Class 2

Part II: Frequency Response of DT LTI Systems

$$x[n] = \sum_k a_k {z_k}^n \longrightarrow \boxed{\begin{array}{c}\textit{DT LTI Systems}\\ (h[n])\end{array}} \longrightarrow y[n] = \sum_k a_k H(z_k) {z_k}^n$$

$$x[n] = z^n \longrightarrow \boxed{h[n]} \longrightarrow y[n] = \sum_{k=-\infty}^{+\infty} h[k] z^{n-k}$$

$$= z^n \sum_{k=-\infty}^{+\infty} h[k] z^{-k}$$

$$= \underbrace{H(z)}_{\text{eigenvalue}} \underbrace{z^n}_{\text{eigenfunction}}$$

$$x[n] = z^n \longrightarrow \boxed{h[n]} \longrightarrow y[n] = \sum_{k=-\infty}^{+\infty} h[k] z^{n-k}$$

$$= z^n \sum_{k=-\infty}^{+\infty} h[k] z^{-k}$$

$$= H(z) z^n$$

$$H(z) = \sum_{-\infty}^{+\infty} h[k] z^{-k}$$

when z is arbitrary complex number, H(z) is System Function.
when z=ejw， H(z) is H(ejw), that we named Frequency Response.

$$e^{j\omega n} \longrightarrow \boxed{\text{DT LTI Systems } (h(n))} \longrightarrow H(e^{j\omega}) e^{j\omega n}$$

$$x[n] \left\{ \begin{array}{l} a_0 e^{j0\left(\frac{2\pi}{N}\right)n} \\ a_1 e^{j1\left(\frac{2\pi}{N}\right)n} \\ a_2 e^{j2\left(\frac{2\pi}{N}\right)n} \\ \dots \\ a_{N-1} e^{j(N-1)\left(\frac{2\pi}{N}\right)n} \end{array} \right. \qquad \left. \begin{array}{r} a_0 H(e^{j0\left(\frac{2\pi}{N}\right)}) e^{j0\left(\frac{2\pi}{N}\right)n} \\ a_1 H(e^{j1\left(\frac{2\pi}{N}\right)}) e^{j1\left(\frac{2\pi}{N}\right)n} \\ a_2 H(e^{j2\left(\frac{2\pi}{N}\right)}) e^{j2\left(\frac{2\pi}{N}\right)n} \\ \dots \\ a_{N-1} H(e^{j(N-1)\left(\frac{2\pi}{N}\right)}) e^{j(N-1)\left(\frac{2\pi}{N}\right)n} \end{array} \right\} y[n]$$

Frequency Response of DT LTI Systems by Matlab

In Lab 2,

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

\+ Initial rest

Causal DT LTI System is uniquely specified by two coefficient vectors:

$A=[a_0,a_1,a_2\ldots,\ a_K]$,$B=[b_0,b_1,b_2\ldots,\ b_M]$.

Syntax 1: y=filter(B, A, x) ----> filters the data in vector X with the filter described by vectors A and B to create the filtered data Y.

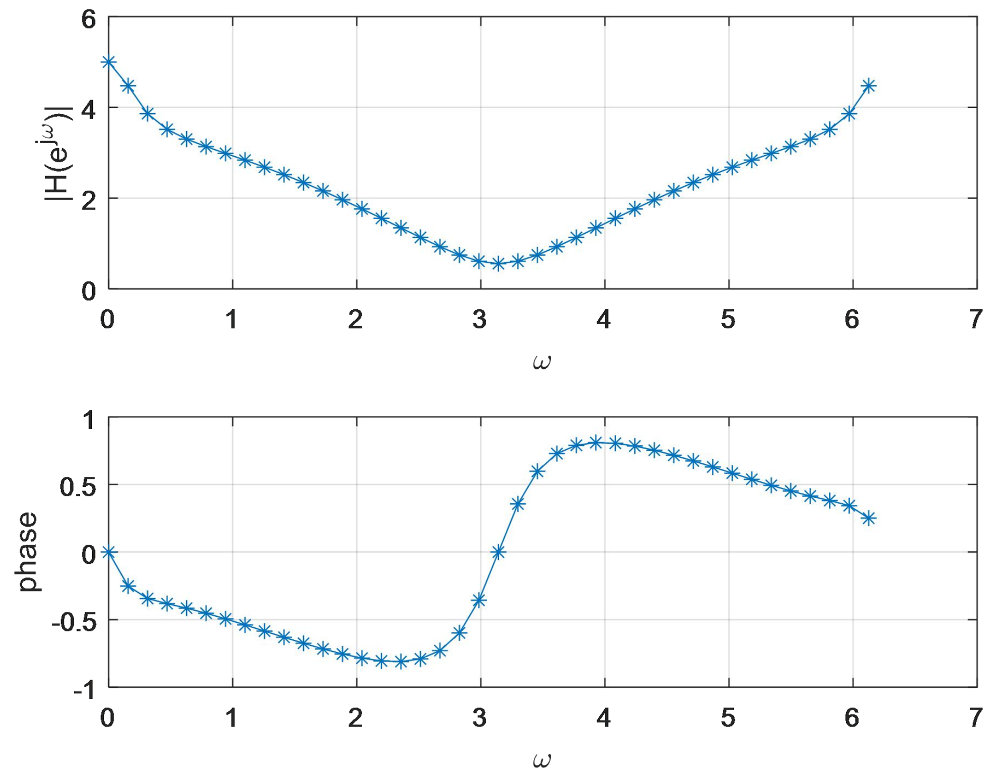
Syntax 2: [H,omega]=freqz(b,a,N,'whole')----> computes the samples of the frequency response at N evenly spaced frequencies from 0 to 2π, that is: $H(e^{j\omega_k}), \omega = (\frac{2\pi}{N})k\ ,\ k = 0, 1, 2 \ldots\ldots N-1$

[H omega] = freqz(b, a, N) ----> computes the samples of the frequency response at N evenly spaced frequencies from 0 to π, that is: $H(e^{j\omega_k}), \omega = (\frac{\pi}{N})k\ ,\ k = 0, 1, 2 \ldots\ldots N\text{-}1$

freqz

Causal DT LTI $y[n]-0.8y[n-1]=2x[n]-x[n-2]$

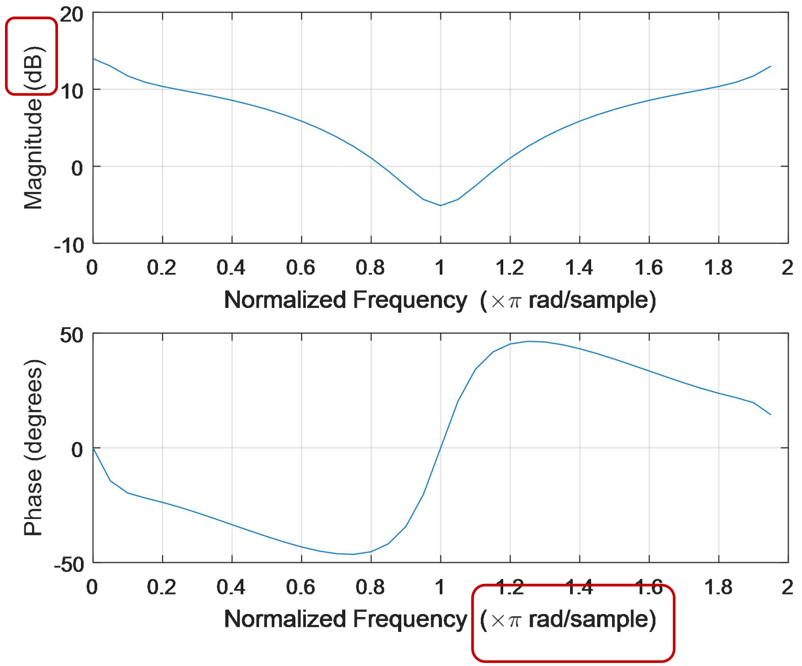
```
%define the vector of coefficient
a1=[1,-0.8];
b1=[2,0,-1];
%figure and plot the frequency response
[H omega]=freqz(b1,a1,40,'whole');
subplot(2,1,1), plot(omega,abs(H),'*-');
xlabel('\omega');ylabel('|H(e^{j\omega})|');
grid;
subplot(2,1,2),plot(omega,angle(H),‘*-’);
xlabel(‘\omega’);ylabel('phase');
grid;
```

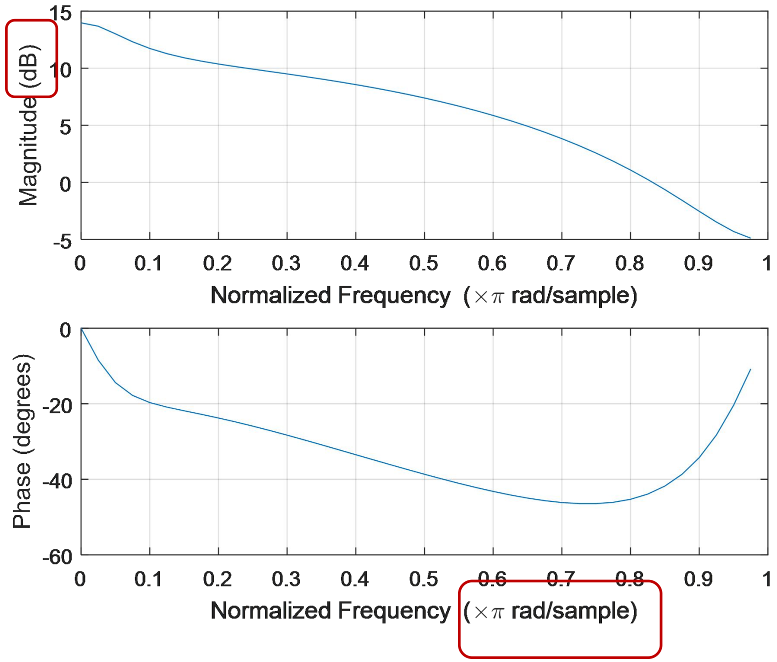


Greek Letters and Special Characters in Graph Text

```
A=[1 -0.8];
B=[2 0 -1];
figure;
freqz(B,A,40, 'whole')
```

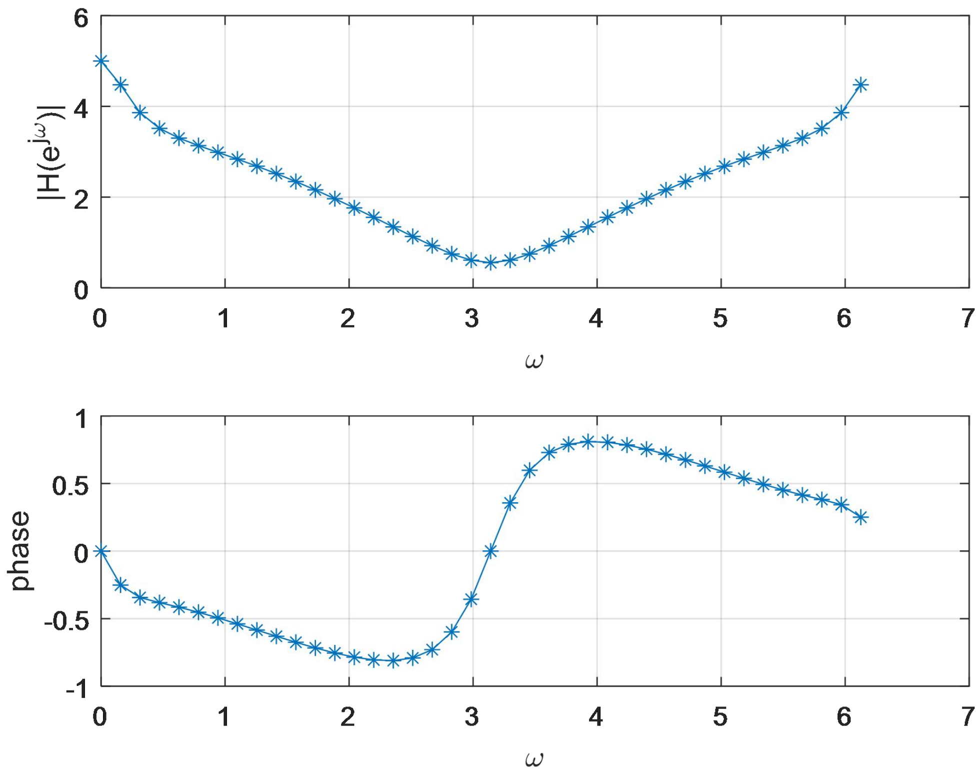
```
A=[1 -0.8];
B=[2 0 -1];
figure;
freqz(B,A,40)
```





[H,omega]=freqz(b,a,N,'whole') the meaning of its outputs:

Causal DT LTI $y[n]-0.8y[n-1]=2x[n]-x[n-2]$



For this DT LTI System, the output $H(e^{jwk})$ is just the eigenvalue to each eigenfunction e^{jwk}, also it is the frequency response.

$$x[n] = \sum_{k=0}^{N-1} e^{jw_k n}$$

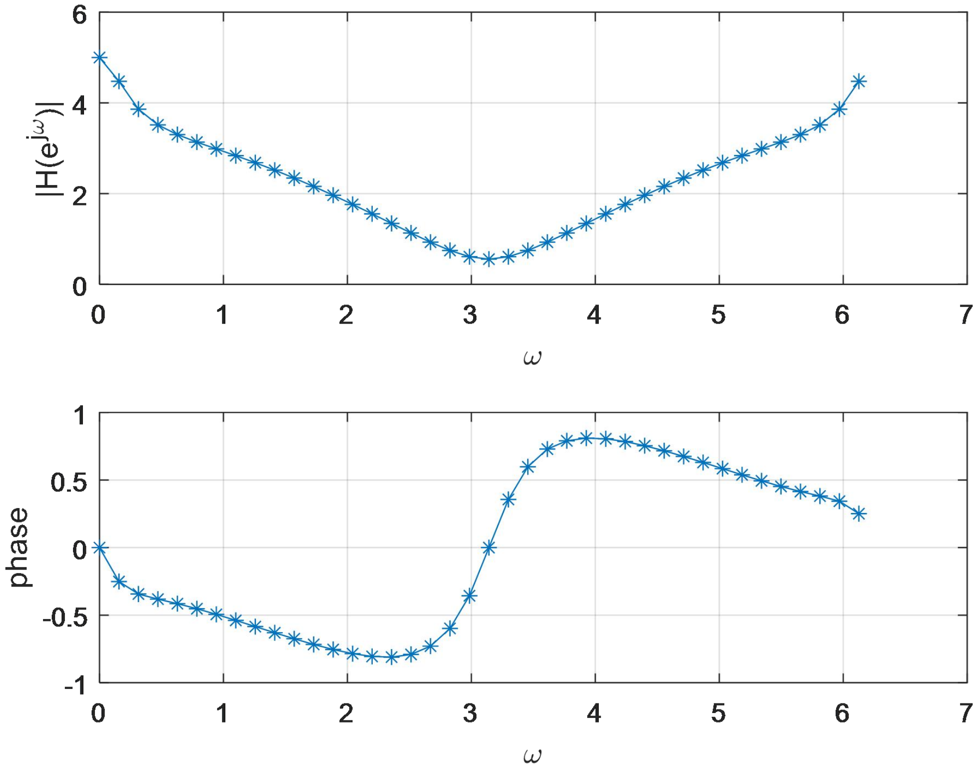
$$y[n] = \sum_{k=0}^{N-1} H(e^{jw_k})e^{jw_k n}$$

$$w_k = \frac{2\pi}{N}k$$

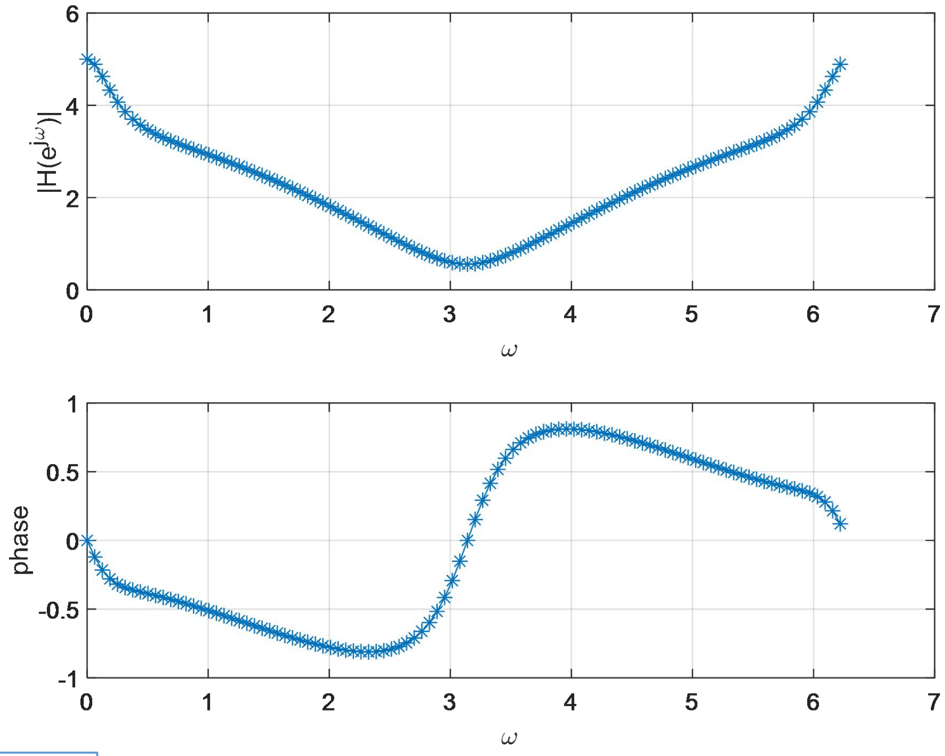
Similar to h[n]

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

Causal DT LTI y[n]-0.8y[n-1]=2x[n]-x[n-2]



N=40



N=100

$$x[n] = \sum_{k=0}^{N-1} e^{jw_k n}$$

$$y[n] = \sum_{k=0}^{N-1} H(e^{jw_k}) e^{jw_k n}$$

$$w_k = \frac{2\pi}{N} k$$

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

$$x[n] = \sum_{k=0}^{N-1} e^{jw_k n}$$

$$y[n] = \sum_{k=0}^{N-1} H(e^{jw_k}) e^{jw_k n}$$

$$w_k = \frac{2\pi}{N} k$$

$$x[n] = \sum_{k} a_k \, e^{jw_k n} \longrightarrow \text{DT LTI Systems } (h(n)) \longrightarrow y[n] = \sum_{k} a_k \, H(e^{jw_k n}) e^{jw_k n}$$

$$e^{j\omega n} \longrightarrow \text{DT LTI Systems } (h(n)) \longrightarrow H(e^{j\omega}) e^{j\omega n}$$

$$x[n] \; \left\{ \begin{array}{l} a_0 e^{j0\left(\frac{2\pi}{N}\right)n} \\ a_1 e^{j1\left(\frac{2\pi}{N}\right)n} \\ a_2 e^{j2\left(\frac{2\pi}{N}\right)n} \\ \dots \\ a_{N-1} e^{j(N-1)\left(\frac{2\pi}{N}\right)n} \end{array} \right.
\qquad
\left. \begin{array}{r} a_0 H\left(e^{j0\left(\frac{2\pi}{N}\right)}\right) e^{j0\left(\frac{2\pi}{N}\right)n} \\ a_1 H\left(e^{j1\left(\frac{2\pi}{N}\right)}\right) e^{j1\left(\frac{2\pi}{N}\right)n} \\ a_2 H\left(e^{j2\left(\frac{2\pi}{N}\right)}\right) e^{j2\left(\frac{2\pi}{N}\right)n} \\ \dots \\ a_{N-1} H\left(e^{j(N-1)\left(\frac{2\pi}{N}\right)}\right) e^{j(N-1)\left(\frac{2\pi}{N}\right)n} \end{array} \right\} \; y[n]$$

■ 3.2 Tutorial: `freqz`

The signals $e^{j\omega n}$ are eigenfunctions of LTI systems. For each value of ω the frequency response $H(e^{j\omega})$ is the eigenvalue of the LTI system for the eigenfunction $e^{j\omega n}$; when the input sequence is $x[n] = e^{j\omega_0 n}$, the output sequence is $y[n] = H(e^{j\omega_0})e^{j\omega_0 n}$. For a causal LTI system described by a difference equation, the command `[H omega]=freqz(b,a,N)` computes the frequency response $H(e^{j\omega})$ at N evenly spaced frequencies between 0 and π, i.e., $\omega_k = (\pi/N)k$ for $0 \le k \le N-1$. The coefficient vectors `a` and `b` specify the difference equation using the same format as Eq. (2.8) in the `filter` tutorial. For the command above, `freqz` returns $H(e^{j\omega_k})$ in `H` and the frequencies ω_k in `omega`. Including the `'whole'` option as `[H omega]=freqz(b,a,N,'whole')` computes the samples of the frequency response $H(e^{j\omega})$ at N evenly spaced frequencies from 0 to 2π, $\omega_k = (2\pi/N)k$ for $0 \le k \le N-1$.

(a). Define `a1` and `b1` to describe the causal LTI system specified by the difference equation $y[n] - 0.8y[n-1] = 2x[n] - x[n-2]$.

(b). Use `freqz` with the coefficients from Part (a) to define `H1` to be the value of the frequency response at 4 evenly spaced frequencies between 0 and π and `omega1` to be those frequencies. The following sample output shows the values each vector should have if you have defined things correctly:

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

3.8 First-Order Recursive Discrete-Time Filters

This exercise demonstrates the effect of first-order recursive discrete-time filters on periodic signals. You will examine the frequency responses of two different systems and also construct a periodic signal to use as input for these systems. This exercise assumes you are familiar with using `fft` and `ifft` to compute the DTFS of a periodic signal as described in Tutorial 3.1. In addition, it is also assumed you are proficient with the `filter` and `freqz` commands described in Tutorials 2.2 and 3.2. Several parts of this exercise require you to generate vectors which should be purely real, but have very small imaginary parts due to roundoff errors. Use `real` to remove these residual imaginary parts from these vectors.

This exercise focuses on two causal LTI systems described by first-order recursive difference equations:

$$\begin{aligned} \text{System 1:} \quad y[n] - 0.8y[n-1] &= x[n], \\ \text{System 2:} \quad y[n] + 0.8y[n-1] &= x[n]. \end{aligned}$$

The input signal $x[n]$ will be the periodic signal with period $N = 20$ described by the DTFS coefficients

$$a_k = \begin{cases} 3/4\,, & k = \pm 1\,, \\ -1/2\,, & k = \pm 9\,, \\ 0\,, & \text{otherwise}\,. \end{cases} \tag{3.10}$$

Part III: Frequency Response of CT LTI Systems

$$x(t)=\sum_k a_k e^{st} \longrightarrow \boxed{\begin{matrix}\textit{CT LTI Systems}\\ (h(t))\end{matrix}} \longrightarrow y(t)=\sum_k a_k H(s_k)e^{st}$$

$$x(t)=e^{st} \longrightarrow \boxed{h(t)} \longrightarrow y(t)=\int_{-\infty}^{+\infty} h(\tau)e^{s(t-\tau)}d\tau$$

$$=\left[\int_{-\infty}^{+\infty} h(\tau)e^{-s\tau}d\tau\right]e^{st}$$

$$=\underbrace{H(s)}_{\textbf{eigenvalue}}\underbrace{e^{st}}_{\textbf{eigenfunction}}$$

Complex exponentials are **eigenfunctions** of LTI systems, i.e., when the input equence is a complex exponential, the output is the same complex exponential only scaled in amplitude by a complex constant. This constant can be computed from the impulse response h(t).

$$\boxed{H(s) = \int_{-\infty}^{+\infty} h(\tau)\, e^{-s\tau} d\tau}$$

$$x(t) = e^{st} \longrightarrow \boxed{h(t)} \longrightarrow y(t) = \int_{-\infty}^{+\infty} h(\tau) e^{s(t-\tau)} d\tau$$

$$= \left[\int_{-\infty}^{+\infty} h(\tau) e^{-s\tau} d\tau\right] e^{st}$$

$$= H(s) e^{st}$$

$$H(s) = \int_{-\infty}^{+\infty} h(\tau)\, e^{-s\tau} d\tau$$

when s is arbitrary complex number, H(s) is System Function.
when s=jw， H(s) is H(jw), that we named Frequency Response.

$$e^{jwt} \qquad\qquad H(jw)e^{jwt}$$

$$x(t) \left\{ \begin{array}{l} a_{-\infty} e^{j-\infty(\frac{2\pi}{T})t} \\ \cdot\ \cdot\ \cdot \\ a_{-2} e^{j-2(\frac{2\pi}{T})t} \\ a_{-1} e^{j-1(\frac{2\pi}{T})t} \\ a_0 e^{j0(\frac{2\pi}{T})t} \\ a_1 e^{j1(\frac{2\pi}{T})t} \\ a_2 e^{j2(\frac{2\pi}{T})t} \\ \cdot\ \cdot\ \cdot \\ a_{\infty} e^{j\infty(\frac{2\pi}{T})t} \end{array} \right. \qquad \left. \begin{array}{l} a_{-\infty} H(j-\infty(\frac{2\pi}{T})) e^{j-\infty(\frac{2\pi}{T})t} \\ \cdot\ \cdot\ \cdot \\ a_{-2} H(j-2(\frac{2\pi}{T})) e^{j-2(\frac{2\pi}{T})t} \\ a_{-1} H(j-1(\frac{2\pi}{T})) e^{j-1(\frac{2\pi}{T})t} \\ a_0 H(j0(\frac{2\pi}{T})) e^{j0(\frac{2\pi}{T})t} \\ a_1 H(j1(\frac{2\pi}{T})) e^{j1(\frac{2\pi}{T})t} \\ a_2 H(j2(\frac{2\pi}{T})) e^{j2(\frac{2\pi}{T})t} \\ \cdot\ \cdot\ \cdot \\ a_{\infty} H(j\infty(\frac{2\pi}{T})) e^{j\infty(\frac{2\pi}{T})t} \end{array} \right\} y(t)$$

Frequency Response of CT LTI Systems by Matlab

Tutorial ：Textook2.4.1

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

Causal CT LTI System Described by Differential Equations is uniquely specified by two coefficient vectors:

$A=[a_K, a_{k-1},\dots a_1,a_0]$,$B=[b_M , b_{M-1}\dots b_1,b_0]$.

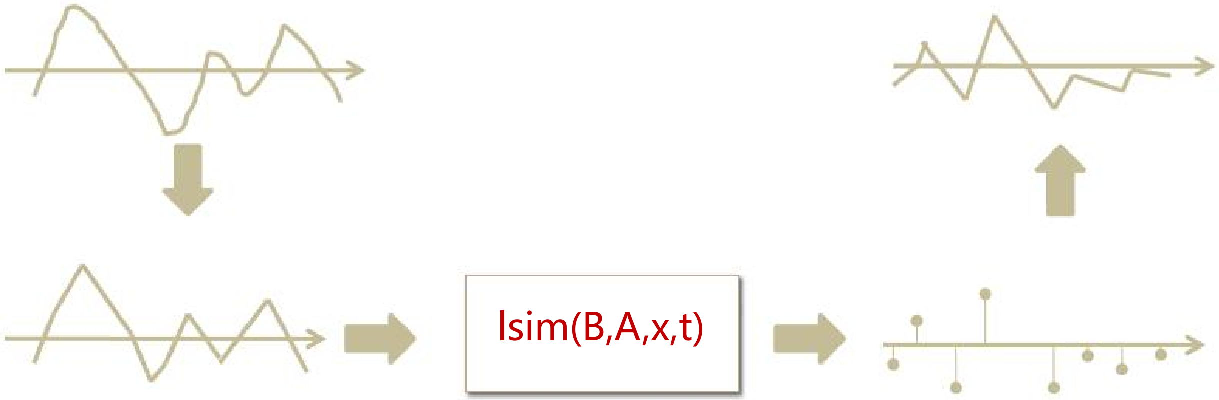
Syntax 1: y=lsim(B,A,x,t); or y=lsim(sys,x,t); sys=tf(B,A); simulates the response of the system to the input signal specified by the vectors x and t. The vector t contains the time samples for the input and output, x contains the values of the input x(t) at each time in t, and y contains the simulated values of the output y(t) at each time in t.

Lab 2.3

Syntax 2: H=freqs(b,a); or H=freqs(b,a,w); computes the frequency response of this system. Detailed content will be introduced in the next chapter;

Coefficient vectors : **A=[a_K, a_{k-1},… a_1,a_0] ,B=[b_M , b_{M-1}… b_1,b_0]**.

Syntax : y=lsim(B,A,x,t) or y=lsim(sys,x,t); sys=tf(B,A); simulates the response of the system to the input signal specified by the vectors x and t. The vector t contains the time samples for the input and output, x contains the values of the input x(t) at each time in t, and y contains the simulated values of the output y(t) at each time in t.



tf

Create transfer function model, convert to transfer function model

$$H(s) = \frac{\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2}.$$

```
zeta = 0.25;
w0 = 3;
H = tf(w0^2,[1,2*zeta*w0,w0^2])

H =

          9
  ---------------
  s^2 + 1.5 s + 9

Continuous-time transfer function.
```

■ 2.3 Tutorial: `lsim` with Differential Equations

The function `lsim` can be used to simulate the output of continuous-time, causal LTI systems described by linear constant-coefficient differential equations of the form

$$\sum_{k=0}^{N} a_k \frac{d^k y(t)}{dt^k} = \sum_{m=0}^{M} b_m \frac{d^m x(t)}{dt^m}. \tag{2.11}$$

To use `lsim`, the coefficients a_k and b_m must be stored in MATLAB vectors `a` and `b`, respectively, in descending[1] order of the indices k and m. Rewriting Eq. (2.11) in terms of the vectors `a` and `b` gives

$$\sum_{k=0}^{N} \mathtt{a}(N+1-k) \frac{d^k y(t)}{dt^k} = \sum_{m=0}^{M} \mathtt{b}(M+1-m) \frac{d^m x(t)}{dt^m}. \tag{2.12}$$

Note that `a` must contain $N+1$ elements, which might require appending zeros to `a` to account for coefficients a_k that equal zero. Similarly, the vector `b` must contain $M+1$ elements. With `a` and `b` defined as in Eq. (2.12), executing

```
>> y = lsim(b,a,x,t);
```

■ 3.3 Tutorial: `lsim` with System Functions

Tutorial 2.3 describes how the `lsim` command can be used to simulate a causal LTI continuous-time system whose input and outputs satisfy a linear constant-coefficient differential equation. The output of a causal LTI system specified by its system function can also be simulated using `lsim`, since the system function uniquely specifies the differential equation relating the input and output of the system. If the system function is put in the form

$$H(s) = \frac{\mathtt{b}(1)s^M + \ldots + \mathtt{b}(M-1)s + \mathtt{b}(M)}{\mathtt{a}(1)s^N + \ldots + \mathtt{a}(N-1)s + \mathtt{a}(N)},$$

the output of the system for an input $x(t)$ can be simulated using `lsim(b,a,x,t)`, where the MATLAB vectors `b` and `a` contain the coefficients of the numerator and denominator polynomials in s, and the vectors `x` and `t` describe the input signal in the same format specified in Tutorial 2.3. Note that $H(s)$ must be a proper fraction, i.e., $N \geq M$.

As an example, consider the system function

$$H(s) = \frac{s + \frac{1}{2}}{s - 2},$$

whose coefficients are defined by the vectors `b=[1 1/2]` and `a=[1 -2]`. The command

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

+

Initial rest

Coefficient vectors : **A=[a_K, a_{k-1},... a_1,a_0] ,B=[b_M , b_{M-1}... b_1,b_0]**.

Laplace Transformation

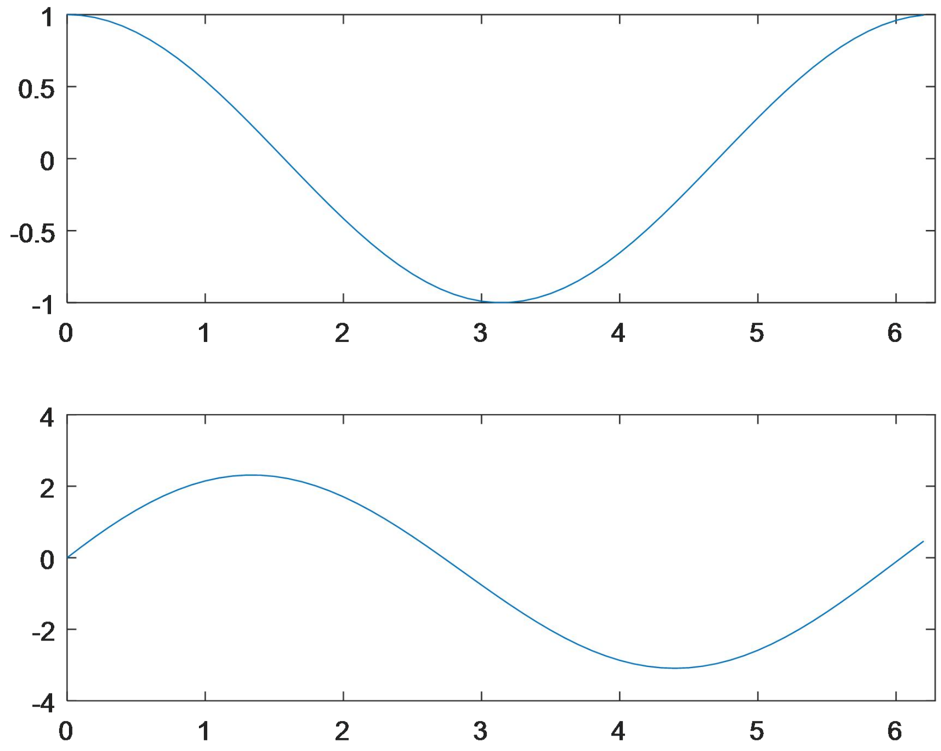
$$\sum_{k=0}^{K} a_k s^k Y(s) = \sum_{m=0}^{M} b_m s^m X(s)$$

$$H(s) = \frac{Y(s)}{X(s)} = \frac{b_M s^M + b_{M-1} s^{M-1} + \cdots + b_1 s^1 + b_0}{a_K s^K + a_{K-1} s^{K-1} + \cdots + a_1 s^1 + a_0}$$

Transfer Function/
System Function

Causal CT LTI   dy(t)/dt+0.3y(t) = 3x(t)

```
%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]);
```

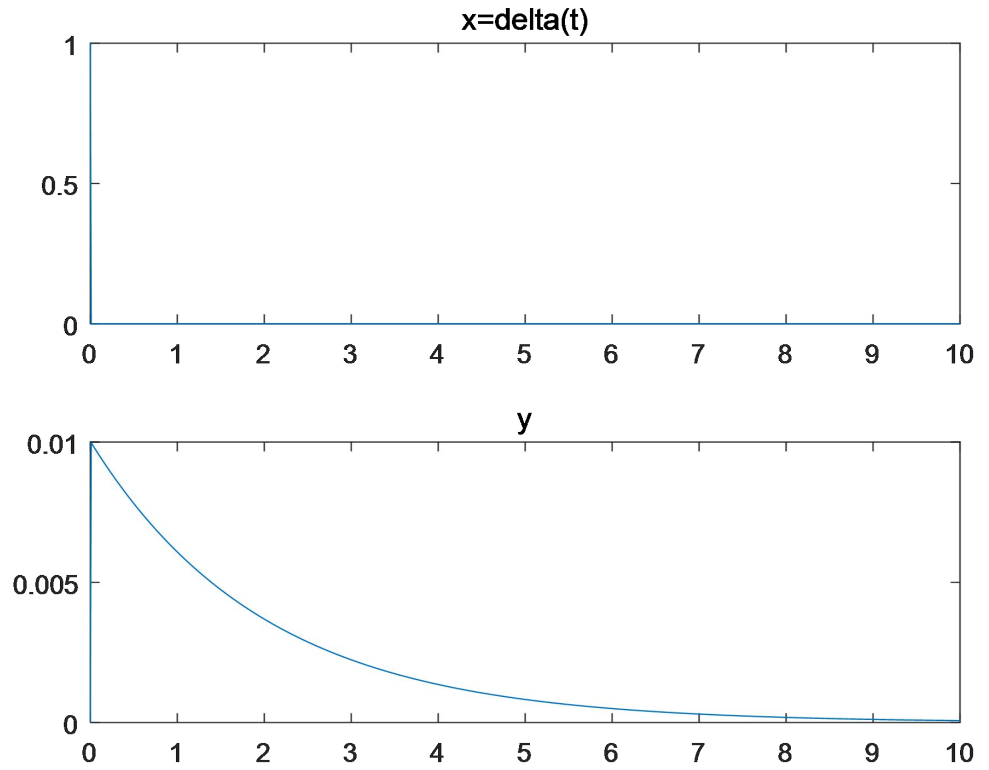


x is is a row vector, the output of lsim is a column vector.

Causal CT LTI $dy(t)/dt+0.5y(t) = x(t)$

```
%dy(t)/dt+0.5y(t) = x(t)
A=[1 0.5];
B=1;
%Sample the input signal x=delta(t):
t=0:0.01:10;
x=[1,zeros(1,length(t)-1)];
y=lsim(B,A,x,t)';
subplot(2,1,1), plot(t,x); xlim([0 10]),title('x=delta(t)')
subplot(2,1,2), plot(t,y); xlim([0 10]),title('y')
figure
impulse(B,A,t)
```

y=impulse(B,A,t)



It is the impulse response of the CT LTI System.

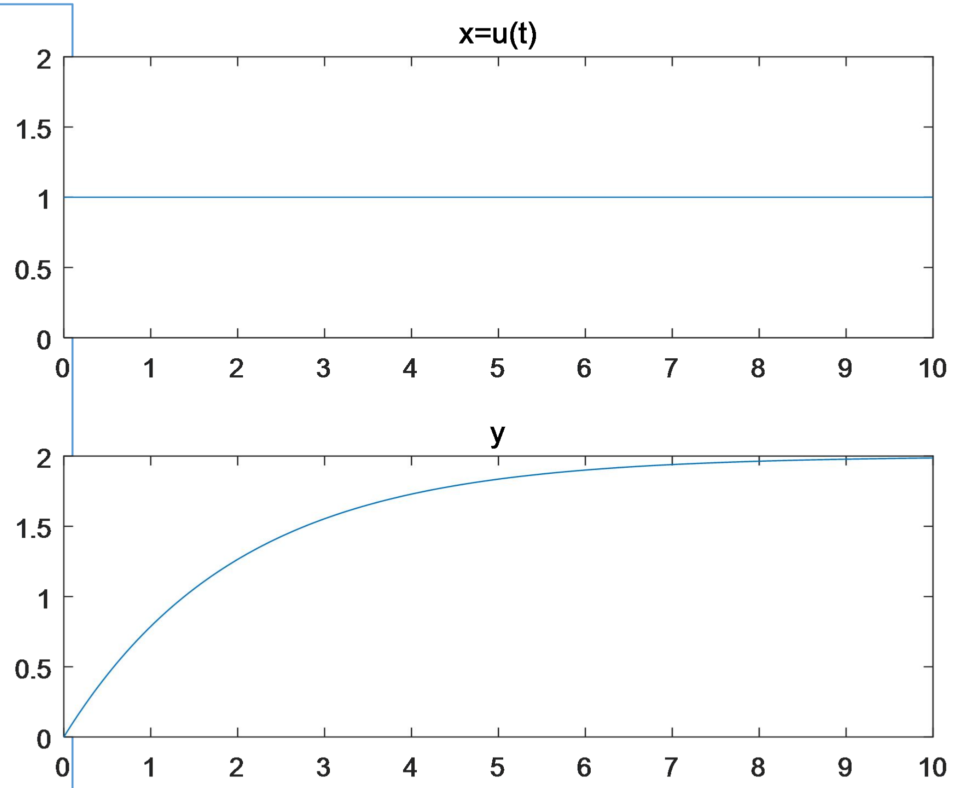
y=impulse(B,A,t)

Causal CT LTI dy(t)/dt+0.5y(t) = x(t)

```
%dy(t)/dt+0.5y(t) = x(t)
A=[1 0.5];
B=1;
%Sample the input signal x=u(t):
t=0:0.01:10;
x=ones(1,length(t));
y=lsim(B,A,x,t)';
```

y=step(B,A,t)

```
subplot(2,1,1), plot(t,x); xlim([0 10]),title('x=u(t)')
subplot(2,1,2), plot(t,y); xlim([0 10]),title('y')
figure
step(B,A,t)
```



It is the step response of the CT LTI System.

y=step(B,A,t)

freqs(Frequency Response of CT LTI Systems)

H=freqs(b,a) or H=freqs(b,a,w), calculate the frequency response of CT LTI Systems.

$$x(t) = \sum_{k} a_k e^{jw_k t}$$

CT LTI Systems

LTI （h(t)）

$$y(t) = \sum_{k} a_k H(jw_k) e^{jw_k t}$$

$$x(t) = \sum_{k=-\infty}^{+\infty} a_k e^{jkw_0 t} = \sum_{k=-\infty}^{+\infty} a_k e^{jk(2\pi/T)t}$$ → ***Synthesis Equation***

$$a_k = \frac{1}{T}\int_T x(t)e^{-jkw_0 t}dt = \frac{1}{T}\int_T x(t)e^{-jk(2\pi/T)}dt$$ → ***Analysis Equation***

{a_k} are often called the Fourier series coefficients or the **spectral coefficients**.

k $\in (-\infty, +\infty)$

>> help freqs

freqs Laplace-transform (s-domain) frequency response.

H = freqs(B,A,W) returns the complex frequency response vector H of the filter B/A:

$$H(s) = \frac{B(s)}{A(s)} = \frac{b(1)s^n + b(2)s^{n-1} + \cdots + b(n+1)}{a(1)s^m + a(2)s^{m-1} + \cdots + a(m+1)}$$

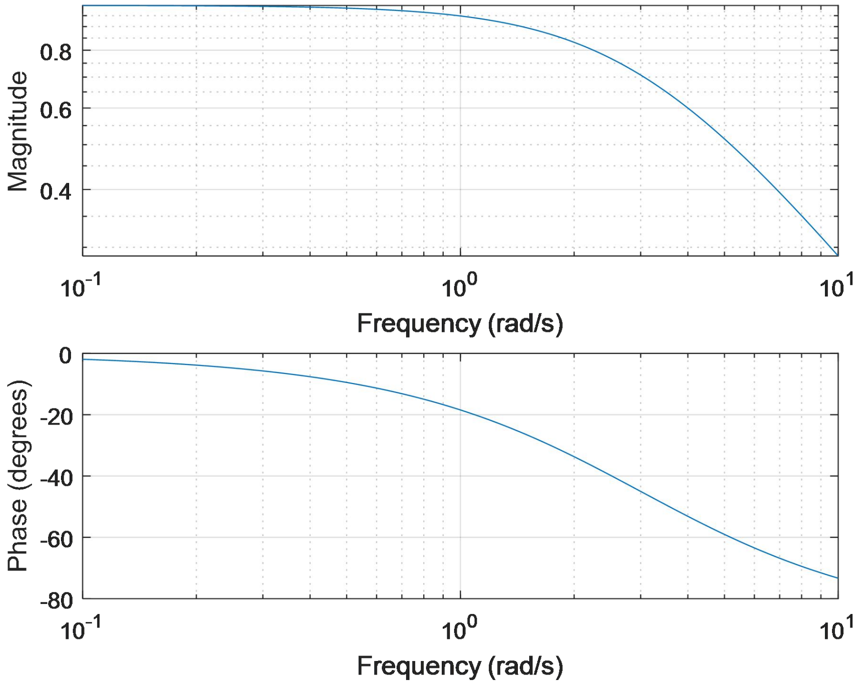
given the numerator and denominator coefficients in vectors B and A. The frequency response is evaluated **at the points specified in vector W (in rad/s)**. The magnitude and phase can be graphed by calling freqs(B,A,W) with no output arguments.

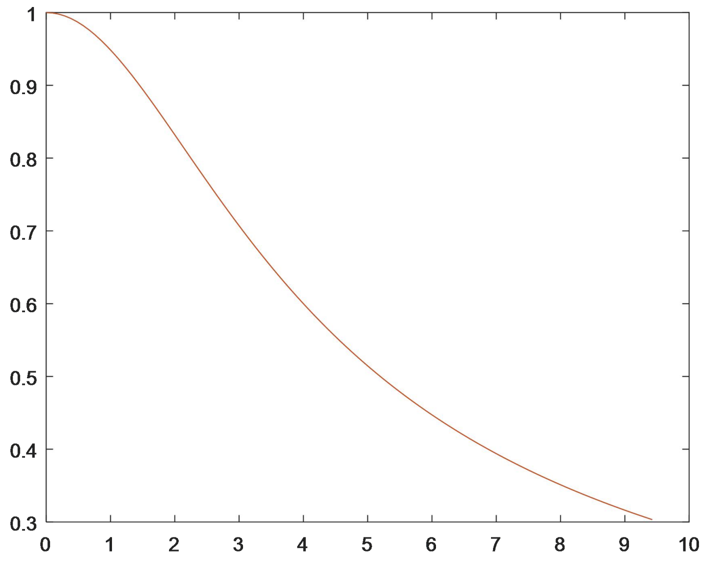
[H,W] = freqs(B,A) automatically picks a **set of 200 frequencies** W on which the frequency response is computed.

An causal、stable CT LTI System can be described by the following differential equation, please plot its frequency response.

$$\frac{dy(t)}{dt} + 3y(t) = 3x(t)$$

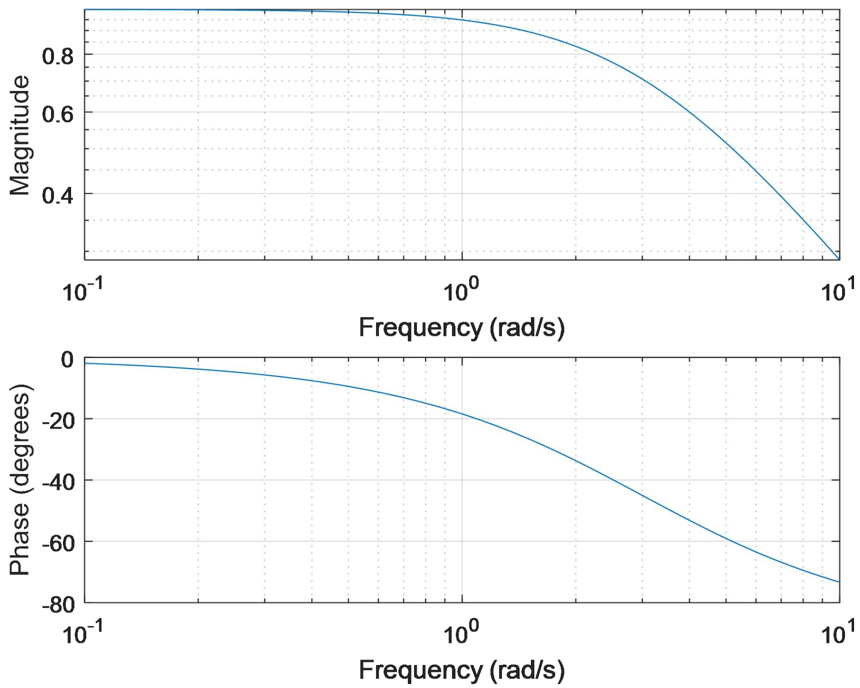
```
 a=[1 3];
 b=3;
%automatically picks a set of  W
 freqs(b,a)

%Specify vector W (in rad/s)
 w=linspace(0,3*pi);
 H=freqs(b,a,w);
 figure
 plot(w,abs(H)) ,
hold on, plot(w,abs(3./(3+1j*w)))
```





H = freqs(B,A,W) the meaning of its outputs:

Causal CT LTI $\frac{dy(t)}{dt} + 3y(t) = 3x(t)$



For this CT LTI System, the output H(jw) is just the eigenvalue to each eigenfunction e^{jwk} , also it is the frequency response.

Similar to h(t)

■ 3.9 Frequency Response of a Continuous-Time System

This exercise demonstrates the effect of the frequency response of a continuous-time system on periodic signals. You will examine the response of a simple linear system to each of the harmonics that compose a periodic signal as well as to the periodic signal itself. In this exercise you will need to use the function `lsim` as discussed in the Tutorial 3.3.

Consider a simple RC circuit that has a system function given by

$$H(s) = \frac{1}{1 + RCs},$$

s=jw

whose input is given by

$$x(t) = \cos(t),$$

and whose output is $y(t)$. For the problems that follow, use `t=linspace(0,20,1000)` for all simulations, and assume that the time constant RC is 1.

Properties of CTFS

Intermediate Problems

(a). Use the function `lsim` to simulate the response of the system $H(s)$ to $x(t)$ over $0 \leq t \leq 20$, storing the response in the signal $y(t)$. Plot the output $y(t)$ and the input $x(t)$ for $10 \leq t \leq 20$ on the same graph, and note the amplitude and phase change from input to output. Can you explain each of these effects in terms of the frequency response? Hint: Use the system function $H(s)$ to determine the exact form of the output $y(t)$ when the input is $x(t) = \cos(t)$.

(b). Now look at the response of the system to the square wave that results from first passing $x(t)$ through a hard-limiter, $x_2(t) = \text{sign}(\cos(t))$. A simple way to use selective indexing to generate the square wave is

Calculate a_k analytical, then only use the previous harmonics to synthesis the input signal approximately. Also, H(jw) is already known.

e^{jwt}	$H(jw)e^{jwt}$
$a_{-\infty}e^{j-\infty(\frac{2\pi}{T})t}$	$a_{-\infty}H(j-\infty(\frac{2\pi}{T}))e^{j-\infty(\frac{2\pi}{T})t}$
$\cdot\ \cdot\ \cdot$	$\cdot\ \cdot\ \cdot$
$a_{-2}e^{j-2(\frac{2\pi}{T})t}$	$a_{-2}H(j-2(\frac{2\pi}{T}))e^{j-2(\frac{2\pi}{T})t}$
$a_{-1}e^{j-1(\frac{2\pi}{T})t}$	$a_{-1}H(j-1(\frac{2\pi}{T}))e^{j-1(\frac{2\pi}{T})t}$
$a_0e^{j0(\frac{2\pi}{T})t}$	$a_0H(j0(\frac{2\pi}{T}))e^{j0(\frac{2\pi}{T})t}$
$a_1e^{j1(\frac{2\pi}{T})t}$	$a_1H(j1(\frac{2\pi}{T}))e^{j1(\frac{2\pi}{T})t}$
$a_2e^{j2(\frac{2\pi}{T})t}$	$a_2H(j2(\frac{2\pi}{T}))e^{j2(\frac{2\pi}{T})t}$
$\cdot\ \cdot\ \cdot$	$\cdot\ \cdot\ \cdot$
$a_{\infty}e^{j\infty(\frac{2\pi}{T})t}$	$a_{\infty}H(j\infty(\frac{2\pi}{T}))e^{j\infty(\frac{2\pi}{T})t}$

x(t) $\quad \widetilde{x(t)} \quad \widetilde{y(t)} \quad$ y(t)

Lab3 Assignments

- Read tutorial: 3.1、3.2
- Assignments：3.5 & 3.8 & 3.10