Xi'an Jiaotong-Liverpool University
西交利物浦大学

EEE220 Instrumentation and Control System

2018-19 Semester 2

Dr. Qing Liu

Email: qing.liu@xjtlu.edu.cn

Office: EE516

Department of Electrical and Electronic Engineering

21 March, 2019

Lecture 10

Outline

Control Systems:

Mathematical Models of Systems: part 2/2

- ❑ Differential Equations of Physical Systems
- ❑ The Transfer Function of Linear Systems
- ❑ **Block Diagram Models**
- ❑ **Signal-Flow Graph Models**
- ❑ **Simulation Tool**

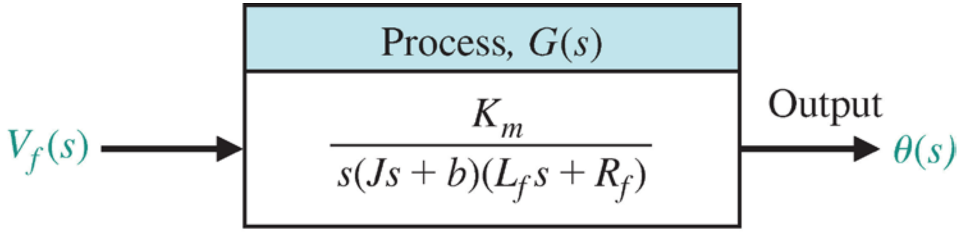
Block Diagram Models

- The dynamic systems are typically represented mathematically by a set of simultaneous **differential equations**; the **Laplace transform** reduces the problem to the solution of a set of linear algebraic equations;

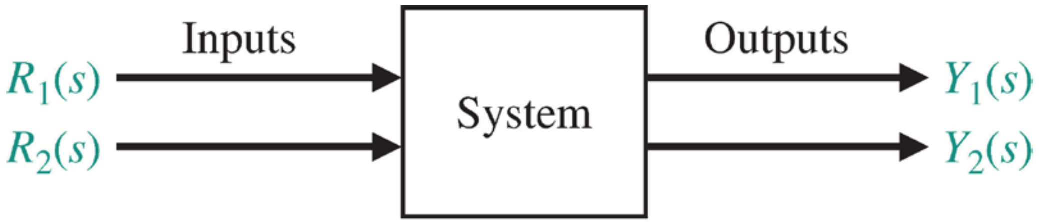
- **Transfer functions** are introduced to represent the relationship between the input and output variables of the system; it is an important relation in control engineering;

- **Block diagram** is a graphic way to represent this important cause-and-effect relationship. It consists of unidirectional, operational blocks that represent the transfer function of the systems of interest.

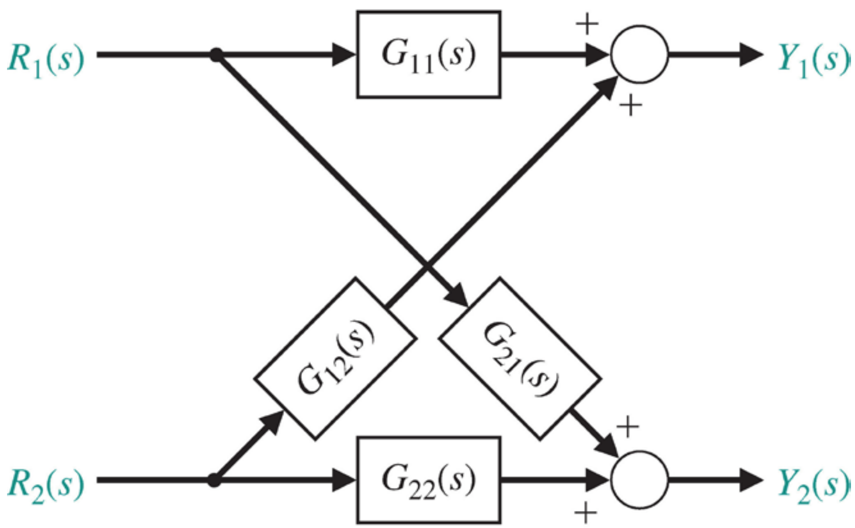
Examples



(a) Block diagram of a DC motor.

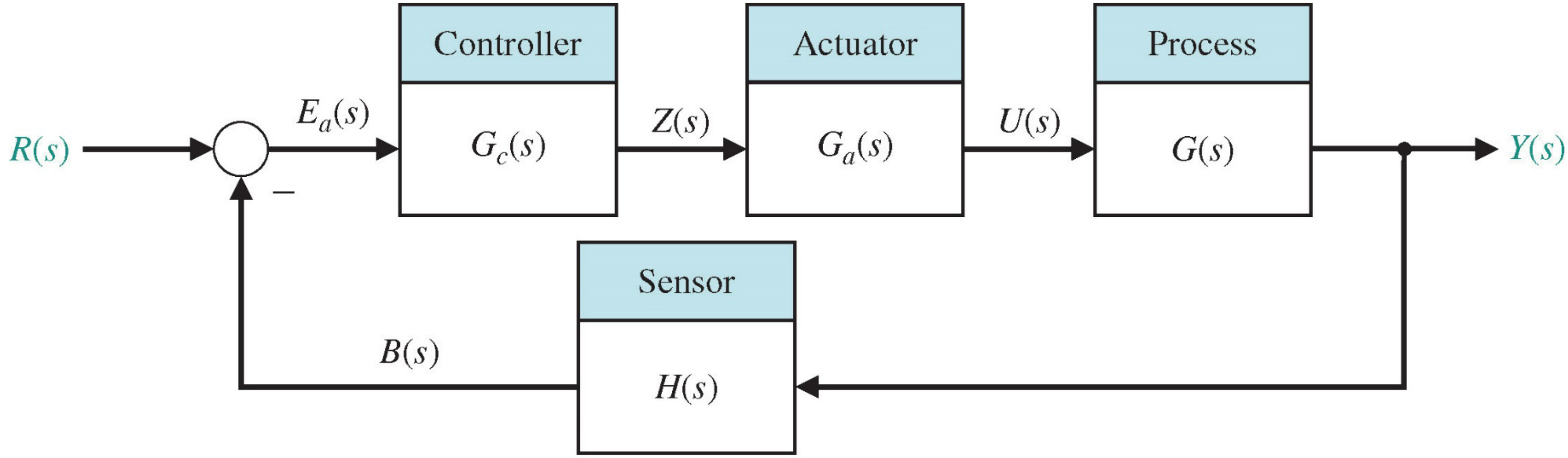


(b) General block representation of two-input, two-output system.



(c) Block diagram of a two-input, two-output interconnected system.

Transfer Function of A Negative Feedback Control System



The error signal or actuating signal $E_a(s)$ is

$$E_a(s) = R(s) - H(s)Y(s)$$

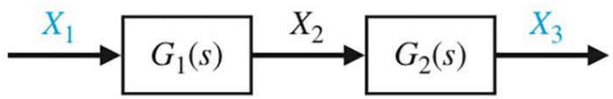
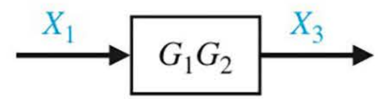
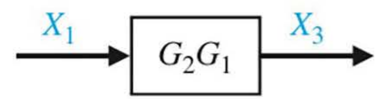
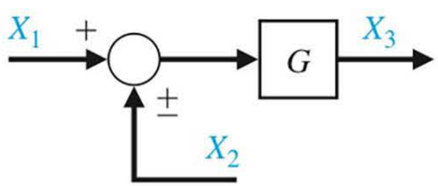
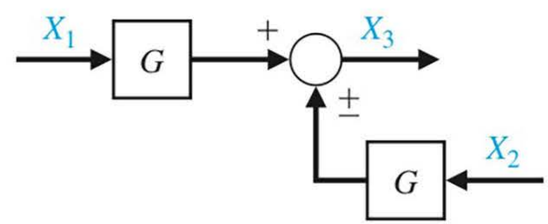
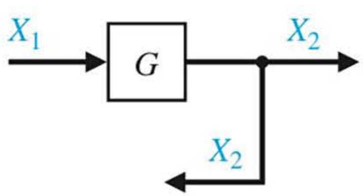
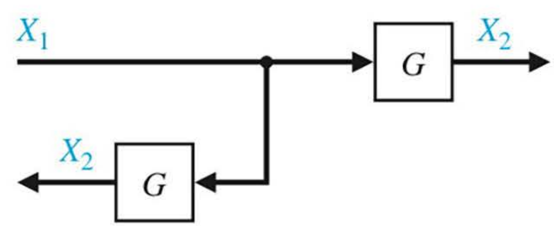
The output signal $Y(s)$ can be represented as

$$Y(s) = E_a(s)G_c(s)G_a(s)G(s)$$

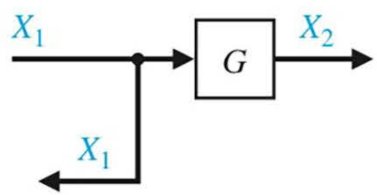
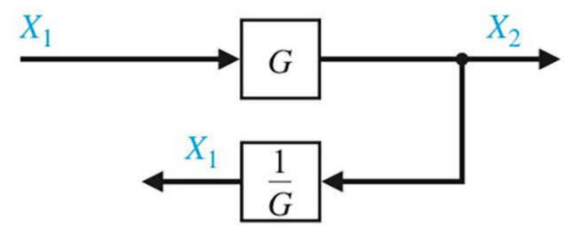
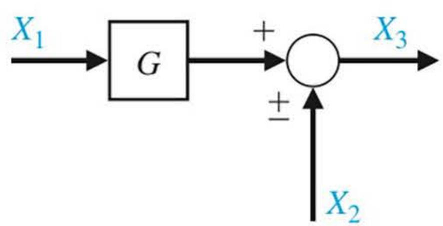
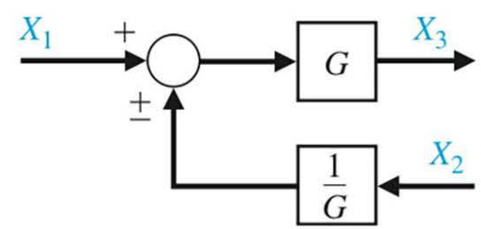
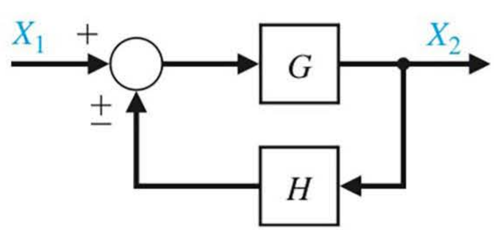
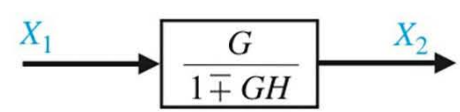
Combining the above two equations, the transfer function of the system is

$$\boxed{\frac{Y(s)}{R(s)} = \frac{G_c(s)G_a(s)G(s)}{1 + G_c(s)G_a(s)G(s)H(s)}}$$

Block Diagram Transformations

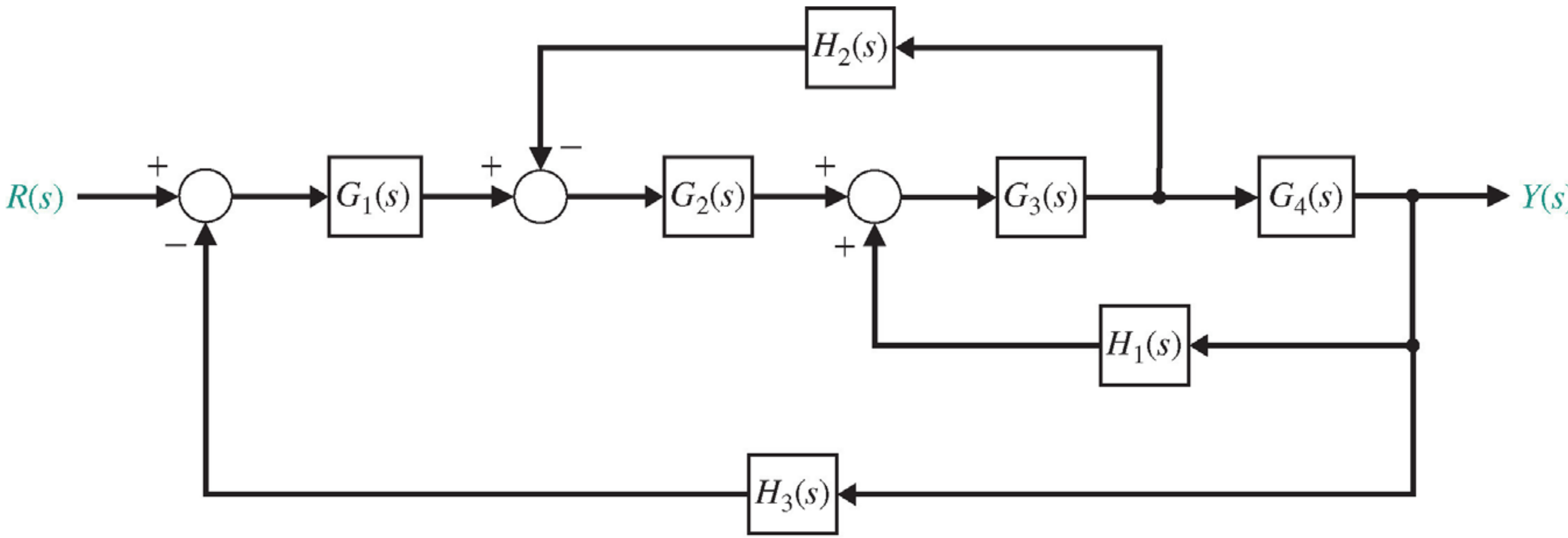
Transformation	Original Diagram	Equivalent Diagram
1. Combining blocks in cascade		 or 
2. Moving a summing point behind a block		
3. Moving a pickoff point ahead of a block		

Block Diagram Transformations (cont'd)

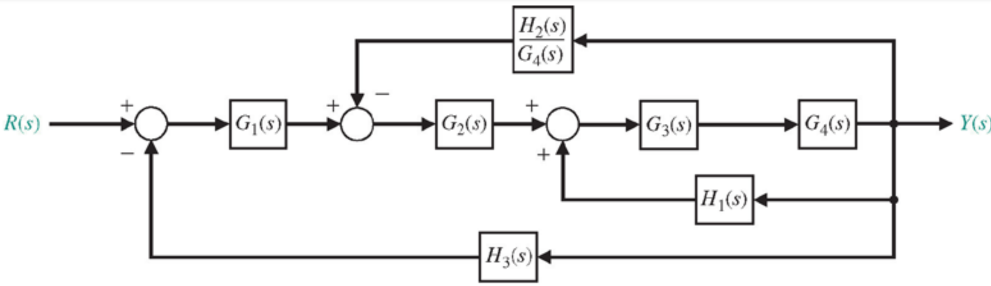
Transformation	Original Diagram	Equivalent Diagram
4. Moving a pickoff point behind a block		
5. Moving a summing point ahead of a block		
6. Eliminating a feedback loop		

Reduction of the Block Diagram

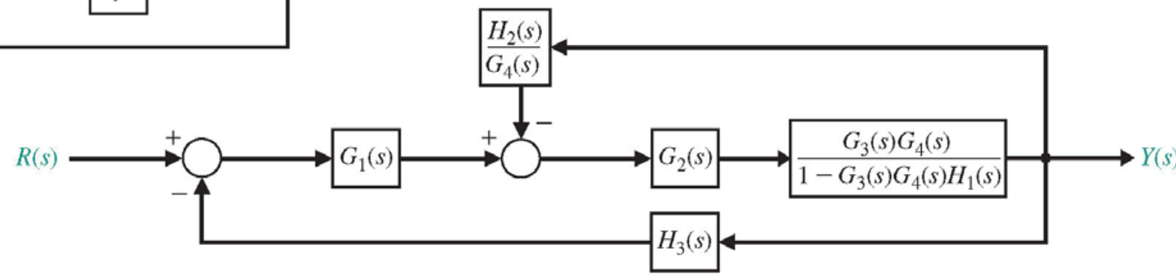
A multi-loop feedback control system



Please note the feedback $H_1G_3G_4$ is a positive feedback; while $G_2G_3H_2$ and $G_1G_2G_3G_4H_3$ are negative feedback loops.

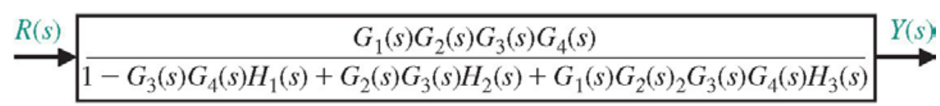


(a)



(b)

- Step 1: move the pickoff point between G3 & G4 after G4 (from original diagram to (a));
- Step 2. eliminate loop 1 (G3G4H1) (from (a) to (b));
- Step 3. eliminate loop 2 (from (b) to (c));
- Step 4. eliminate loop 3 (from (c) to (d)).

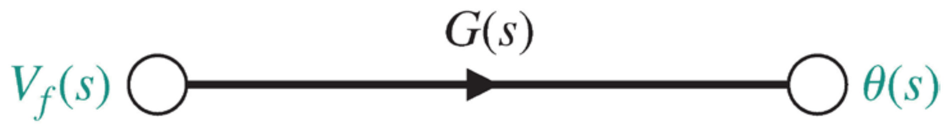


(d)

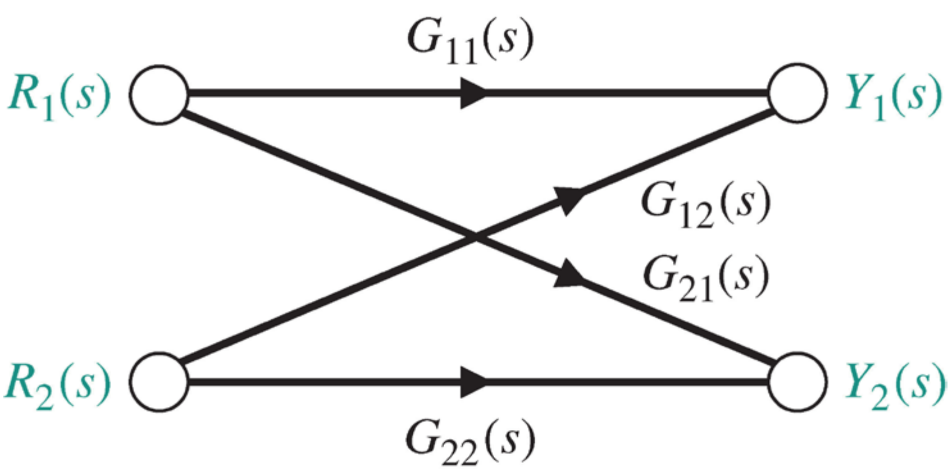
Signal-Flow Graph Models

- A **signal-flow graph** is a diagram consisting of **nodes** that are connected by several **directed branches** and is a graphic representation of a set of linear relations;
- Signal-flow graph is particularly useful for feedback control systems because feedback theory is primarily concerned with the flow and processing of signals in the system;
- The basic element of a signal-flow graph is a unidirectional path segment called a **branch**, which relates the dependency of input and an output variable in a manner equivalent to a **block** of a block diagram.
- For complex systems, the block diagram method can become difficult to complete. By using the signal-flow graph model, the reduction procedure (used in the block diagram method) is not necessary to determine the relationship between system variables.

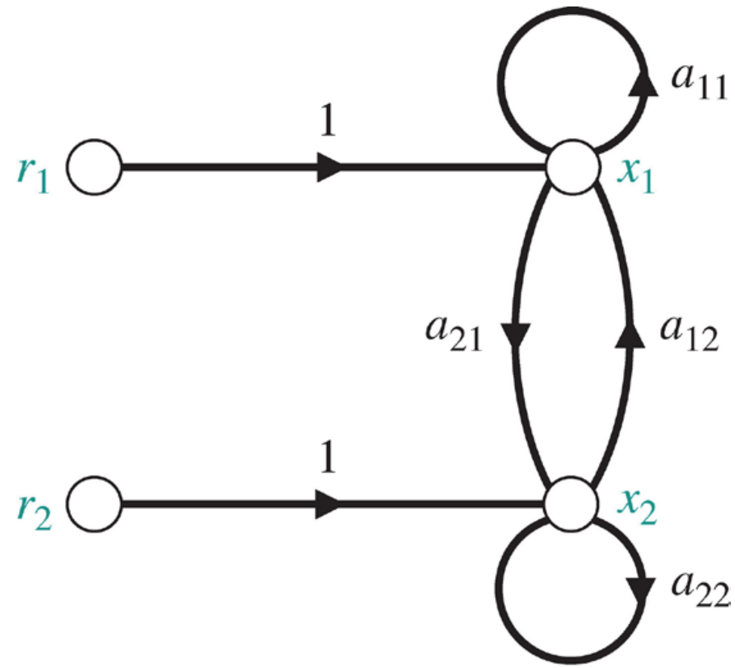
Examples



(a) Signal-flow graph of a DC motor.



(b) Signal-flow graph of two-input, two-output interconnected system.

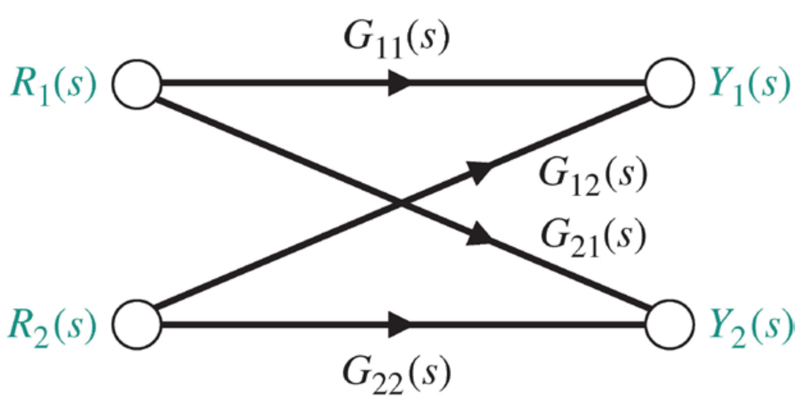


(c) Signal-flow graph of two algebraic equations.

Basic Concepts

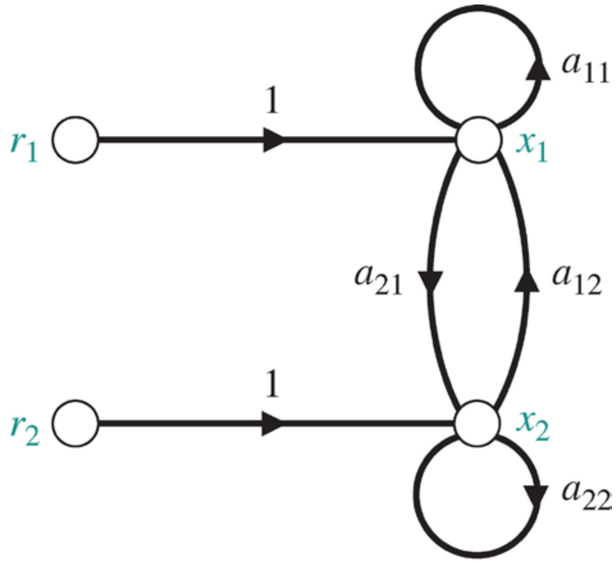
- **Nodes**: are the input and output points or junctions; all branches leaving a node will pass the nodal signal to the output node of each branch (unidirectional); the summation of all signals entering a node is equal to the node variable;
- A **Path**: is a branch or a continuous sequence of branches that can be traversed from one node (signal) to another node (signal);

- A **Loop**: is a closed path that originates and terminates on the same node, with no node been met twice along the path;
- Two loops are said to be **nontouching** if they don't have a common node; two touching loops share one or more common nodes;



Signal-flow graph of two-input, two-output interconnected system.

Determine Transfer Function from the Graph



$$a_{11}x_1 + a_{12}x_2 + r_1 = x_1$$

$$a_{21}x_1 + a_{22}x_2 + r_2 = x_2.$$

$$x_1(1 - a_{11}) + x_2(-a_{12}) = r_1,$$

$$x_1 = \frac{(1 - a_{22})r_1 + a_{12}r_2}{(1 - a_{11})(1 - a_{22}) - a_{12}a_{21}} = \frac{1 - a_{22}}{\Delta}r_1 + \frac{a_{12}}{\Delta}r_2,$$

$$x_2 = \frac{(1 - a_{11})r_2 + a_{21}r_1}{(1 - a_{11})(1 - a_{22}) - a_{12}a_{21}} = \frac{1 - a_{11}}{\Delta}r_2 + \frac{a_{21}}{\Delta}r_1.$$

Where: $\Delta = (1 - a_{11})(1 - a_{22}) - a_{12}a_{21} = 1 - a_{11} - a_{22} + a_{11}a_{22} - a_{12}a_{21}.$

Mason's Signal-flow Gain Formula

In general, the linear dependence $T_{ij}(s)$ between the independent variable x_i (often called the input variable) and a dependent variable x_j is given by:

$$\boxed{T_{ij} = \frac{\sum_k P_{ijk}\Delta_{ijk}}{\Delta}}$$

The summation is taken over all possible k paths from x_i to x_j.

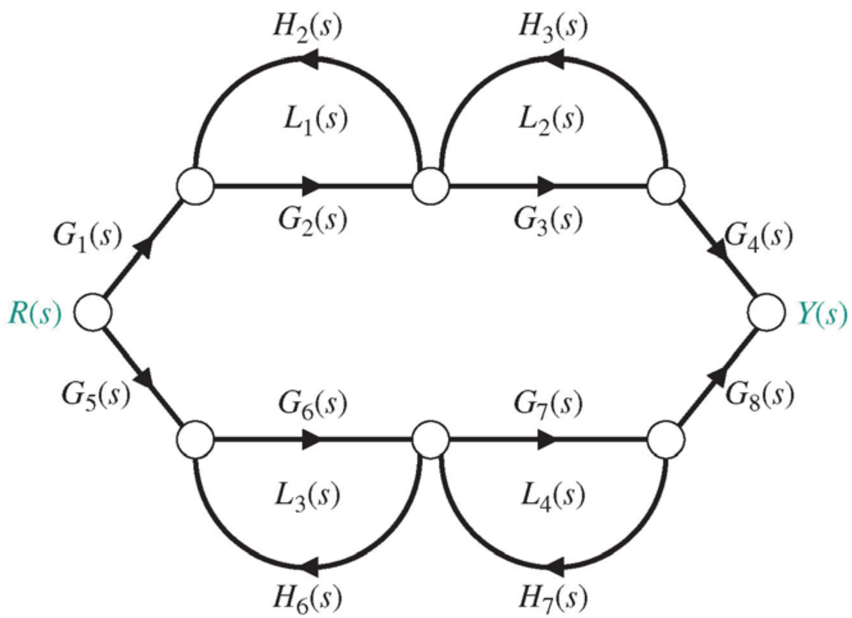
- $\boldsymbol{P_{ijk}}$: is the path gain defined as the product of gains of the branches of the path, traversed in the direction of the arrows with no node encountered more than once;
- Δ_{ijk}: cofactor, is the determinant with the loops touching the kth path removed.
- Δ: the determinant, is:

$$\Delta = 1 - \sum_{n=1}^{N} L_n + \sum_{\substack{n,\, m \\ \text{nontouching}}} L_n L_m - \sum_{\substack{n,\, m,\, p \\ \text{nontouching}}} L_n L_m L_p + \cdots,$$

Where L_q equals the value of qth loop transmittance, therefore:

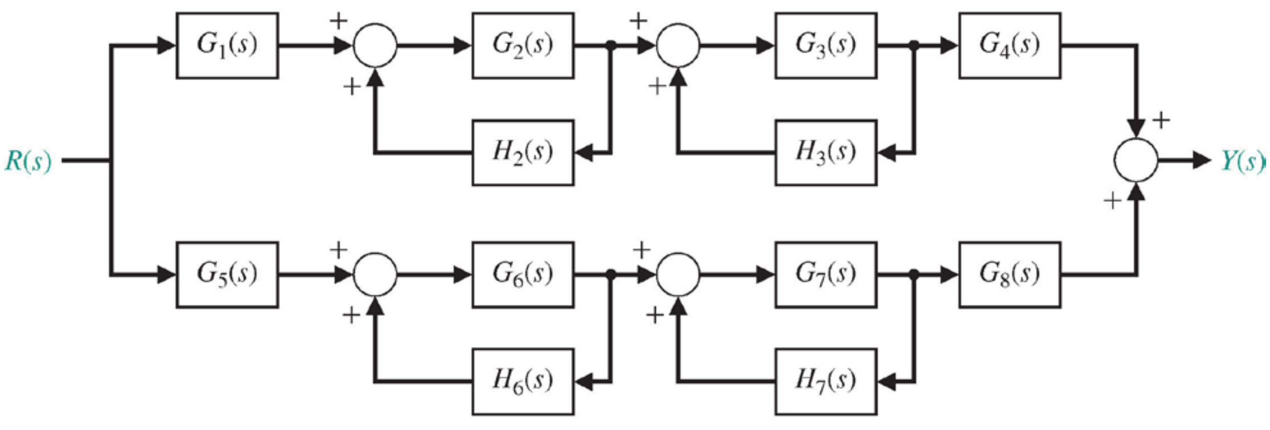
$$\begin{aligned} \Delta = 1 &- (\text{sum of all different loop gains}) \\ &+ (\text{sum of the gain products of all combinations of two nontouching loops}) \\ &- (\text{sum of the gain products of all combinations of three nontouching loops}) \\ &+ \cdots. \end{aligned}$$

Mason’s Rule: Application



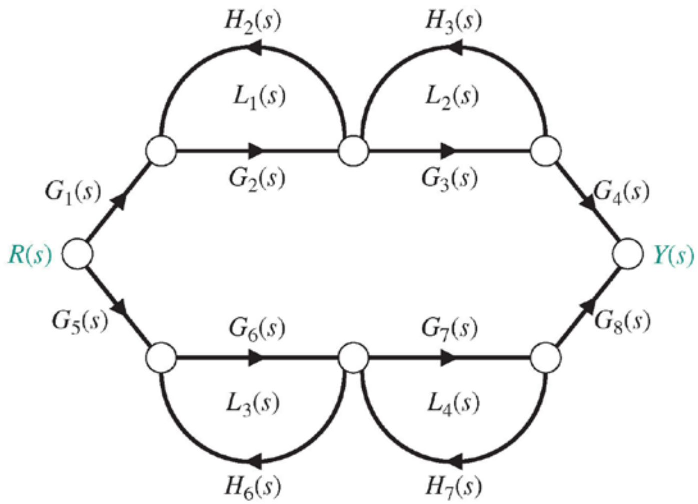
(a)

How to obtain Transfer Function between Input R(s) and Output Y(s)?



(b)

Example 10.1



$$\boxed{T_{ij} = \frac{\sum_k P_{ijk}\Delta_{ijk}}{\Delta}}$$

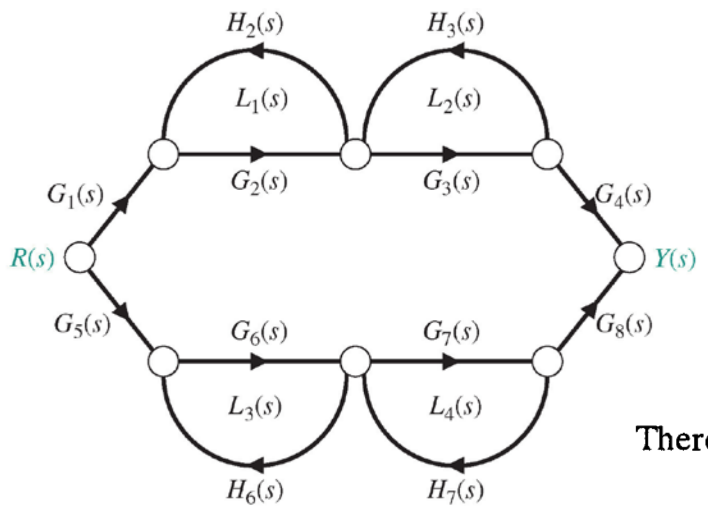
1. How many possible paths? $P_1 = G_1G_2G_3G_4$ (path 1) and $P_2 = G_5G_6G_7G_8$ (path 2).

2. How many loops? There are four self-loops:

$$L_1 = G_2H_2, \quad L_2 = H_3G_3, \quad L_3 = G_6H_6, \quad \text{and} \quad L_4 = G_7H_7.$$

3. How many groups of 2 nontouching loops? Loops L_1 and L_2 do not touch L_3 and L_4.

4. How many groups of 3 nontouching loops? None.



$$\boxed{T_{ij} = \frac{\sum_k P_{ijk}\Delta_{ijk}}{\Delta}}$$

There are four self-loops:

$$L_1 = G_2H_2, \quad L_2 = H_3G_3, \quad L_3 = G_6H_6, \quad \text{and} \quad L_4 = G_7H_7.$$

Loops L_1 and L_2 do not touch L_3 and L_4. Therefore, the determinant is

$$\Delta = 1 - (L_1 + L_2 + L_3 + L_4) + (L_1L_3 + L_1L_4 + L_2L_3 + L_2L_4).$$

The cofactor of the determinant along path 1 is evaluated by removing the loops that touch path 1 from Δ. Hence, we have

$$L_1 = L_2 = 0 \quad \text{and} \quad \Delta_1 = 1 - (L_3 + L_4).$$

Similarly, the cofactor for path 2 is

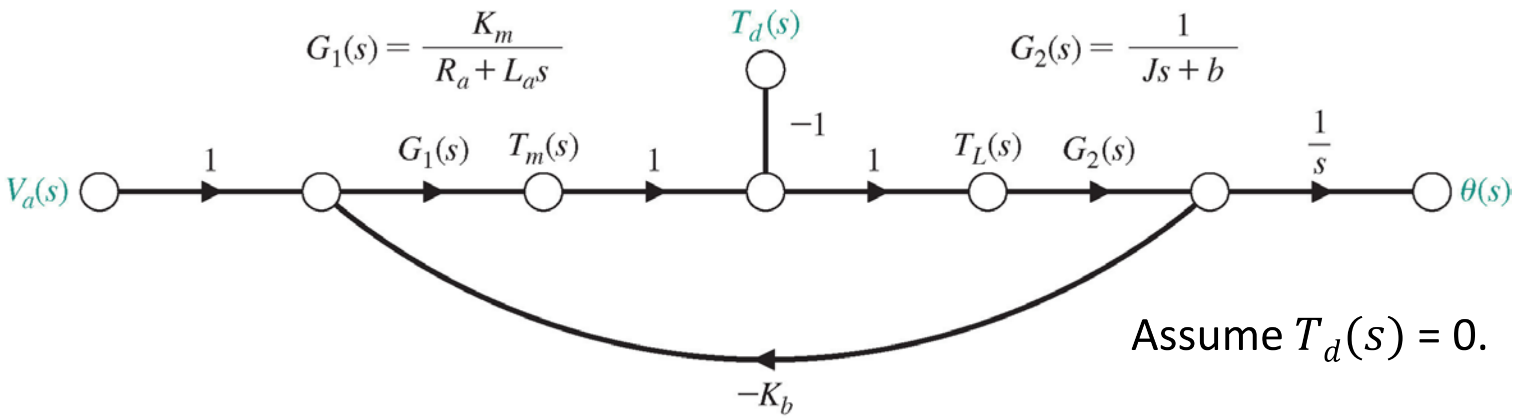
$$\Delta_2 = 1 - (L_1 + L_2).$$

Therefore, the transfer function of the system is

$$\frac{Y(s)}{R(s)} = T(s) = \frac{P_1\Delta_1 + P_2\Delta_2}{\Delta}$$

$$= \frac{G_1G_2G_3G_4(1 - L_3 - L_4) + G_5G_6G_7G_8(1 - L_1 - L_2)}{1 - L_1 - L_2 - L_3 - L_4 + L_1L_3 + L_1L_4 + L_2L_3 + L_2L_4}$$

Example 10.2



Assume $T_d(s) = 0$.

Signal-flow graph of the armature-controlled DC motor

There is only 1 forward path, which touches one loop:

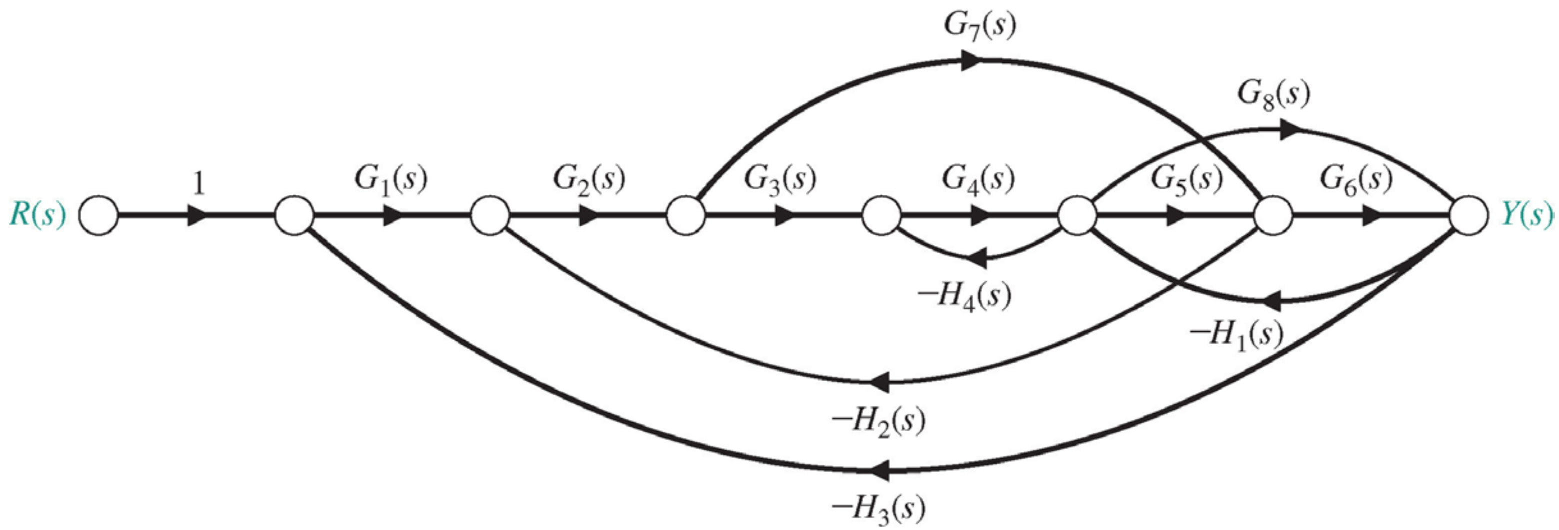
$$P_1(s) = \frac{1}{s}G_1(s)G_2(s) \quad \text{and} \quad L_1(s) = -K_bG_1(s)G_2(s).$$

Therefore, the transfer function is

$$T(s) = \frac{P_1(s)}{1 - L_1(s)} = \frac{(1/s)G_1(s)G_2(s)}{1 + K_bG_1(s)G_2(s)} = \frac{K_m}{s[(R_a + L_as)(Js + b) + K_bK_m]}.$$

Example 10.3

Consider a reasonably complex system that would be difficult to reduce by block diagram techniques:



$$P_1 = G_1G_2G_3G_4G_5G_6, \qquad P_2 = G_1G_2G_7G_6, \quad \text{and} \quad P_3 = G_1G_2G_3G_4G_8.$$

The feedback loops are

$$L_1 = -G_2G_3G_4G_5H_2, \qquad L_2 = -G_5G_6H_1, \qquad L_3 = -G_8H_1, \qquad L_4 = -G_7H_2G_2,$$
$$L_5 = -G_4H_4, \qquad L_6 = -G_1G_2G_3G_4G_5G_6H_3, \qquad L_7 = -G_1G_2G_7G_6H_3, \quad \text{and}$$
$$L_8 = -G_1G_2G_3G_4G_8H_3.$$

Loop L_5 does not touch loop L_4 or loop L_7, and loop L_3 does not touch loop L_4; but all other loops touch. Therefore, the determinant is

$$\Delta = 1 - (L_1 + L_2 + L_3 + L_4 + L_5 + L_6 + L_7 + L_8) + (L_5L_7 + L_5L_4 + L_3L_4).$$

The cofactors are

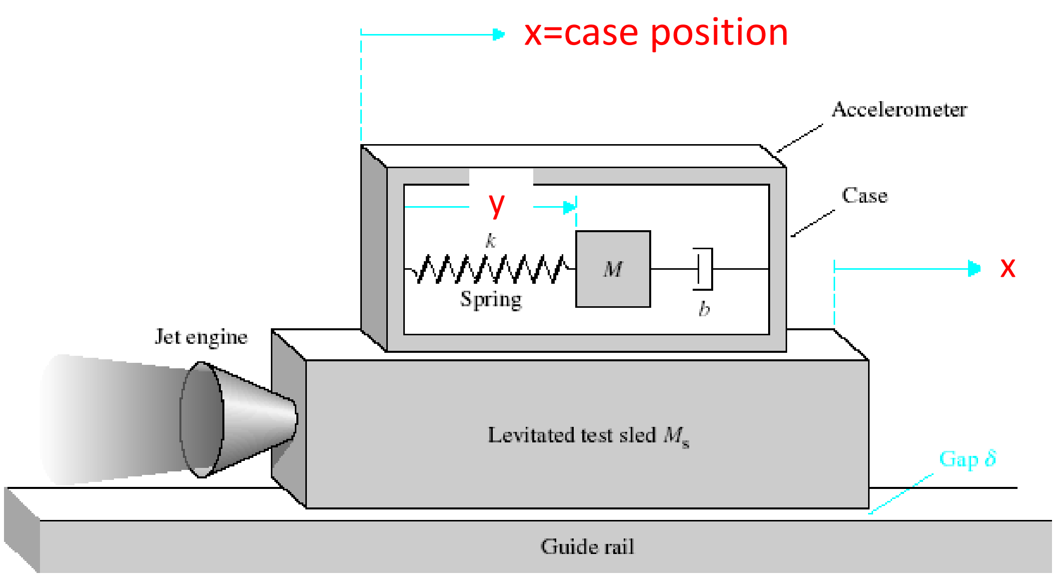
$$\Delta_1 = \Delta_3 = 1 \quad \text{and} \quad \Delta_2 = 1 - L_5 = 1 + G_4H_4.$$

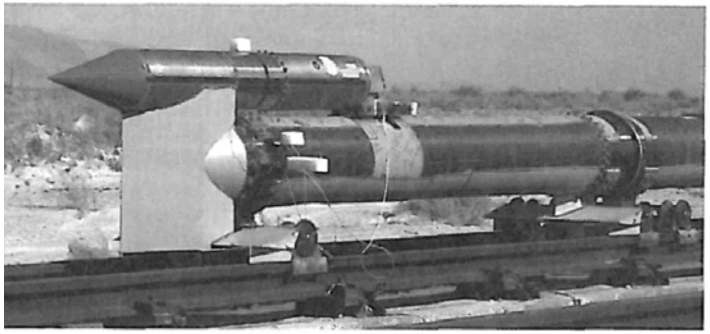
Finally, the transfer function is

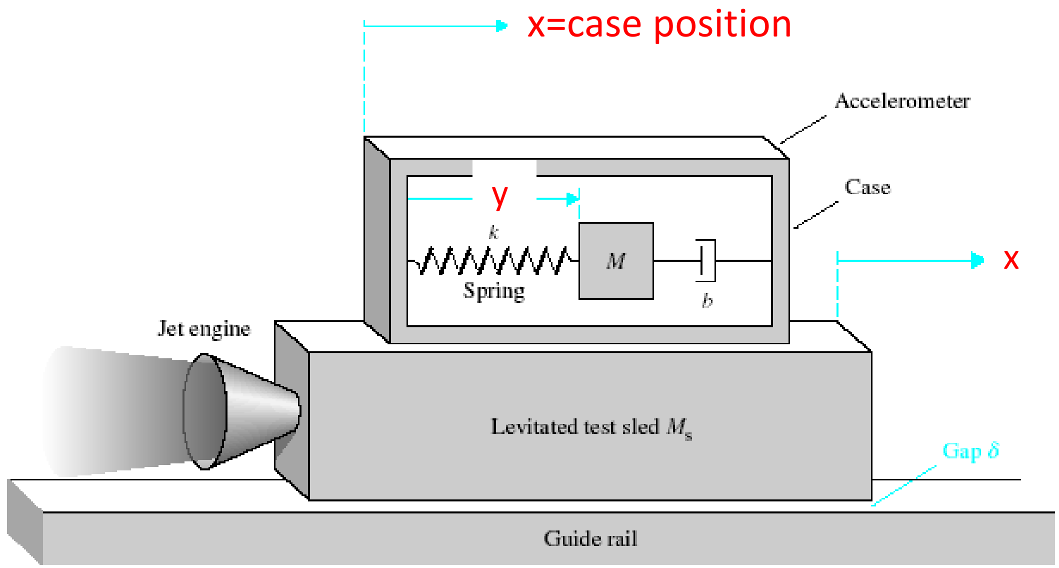
$$T(s) = \frac{Y(s)}{R(s)} = \frac{P_1 + P_2\Delta_2 + P_3}{\Delta}.$$

Real-world Example: Mechanical Accelerometer

- A mechanical accelerometer is used to measure the acceleration of a rocket test sled. The test sled maneuvers above a guide rail a small distance δ.
- The accelerometer provides a measurement of the acceleration $a(t)$ of the sled, since the position *y* of the mass *M*, with respect to the accelerometer case, is proportional to the acceleration of the case (and the sled).
- The goal is to design an accelerometer with an appropriate dynamic responsiveness. We wish to design an accelerometer with an acceptable time for the desired measurement characteristic.







The sum of the forces acting on the mass is

$$-b\frac{dy}{dt} - ky = M\frac{d^2}{dt^2}(y + x)$$

or

$$M\frac{d^2y}{dt^2} + b\frac{dy}{dt} + ky = -M\frac{d^2x}{dt^2}.$$

Since

$$M_s\frac{d^2x}{dt^2} = F(t),$$

is the engine force, we have

$$M\ddot{y} + b\dot{y} + ky = -\frac{M}{M_s}F(t),$$

or

$$\ddot{y} + \frac{b}{M}\dot{y} + \frac{k}{M}y = -\frac{F(t)}{M_s}.$$

We select the coefficients where $b/M = 3$, $k/M = 2$, $F(t)/M_s = Q(t)$, and we consider the initial conditions $y(0) = -1$ and $\dot{y}(0) = 2$. We then obtain the Laplace transform equation, when the force, and thus $Q(t)$, is a step function, as follows:

$$(s^2Y(s) - sy(0) - \dot{y}(0)) + 3(sY(s) - y(0)) + 2Y(s) = -Q(s).$$

Since $Q(s) = P/s$, where P is the magnitude of the step function, we obtain

$$(s^2Y(s) + s - 2) + 3(sY(s) + 1) + 2Y(s) = -\frac{P}{s},$$

or

$$(s^2 + 3s + 2)Y(s) = \frac{-(s^2 + s + P)}{s}.$$

Thus the output transform is

$$Y(s) = \frac{-(s^2 + s + P)}{s(s^2 + 3s + 2)} = \frac{-(s^2 + s + P)}{s(s + 1)(s + 2)}.$$

Expanding in partial fraction form yields

$$Y(s) = \frac{k_1}{s} + \frac{k_2}{s + 1} + \frac{k_3}{s + 2}.$$

We then have

$$k_1 = \left. \frac{-(s^2 + s + P)}{(s+1)(s+2)} \right|_{s=0} = -\frac{P}{2}.$$

Similarly, $k_2 = +P$ and $k_3 = \dfrac{-P-2}{2}$. Thus,

$$Y(s) = \frac{-P}{2s} + \frac{P}{s+1} + \frac{-P-2}{2(s+2)}.$$

Therefore, the output measurement is

$$y(t) = \frac{1}{2}[-P + 2Pe^{-t} - (P+2)e^{-2t}], \quad t \geq 0.$$

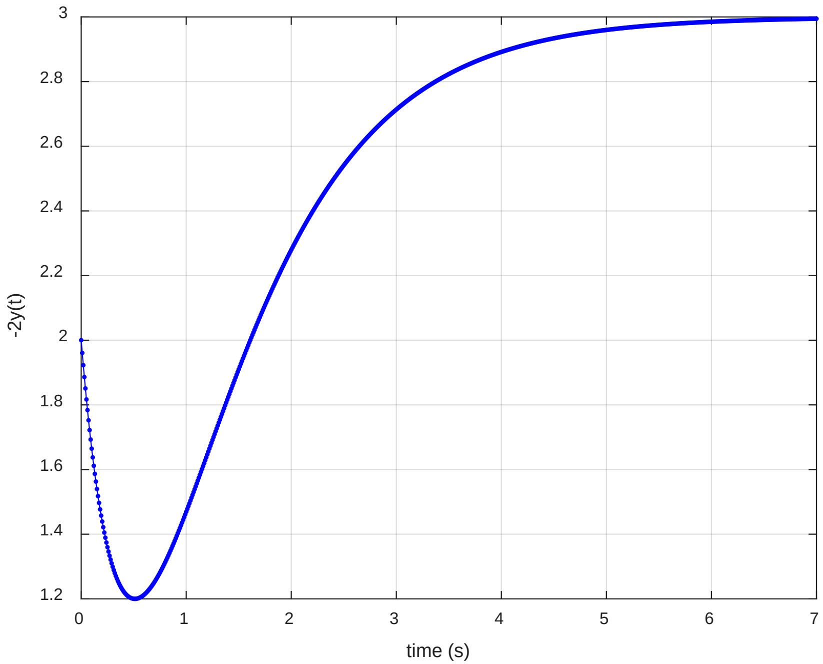
Time-domain Response Simulation by Matlab

Command Window

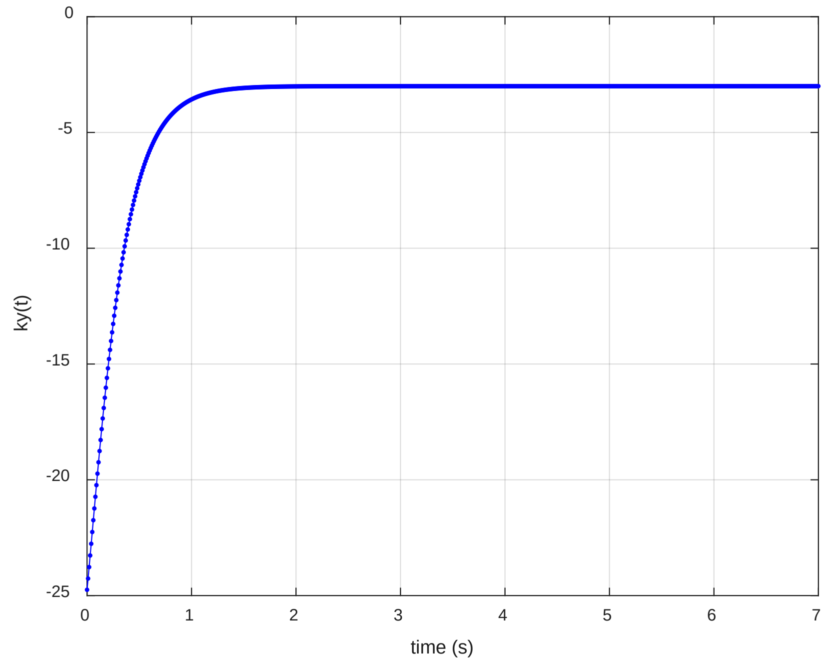
New to MATLAB? See resources for Getting Started.

```
>> t = 0:0.01:7;
>> y = 1/2*(-P+2*P*exp(-t)-(P+2)*exp(-2*t));
>> figure,plot(t,-2*y,'b.-'); grid on; xlabel('time (s)'); ylabel('-2y(t)');
>>
```

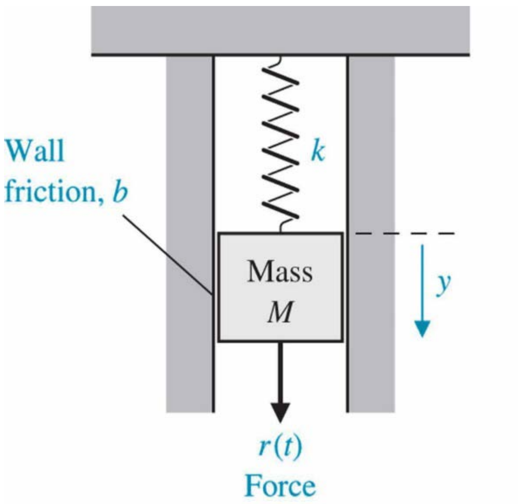
$b/M = 3, k/M = 2,$



$b/M = 12 \quad k/M = 32$



Simulation of System Using Matlab



$$Y(s) = \frac{(M\, s + b)\cdot y_o}{M\, s^2 + b\, s + k} = \frac{p(s)}{q(s)}$$

$$y(s) = \frac{\left(s + \frac{b}{M}\right)\cdot(y_o)}{\left[s^2 + \left(\frac{b}{M}\right)\cdot s + \frac{k}{M}\right]} = \frac{(s + 2\cdot\zeta\cdot\omega_n)}{s^2 + 2\cdot\zeta\cdot\omega_n s + \omega_n^2}$$

$$s1 = -(\zeta\cdot\omega_n) + \omega_n\cdot\sqrt{\zeta^2 - 1}$$

$$\omega_n = \sqrt{\frac{k}{M}} \qquad \zeta = \frac{b}{\left(2\cdot\sqrt{k\cdot M}\right)}$$

$$s2 = -(\zeta\cdot\omega_n) - \omega_n\cdot\sqrt{\zeta^2 - 1}$$

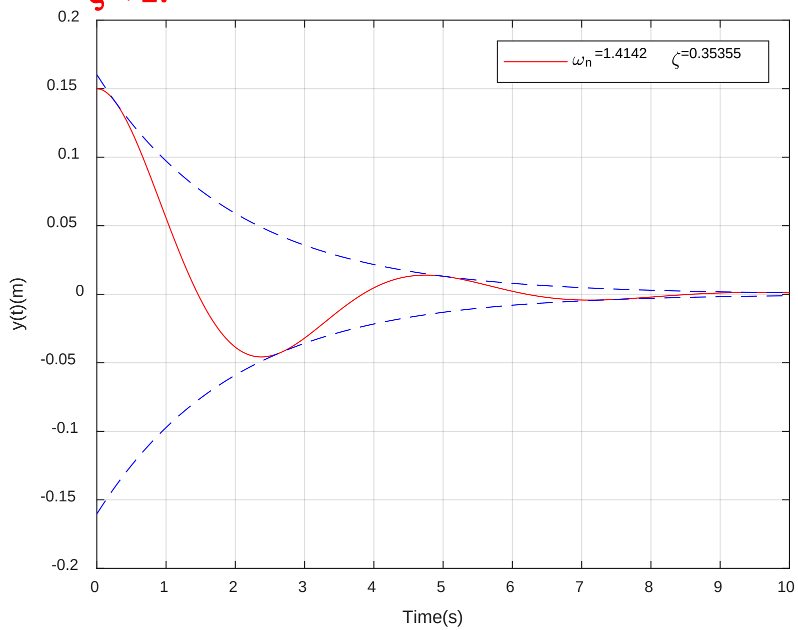
- $y(0) = 0.15\ \text{m}, \quad \omega_n = \sqrt{2}\dfrac{\text{rad}}{\text{sec}}, \quad \zeta = \dfrac{1}{2\sqrt{2}} \quad \left(\dfrac{k}{M} = 2, \dfrac{b}{M} = 1\right).$

Command Window

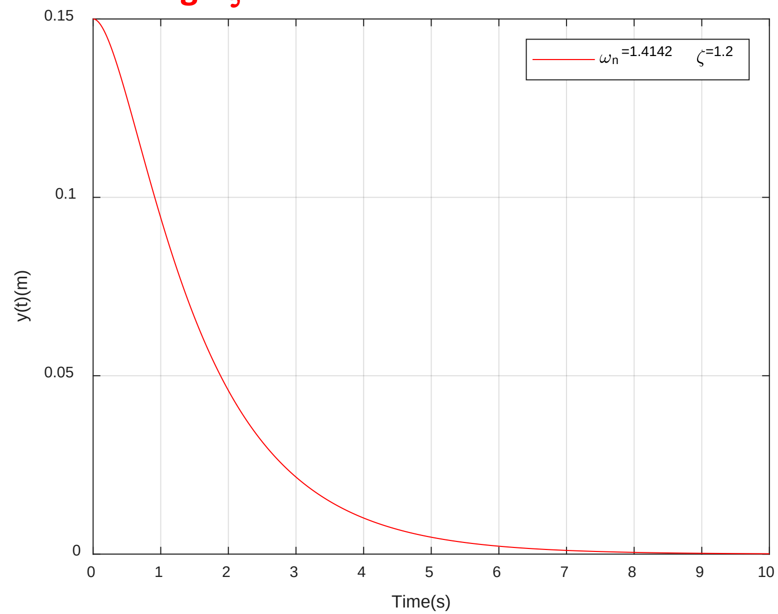
New to MATLAB? See resources for Getting Started.

```
>> y0=0.15;
>> wn=sqrt(2);
>> zeta=1/(2*sqrt(2));
>> t=[0:0.01:10];
>> c=(y0/sqrt(1-zeta^2));
>> y=c*exp(-zeta*wn*t).*sin(wn*sqrt(1-zeta^2)*t+acos(zeta));
>> bu=c*exp(-zeta*wn*t);bl=-bu;
>> figure,plot(t,y,'r-',t,bu,'b--',t,bl,'b--'),grid; xlabel('Time(s)'), ylabel('y(t)(m)');
```

ς < 1:



Change ς to be >1:



Useful functions in Control Toolbox

- Calculate roots for polynomial (**roots, poly**)

$$p(s) = s^3 + 3s^2 + 4$$

```
>> p=[1 3 0 4]

p =

     1     3     0     4

>> r = roots(p)

r =

  -3.3553 + 0.0000i
   0.1777 + 1.0773i
   0.1777 - 1.0773i

>> p = poly(r)

p =

    1.0000    3.0000   -0.0000    4.0000
```

- Multiply and evaluate polynomial (**conv, polyval**)

Using **conv** and **polyval** to multiply and evaluate the polynomials $(3s^2 + 2s + 1)(s + 4)$.

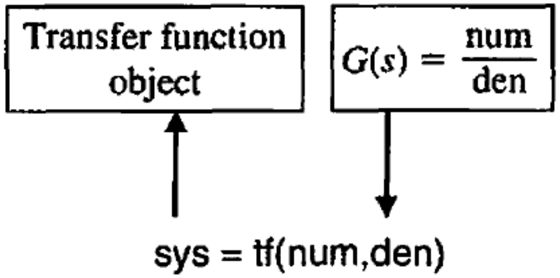
```
>>p=[3 2 1]; q=[1 4];
>>n=conv(p,q)
n=
  3   14   9   4

>>value=polyval(n,-5)
value =

  -66
```

Multiply p and q.

$n(s) = 3s^3 + 14s^2 + 9s + 4$

Evaluate $n(s)$ at $s = -5$.

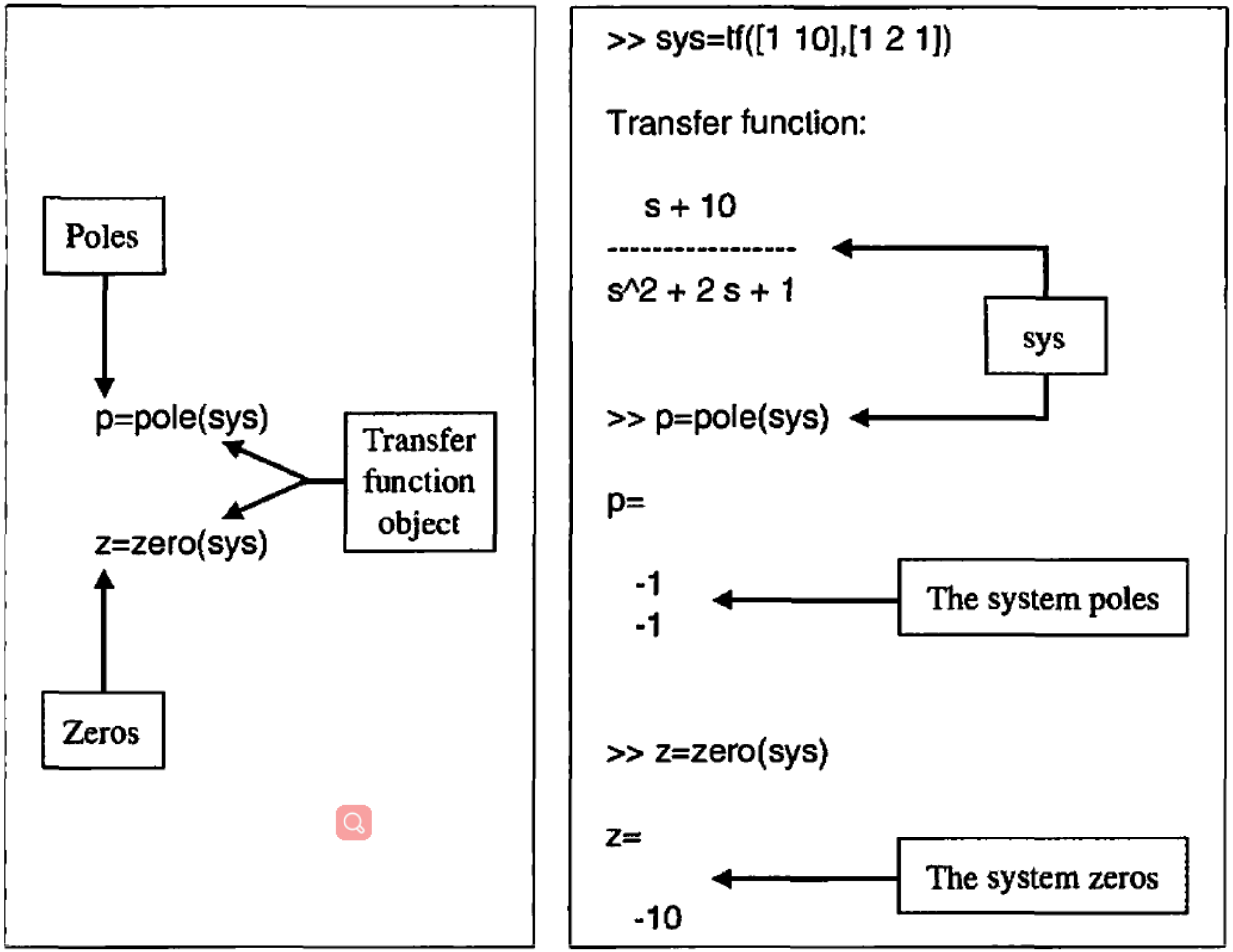
- Create transfer functions and add transfer functions (**tf**)




```
>> num1 = [10]; den1=[1 2 5];
>> sys1 = tf(num1, den1)

sys1 =

       10
  -------------
  s^2 + 2 s + 5

Continuous-time transfer function.

>> sys2 = tf([1], [1 1])

sys2 =

    1
  -----
  s + 1

Continuous-time transfer function.
```

```
>> sys=sys1+sys2

sys =

      s^2 + 12 s + 15
  ---------------------
  s^3 + 3 s^2 + 7 s + 5

Continuous-time transfer function.
```

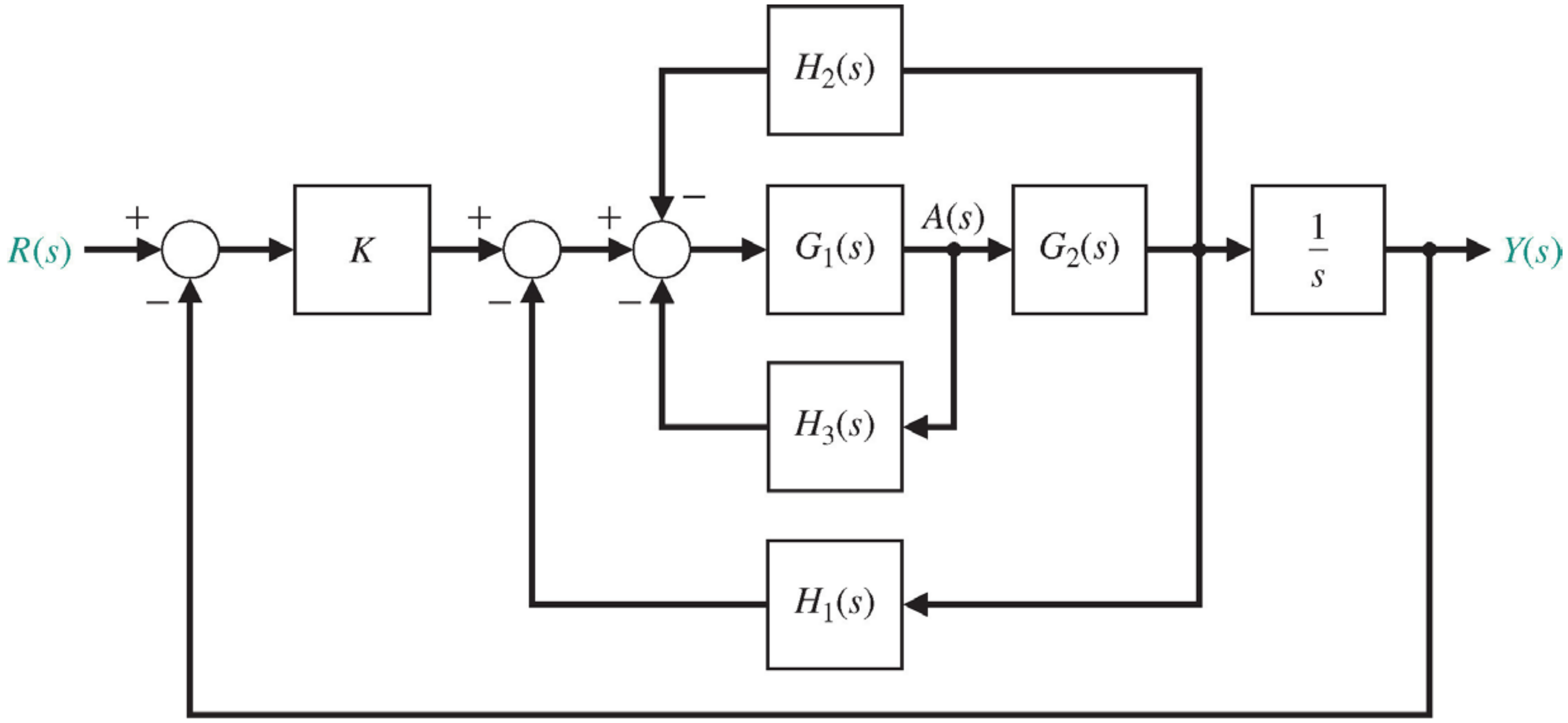
o Compute pole and zero locations of a linear system (**pole, zero**)



- Pole-zero map (**pzmap**);
- Block diagram (**series, parallel**);
- Feedbak system (**feedback**)
- Time-domain response (**step, impulse**);
- …. (Please refer to the manual of control system toolbox of Matlab)

Quiz 10.1

Determine the transfer function for the following system:



Thank You !