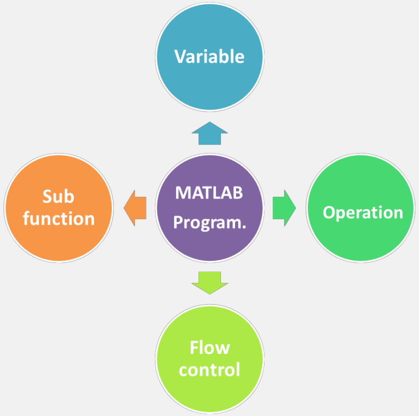
Signals and Systems (Lab)

Lab 2: Linear Time-Invariant Systems

Dr. Wu Guang
wug@sustc.edu.cn
Electrical & Electronic Engineering
Southern University of Science and Technology

Review

- ✓ What's MATLAB
- ✓ Every Variable is a Matrix
- ✓ Matrices Operations
- ✓ Flow Control
- ✓ User Defined Functions
- ✓ Display Facilities

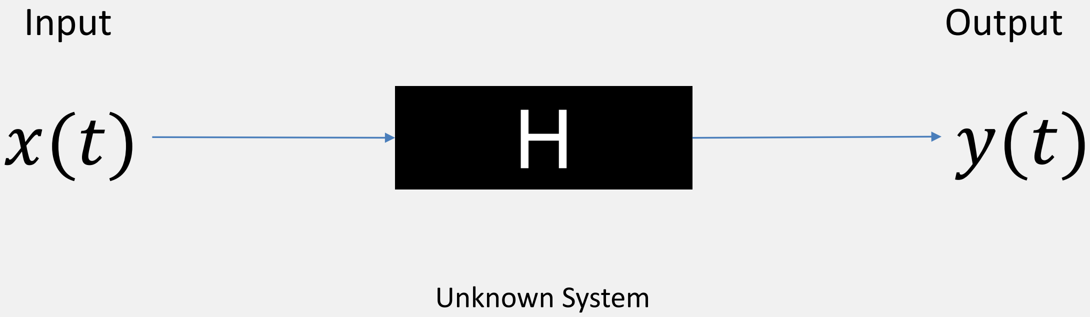


Feedback

Overview

- Verify the property of ***convolution***
- Verify the property of LTI systems
- Design a LTI system for echo cancellation

What's the output of the system ?



Two steps to calculate the output

- Step 1: to measure the system function:

$$\delta(t) \longrightarrow \boxed{H} \longrightarrow h(t)$$

- Step 2: to represent the $x(t)$ as a combination of the $\delta(t)/\delta(t-kT_s)$:

$$\sum_k I_k \delta(t-kT_s) \longrightarrow \boxed{H} \longrightarrow \sum_k I_k h(t-kT_s)$$

Requirement

- The system H is Linear Time Invariant (LTI):

$$I_k\delta(t) \longrightarrow \boxed{H} \longrightarrow I_k h(t)$$

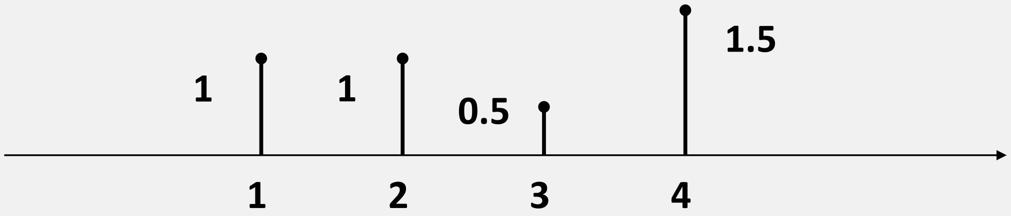
$$\delta(t - kT_s) \longrightarrow \boxed{H} \longrightarrow h(t - kT_s)$$

- The definition of convolution:

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

Exercise 1: Discrete-Time signal

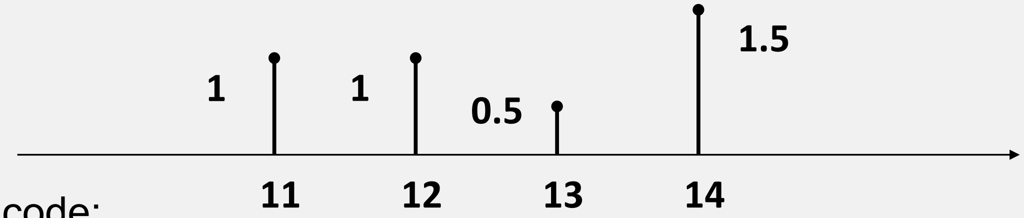
- Discrete-Time signal (DT signal):



- Matlab code:

```
x=[1 1 0.5 1.5]
stem(x)
```

- Another DT signal:



- Matlab code:

```
x=[1 1 0.5 1.5]
stem([11 12 13 14], x)
```

Signal value

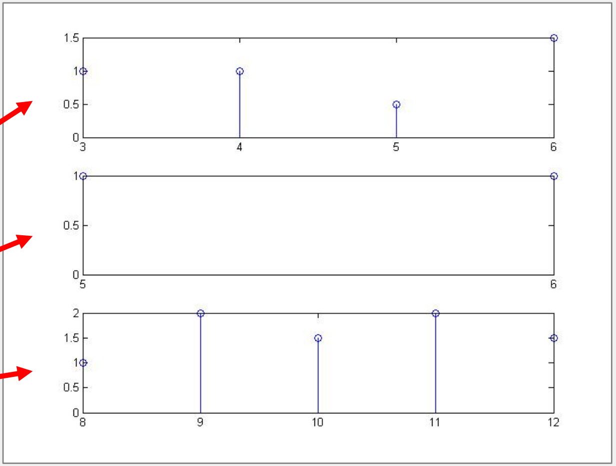
```
x=[1 1 0.5 1.5]
nx=[11 12 13 14]
stem(nx, x)
```

Non-zero region of time index

Exercise 2: Calculate Convolution by **conv()**

- Step 1: Define u, nu and v, nv
- Step 2: Calculate the signal values of w[n] by **w=conv(u,v)**
- Step 3: Calculate the non-zero interval of w[n] by **nw=nu(1)+nv(1):nu(end)+nv(end)**

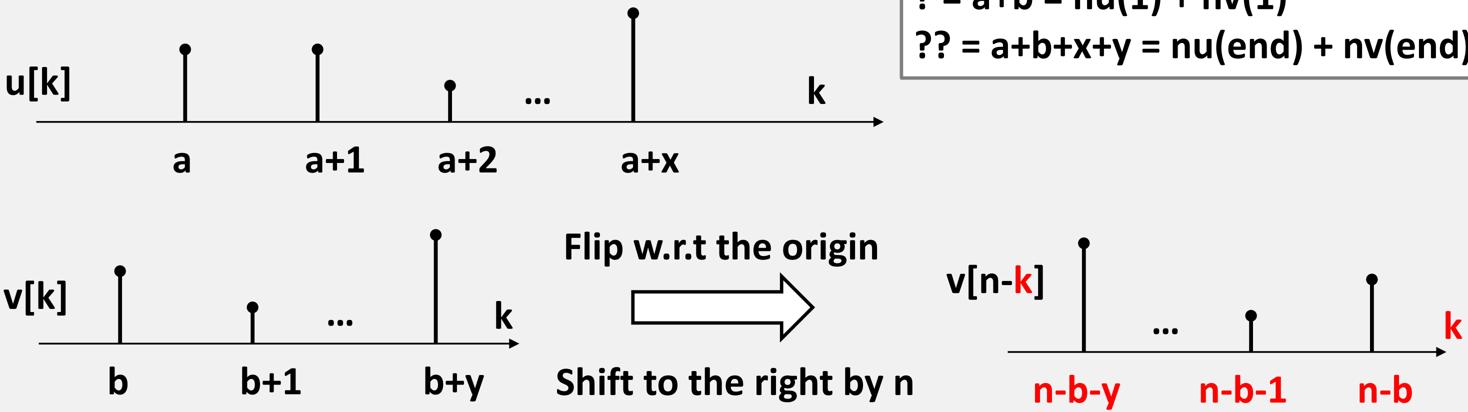
```
u=[1 1 0.5 1.5]
nu=3:6
v=[1 1]
nv=5:6
w=conv(u,v)
nw=nu(1)+nv(1):nu(end)+nv(end)
subplot(3,1,1), stem(nu,u)
subplot(3,1,2), stem(nv,v)
subplot(3,1,3), stem(nw,w)
```



DT Signal Convolution

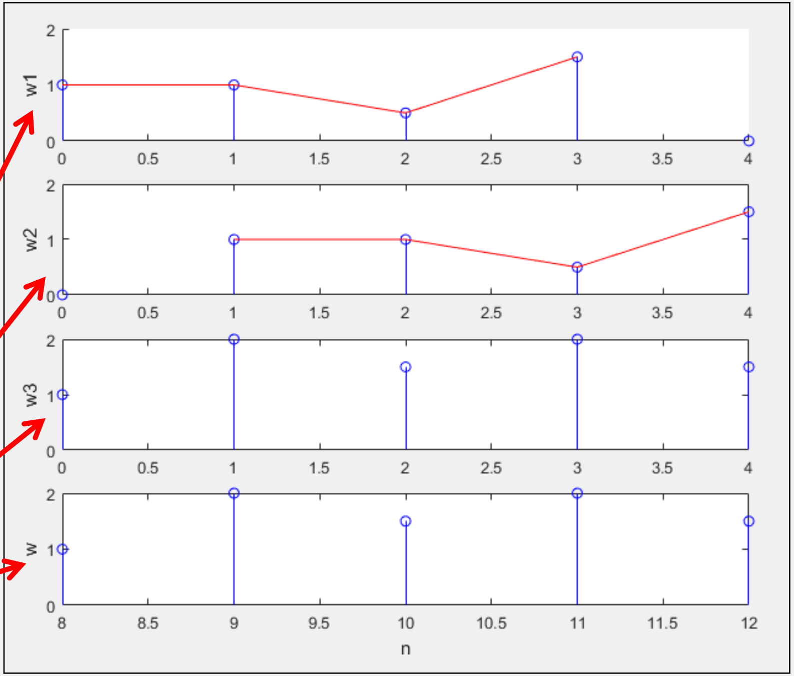
- $w[n] = u[n] * v[n] = \sum_{k=-\infty}^{+\infty} u[k]v[n-k]$
  - Non-zero interval of DT signal u: **nu=a:a+x**
  - Non-zero interval of DT signal v: **nv=b:b+y**
  - Non-zero interval of w[n]: **nw=?:??**

n-b=a => n=a+b
n-b-y=a+x => n=a+b+y+x
Hence,
nw=[a+b:a+b+x+y]
? = a+b = nu(1) + nv(1)
?? = a+b+x+y = nu(end) + nv(end)



Exercise 3: Another method

```
nu=3:6
nv=5:6
nw=nu(1)+nv(1):nu(end)+nv(end)
u=[1 1 0.5 1.5]
v=[1 1]
w1=[u 0]
w2=[0 u]
w3=w1+w2
w=conv(u,v)
subplot(4,1,1), stem([0:4],w1)
subplot(4,1,2), stem([0:4],w2)
subplot(4,1,3), stem([0:4],w3)
subplot(4,1,4), stem(nw,w)
```



LTI System by Difference Equation

- Reading assignment: textbook 2.4.

- Causal DT LTI system can be specified by a linear constant-coefficient difference equation:

$$\boxed{\sum_{k=0}^{K} a_k y[n-k] = \sum_{m=0}^{M} b_m x[n-m]}$$

- For example:
  - $y[n]=0.5x[n]+x[n-1]+2x[n-2]$;

    h[n]=? Finite Impulse Response (FIR)
  - $y[n]-0.8y[n-1]=2x[n]$;

    h[n]=? Infinite Impulse Response (IIR)

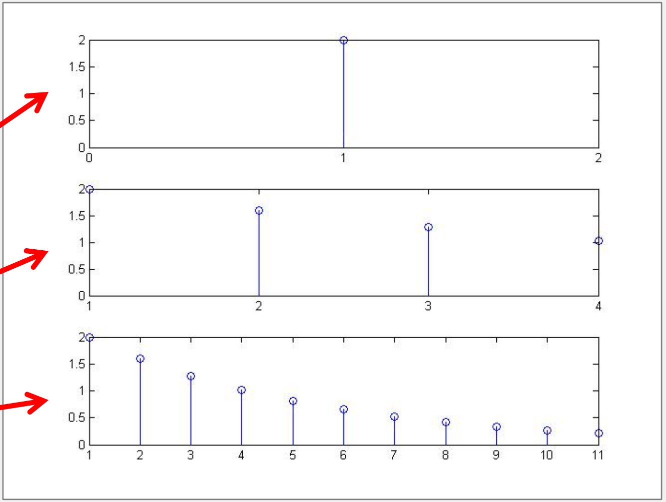
- **Causal** DT LTI system is uniquely specified by two vectors: $A=[a_0\ a_1\ a_2\ \ldots\ a_K]$ and $B=[b_0\ b_1\ \ldots\ b_M]$
  - A=[1] B=[0.5 1 2]
  - A=[1 -0.8] B=[2]

Exercise 2: Calculate Output Signal by filter()

- Syntax: y=filter(B, A, x)
- System: specify by the coefficient vector A and B
- x and y share the same range of time indices
  - Output signal y may be truncated

y[n]-0.8y[n-1]=2x[n]

```
A = [1 -0.8]
B = 2
X1 = [1]
Y1 = filter(B, A, X1);
X2 = [1 0 0 0]
Y2 = filter(B, A, X2);
X3 = [1 0 0 0 0 0 0 0 0 0 0]
Y3 = filter(B, A, X3);
subplot(3,1,1), stem(Y1)
subplot(3,1,2), stem(Y2)
subplot(3,1,3), stem(Y3)
```

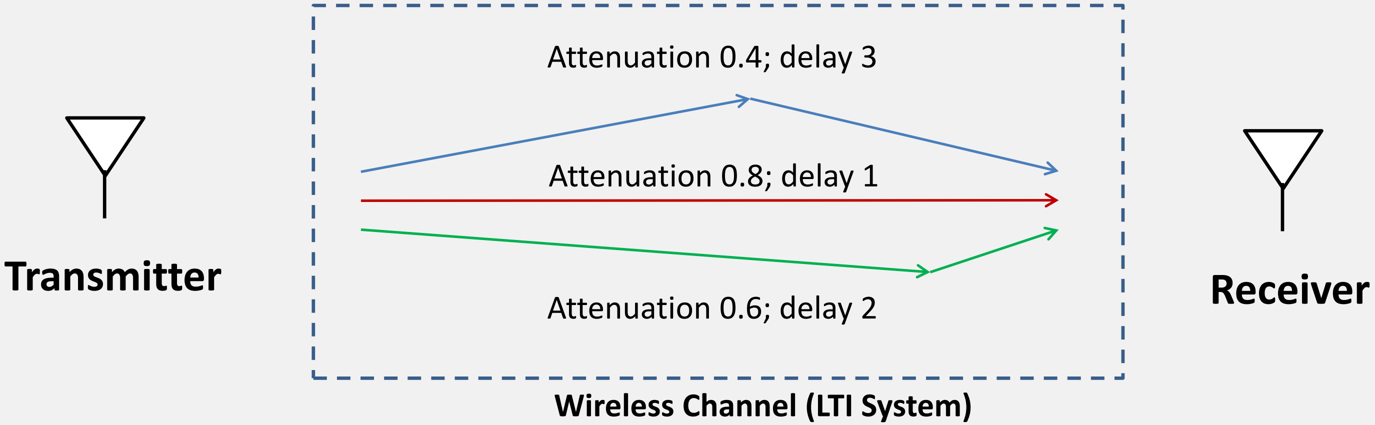


Summary

	conv	filter
Scenario	• Finite signal • Finite impulse response	• Finite signal • Infinite/finite impulse response
Remarks	• nw=nu(1)+nv(1):nu(end)+nv(end)	• x and y share the same range of time indices • Truncated output

- **Any question ?**

Application: Simplified Wireless Channel



Impulse response: $h1[n] = 0.8\delta[n-1] + 0.6\delta[n-2] + 0.4\delta[n-3]$

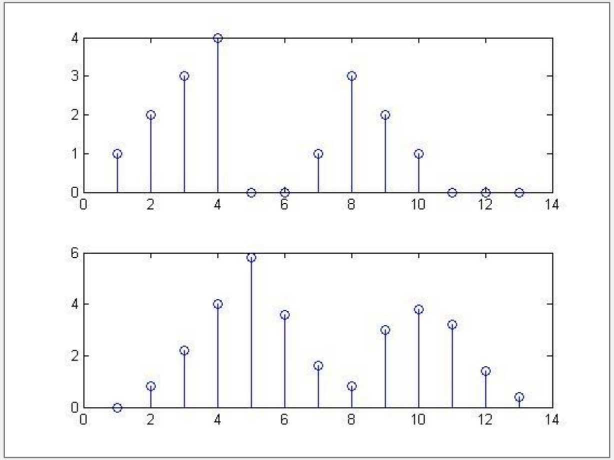
Difference equation: $y[n] = 0.8x[n-1] + 0.6x[n-2] + 0.4x[n-3]$

Generate Tx Signal:

```
x=[1 2 3 4 0 0 1 3 2 1 0 0 0];
```

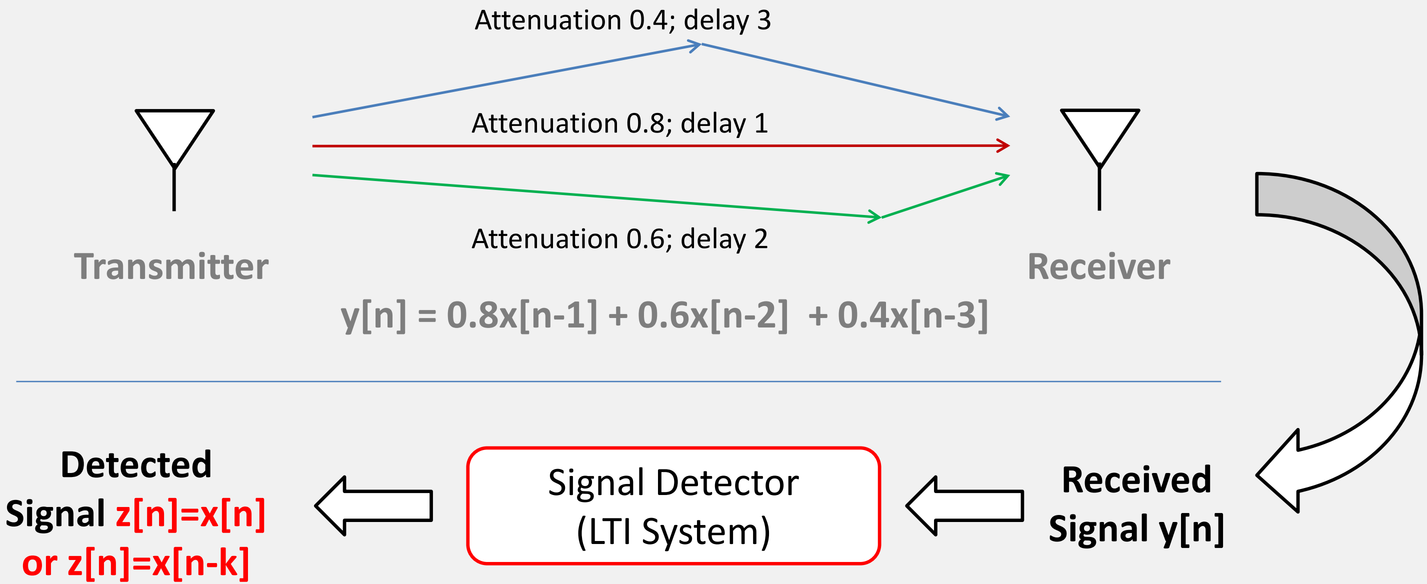
Generate Rx Signal:

```
A1 = 1;
B1 = [0 0.8:-0.2:0.4];
y = filter(B1, A1, x);
subplot(2,1,1), stem(x);
subplot(2,1,2), stem(y);
```



Signal detection:

How to recover the transmission signal from the received signal ?



Impulse response: $h2[n] * h1[n] = \delta[n]$ or $\delta[n-k]$

Difference Equation: $0.8z[n-1] + 0.6z[n-2] + 0.4z[n-3] = y[n]$

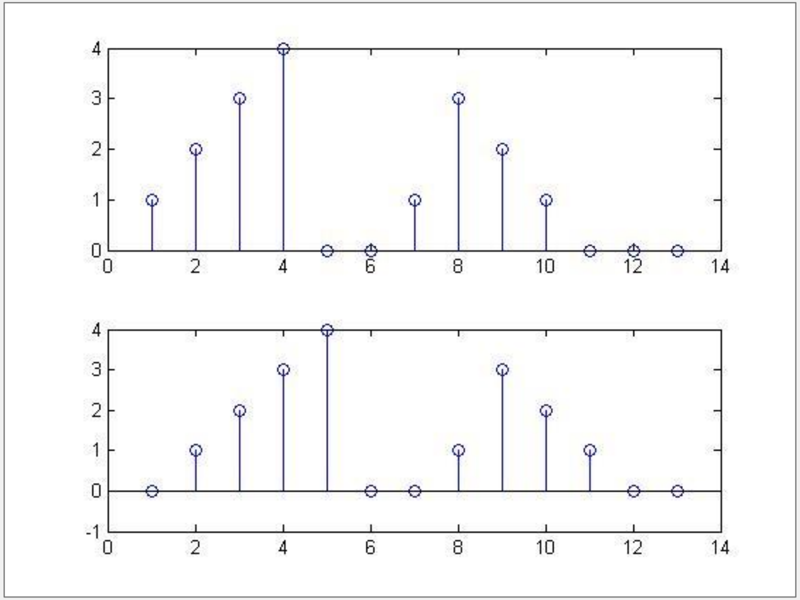
$$\boxed{z'[n] = z[n-1] \quad \Rightarrow \quad 0.8z'[n] + 0.6z'[n-1] + 0.4z'[n-2] = y[n]}$$

Generate Detected Signal:

```
A2 = 0.8:-0.2:0.4;
B2 = 1;
z = filter(B2, A2, y);
```

Compare the two signals:

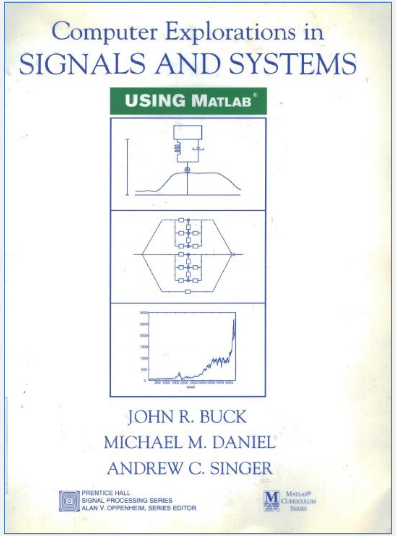
```
subplot(2,1,1), stem(x);
subplot(2,1,2), stem(z);
```



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Lab Assignment 2

- Read tutorial 2.1 & 2.2 & 2.3 by yourself
- 2.4 & 2.5 & 2.10
  - The sound file for 2.10: **lineup.mat**
- Submit your report 3.26 /3.27

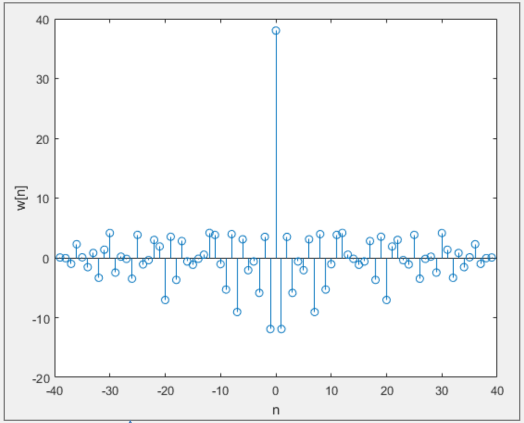


南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Signal Auto-Correlation

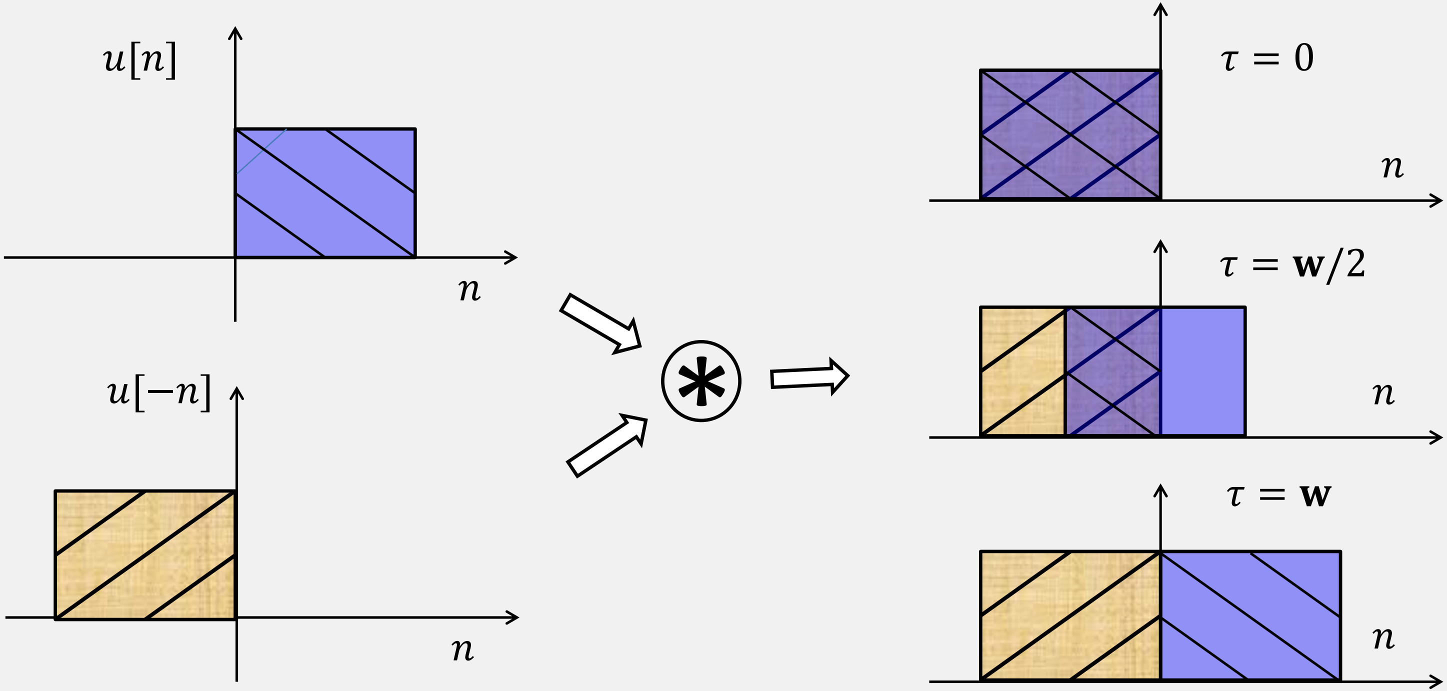
- Auto-correlation of u[n]: $w[n] = u[n] * u[-n]$

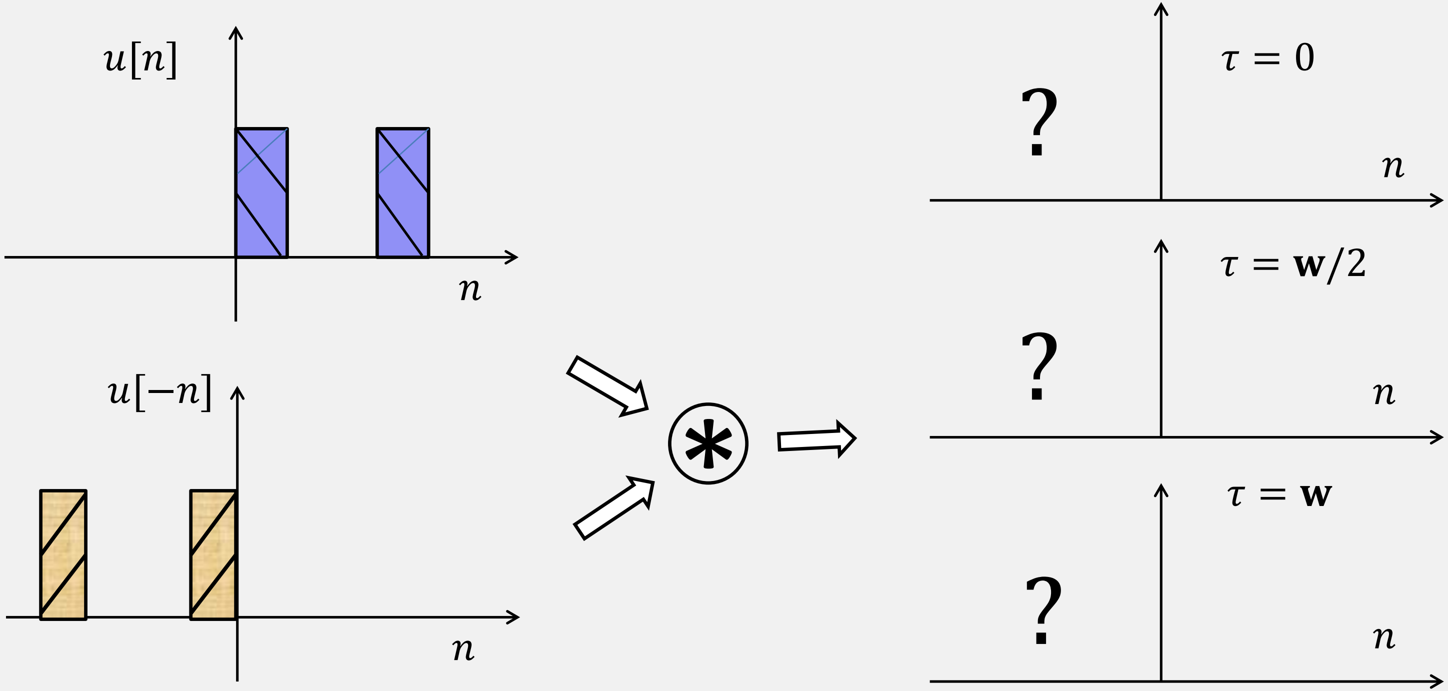
```
u=randn(1,40);
nu = 1:40;
v=u(end:-1:1);
nv=-40:-1;
w=conv(u,v);
nw=nu(1)+nv(1):nu(end)+nv(end);
stem(nw,w)
```



Auto-correlation of a random signal has a high peak at the origin

Understanding auto-correlation

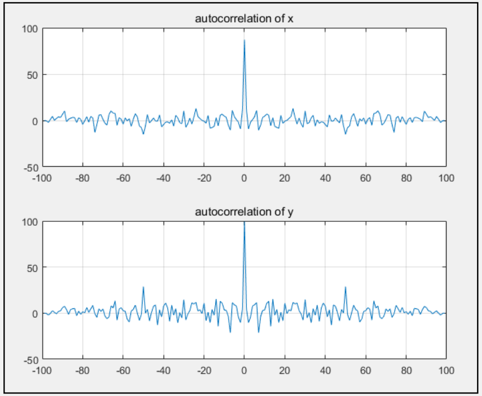




$u[n]$
n
$u[-n]$
n
$\circledast$
?
$\tau = 0$
n
?
$\tau = \mathbf{w}/2$
n
?
$\tau = \mathbf{w}$
n

Signal Auto-Correlation

- NX = 100;
- x = randn(1,NX);
- N =50;
- alpha = 0.9;
- yex = filter([1,zeros(1,N-1),alpha],1,x);
- Rxx = conv(x,fliplr(x));
- Ryy = conv(yex,fliplr(yex));
- figure;subplot(212);
- plot([-NX+1:NX-1],Ryy); grid on; title('autocorrelation of y');
- subplot(211);
- plot([-NX+1:NX-1],Rxx); grid on; title('autocorrelation of x');



Suppose that you were given $y[n]$ but did not know the value of the echo time, N, or the amplitude of the echo, α. Based on Eq. (2.21), can you determine a method of estimating these values? Hint: Consider the output $\mathbf{y}$ of the echo system to be of the form:

$$y[n] = x[n] * (\delta[n] + \alpha\delta[n-N])$$

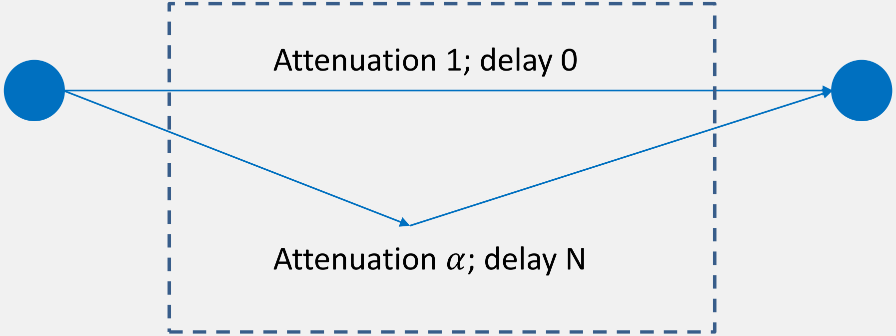
and consider the signal,

$$R_{yy}[n] = y[n] * y[-n].$$

This is called the autocorrelation of the signal $y[n]$ and is often used in applications of echo-time estimation. Write $R_{yy}[n]$ in terms of $R_{xx}[n]$ and also plot $R_{yy}[n]$. You will

$$R_{yy}[n] = x[n] * (\delta[n] + \alpha\delta[n-N]) \; * \; x[-n] * (\delta[n] + \alpha\delta[n+N])$$
$$= R_{xx} * ((1+\alpha^2)\delta[n] + \alpha\delta[n-N] + \alpha\delta[n+N])$$

$$y[n] = x[n] + \alpha x[n - N]$$



$x[n]$ is known:

$$\boxed{R_{yy}[n] = R_{xx} * ((1+\alpha^2)\delta[n] + \alpha\delta[n-N] + \alpha\delta[n+N])}$$

$$\boxed{y[n] = x[n] + \alpha x[n-N]}$$

$$\boxed{\hat{y}[n] = x[n] + \hat{\alpha} x[n-N]}$$

$$\boxed{\min\{y[n] - \hat{y}[n]\}}$$

- **Question ?**