

## **Linear Convolution using Circular Convolution and Vice versa**

### **Aim**

1. To perform Linear Convolution using Circular Convolution.
2. To perform Circular Convolution using Linear Convolution.

### **Theory**

#### **Performing Linear Convolution Using Circular Convolution**

Method:

1. Zero-Padding:
  - Pad both sequences  $x[n]$  and  $h[n]$  with zeros to a length of at least  $2N-1$ , where  $N$  is the maximum length of the two sequences. This ensures that the circular convolution will not wrap around and introduce artificial periodicity.
2. Circular Convolution:
  - Perform circular convolution on the zero-padded sequences.
3. Truncation:
  - Truncate the result of the circular convolution to the length  $N_1 + N_2 - 1$ , where  $N_1$  and  $N_2$  are the lengths of the original sequences  $x[n]$  and  $h[n]$ , respectively.

#### **Performing Circular Convolution Using Linear Convolution**

Method:

1. Zero-Padding:
  - Pad both sequences  $x[n]$  and  $h[n]$  to a length of at least  $2N-1$ , where  $N$  is the maximum length of the two sequences.
2. Linear Convolution:
  - Perform linear convolution on the zero-padded sequences.
3. Modulus Operation:
  - Apply the modulus operation to the indices of the linear convolution result, using the period  $N$ . This effectively wraps around the ends of the sequence, making it circular.

## **Program**

### **a)Linear convolution using circular convolution**

```
clc;
clear ;
close all;
x=[1 2 3 4];
h=[1 1 1];
n=length(x)+length(h)-1;
x=[x zeros(1,n-length(x))];
h=[h zeros(1,n-length(h))];
x1=fft(x);
h1=fft(h);
y1=x1.*h1;
y=ifft(y1);
disp("Linear Convolution using Circular Convolution :");
disp(y);
```

### **b)Circular convolution using linear convolution**

```
clc;
clear all;
close all;
x=[1 2 3 4];
h=[1 1 1];
y=conv(x,h);
conv_len=max(length(x),length(h));
result=[y(1:conv_len)];
new_arr=[y(conv_len+1:length(y)) zeros(1,length(y)-conv_len)];
```

```
result=result+new_arr;
```

```
disp("Circular convolution using Linear Convolution:")
```

```
disp(result);
```

### **Result**

Performed a) Linear Convolution using Circular Convolution; b) Circular Convolution using Linear Convolution and verified result.

## **Observation**

### **a)Linear convolution using circular convolution**

Linear Convolution using Circular Convolution:

1   3   6   9   7   4

### **b)Circular convolution using linear convolution**

Circular convolution using Linear Convolution:

8   7   6   9