.
   dftOut = [];
   % now iterate over the columns.
   for col = 1:numcols
      current_column = matr(:, col);
      % multiply by the DFT matrix and store the result as column
in the dftOut matrix.
      dftOut(:, size(dftOut,2)+1) = D*current_column;
   end
   % the final result is dftOut.
end
```

# User Defined DFT matrix generator

```
function dftmtxout = UDFT(N)
```

```
        % define a dft matrix dftmtxout.
        dftmtxout = [];

        % now iterate over rows which represent 'k' in the DFT matrix
and then iterate over columns which represent
        % 'n' values in the DFT matrix.
        for k = 1:N
           row = [];
           for n = 1:N
               row = [row exp(-j*2*pi*(k-1)*(n-1)/N)];
           end
           dftmtxout = [dftmtxout; row];
        end
     end
```

# Function to show Chirp Function Usage - Question 1

```
     function y = question1()
        % we require 2000 samples and the sampling freq = fs = 500 Hz.
        % 2000 X (1/500.)s = 4 sec
        fs = 500;
        t = 0: 1/fs: 4- (1/fs);

        % use chirp now.
        y = chirp(t, 0, 2, 125);
        % plot this figure.
        figure;
        subplot(1,1,1);
        plot(t, y);
        grid on;
        xlabel('time in seconds');
        ylabel('Amplitude of $x(n)$', 'interpreter', 'latex');
        title('The chirp cosinusoid $ \displaystyle x(n) = cos\Big(2\pi
\Big(\frac{f_{i} + \beta n}{f_{s}} \Big) n \Big)$ sampled at
$f_{s}=500Hz$', 'interpreter', 'latex');

     end
```

# Function for dividing Signal into overlapping blocks - Question 2

```
     function block_matrix = block_it(signal, N, M)
         % duration of the signal.
         L = length(signal);

        % compute the zeros needed to be added.
        zeros_needed = N - rem(L, M);
        signal = [signal zeros(1, zeros_needed)];
```

```
        % count the number of iterations.
        uplimit = L/M;
        % define a matrix.
        block_matrix = [];

        % initialise the iterations.
        for i = 1:uplimit
            % get the current block.
            start_index = (i-1)*M + 1;
            stop_index  = (i-1)*M + N;

            current_block = signal(start_index:stop_index);

            % add this as a row the matrix.
            block_matrix = [block_matrix; current_block];

            % exit the loop after all the iterations are completed.
        end
        % take the transpose of the matrix.
        block_matrix = transpose(block_matrix);
    end

end
```

# Comments on the STFT of the Chirp Cosinusoid

It is clear from the 2D plots that as we increase the value of $M$ from 1 to 100, the quality of the line in the spectrogram decreases. That happens because when $M = 1$, we are having the maximum overlap and thats why we get a sharper resolution for the change in frequency over time. **M is similar to the time derivative in the case of continuous time**. The more smaller the $M$ is, the more precise rate of change can be observed in the frequency of the cosinusoid. If $M = 100$, which is the case of least overlap, we are actually missing out the important changes in the frequency. So, we won't observe those changes which might have occured in between. However, that is the important part. We might not always need to make $M = 1$, because that might not even be required. What is required is that we are able to tune $M$ in range 1 to $N$ (here, 100) to get the required graph for change in frequency over time. Now when $N = 1000$ and $M = 1$, we are actually dividing the signal into 1000 rows and each time we are observing the change in frequency by setting the time slot $M = 1$ for good precision. Since the chirp is a linear increase in the frequency of cosinusoid (atleast in this case), we observe that the 2D and 3D plots are straight lines, i.e., the DFT coefficient magnitude increase with time and therefore the **center frequency** also increases, this is the implication over here.

# Comments on STFT of the inputSignal.mat signal.

This new signal has a single frequency or some set of frequencies at certain time. Most of the time the frequency present in it is 0 Hz or D.C. value. **The comments made about chirp cosinusoid made above apply same here as well**. What's new here is that there is only a single frequency present in the signal starting at sample value 500 and then the same frequency sharply decreases to 0 Hz again, that is the D.C. value again and remains that throughout. That's why we see a single dot like spectrum at only a particular

$m$ value which is a particular instant of time. Also, one more interesting feature to note here is that when $N = 100$, as we increase $M$ from 1 to 100, the degree of precision decreases in the sense ripples in the plot decreases. That is there are more ripples in $M = 1$ and almost no ripples when $M = 100$.

*Published with MATLAB® R2017b*