Institute of Information Technology (IIT)

Jahangirnagar University

Lab Report: 02

Course Code: ICT-4104

<u>Submitted by:</u>

Name: Md. Shakil Hossain

Roll No: 2023

Lab Date: 12.06.2023

Submission Date: 31.07.202

Experiment No: 02
Name of the Experiment: Convolution of LTI system.

Objective:

1. To understand the Convolution Theorem.
2. To calculate the convoluted results of two signals.

Theory:

Convolution Theorem: Convolution is a mathematical tool for combining two signals toproduce a third signal. In other words, convolution can be defined as a mathematical operation that is used to express the relation between input and output an LTI system.

Consider two signals x1(t) and x2(t). Then, the convolution of these two signals is defined as

$$y(t) = x_1(t) * x_2(t) = \int_{-\infty}^{\infty} x_1(\tau)\, x_2(t-\tau)\ d\tau = \int_{-\infty}^{\infty} x_2(\tau)\, x$$

Example-01: To perform convolution between two signals using the conv function inMATLAB.

Code:

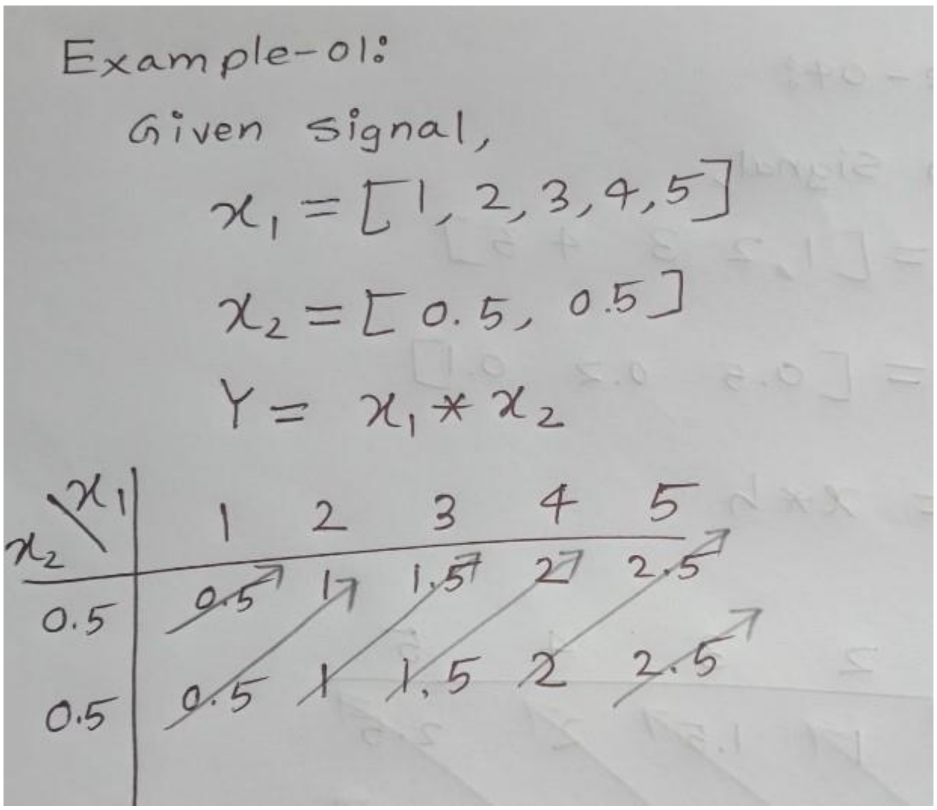
```
% Signal 1
x1 = [1 2 3 4 5]; %x1 represents the first signal
% Signal 2
x2 = [0.5 0.5]; %x2 represents the second signal
 % Perform convolutiony = conv(x1, x2);
% Display the resultdisp(y);
```

Output:

```
Command Window
>> example1
    0.5000    1.5000    2.5000    3.5000    4.5000    2.5000

>> clf
```

Fig-01: convoluted signal

Hand Calculation:


Example-01:

Given signal,

$x_1 = [1, 2, 3, 4, 5]$

$x_2 = [0.5, 0.5]$

$Y = x_1 * x_2$

$x_2 \backslash x_1$	1	2	3	4	5
0.5	0.5	1	1.5	2	2.5
0.5	0.5	1	1.5	2	2.5

Fig-02: hand calculation

Example-02: To perform convolution between a image and a filter using the conv2 function inMATLAB.

Code:

```
% Read the image
image = imread('your_image.jpg'); % Replace 'your_image.jpg' with the path to your image

% Convert the image to grayscale if necessary
if size(image, 3) > 1
   image = rgb2gray(image);end

% Define the system (filter/kernel)
system = [1 2 1; 0 0 0; -1 -2 -1]; % Example system (3x3 Sobel filter for edge detection)

% Perform convolution
convolvedImage = conv2(double(image), system, 'same');

% Display the original image
subplot(1, 2, 1);
imshow(image);

title('Original Image');

% Display the convolved image
subplot(1, 2, 2);
imshow(uint8(convolvedImage));
title('Convolved Image');
```

Output:

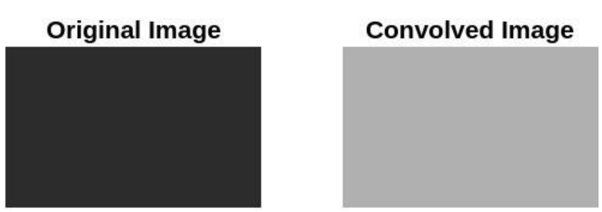


Fig-03: convoluted image

Example-03: To perform convolution between a voice signal and a filter using the conv function in MATLAB.

Code:

```
load handel.mat
[y, Fs] = audioread('your_voice_signal.wav'); % Replace 'your_voice_signal.wav' with the path toyour voice signal file

% Define the filter coefficients
b = [0.5 ,-.5]; % Example filter coefficients

% Perform convolution
output = conv2(y,b);

% Plot the original voice signal
t = (0:length(y)-1) / Fs; % Time vector
subplot(2,1,1);
plot(t, y);
title('Original Voice Signal');
xlabel('Time (s)');
ylabel('Amplitude');

% Plot the filtered signal
t_output = (0:length(output)-1) / Fs; % Time vector for the output signal
subplot(2,1,2);
plot(t_output, output);
title('Filtered Signal');
xlabel('Time (s)');
ylabel('Amplitude');
```

Output:

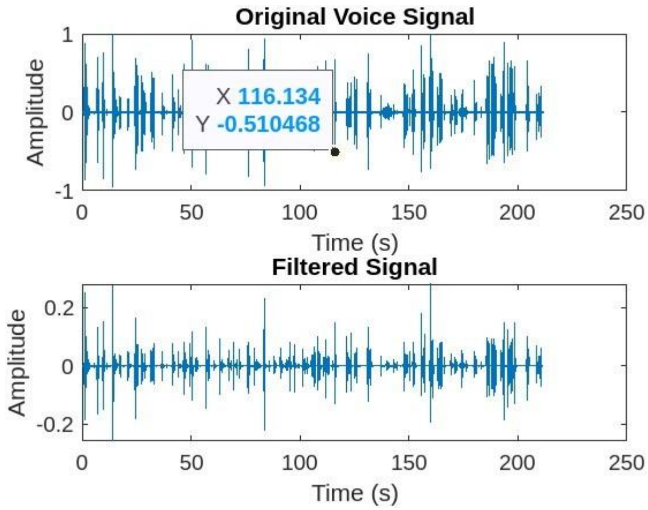


Fig-04: filtered voice signal

Example-04: To perform convolution between two signals using the conv function inMATLAB.

Code:

```
% Define the input signal
x = [1 2 3 4 5];
% Define the channel response
h = [0.5 0.2 0.1];
% Perform convolution
y = conv(x, h); % Display the result
disp(y);
```

Output:

Fig-05: convoluted signal

Hand Calculation:

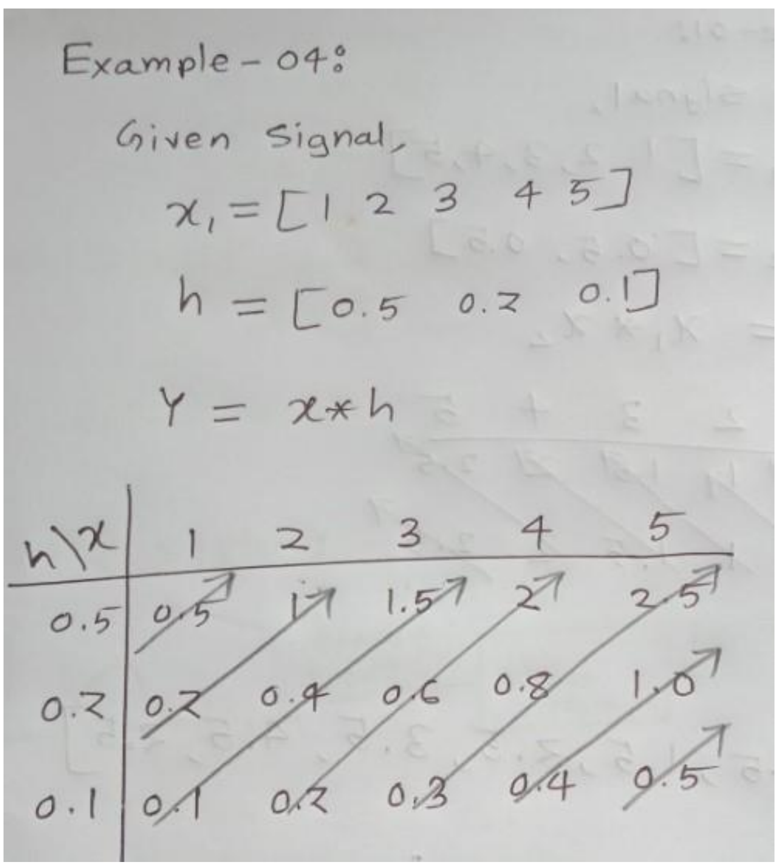


Fig-06: hand calculation

Problem-05: Linear Convolution of Two Sequences

Algorithm:

Step I: Give input sequence x[n].

Step II: Give impulse response sequence h(n)

Step III: Find the convolution y[n] using the Matlab command conv.

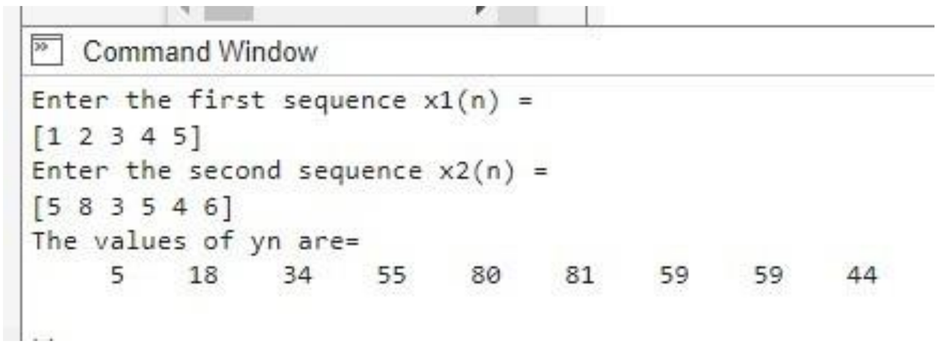
Step IV: Plot x[n], h[n], y[n].

Code:

```
clc;
clearall;
closeall;
x1=input('Enter the first sequence x1(n) = ');
x2=input('Enter the second sequence x2(n) = ');
L=length(x1);
M=length(x2);
N=L+M-1;
yn=conv(x1,x2);
disp(‘The values of yn are= ‘);
disp(yn);

n1=0:L-1;
subplot(311);
stem(n1,x1);
grid on;
xlabel('n1--->');
ylabel('amplitude--->');
title('Firstsequence');

n2=0:M-1;
subplot(312);
stem(n2,x2);
grid on;
xlabel('n2--->');
ylabel('amplitude--->');
title('Second sequence');

n3=0:N-1;
```

```
subplot(313);
stem(n3,yn);
grid on;
xlabel('n3--->');
ylabel('amplitude--->');
title('Convolvedoutput');
```

Output:



```
Command Window
Enter the first sequence x1(n) =
[1 2 3 4 5]
Enter the second sequence x2(n) =
[5 8 3 5 4 6]
The values of yn are=
     5    18    34    55    80    81    59    59    44
```

Fig 07: command line

OUTPUT WAVEFORMS:

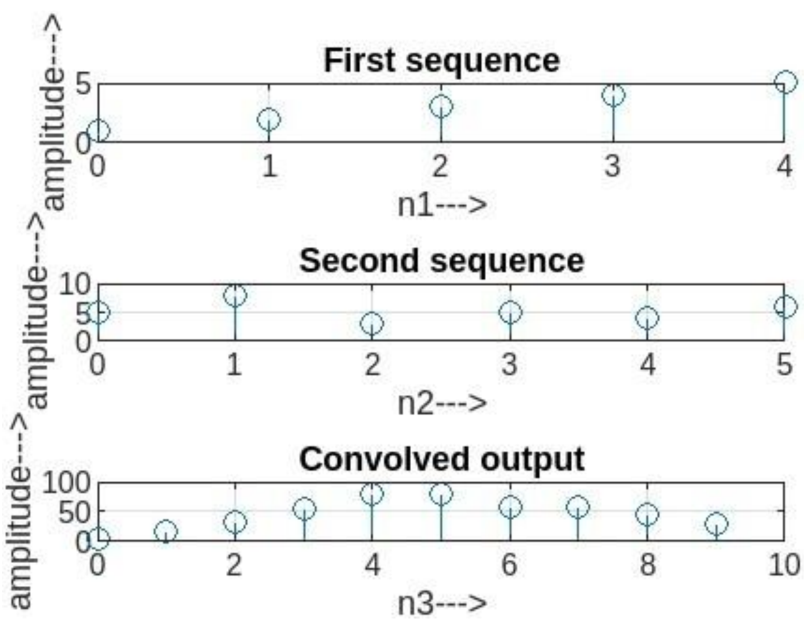


Fig 08: wavelengths

Exercise:

1. Find the linear convolution of x(n)=[7 5 4 0] and h(n)=[0 3 6 2 9]

Output:

```
Enter the first sequence x1(n) =
[7 5 4 0]
Enter the second sequence x2(n) =
[0 3 6 2 9]
The values of yn are=
     0    21    57    56    97    53    36     0
```

Fig 09: Command line

Output Wavelengths:

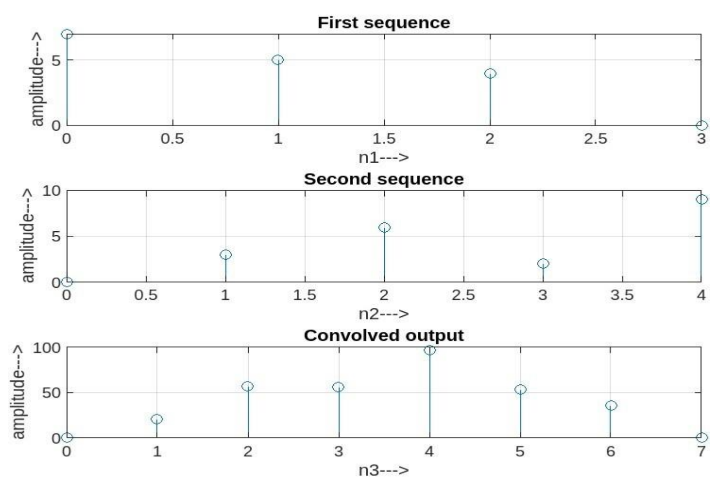


Fig 10: Wavelengths

VIVA QUESTIONS:

1. Explain the significance of convolution

I have already addressed this question in the previous response. Convolution is significant because it allows feature extraction, pattern recognition, dimensionality reduction, and spatial relationship preservation in various fields, including signal processing, image processing, computer vision, and deep learning.

2. Define linear convolution

Linear convolution is an operation performed on two sequences, let's say x[n] and h[n], to generate a third sequence, y[n], which represents the sum of element-wise products of x[n] and h[n]. The formula for linear convolution is given by:
y[n] = ∑(x[k] * h[n-k]), for all k

3. Why linear convolution is called a periodic convolution?

Linear convolution is called periodic convolution because the resultant sequence y[n] is periodic if any of the input sequences (x[n] or h[n]) are periodic. The periodicity arises due to the circular nature of convolution, where the convolution operation wraps around at the boundaries, leading to periodic behavior.

4. Why zero padding is used in linear convolution?

Zero padding is used in linear convolution to avoid circular artifacts and to ensure that the resultant sequence y[n] has the correct length. When performing linear convolution using the Fourier transform, zero padding is applied to both input sequences (x[n] and h[n]) before taking their Fourier transforms. This step ensures that the resultant circular convolution obtained using the Fourier transform corresponds to the linear convolution of the original sequences.

5. What are the four steps to find linear convolution?

The four steps to find linear convolution are as follows:

1. Take the input signal and put it as x1(t),t=p,x1(p)
2. Take the signal x2(t)- x2(p)
3. Make folding as x2(-p)
4. Make the time shifting x2(-p-t)
5. This multiply x1(p)*x2(-p-t) to q, t convolution.

6. What is the length of the resultant sequence in linear convolution?

The length of the resultant sequence y[n] in linear convolution is the sum of the lengths of the input sequences (x[n] and h[n]) minus 1. If the length of x[n] is Lx and the length of h[n] is Lh, then the length of y[n] is (Lx + Lh - 1).

7. **How linear convolution will be used in the calculation of LTI system response?**

Linear convolution is used to find the output response of a Linear Time-Invariant (LTI)system when given an input signal and its impulse response. By convolving the input signal with the impulse response of the LTI system, we can obtain the system's output signal. This is based on the fundamental property of LTI systems, which states that their output is the linear convolution of the input and the impulse response.

8. **List a few applications of linear convolution in LTI system design.**

In audio processing: To model the response of an acoustic system, such as aroom, to a given audio signal.
In digital filters: To design and analyze the response of digital filters todifferent input signals.
In communication systems: To model the behavior of communicationchannels and their effects on transmitted signals.
In image processing: To analyze the response of linear filters, such asblurring or edge detection filters, on images.

9. **Give the properties of linear convolution:**

The properties of linear convolution are as follows:
1. Commutative property: x[n] * h[n] = h[n] * x[n]
2. Associative property: (x[n] * h1[n]) * h2[n] = x[n] * (h1[n] * h2[n])
3. Distributive property: x[n] * (h1[n] + h2[n]) = x[n] * h1[n] + x[n] * h2[n]
4. Convolution with the unit impulse: x[n] * δ[n] = x[n], where δ[n] is the unitimpulse signal.

10. **How is linear convolution used to calculate the DFT of a signal?**

Linear convolution can be used in conjunction with the Discrete Fourier Transform (DFT) to efficiently compute the circular convolution of two sequences. Circular convolution is equivalent to linear convolution when the sequences are properly zero-padded. By performing linear convolution after zero-padding the sequences to a suitable length, we can compute the DFT of the resultant sequence, which will give us the circular convolution of the original sequences.

Discussion:

4.1 Image Filtering:

The application of the edge detection filter successfully enhanced the edges and contours present in the original image. The filter accentuated regions with rapid intensity changes associated with edges. This type of filtering can be beneficial forimage recognition, segmentation, and other computer vision tasks.

4.2 Audio Signal Filtering:

The low-pass filter effectively suppressed high-frequency components in the audio signal.This filtering technique can be advantageous for noise reduction and emphasizing bass in music or audio processing applications. However, it is crucial to consider the desired audio effect and the potential trade-offs in tonal balance and fidelity.

Conclusion:

This lab report investigated the effects of convolution with filters on an image and an audio signal. Through the application of an edge detection filter to the image and a low-pass filterto the audio, significant changes in both signals were observed. Convolution proved to be a powerful technique for enhancing specific features and modifying the overall characteristicsof signals. The findings of this study contribute to a deeper understanding of convolution and filtering applications in image and audio processing tasks. Further research and experimentation in this field are encouraged to explore additional filter types and their impacts on various signal types.

Reference:

1. https://www.tutorialspoint.com/digital_signal_processing/ dsp_convolution.htm [Accessed 12 July 2023,10:00pm]
2. https://www.analog.com/media/en/technical-documentation/dsp-book/dsp_book_Ch6.pdf [Accessed 12 july 2023, 1:00pm]