# Lab 3: Linear Convolution

**Students are divided into group of two persons to do this lab assignment.**

**Please wrap the e-file of your report & MATLAB files in a compressed file and then e-submit this file via TA email address by the due day.**

**Thank you.**

## 1. Goals

The goal of this lab is to perform linear convolution by various methods.

## 2. Lab resource

- PC with Matlab

## 2. Background

In a linear time invariant (LTI) system, the response of system can be calculated from the input x(n) and the impulse response h(n) as **linear convolution** below

$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] \qquad (1)$$

It is said that one of the most important operations in LTI systems is the linear convolution. Linear convolution can be evaluated in many different ways. If the sequences are mathematical functions (of finite or infinite duration), then we can analytically evaluate (1) for all n to obtain a functional form of y(n); or we can indirectly calculate by the use of Z-Transform and Inverse Z-Transform as the scheme in Figure 1.

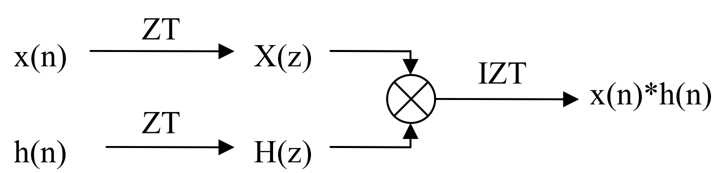


**Figure 1.** Linear convolution calculating scheme via Z-Transform

In practice, there is an efficient way to evaluate linear convolution that is using Discrete Fourier Transform (DFT). An algorithm proposed by Cooley and Tukey in 1965 to reduce the amount of computations involved in the DFT led to the explosion of using the DFT. It also led to the development of many other efficient algorithms that are known as Fast Fourier Transform (FFT) algorithms.

However, there is one problem. The DFT operations result in a circular convolution instead of linear convolution as we desire. In order to use the DFT for linear convolution, we must choose the length of sequences properly. If the length of x(n) is Nx, the length of h(n) is Nh, then the length of x(n) and h(h) should be extended up to Ny = Nx + Nh -1 before doing the circular convolution between new x(n) and new h(n). Figure 2 shows the scheme of doing linear convolution using circular convolution.

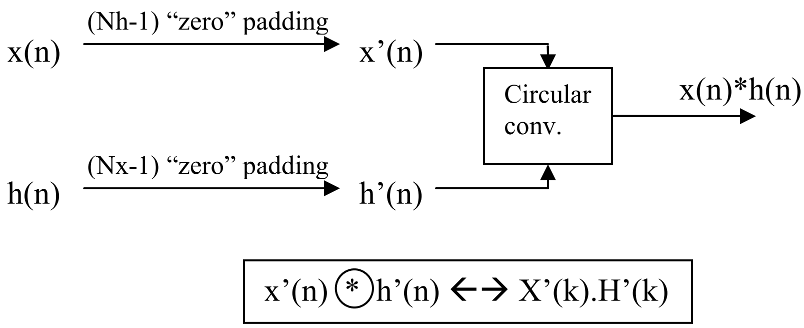


**Figure 2.** Linear convolution calculating scheme via circular convolution

# 4. Assignment

## 4.1. Matlab function "conv" (25%)

In the problems below, let x(n) = u(n-30) – u(n-50) and have time signal start at zero and last for 100 samples. Find the output of each of the LTI systems below.

You will need to do convolution of y(n) = x(n)*h(n) and can use the Matlab function "**conv**" which implements discrete-time convolution. The "**conv**" function takes two vector arguments. Plot x(n), h(n) and y(n) together using "**subplot**" for each part. Since these are discrete-time signals, you should plot them using the **stem** command. You should specify that you want only the first 100 points of y(n) in the plot, e.g. "**stem(y(1:100))**".

- the system has the impulse response h1(n) = u(n-10) – u(n-20)
- the system has the impulse response h2(n) = $(0.9)^n$u(n)
- the system has the impulse response h3(n) = δ(n) – δ(n-1)

## 4.2. Matrix-vector Multiplication to perform Linear Convolution (25%)

When x(n) and h(n) are finite duration Nx and Nh, respectively, then their linear convolution (1) can also be implemented using matrix-vector multiplication. If elements of y(n) and x(n) are arranged in column vectors **x** and **y** respectively, then from (1) we obtain

$$\mathbf{y} = \mathbf{H}\mathbf{x}$$

where linear shifts in h(n-k) for n = 0, 1, …, Nh-1 are arranged as rows in the matrix **H**. This matrix has an interesting structure and is called a *Toeplitz* matrix. To investigate this matrix, consider the sequence

x = [1 2 3 4] and h = [3 2 1]

- Determine the linear convolution y(n) = x(n)*h(n)

- Express x(n) as a 4x1 column vector x and y(n) as a 6x1 column vector y. Now determine the 6x4 matrix **H** so that **y = Hx**

- Characterize the matrix **H**. Can you give the definition of a Toeplitz matrix?

- What can you say about the first column and the first row of **H**?

- Develop an alternate Matlab function to implement linear convolution. The format of the function should be

```
function [y,H]=conv_tp(h,x)
% Linear Convolution using Toeplitz Matrix
% -------------------------------------------------------------------------------
% [y,H] = conv_tp(h,x)
% y = Output sequence in column vector form
% H = Toeplitz matrix corresponding to sequence h so that y =Hx
% h = Impulse Response sequence in column vector form
% x = Input sequence in column vector form
% -------------------------------------------------------------------------------
```

## 4.3. Z-transform and Inverse Z-Transform (25%)

In general, by using the definition of Z-Transform, Z-Transform table, and its properties, one can determine Z-Transform of common sequences. For example, using Z-Transform table, the properties of time-shifting property and differentiation in the Z-domain, we can determine the Z-Transform of

$$x(n) = (n-2)(0.5)^{n-2}\cos\left[\frac{\pi}{3}(n-2)\right]u(n-2)$$

as below

$$X(z) = \frac{0.25z^{-3} - 0.5z^{-4} + 0.0625z^{-5}}{1 - z^{-1} + 0.75z^{-2} - 0.25z^{-3} + 0.0625z^{-4}}$$

To check that the above X(z) is indeed the correct expression, let us use Matlab to compute the first 10 samples of the sequence x(n) corresponding to X(z) as below

```
>> b = [0 0 0 0.25 -0.5 0.0625]; a = [1 -1 0.75 -0.25 0.0625];
>> delta = [1 0 0 0 0 0 0 0 0 0];
>> n = 0:9;
>> x = filter(b,a,delta)
x =    0         0        0        0.25000000000000       -0.25000000000000       -0.37500000000000
-0.12500000000000      0.07812500000000      0.09375000000000      0.02734375000000
>> u = [0 0 ones(1,8)];
>> x = (n-2).*((1/2).^(n-2)).*cos(pi*(n-2)/3).*u
x =    0         0        0        0.25000000000000       -0.25000000000000       -0.37500000000000
-0.12500000000000      0.07812500000000      0.09375000000000      0.02734375000000
```

For the Inverse of Z-Transform problem, the most practical approach is to use the partial fraction expansion method. It makes the Z-Transform table. The Z-Transform, however, must be a rational function of $z^{-1}$.

A Matlab function **residuez** is available to compute the residue part and the direct terms of a rational function in $z^{-1}$. Type **help residuez** in Matlab command window to learn the information about this **residuez** function.

- Calculate the linear convolution y(n) = x(n)*h(n) where

$$x(n) = (n-2)(0.5)^{n-2}\cos\left[\frac{\pi}{3}(n-2)\right]u(n-2)$$

and

$$h(n) = \left(\frac{1}{3}\right)^{n} u(n) + \left(\frac{1}{2}\right)^{-n} u(-n-1)$$

Hand in your manual solution together with the Matlab code.

### 4.4. Circular Convolution (25%)

Do the same linear convolutions in problem 1 using the circular convolution. Note that circular convolution should be calculated via DFT. You will need to use the Matlab functions **fft** and **ifft** to do this assignment. You can type **help fft** and **help ifft** to get information about these Matlab functions. Compare the results of this assignment with the results getting from assignment 1.

**GOOD LUCK!!!**