Digital Signal Processing [Lab-7]

Table of Contents

- Authors: *Kshitij Srivastava(1510110200)* and *Nilambar Saha(partner)(1510110246)*
- Lab Instructor: *Dr. Ravi Kant Saini*

Objective:

Time-Frequecy analysis and Sliding DFT (In this lab we found the time frequency analysis of a sequence by dividing into the blocks and we uing DFT to compute indicidual blocks. In the sliding DFT, we use compute the first DFT by normal DFT and compute futher DFT of blocks by subtracting and multiplication)

Program:

```
clc;
clear all;
close all;

% * |*Matlab Commands for finding the input from the equation*|
num_samples=10/(1/500);%total time of sample divided by time for one sample
w0=2*pi*10;% Finding the angular frequency of the sweeping frequency f0
w1=2*pi*200;% Finding the angular frequency of the sweeping frequency f1

k=(w1-w0)/10; % Rate in the change in the digital frequency

i=1;
for n=0:1/500:10-(1/500)
    phase= (k*n*n*0.5) + (w0*n) ;%Finding the phase
    x(i)=cos(phase); %Finding the input signal
    i=i+1;%variable for the index
end
```

- **Matlab Commands for dividing the total signal into blocks**

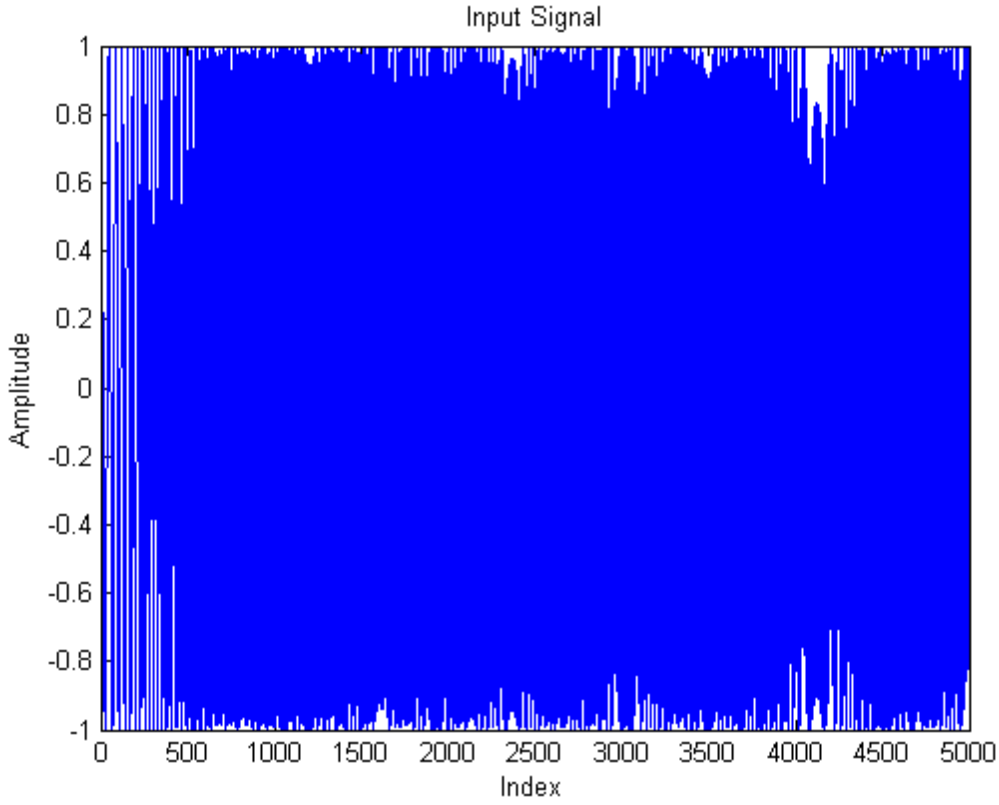
```
no_blocks=0.1/(1/500); % Finding the total number of blocks
no_overlap=no_blocks*0.98; % Finding the overlapped blocks

input_block=zeros(50,5000-50+1);%  matrix of 51x4951 for input blocks
for i=1:4951
    block=x(i:i+49);% Selecting i to i+49 elements of input x
    input_block(:,i)=block;%Placing the block in the matrix's ith column
end
```

- **Matlab Commands computing the DFT of block matrix**

```
output_block=zeros(50,5000-50+1);% Creating a output matrix for input blocks
 j=1;
 for i=1:4951
     block=input_block(:,i);
     fft_block=fft(block);%Finding the DFT of the block input
    output_block(:,i)=fft_block;%Placing DFT output in the block output
 end

 output_block_abs=abs(output_block);%Finding the absolute values of output
 %matrix

 % * |*Matlab Commands for Sliding DFT*|
 out_block_sliding=zeros(50,4951);%Output matrix of sliding DFT
multipli_term=zeros(1,50);% Initializing multiplication term to zero
subtract_term=zeros(1,50);%Initializing Subtraction term to zero

for k=0:49 % making of multiplication term with k from 0 to 49
    multipli_term(k+1)=exp((2*pi*k*1i)/50);
end

X_block=zeros(1,50);
for i=1:4951
    if i==1 % For first block
        X_block=fft(x(1:50));%Finding The DFT of the first Block
        out_block_sliding(:,i)=X_block; %placing the 1st block in 1st column
    else % Fir blocks other than the first
    subtract=-x(i-1)+x(50+i-1);%Subtraction of first and addition of new term
    for j=1:50
        subtract_term(j)=subtract;%Making of subract term
    end
    X_block= (X_block-subtract_term).* multipli_term;
    %New X_block term is made from subtraction of X_block from subtract
    %term and element by element multiplication with multiplication term
    out_block_sliding(:,i)=X_block;%Storing the X_block in the ith column

    end
end

out_block_sliding_abs=abs(out_block_sliding);% Finding the absolute of the
%out_block_sliding matrix
```

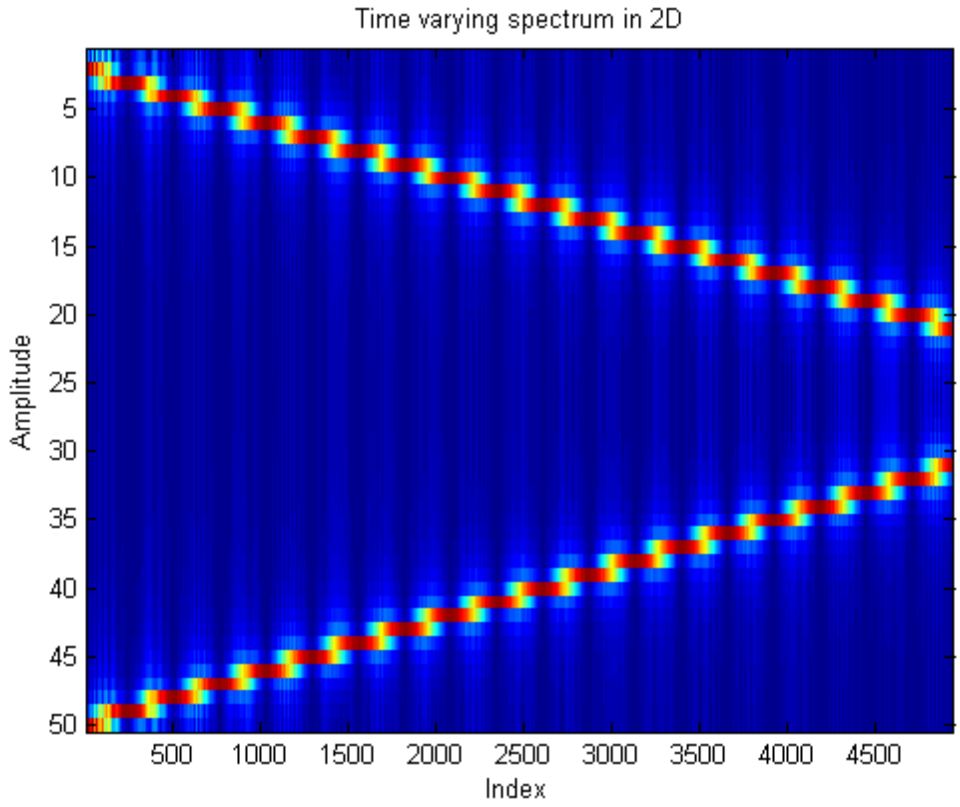
Results:

```
• *Plot for the Question No 1a-The input signal *

figure;plot(x);
title('Input Signal');xlabel('Index');ylabel('Amplitude');
```



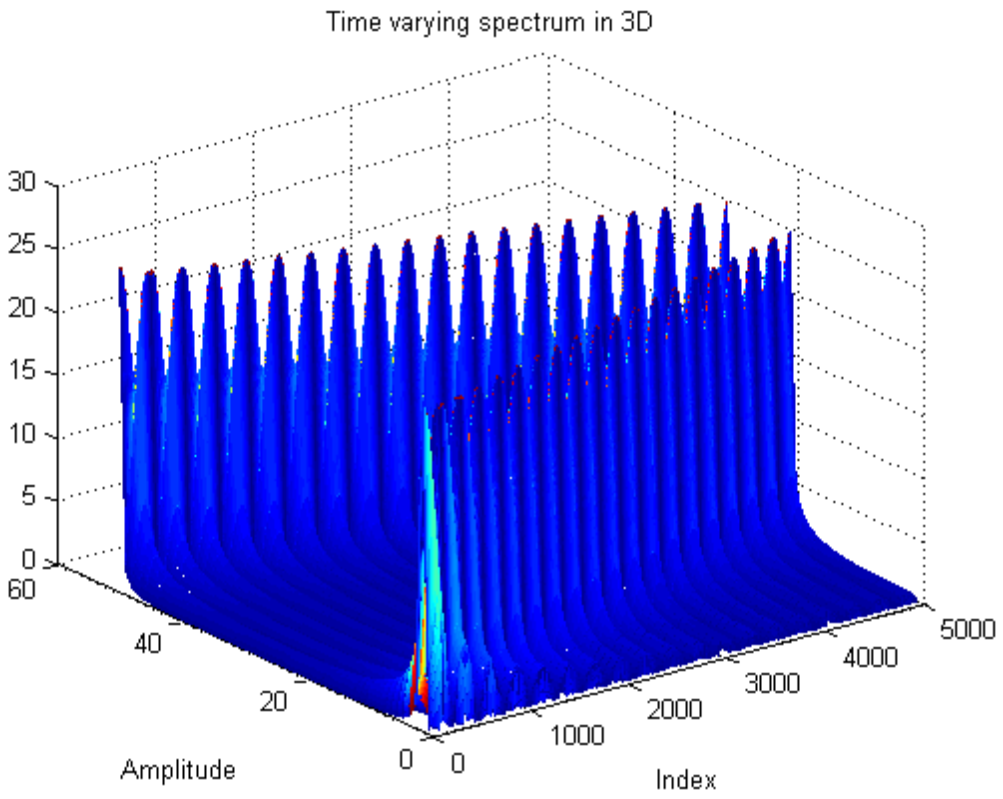
- **Plot for the Question No 1d-The time varying spectrum in 2D**

```
figure;imagesc(output_block_abs);
title('Time varying spectrum in 2D');xlabel('Index');ylabel('Amplitude');
```



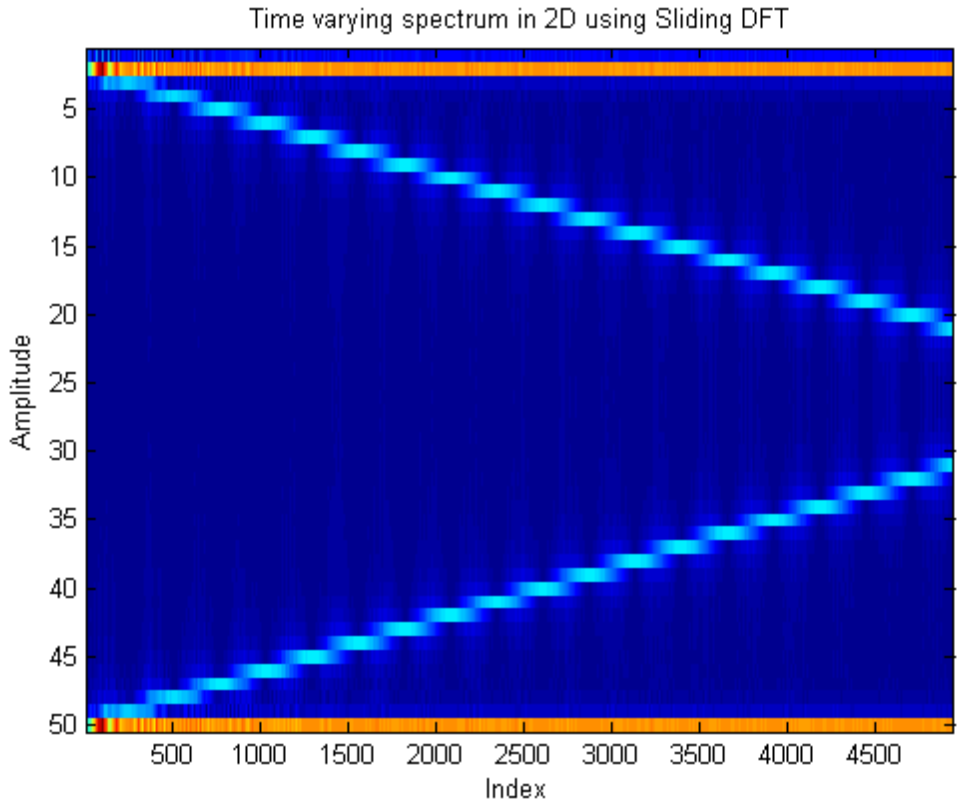
- **Plot for the Question No 1e-The time varying spectrum in 3D**

```
figure;mesh(output_block_abs)
title('Time varying spectrum in 3D');xlabel('Index');ylabel('Amplitude');
```



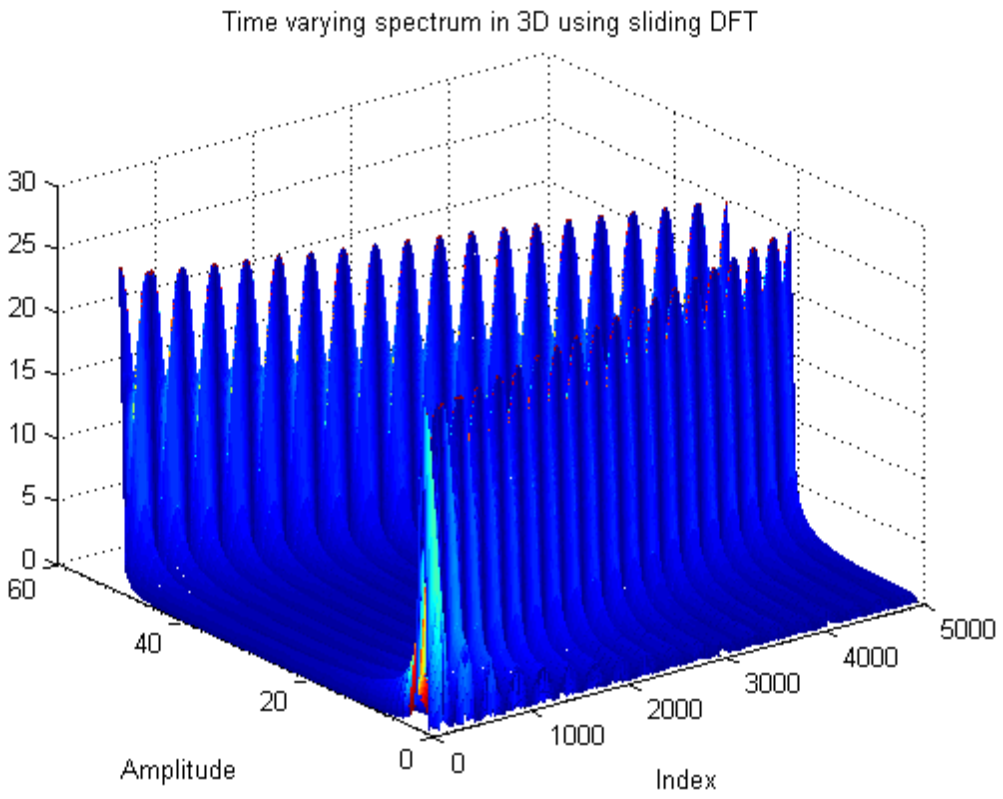
- *Question No 2-The time varying spectrum in 2D using sliding DFT *

```
figure;imagesc(out_block_sliding_abs);
title('Time varying spectrum in 2D using Sliding DFT');
xlabel('Index');ylabel('Amplitude');
```



- *Question No 2-The time varying spectrum in 3D using sliding DFT *

```
figure;mesh(output_block_abs);
title('Time varying spectrum in 3D using sliding DFT');
xlabel('Index');ylabel('Amplitude');
```



- **Question No 3-Saving in the number of multiplications**

```
%multiplication in the time frequecy response is 4951*50*50= 12377500
%As square(N) is the time complexity of finding DFT

%total Multiplication in the the sliding DFT is 4951*50= 247550
% N is the time complexity of finding the DFT

%So total savings is 12129950
```

Published with MATLAB® R2014a