南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

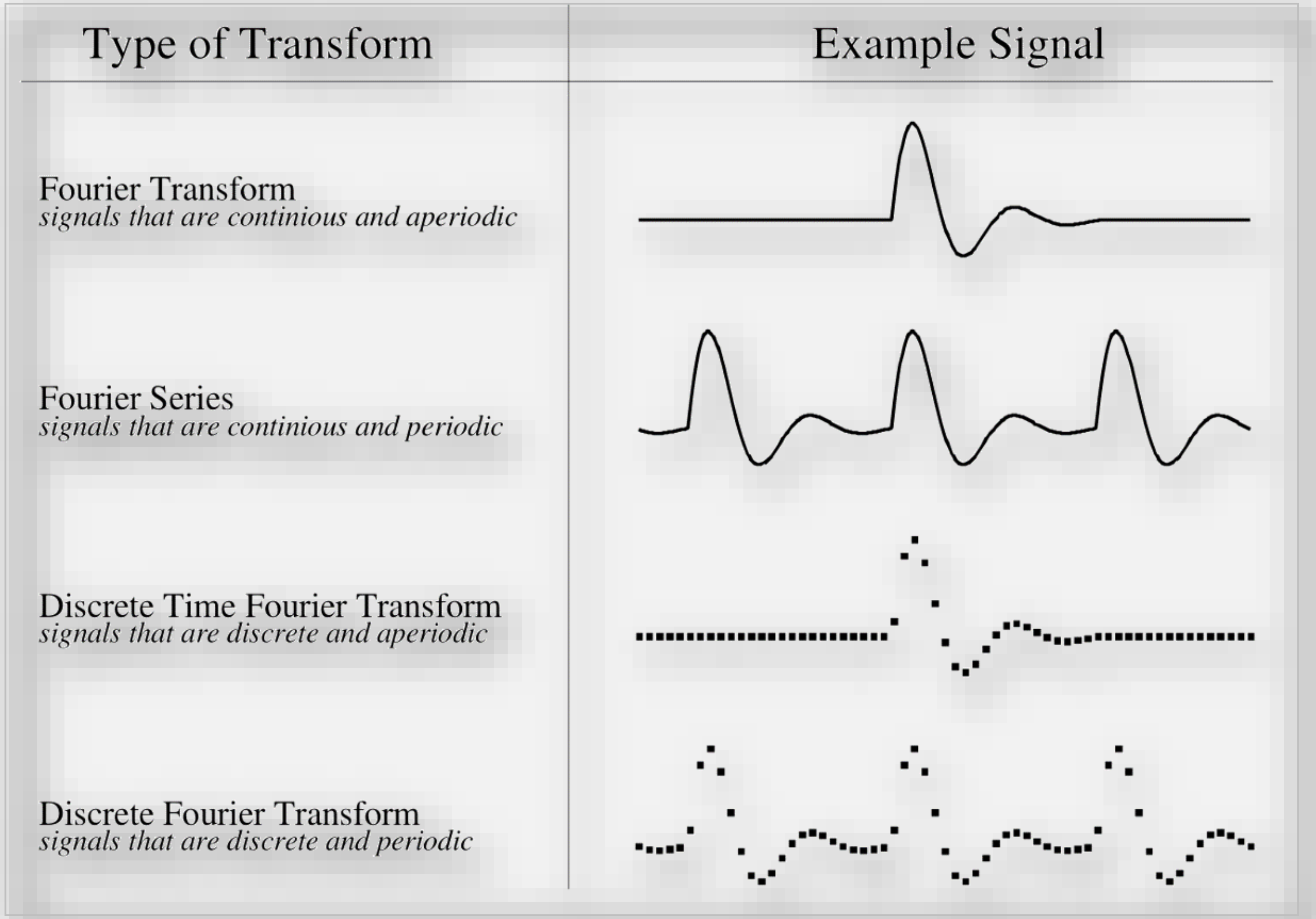
Signals and Systems (Lab)

Lab 3：Fourier Series Representation of Periodic Signals

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

Feedback

Baron Jean Baptiste Joseph Fourier



Baron Jean Baptiste Joseph Fourier
法国数学家、物理学家，1768-1830

Overview

- In Lab 3, you will
  - Verify the **frequency** property of convolution.
  - Verify the **frequency** property of LTI systems.

- In this tutorial, you will learn
  - How to calculate the output of DT LTI system in frequency domain.
  - How to calculate the output of CT LTI system.
  - How to calculate the DTFS of signal.

Eigenvalue and eigenvector

$$AX = \underbrace{\lambda}_{\text{eigenvalue}} \underbrace{X}_{\text{eigenvector}}$$

The eigenvector X has only scalar multiplication under the mapping A.

Complex Exponentials

$$x(t) = e^{st} \longrightarrow \boxed{h(t)} \xrightarrow{h(t) * e^{st}}$$

$$\begin{aligned} y(t) &= \int_{-\infty}^{+\infty} h(\tau) e^{s(t-\tau)} d\tau \\ &= \left[\int_{-\infty}^{+\infty} h(\tau) e^{-s\tau} d\tau \right] e^{st} \\ &= \underbrace{H(s)}_{\text{eigenvalue}} \underbrace{e^{st}}_{\text{eigenfunction}} \end{aligned}$$

$s = j\omega$ - purely imaginary,
i.e. signals of the form $e^{j\omega t}$

DT LTI System

$$e^{j\omega n} \rightarrow \boxed{\text{LTI System: H(.)}} \rightarrow H(e^{j\omega})e^{j\omega n}$$

$$a_0 e^{j0\left(\frac{2\pi}{N}\right)n} \qquad a_0 H\left(e^{j0\left(\frac{2\pi}{N}\right)}\right) e^{j0\left(\frac{2\pi}{N}\right)n}$$

$$a_1 e^{j1\left(\frac{2\pi}{N}\right)n} \qquad a_1 H\left(e^{j1\left(\frac{2\pi}{N}\right)}\right) e^{j1\left(\frac{2\pi}{N}\right)n}$$

$$a_2 e^{j2\left(\frac{2\pi}{N}\right)n} \Rightarrow \boxed{\text{LTI System: H(.)}} \Rightarrow a_2 H\left(e^{j2\left(\frac{2\pi}{N}\right)}\right) e^{j2\left(\frac{2\pi}{N}\right)n}$$

$$\ldots \qquad \ldots$$

$$a_{N-1} e^{j(N-1)\left(\frac{2\pi}{N}\right)n} \qquad a_{N-1} H\left(e^{j(N-1)\left(\frac{2\pi}{N}\right)}\right) e^{j(N-1)\left(\frac{2\pi}{N}\right)n}$$

- Can we calculate response of an LTI system in another way ?

In Lab 2, difference equation is like …

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

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

- Causal DT LTI system is uniquely specified by two vectors: $A=[a_0\ a_1\ a_2\ \dots\ a_K]$ and $B=[b_0\ b_1\ \dots\ b_M]$

We can use filter() to calculate h[n]

- ✓ 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]

Calculate Frequency Response

- In this Lab, we will use freqz() to calculate Frequency Response :

```
[H omega] = freqz(b, a, N);
```

$$H(e^{j\omega_k}) \quad \omega_k = \left(\frac{\pi}{N}\right)k, 0 \leq k \leq N-1$$

```
[H omega] = freqz(b, a, N, 'whole');
```

$$H(e^{j\omega_k}) \quad \omega_k = \left(\frac{2\pi}{N}\right)k, 0 \leq k \leq N-1$$

Exercise 1：Frequency response

- **Consider LTI System:**

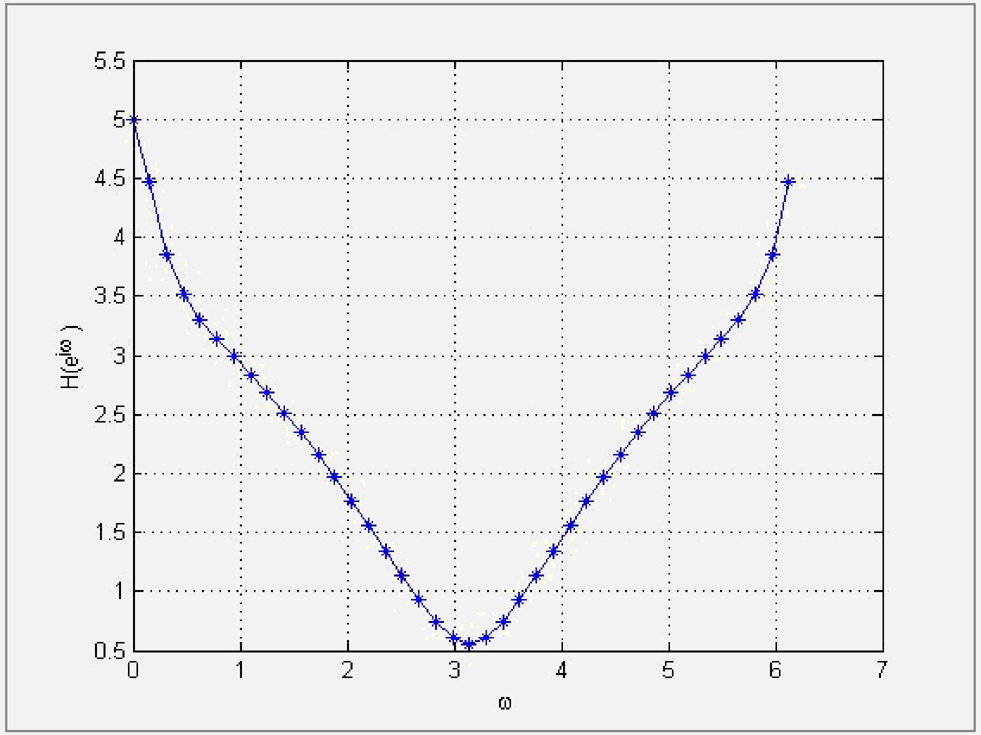
y[n]-0.8y[n-1]=2x[n]-x[n-2]

Define the vector of coefficients:

```
A=[1 -0.8];
B=[2 0 -1];
```

Plot the frequency response:

```
[H Omega] = freqz(B, A, 40, 'whole');
plot(Omega, abs(H), '*-');
xlabel('\omega');
ylabel('H(e^{j\omega} )');
grid;
```



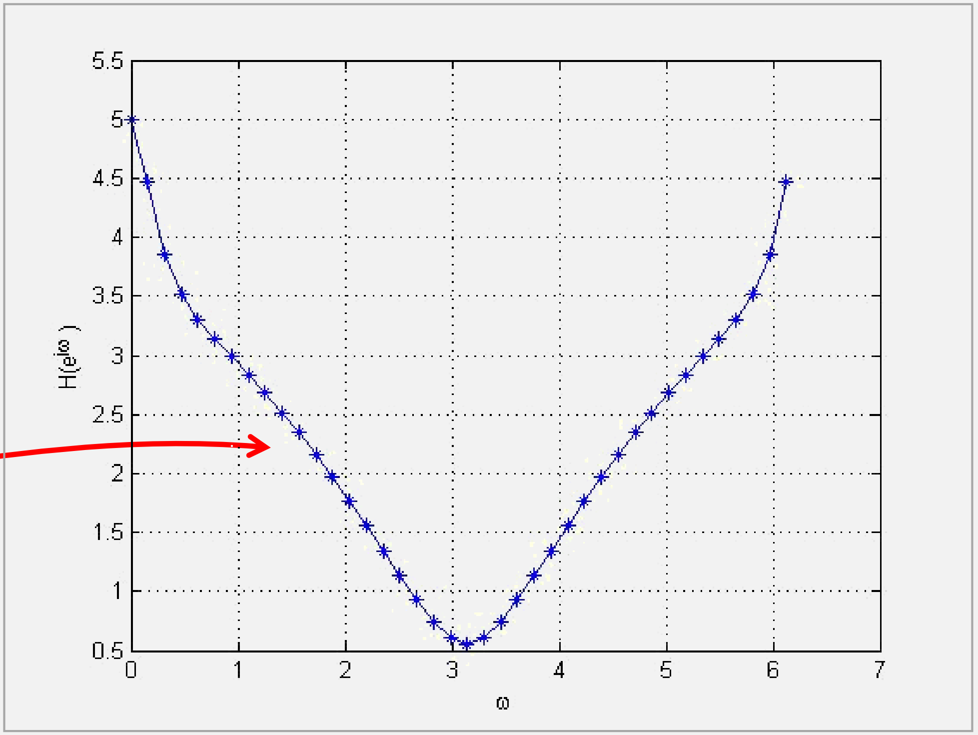
$$a_0 H(e^{j0\left(\frac{2\pi}{N}\right)})e^{j0\left(\frac{2\pi}{N}\right)n}$$

$$a_1 H(e^{j1\left(\frac{2\pi}{N}\right)})e^{j1\left(\frac{2\pi}{N}\right)n}$$

$$a_2 H(e^{j2\left(\frac{2\pi}{N}\right)})e^{j2\left(\frac{2\pi}{N}\right)n}$$

...

$$a_{N-1} \underline{H(e^{j(N-1)\left(\frac{2\pi}{N}\right)})}e^{j(N-1)\left(\frac{2\pi}{N}\right)n}$$



Overview

- In Lab 3, you will
  - Verify the **frequency** property of convolution.
  - Verify the **frequency** property of LTI systems.

- In this tutorial, you will learn
  - How to calculate the output of DT LTI system in frequency domain.
  - **How to calculate the output of CT LTI system.**
  - How to calculate the DTFS of signal.

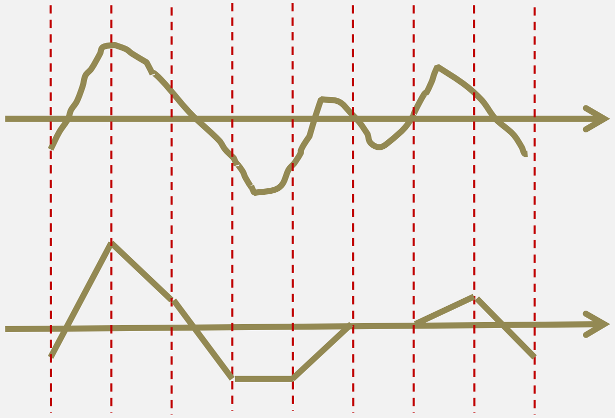
CT LTI System by Differential Equation

- Reading assignment: textbook 2.4.1.
- Causal CT LTI system can be specified by a linear constant-coefficient differential equation:

$$\boxed{\sum_{k=0}^{K} a_k \frac{d^k y(t)}{dt^k} = \sum_{m=0}^{M} b_m \frac{d^m x(t)}{dt^m}}$$

- Coefficient vectors:

$$\boxed{\begin{aligned} \boldsymbol{A} &= [\boldsymbol{a_K}\ \boldsymbol{a_{K-1}} \cdots \boldsymbol{a_0}] \\ \boldsymbol{B} &= [\boldsymbol{b_K}\ \boldsymbol{b_{K-1}} \cdots \boldsymbol{b_0}] \end{aligned}}$$



Attention !

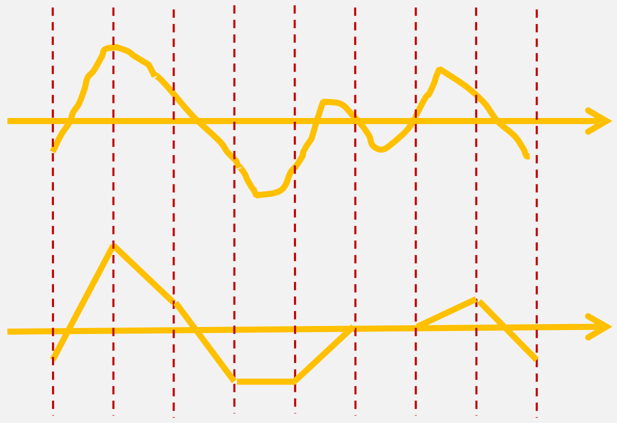
- CT LTI system by differential equation
  - $\sum_{k=0}^{K} a_k \frac{d^k y(t)}{dt^k} = \sum_{m=0}^{M} b_m \frac{d^m x(t)}{dt^m}$
  - $A = [a_K, a_{K-1}, \dots a_0]$
  - $B = [b_M, b_{M-1}, \dots b_0]$
- DT LTI system by difference equation
  - $\sum_{k=0}^{K} a_k y[n-k] = \sum_{m=0}^{M} b_m x[n-m]$
  - $A = [a_0, a_1, \dots a_K]$
  - $B = [b_0, b_1, \dots b_M]$

How to simulate CT systems ?

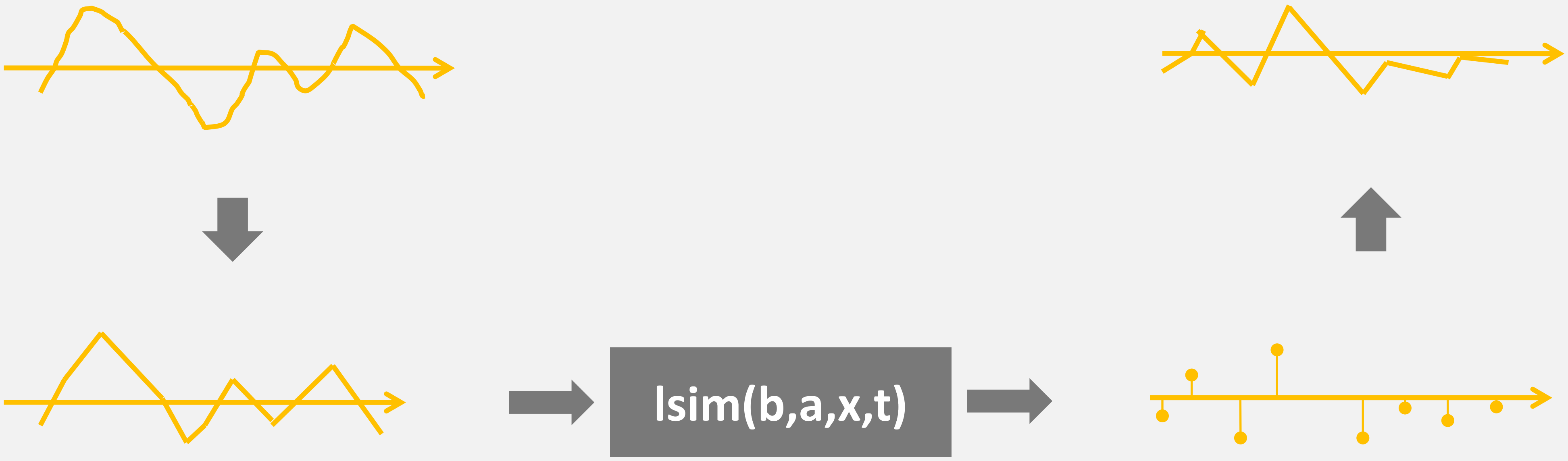
- How to simulate CT systems via Matlab?
- **lsim():** generate sampled output according to sampled input signal and CT system function
- **Syntax:** lsim(b,a,x,t)
- Sampled input signal

  Vector of sampling time: t

  Vector of sampled value: x



Simulation process



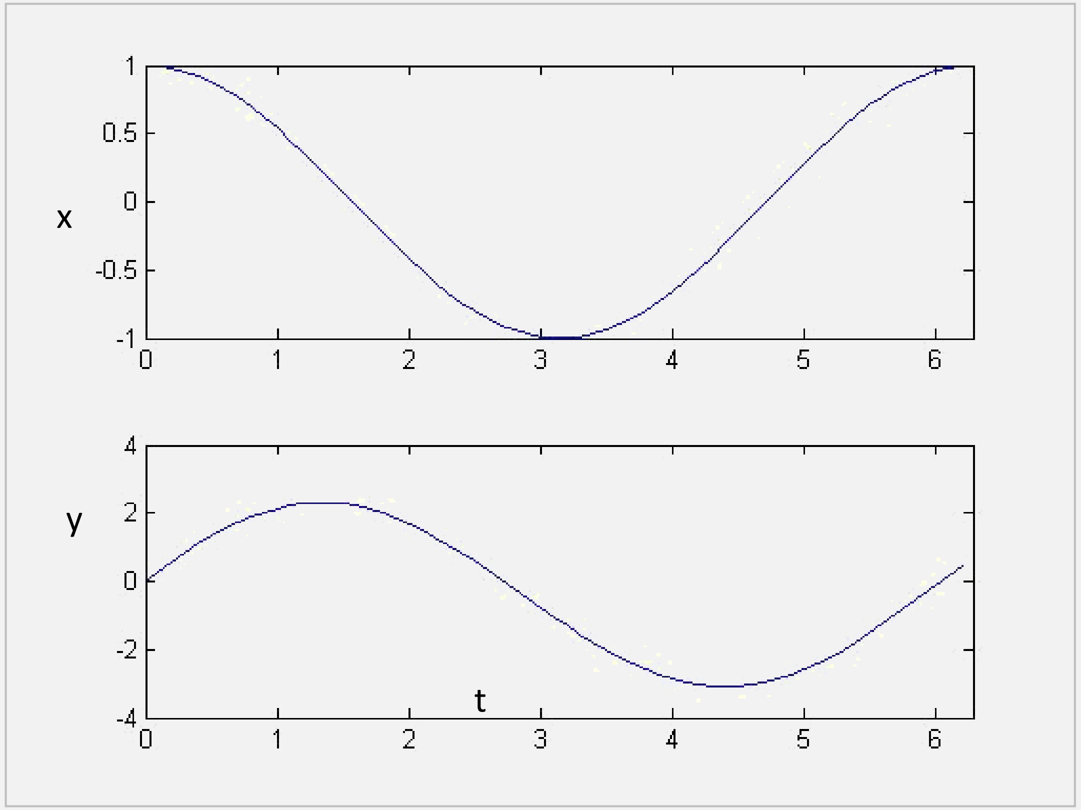
Exercise 2: CT System

- Consider LTI System: 0.3y(t)+dy(t)/dt = 3x(t)

```
A=[1 0.3];
B=3;
```

Sample the input signal x=cos(t):

```
t=0:0.1:2*pi;
x=cos(t);
y=lsim(B,A,x,t)';
subplot(2,1,1), plot(t,x);
xlim([0 2*pi]);
subplot(2,1,2), plot(t,y);
xlim([0 2*pi]);
```



Tips

- Differential equation

$$\sum_{k=0}^{K} a_k \frac{d^k y(t)}{dt^k} = \sum_{m=0}^{M} b_m \frac{d^m x(t)}{dt^m}$$

- System function

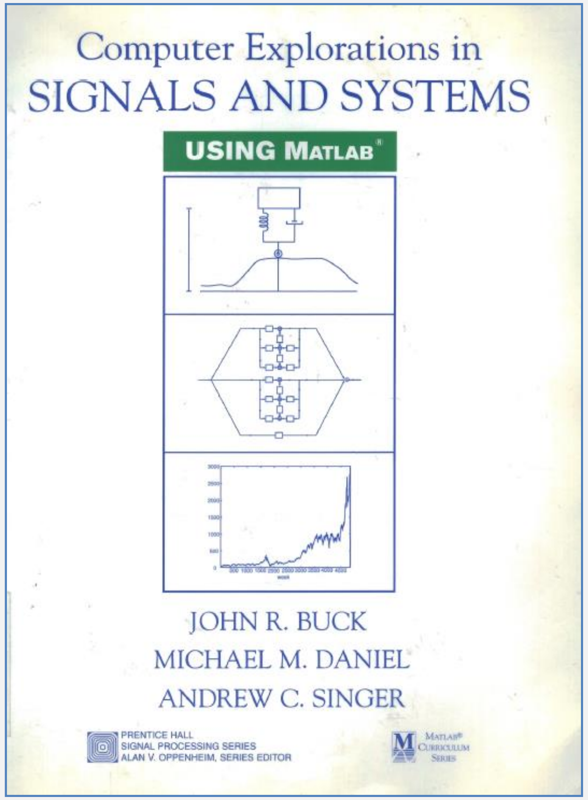
$$\Updownarrow$$

Tutorial 2.3 & 3.3

$$H(s) = \frac{\sum_{m=0}^{M} b_m s^m}{\sum_{k=0}^{K} a_k s^k}$$

Lab Assignment 3 (a)

- Read tutorial 3.2 & 3.3 by yourself
- 3.8 & 3.9
- Submit your report.



- 3.9(c)

Advanced Problems

(c). Analytically calculate the CTFS for the square wave x2. You may find it helpful to first find a relationship between the signal $x_2(t)$ and the signal $s(t)$ defined in Eq. (3.9). Use the ten lowest frequency nonzero CTFS coefficients of x2 to create the first 5 harmonic components individually. For example if you have the positive and

$$s(t) = \begin{cases} 1\,, & |t| < T/4\,, \\ 0\,, & T/4 \le |t| \le T/2 \end{cases} \tag{3.9}$$

CTFS coefficients a_k given by

$$a_k = \frac{\sin(\pi k/2)}{\pi k}.$$

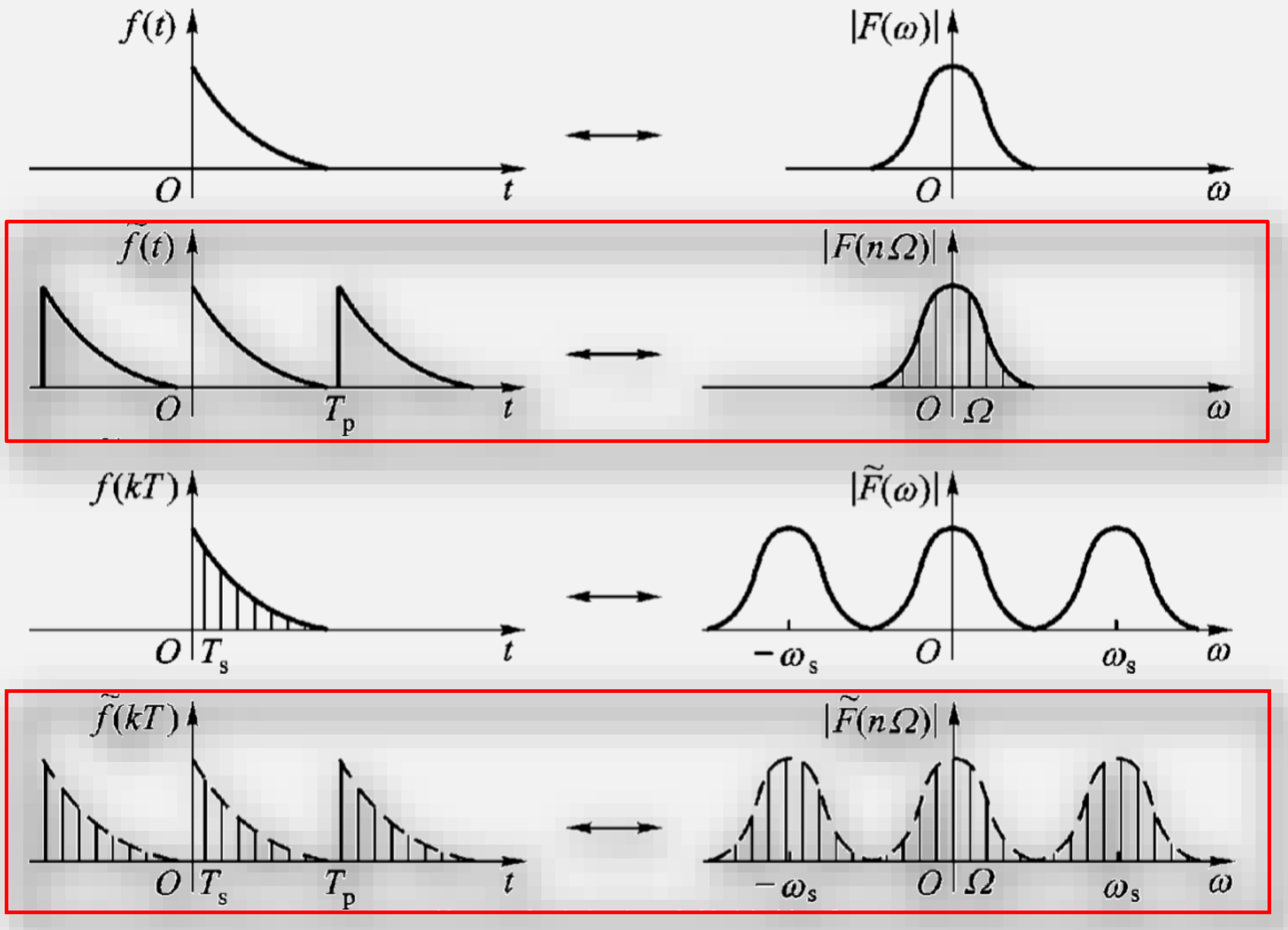
Overview

- In Lab 3, you will
  - Verify the **frequency** property of convolution.
  - Verify the **frequency** property of LTI systems.

- In this tutorial, you will learn
  - How to calculate the output of DT LTI system in frequency domain.
  - How to calculate the output of CT LTI system.
  - **How to calculate the DTFS of signal.**

FS

DTFS



Fourier Series

- Periodic signal with period T or N
- Synchesis equation:

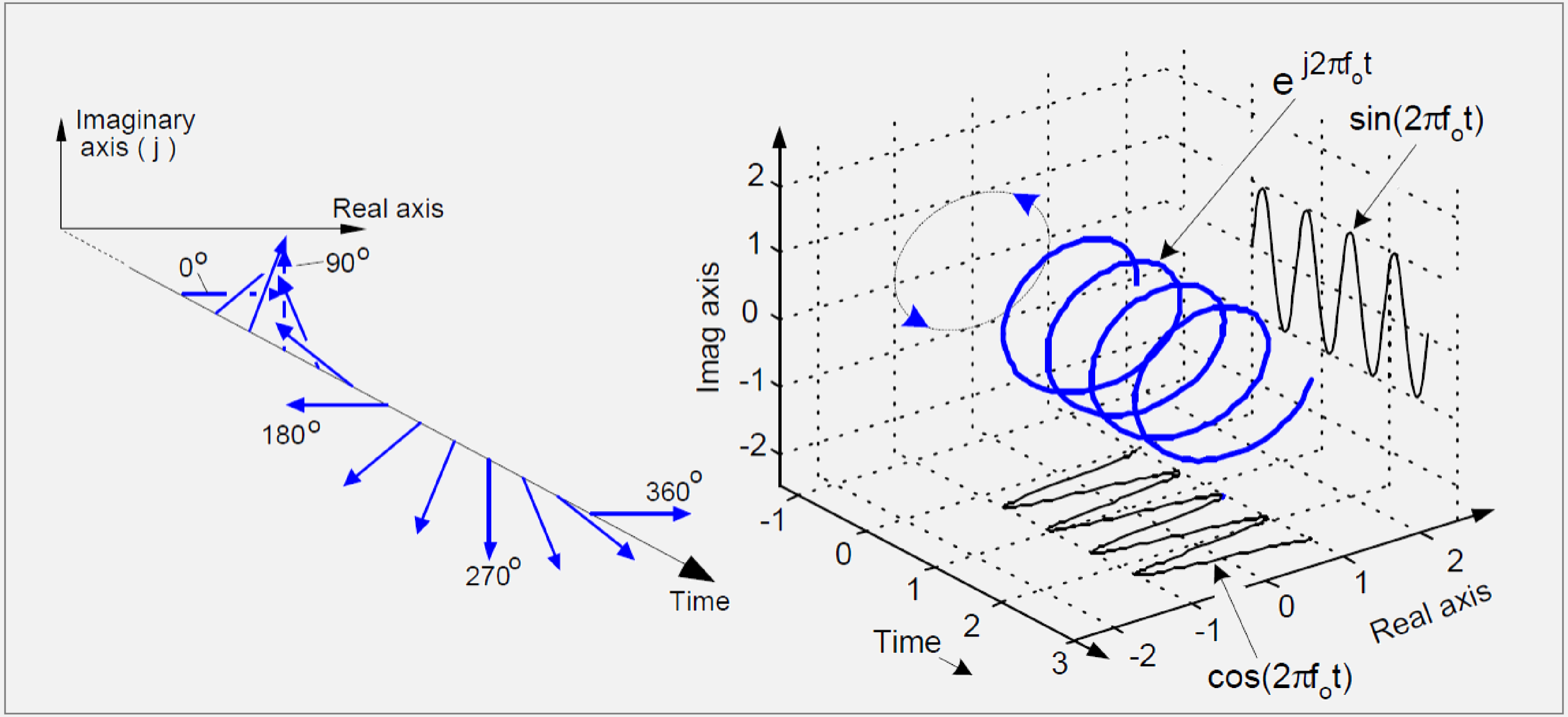
$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\left(\frac{2\pi}{T}\right)t} \quad \text{v.s.} \quad x[n] = \sum_{k=0}^{N-1} a_k e^{jk\left(\frac{2\pi}{N}\right)n}$$

  – Summation of ***N harmonic components***

- Analysis equation:

$$a_k = \frac{1}{T}\int_T x(t)e^{-jk\left(\frac{2\pi}{T}\right)t}dt \quad \text{v.s.} \quad a_k = \frac{1}{N}\sum_{n=0}^{N-1} x[n]e^{-jk\left(\frac{2\pi}{N}\right)n}$$

Understanding $e^{j2\pi f_0 t}$



Quadrature Signals: Complex, But Not Complicated--Richard Lyons

Matlab Function: fft()

- fft(): compute DTFS coefficients from signals

```
>> help fft
                 N
    X(k) =      sum  x(n)*exp(-j*2*pi*(k-1)*(n-1)/N), 1 <= k <= N.
                n=1
```

Compare with our definition:

$$a_k = \sum_{n=1}^{N} \mathrm{x[n]} e^{-j(k-1)\left(\frac{2\pi}{N}\right)(n-1)} \quad \textbf{v.s.} \quad a_k = \frac{1}{N}\sum_{n=0}^{N-1} \mathrm{x[n]} e^{-jk\left(\frac{2\pi}{N}\right)n}$$

Calculate the DTFS of vector x:

```
a = (1/N) * fft(x)
```

Matlab Function: ifft()

- ifft(): reconstruct signals from DTFS coefficients

```
>> help fft
                      N
    x(n) = (1/N) sum  X(k)*exp( j*2*pi*(k-1)*(n-1)/N), 1 <= n <= N.
                     k=1
```

Compare with our definition:

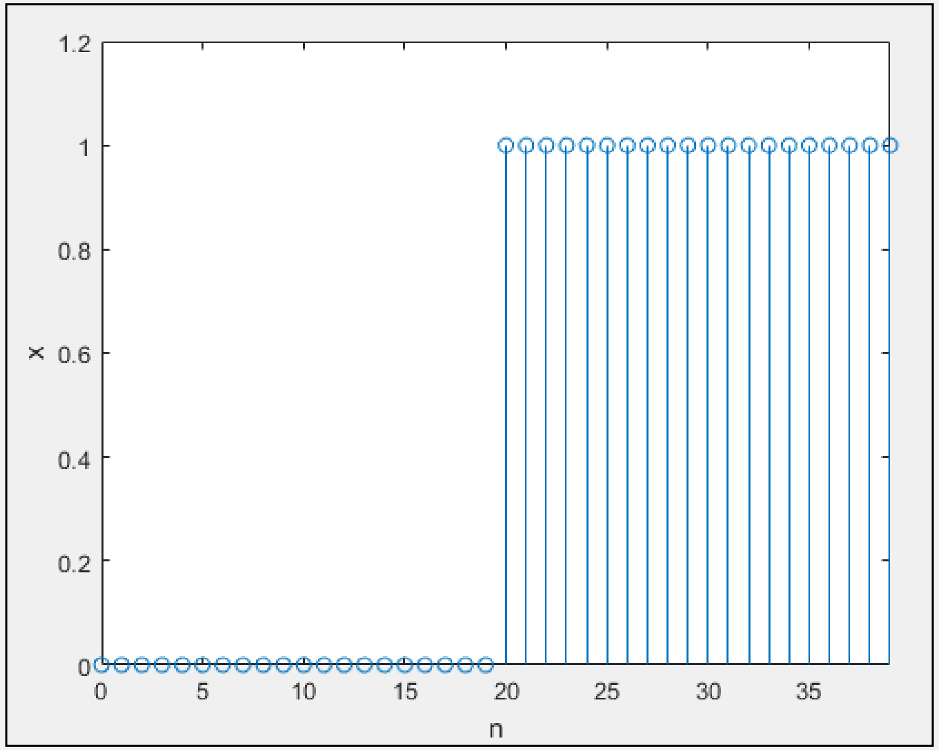
$$\mathbf{x[n]} = \frac{1}{N}\sum_{k=1}^{N} a_k e^{j(k-1)\left(\frac{2\pi}{N}\right)(n-1)} \quad \textbf{v.s.} \quad \mathbf{x[n]} = \sum_{k=0}^{N-1} a_k e^{jk\left(\frac{2\pi}{N}\right)n}$$

Calculate the DTFS of vector x: $\boxed{\text{x = N * ifft(a)}}$

Example

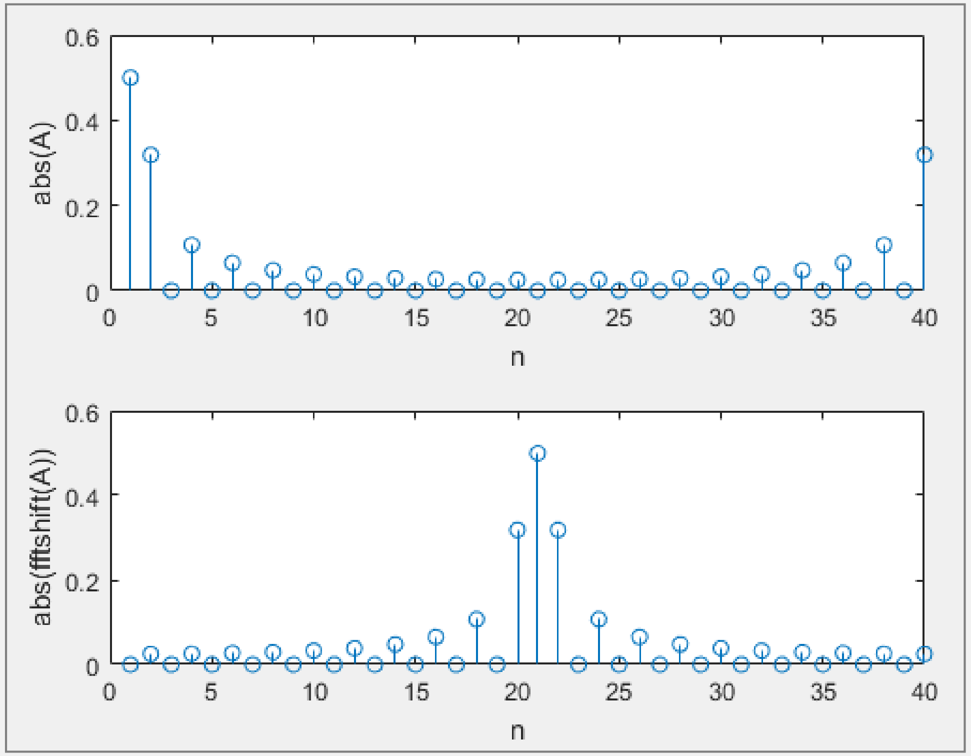
- Periodic DT rectangular wave with ***period = 40***

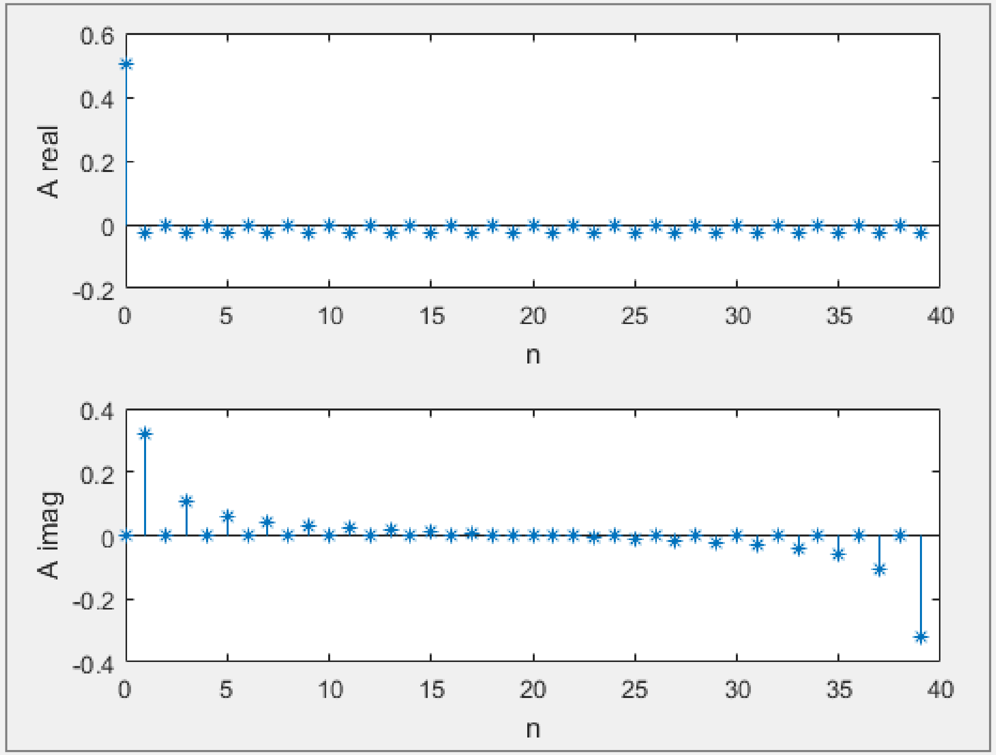
```
x=[zeros(1,20)
ones(1,20)];
stem(0:39, x);
xlim([0 39]);
ylim([0 1.2]);
```



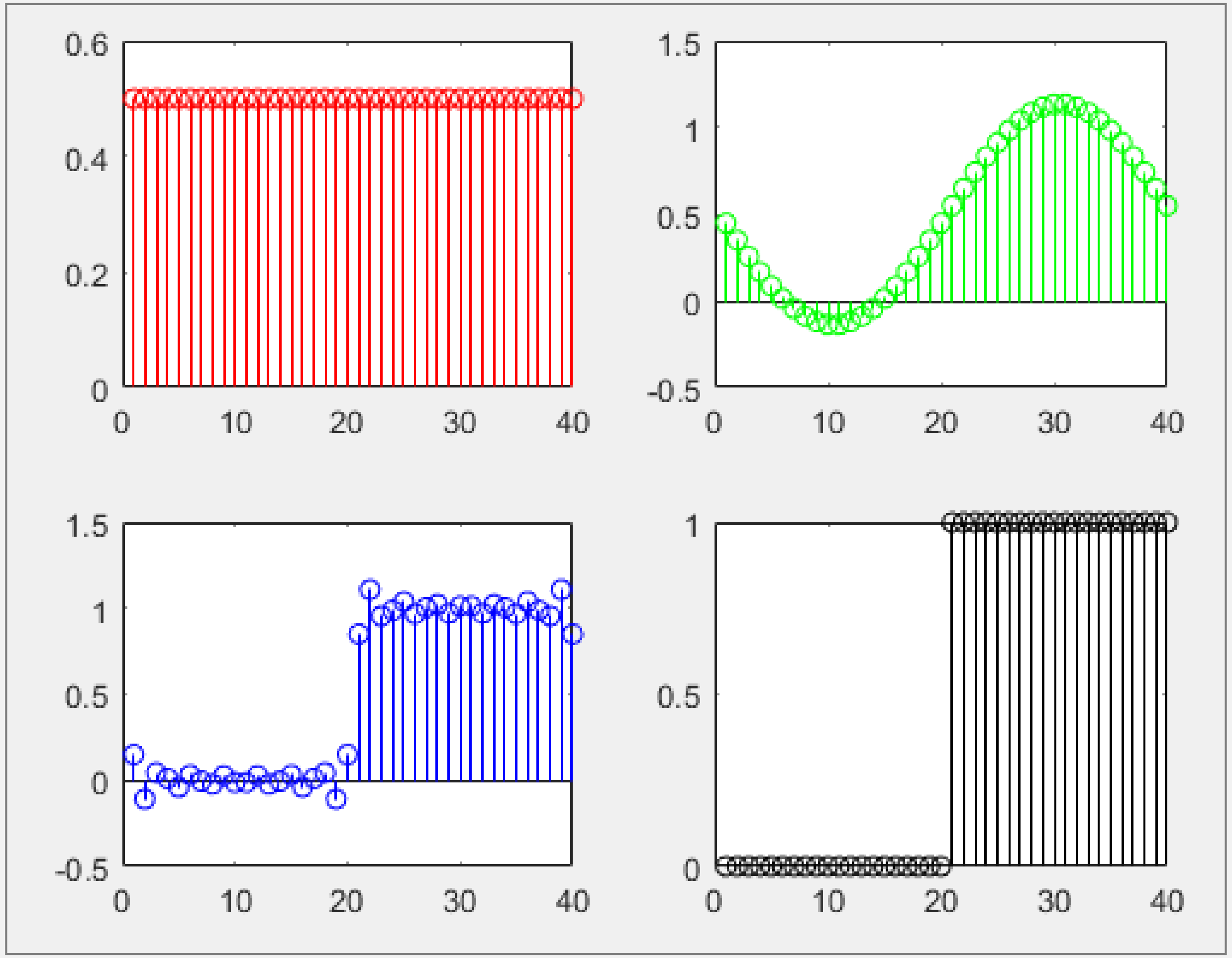
```
A = fft(x) / length(x);
figure(1)
subplot(2,1,1),stem(abs(A));
subplot(2,1,2),stem(abs(fftshift(A)));
```

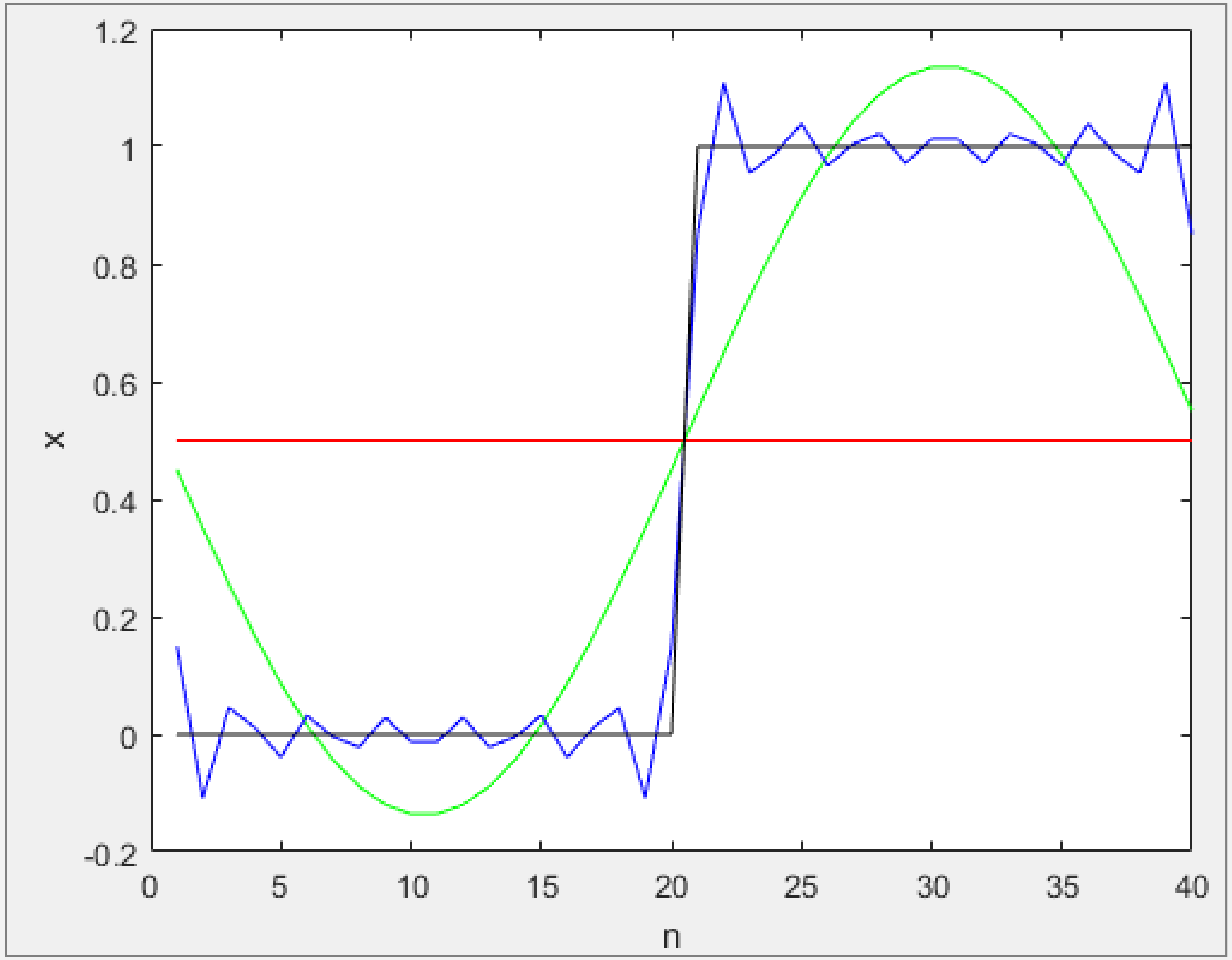
```
A = fft(x) / length(x);
figure(2)
subplot(2,1,1), stem(0: length(x)-1,real(A),'*-');
subplot(2,1,2), stem(0: length(x)-1,imag(A),'*-');
```





```
A1 = [A(1) zeros(1,39)];
A2 = [A(1) A(2) zeros(1,37) A(40)];
A3 = [A(1) A(2:15) zeros(1,11) A(27:40)];
subplot(2,2,1), stem(1:40,ifft(A1)*40, 'r');
subplot(2,2,2), stem(1:40,ifft(A2)*40, 'g');
subplot(2,2,3), stem(1:40,ifft(A3)*40, 'b');
subplot(2,2,4), stem(1:40,x, 'k');
```






```
plot(1:40,ifft(A1)*40, 'r', 1:40,ifft(A2)*40, 'g', 1:40,ifft(A3)*40, 'b', 1:40,x, 'k');
```

Complexity Analysis

- Suppose we know the matrix $E(n,k) = e^{-jk\left(\frac{2\pi}{N}\right)n}$

- How many multiplications & additions are needed to calculate Fourier series $[a_0, a_1, \ldots, a_{N-1}]$

$$a_0 = \frac{1}{N}\sum_{n=0}^{N-1} \mathrm{x[n]}E(n,0)$$

$$\ldots$$

$$a_{N-1} = \frac{1}{N}\sum_{n=0}^{N-1} \mathrm{x[n]}E(n,N-1)$$

Each: N+1 $\times$; N-1 $+$

Total: (N+1)N $\times$; N(N-1) $+$

Complexity Analysis

- Fast Fourier Transform:

  Calculation of Fourier series (transform) can be speeded up

  – Complexity reduces to O(NlogN)

N=4

$$a_0=(x[0]E(0,0)+x[1]E(1,0)+x[2]E(2,0)+x[3]E(3,0))/N$$

$$a_2=(x[0]E(0,2)+x[1]E(1,2)+x[2]E(2,2)+x[3]E(3,2))/N$$

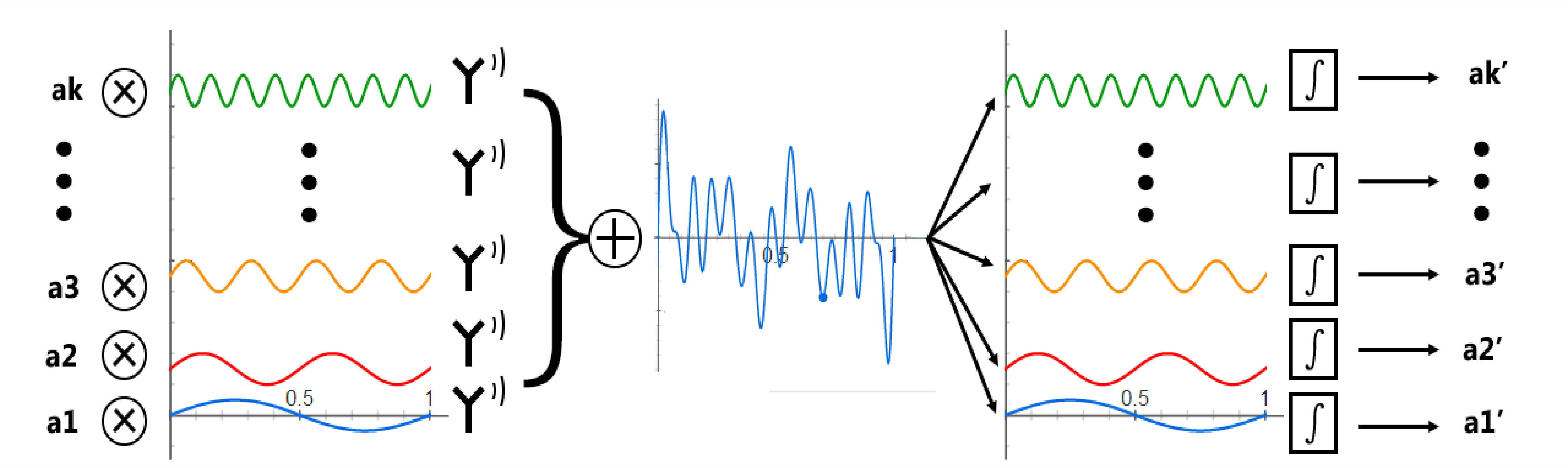
$$a_1=(x[0]E(0,1)+x[1]E(1,1)+x[2]E(2,1)+x[3]E(3,1))/N$$

$$a_3=(x[0]E(0,3)+x[1]E(1,3)+x[2]E(2,3)+x[3]E(3,3))/N$$

To calculate the DFT and FFT of a 1024*1024 image:

CPU	Clock Frequency	DFT	FFT
1941	60 Hz	152.3 y	271.4 d
1971 (4004)	108KHz	30.8 d	3.6 h
1978 (8086)	10MHz	8.0 h	2.3 min
1982 (80286)	20MHz	4.0 h	1.2min
1985 (80386)	33MHz	2.4h	42.6s
1989 (80486)	100MHz	48.0min	14.1s
1995 (Pentium)	200MHz	24.0min	7.0s
1999 (Pentium Ⅲ)	450MHz	10.7min	3.1s
2000 (Pentium 4)	1.4GHz	3.4min	1.0s
2001 (Pentium 4)	2GHz	2.4min	0.7s

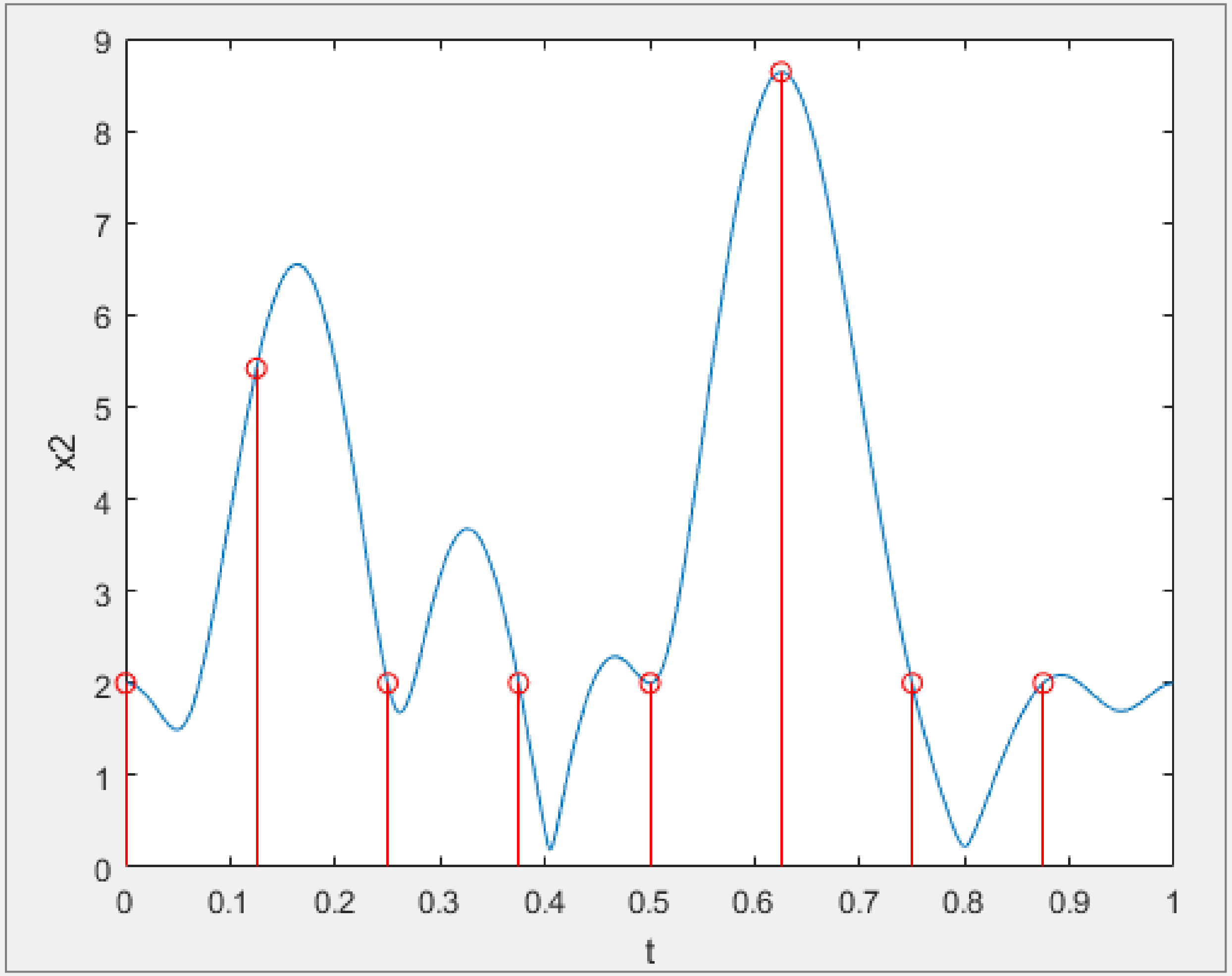
OFDM-Project



IFFT **FFT**

```
N=8;
x=randi([0 3],1,N);
x1=qammod(x,4);
f=1:N;
t=0:0.001:1-0.001;
w=2*pi*f'*t;
y1=x1*exp(j*w);
x2=ifft(x1,N);
plot(t,abs(y1));
hold on
stem(0:1/N:1-1/N,abs(x2)*N,'-r')
xlabel('t')
ylabel('x2')
x3=fft(x2)
```

help



Tips

- ***Periodic convolution*** in time domain is equivalent to multiplication in frequency domain

$$\boldsymbol{x[n]\otimes\hat{h}[n] = \sum_{r=0}^{N-1} x[r]\hat{h}[n-r] \Leftrightarrow \ N a_k h_k}$$

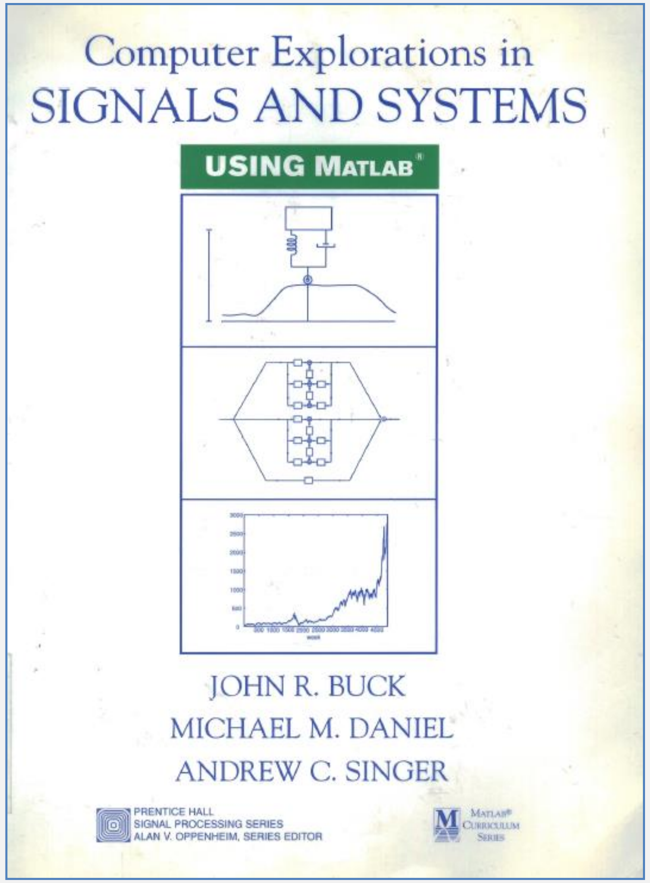
Table 3.2 of Textbook

$$\boldsymbol{y[n] = x[n] * h[n] \ = \ x[n]\otimes\hat{h}[n]}$$

$\hat{h}[n]$ is a periodic version of h[n]

Lab Assignment 3 (b)

- Read tutorial 3.2 & 3.3 by yourself
- 3.5 & 3.10
- Submit your report



- **Question ?**