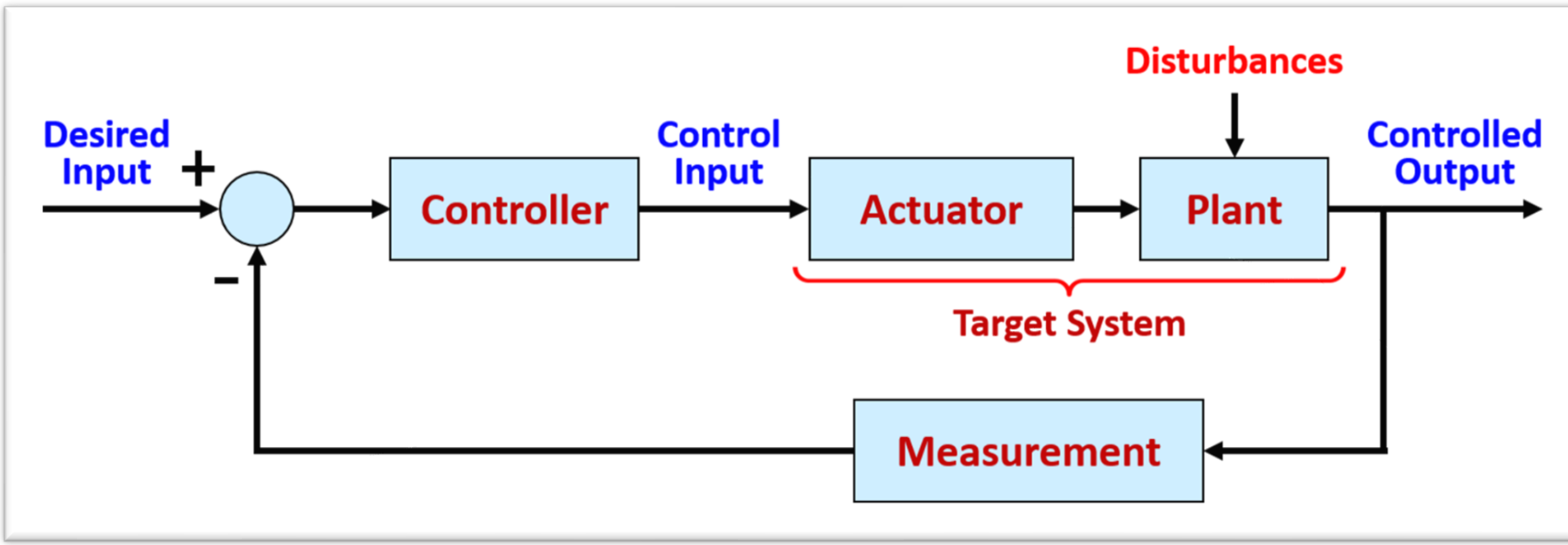
Dynamic Response 2

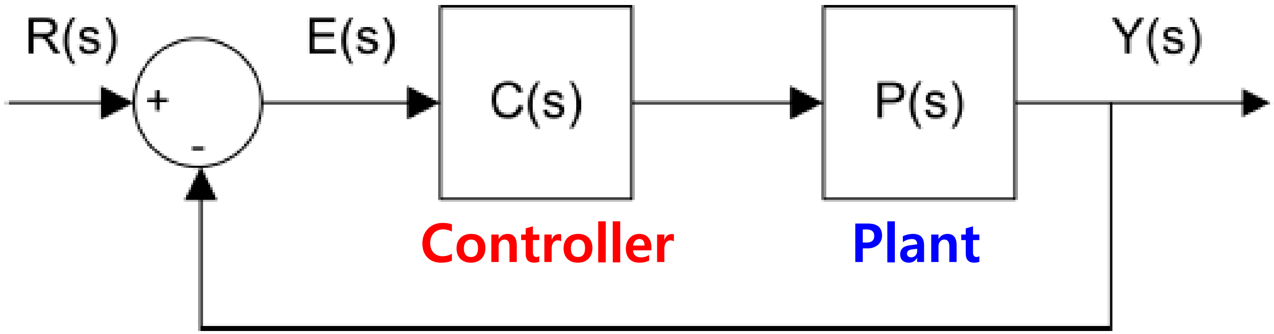
Lecture 5:

- **Pole Placement Design Method**
- Stability



Prof. Seunghoon Woo
Department of Automotive Engineering | College of Automotive Engineering
KOOKMIN UNIVERSITY

Pole Placement Method

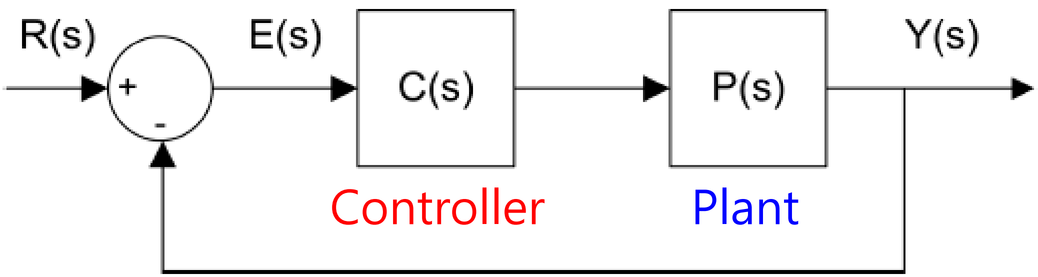


- **Manual Tuning** → ✓ Time-consuming effort !!
- **Automatic Tuning Method** → ✓ Indirect PID tuning ✓ Still needs fine-tuning effort
- **Now, Pole Placement Method!!**

→ Based on the **pole locations**, the **desired characteristic equation of the closed-loop** can be produced by controller.

Pole Placement Method (cont'd)

- **Case 1: First-order System + PI control**



$$C(s)_{PI} = k_p + \frac{k_i}{s}$$

$$P(s) = \frac{b}{s+a}$$

- The **actual** characteristic eq. of closed-loop system (PI-control)

$$\frac{Y(s)}{R(s)} = \frac{CP}{1+CP} = \frac{\dfrac{k_p s + k_i}{s}\,\dfrac{b}{s+a}}{1 + \dfrac{k_p s + k_i}{s}\,\dfrac{b}{s+a}} = \frac{k_p b s + k_i b}{\boxed{s^2 + (a + k_p b)s + k_i b}}$$

- The **desired** characteristic eq. of closed-loop system

$$T(s) = \frac{\omega_n^{\ 2}}{\boxed{s^2 + 2\varsigma\omega_n s + \omega_n^{\ 2}}}$$

Pole Placement Method (cont'd)

- **Case 1: First-order System + PI control (cont'd)**
  - Step 0: Model matching condition

$$\underbrace{s^2 + (a + k_p b)s + k_i b}_{\text{Actual C-L system}} = \underbrace{s^2 + 2\varsigma\omega_n s + {\omega_n}^2}_{\text{Desired C-L system}}$$

  - Step response requirement can be satisfied by PI-control gains.

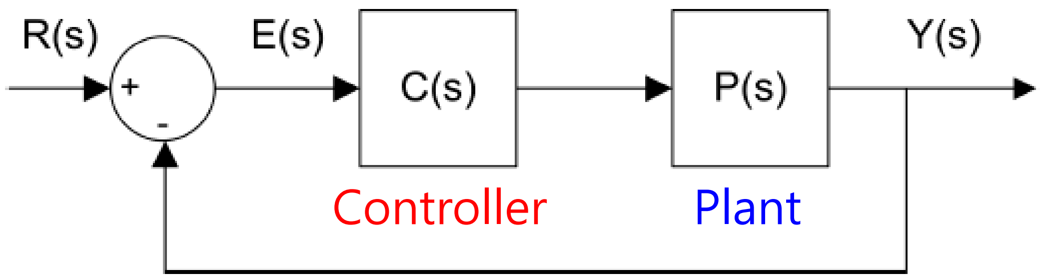
(1) Overshoot $\quad M_p = 1 + \exp\left(-\zeta\omega_n T_p\right) = 1 + \exp\left(-\dfrac{\pi\zeta}{\sqrt{1-\zeta^2}}\right)$

(2) Rising time $\quad T_r = \dfrac{\pi - \varphi}{\omega_d} \quad where,\ \omega_d = \omega_n\sqrt{1 - \varsigma^2},\ \cos\varphi = \varsigma$

(3) Settling time $\quad T_s \approx \dfrac{4.6}{\varsigma\omega_n} = \dfrac{4.6}{\sigma}$

Pole Placement Method (cont'd)

Case 1: First-order System + PI control (cont'd)



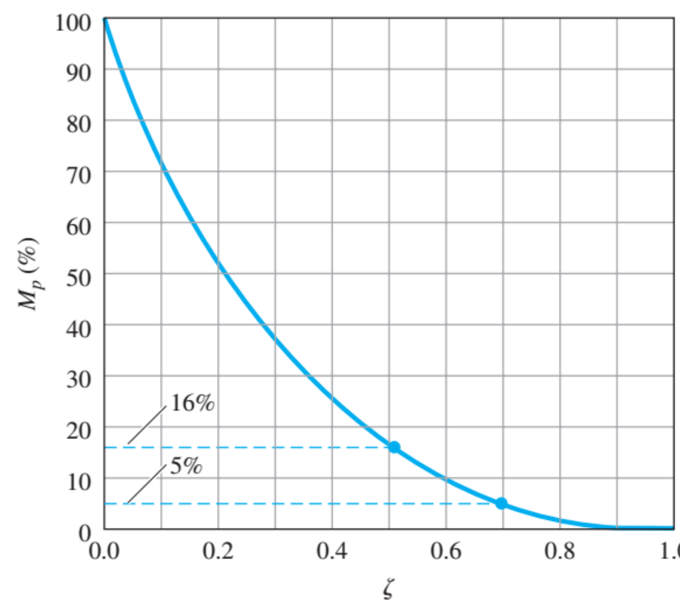
$$C(s)_{PI} = k_p + \frac{k_i}{s}$$

$$P(s) = \frac{b}{s+a}$$

- Step 1: step response requirement

(1) Overshoot = 10%

$$M_p = \exp\left(-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}\right) \quad \Rightarrow \quad \zeta \approx 0.6$$



(2) Rising time = 0.1 sec

$$T_r = \frac{\pi - \varphi}{\omega_n\sqrt{1-\varsigma^2}}$$

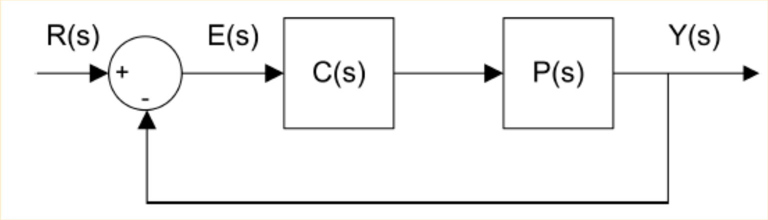
$$where, \cos\varphi = \varsigma$$

$$\omega_n = \frac{3.14 - \cos^{-1}(0.6)}{0.1 \times \sqrt{1-0.6^2}} = 27.7\,[\mathrm{rad/s}]$$

Pole Placement Method (cont'd)

- **Case 1: First-order System + PI control (cont'd)**
  - Step 2: Check desired pole locations

$$\left.\begin{aligned} & s^2 + 2\varsigma\omega_n s + \omega_n^2 = 0 \\ & \Rightarrow s^2 + 2(0.6)(27.7)s + 27.7^2 = 0 \end{aligned}\right\} \quad s_{1,2} = -16.6 \pm 22.1j$$



$$C(s)_{PI} = k_p + \frac{k_i}{s} \qquad P(s) = \frac{b}{s+a}$$

  - Step 3: Calculate PI gains
    based on model matching

$$s^2 + \underline{(a + k_p b)}s + k_i b$$

$$= s^2 + \underline{2\varsigma\omega_n}s + \omega_n^2$$

$$\left.\begin{aligned} & s^1 \rightarrow k_p = \frac{1}{b}(2\zeta\omega_n - a) \\ & s^0 \rightarrow k_i = \frac{\omega_n^2}{b} \end{aligned}\right\}$$

$a = 30$

$b = 30$

$\zeta \approx 0.6$

$\omega_n = 27.7\,[\text{rad/s}]$

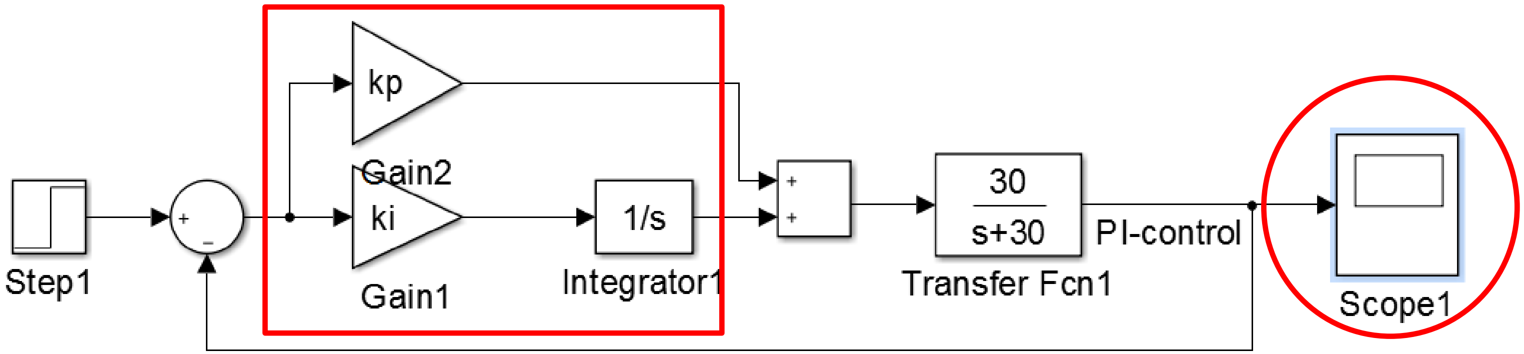
$\Rightarrow$

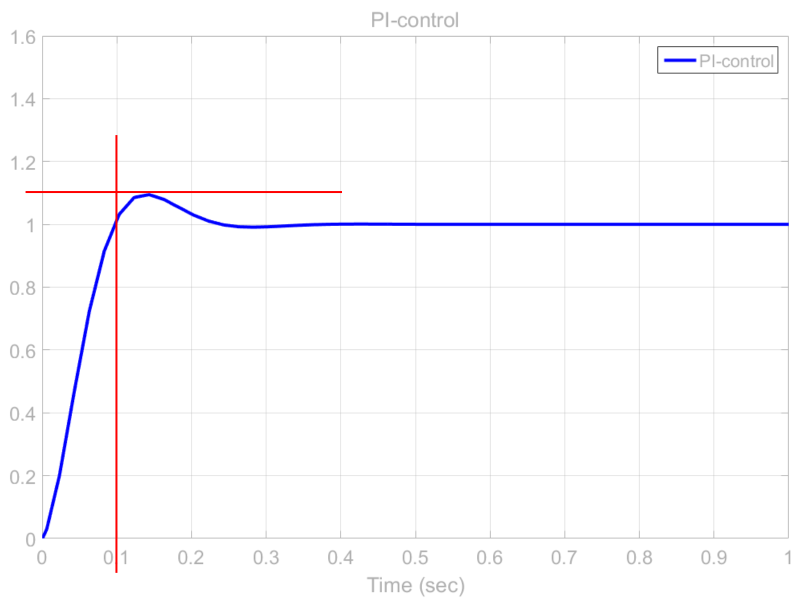
$k_p = 0.1$

$k_i = 25.5$

Pole Placement Method (cont'd)

- **Case 1: First-order System + PI control (cont'd)**
  - Step 4: Simulation model & result



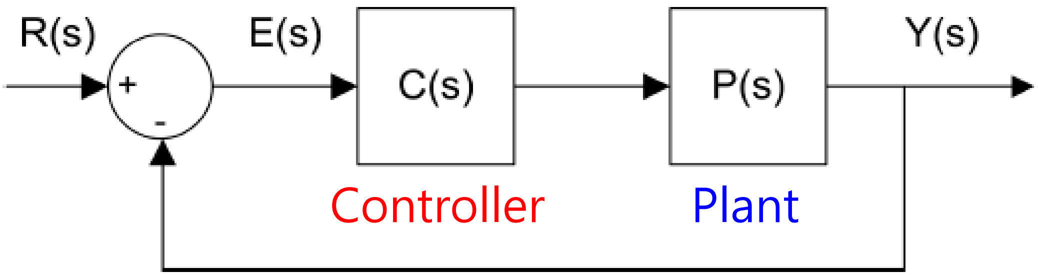


Rising time = 0.1 sec

Overshoot = 10 %

Pole Placement Method (cont'd)

- **Case 2: Second-order System + PID control**



$$C(s)_{PID} = k_p + \frac{k_i}{s} + k_d s$$

$$P(s) = \frac{b}{s(s+a)}$$

- The **actual** characteristic eq. of closed-loop system (PID-control)

$$\frac{Y(s)}{R(s)} = \frac{CP}{1+CP} = \frac{\dfrac{k_d s^2 + k_p s + k_i}{s}\,\dfrac{b}{s^2 + as}}{1 + \dfrac{k_d s^2 + k_p s + k_i}{s}\,\dfrac{b}{s^2 + as}} = \frac{b(k_d s^2 + k_p s + k_i)}{\boxed{s^3 + (bk_d + a)s^2 + (bk_p)s + bk_i}}$$

- The **desired** characteristic eq. of closed-loop system

$$T(s) = \frac{\omega_n^2}{\boxed{s^2 + 2\varsigma\omega_n s + \omega_n^2}}$$

Different order !!. How ??

Pole Placement Method (cont'd)

- **Case 2: Second-order System + PID control (cont'd)**
  - 2nd-order desired C-L system vs. 3rd-order controlled C-L system
  - So, we need a real pole to make 3rd-order desired C-L system !!

Faster (decay) pole

➡ Minor pole

vs.

Slower (decay) pole

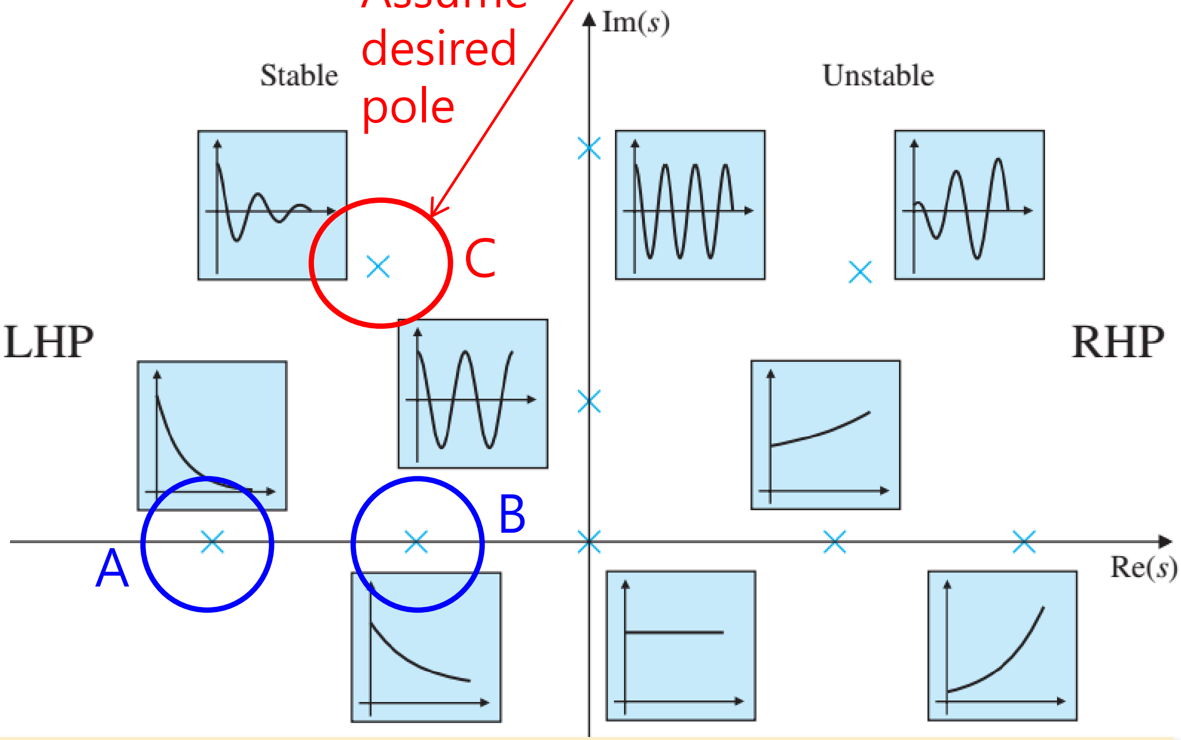
➡ Dominant pole

Slower → Faster:

B → C → A

$$s^2 + 2\varsigma\omega_n s + \omega_n^2 = 0$$

Assume desired pole



Thus, real pole A (faster) does NOT longer affect to desired pole C !!

Pole Placement Method (cont'd)

Case 2: Second-order System + PID control (cont'd)

- Step 0: Model matching condition with a minor (faster) pole

$$\underbrace{s^3 + (bk_d + a)s^2 + (bk_p)s + bk_i}_{\text{Actual C-L system}} = \underbrace{\overbrace{(s+p)}^{\text{minor}}\overbrace{(s^2 + 2\varsigma\omega_n s + \omega_n^2)}^{\text{dominant}}}_{\text{Desired C-L system}}$$

$$s_{1,2} = -5.75 \pm 11.4j$$

$$= s^3 + \{(p) + 2\varsigma\omega_n\}s^2 + \{2\varsigma\omega_n(p) + \omega_n^2\}s + \omega_n^2(p)$$

$s^2 \rightarrow k_d = \frac{1}{b}\{(p) + 2\varsigma\omega_n - a\}$

$s^1 \rightarrow k_p = \frac{1}{b}\{2\varsigma\omega_n(p) + \omega_n^2\}$

$s^0 \rightarrow k_i = \frac{1}{b}\{\omega_n^2(p)\}$

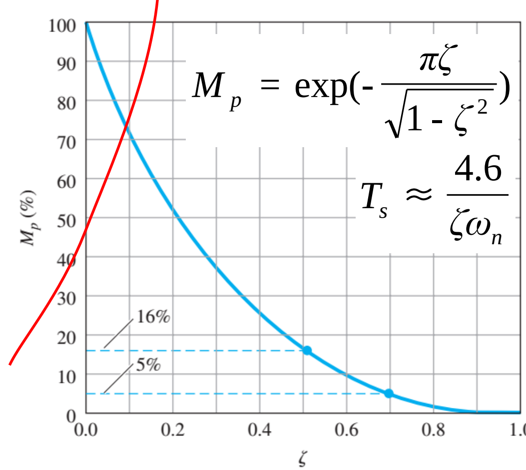
$a = 10$

$b = 10$

$p = 5.75$

$$\boxed{k_d = 0.73 \quad k_p = 22.9 \quad k_i = 93.9}$$

- Step 1: step response requirement
  - (1) Overshoot = 20% $\rightarrow$ $\zeta \approx 0.45$
  - (2) Settling Time = 0.8 sec $\rightarrow$ $\omega_n = 12.8\,[\text{rad/s}]$



Pole Placement Method (cont'd)

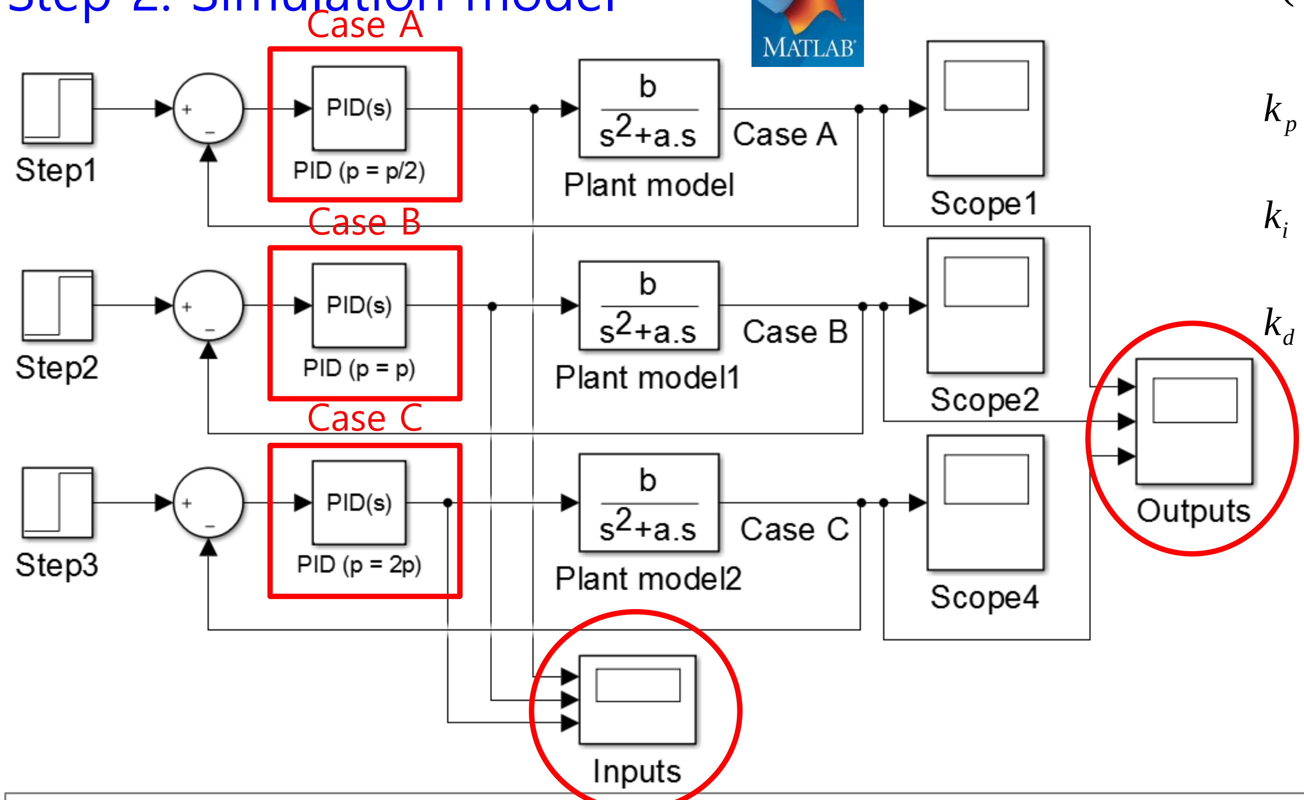
- **Case 2: Second-order System + PID control (cont'd)**
  - Step 2: Simulation model

$$C(s)_{PID} = k_p + \frac{k_i}{s} + k_d s$$

$$k_p = \frac{1}{b}\{2\varsigma\omega_n(p) + \omega_n{}^2\}$$

$$k_i = \frac{1}{b}\{\omega_n{}^2(p)\}$$

$$k_d = \frac{1}{b}\{(p) + 2\varsigma\omega_n - a\}$$



- Case A: p = 5.75 / 2 (smaller dominant pole)
- Case B: p = 5.75 (near dominant pole)
- Case C: p = 5.75 x 2 (two times far from dominant pole)

Pole Placement Method (cont'd)

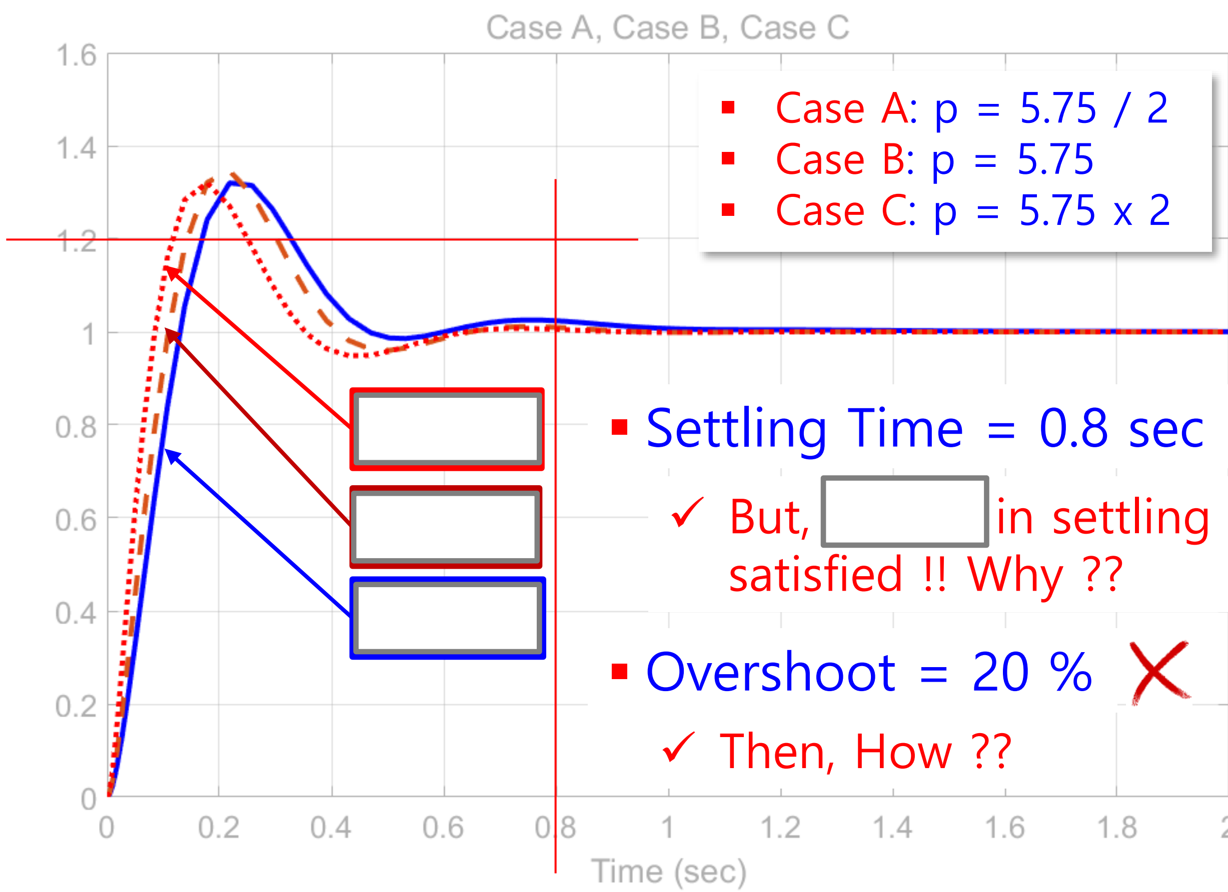
- **Case 2: Second-order System + PID control (cont'd)**
  - Step 3: Simulation result (1) : Outputs

$$C(s)_{PID} = k_p + \frac{k_i}{s} + k_d s$$

$$k_p = \frac{1}{b}\{2\varsigma\omega_n(p) + \omega_n^{\,2}\}$$

$$k_i = \frac{1}{b}\{\omega_n^{\,2}(p)\}$$

$$k_d = \frac{1}{b}\{(p) + 2\varsigma\omega_n - a\}$$



- Settling Time = 0.8 sec ✓
  - ✓ But, $\boxed{}$ in settling time is NOT satisfied !! Why ??
- Overshoot = 20 % ✗
  - ✓ Then, How ??

Pole Placement Method (cont'd)

- **Case 2: Second-order System + PID control (cont'd)**
  - Step 3: Simulation result (1) : Outputs (cont'd)

$$C(s)_{PID} = k_p + \frac{k_i}{s} + k_d s$$

  - Which controller contributes to reduce overshoot ??

$$k_p = \frac{1}{b}\{2\varsigma\omega_n(p) + \omega_n{}^2\}$$

$$k_i = \frac{1}{b}\{\omega_n{}^2(p)\}$$

$$\boxed{k_d = \frac{1}{b}\{(p) + 2\varsigma\omega_n - a\} \times \beta}$$

$\Rightarrow$

$\beta > 1$: increasing damping

$\beta < 1$: decreasing damping

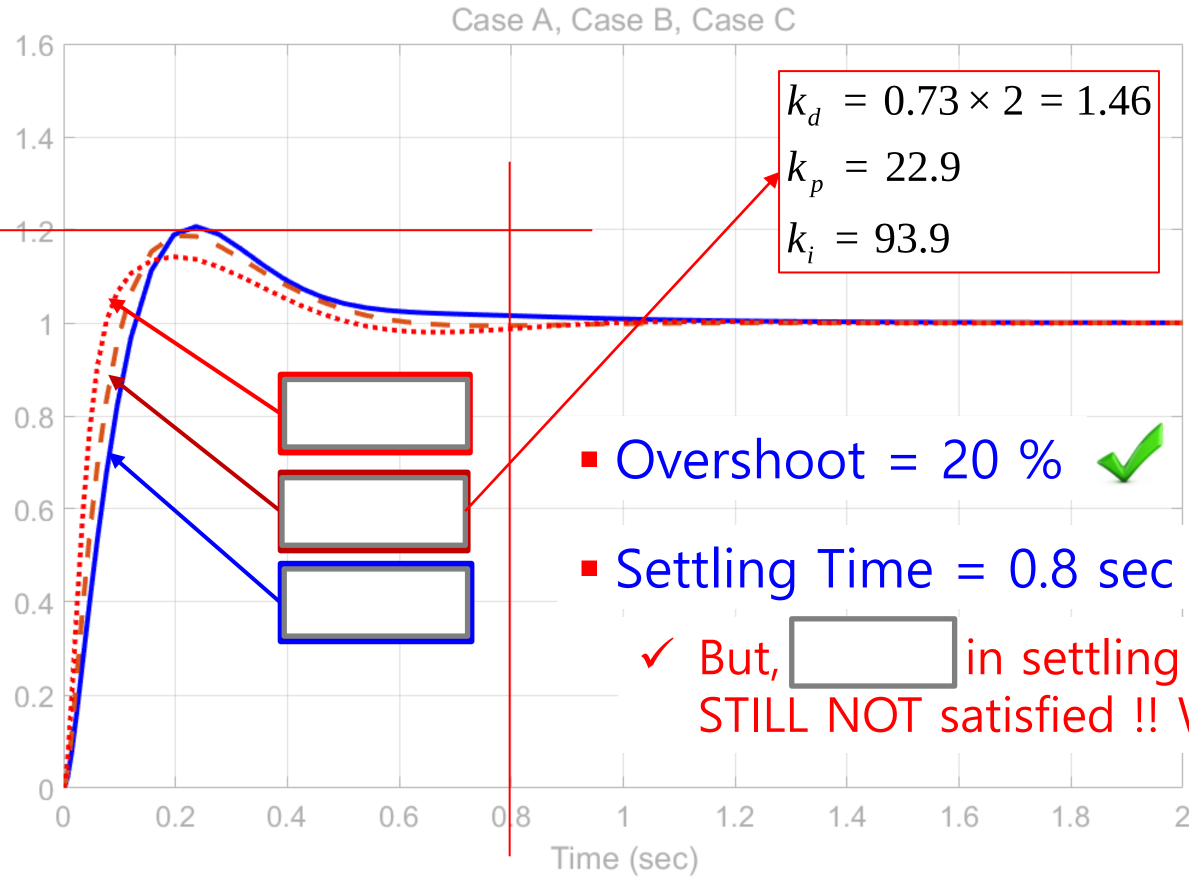
where,

β : weighting factor

Pole Placement Method (cont'd)

- **Case 2: Second-order System + PID control (cont'd)**
  - Step 3: Simulation result (1) : Outputs

$$C(s)_{PID} = k_p + \frac{k_i}{s} + k_d s$$



$$k_d = 0.73 \times 2 = 1.46$$
$$k_p = 22.9$$
$$k_i = 93.9$$

$$k_p = \frac{1}{b}\{2\varsigma\omega_n(p) + \omega_n^2\}$$

$$k_i = \frac{1}{b}\{\omega_n^2(p)\}$$

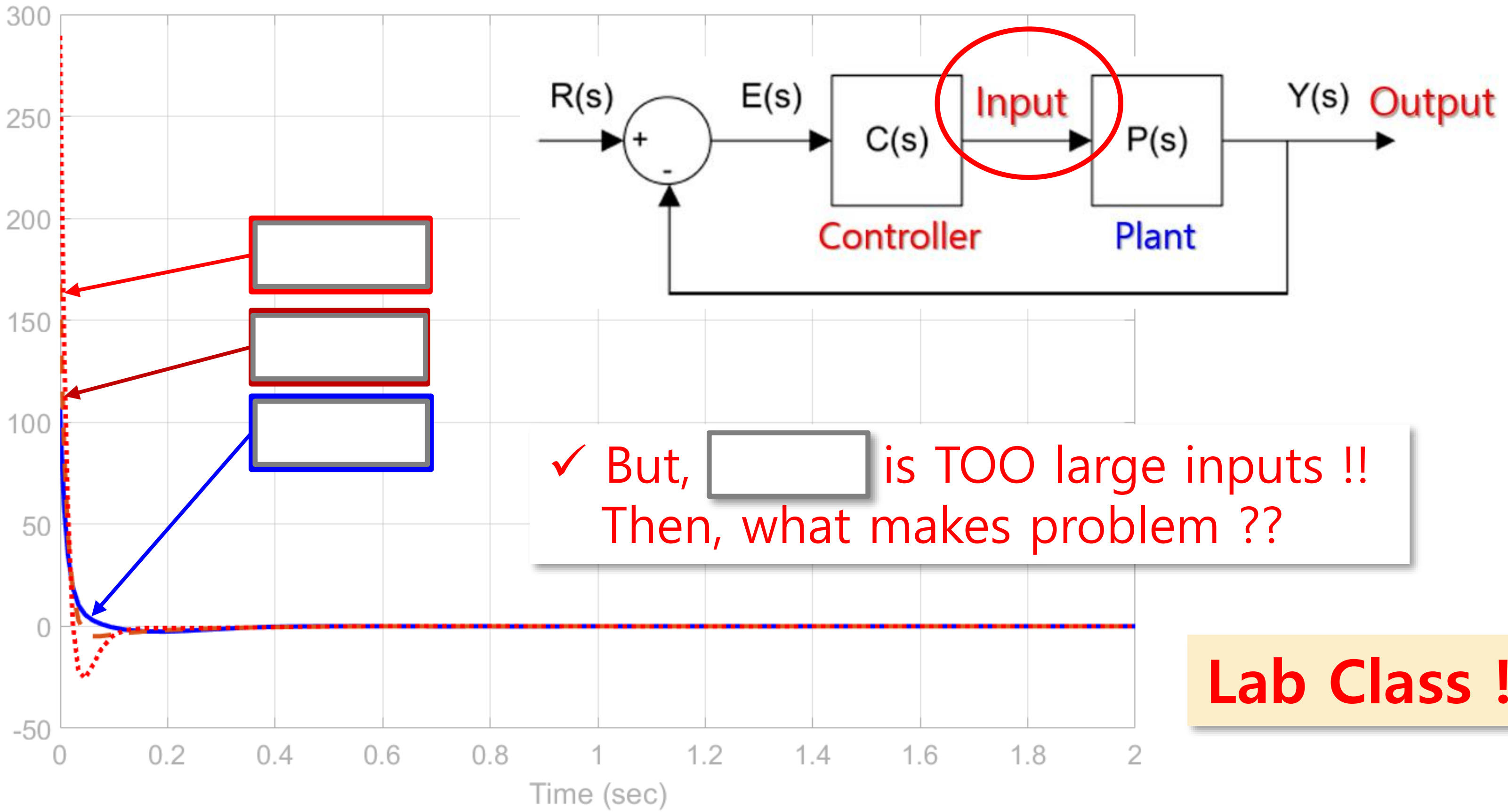
$$k_d = \frac{1}{b}\{(p) + 2\varsigma\omega_n - a\} \times \beta$$

$$\boxed{\beta = 2}$$

- Overshoot = 20 % ✓
- Settling Time = 0.8 sec ✓
  - ✓ But, [ ] in settling time is STILL NOT satisfied !! Why ??

Pole Placement Method (cont'd)

- **Case 2: Second-order System + PID control (cont'd)**
  - Step 3: Simulation result (2) : Inputs



✓ But, [ ] is TOO large inputs !!
Then, what makes problem ??

Lab Class !!

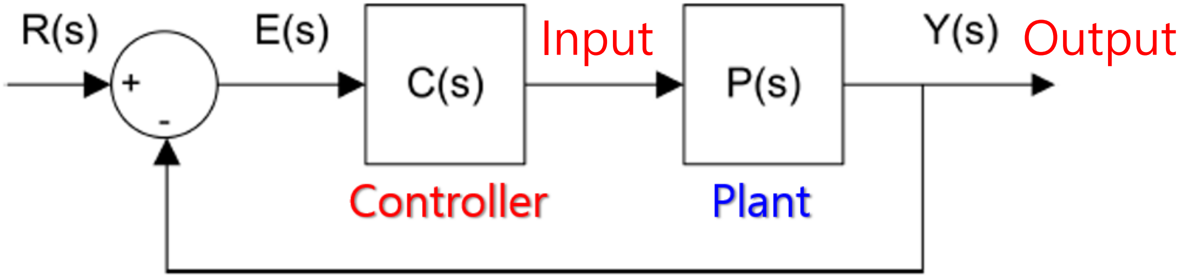
Pole Placement Method (cont'd)

- **Case 2: Second-order System + PID control (cont'd)**
  - Step 3: Simulation result (2) : Inputs (cont'd)

Lab Class !!

For example, DC-motor position control

$$P(s) = \frac{\Theta(s)}{V_a(s)} = \frac{Output}{Input} = \frac{Position}{Voltage} = \frac{K}{s(\tau s + 1)}$$

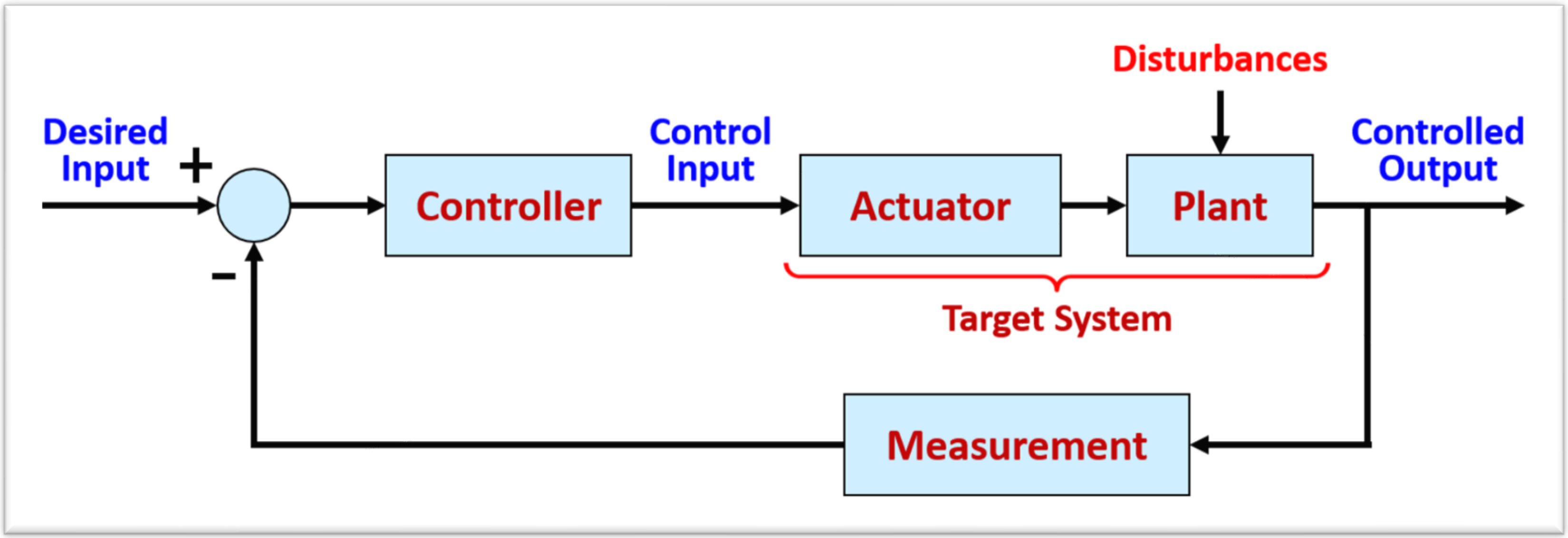


Thus, we should check whether the input capability (or limitation) to perform the controlled output !!

Dynamic Response 2

Lecture 5:

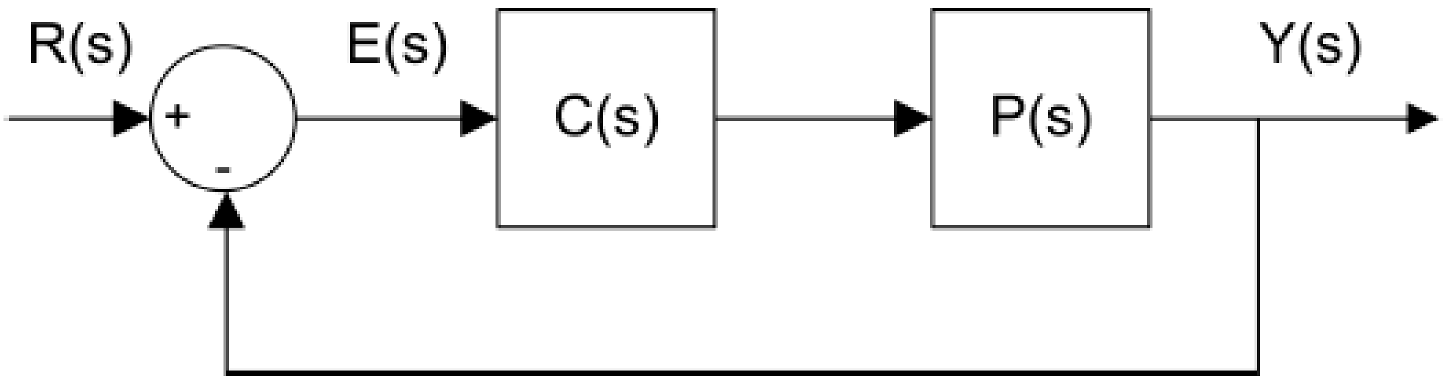
- Pole Placement Design Method
- Stability



Prof. Seunghoon Woo
Department of Automotive Engineering | College of Automotive Engineering
KOOKMIN UNIVERSITY

What is the Stability ??

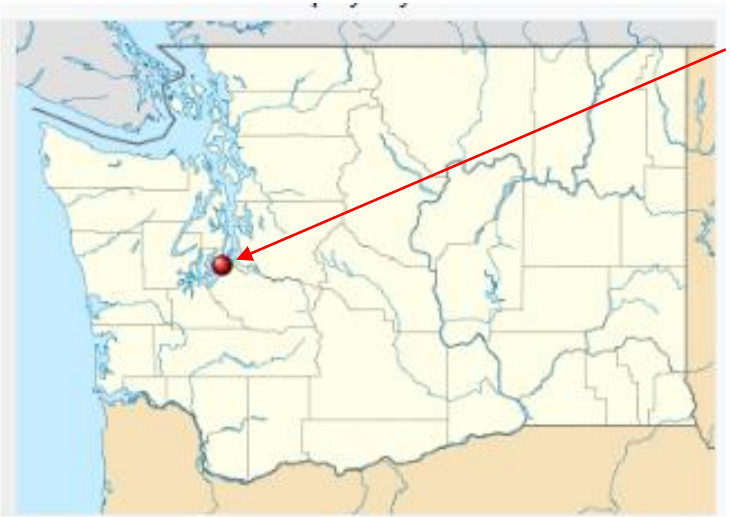
- Utmost important specification in control design !!
- Unstable systems have to be stabilized by feedback.
- Unstable closed-loop systems are useless !!
- What happens if a system is unstable?
  - may hit mechanical/electrical/biological/chemical "stops" (handling with saturation scheme)
  - may break down or burn out



What happened if a system is unstable?

- Tacoma Narrows Bridge (July 1-Nov.7, 1940)

  https://www.youtube.com/watch?v=XggxeuFDaDU

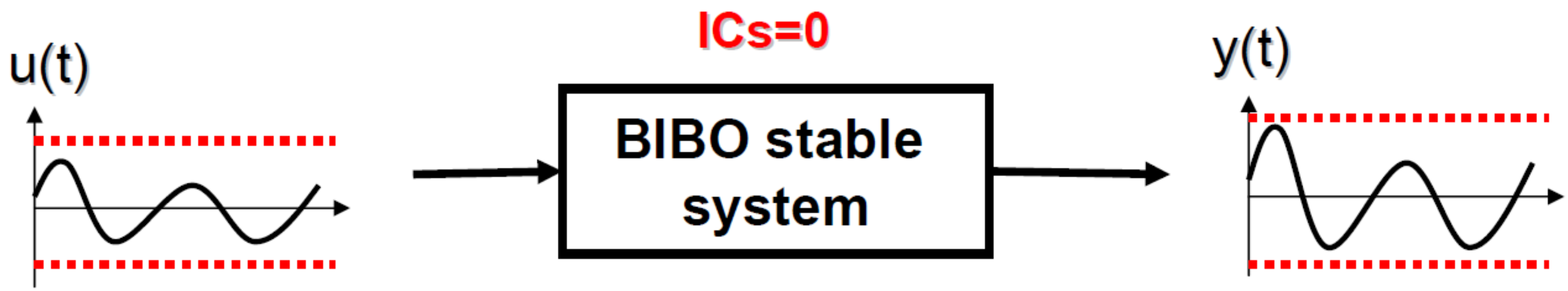


Tacoma city @ Washington State, USA

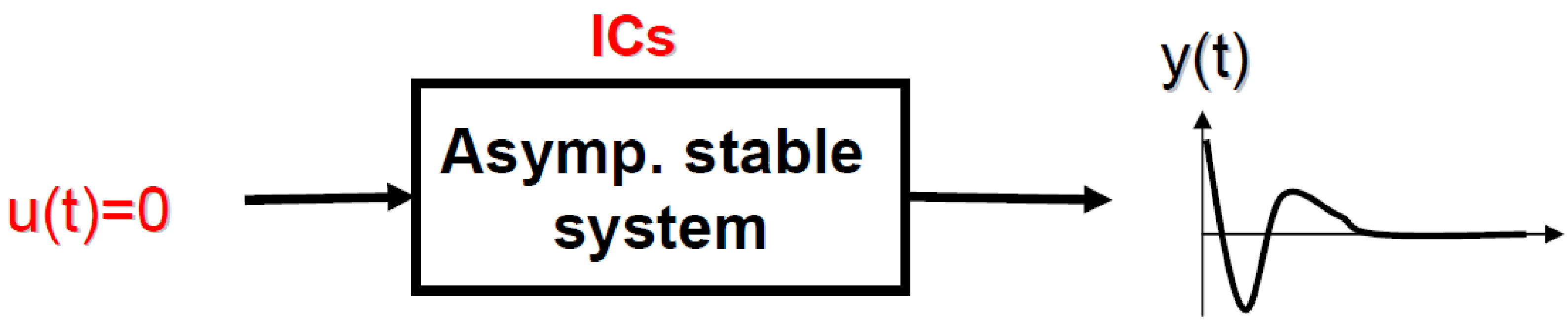
2008…

Mathematical Definitions of Stability

- BIBO (Bounded-Input-Bounded-Output) stability:
  Any bounded input generates a bounded output.



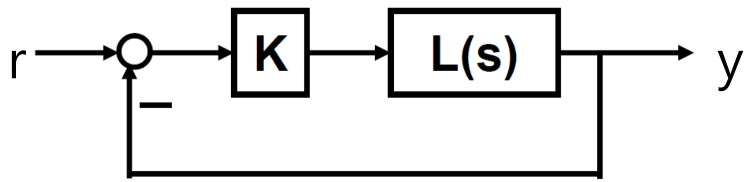
- Asymptotic stability:
  Any ICs generates y(t) converging to zero.



Stability Summary

Let s_i be poles of rational G(s)

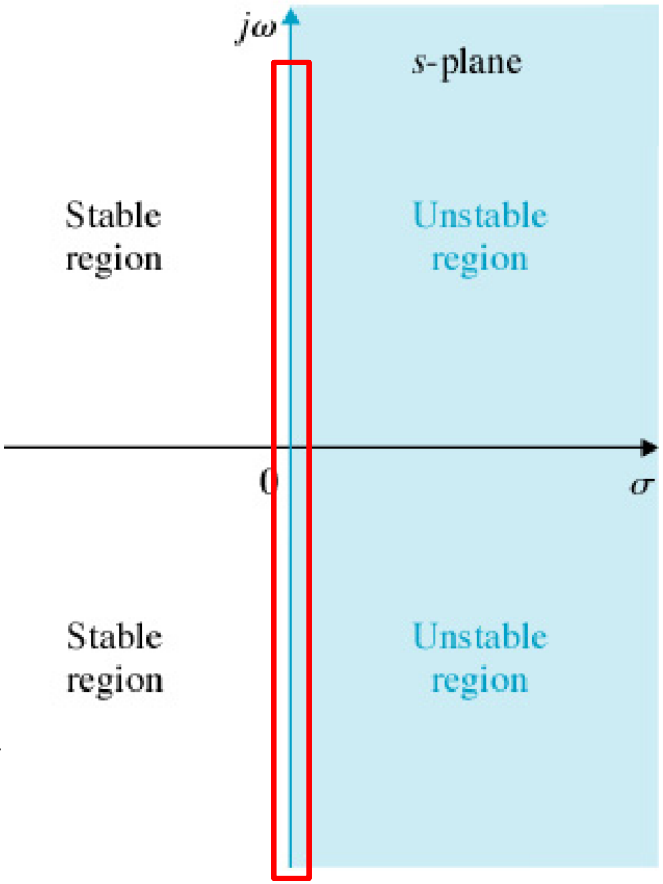
$$G(s) = 1 + KL(s)$$



- (BIBO, asymptotically) stable if

  $\mathrm{Re}(s_i)<0$ for all i.

- Marginally stable if
  - $\mathrm{Re}(s_i)<=0$ for all i, and
  - simple root for $\mathrm{Re}(s_i)=0$
- Unstable if
  - it is neither stable nor marginally stable.



Routh-Hurwitz Criterion

- This is for LTI systems with a polynomial denominator (without sin, cos, exponential etc.)
- It determines if all the roots of a polynomial-
  - lie in the LHP (left half-plane),
  - or equivalently, have negative real parts.
- It also determines the number of roots of a polynomial in the RHP (right half-plane).
- But, it does NOT explicitly compute the roots.

Edward Routh

Edward John Routh (1831–1907)

Born 20 January 1831
Quebec, Canada

Died 7 June 1907 (aged 76)
Cambridge, England

Adolf Hurwitz

Born 26 March 1859
Hildesheim, Kingdom of Hanover (now part of Germany)

Died 18 November 1919 (aged 60)
Zürich, Switzerland

Polynomial and an Assumption

- Consider a polynomial of characteristic eq.

$$Q(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0$$

- Assume $a_0 \neq 0$
  - If this assumption does not hold, Q can be factored as

$$Q(s) = s^m \underbrace{(\hat{a}_{n-m} s^{n-m} + \cdots + \hat{a}_1 s + \hat{a}_0)}_{\hat{Q}(s)}$$

where $\hat{a}_0 \neq 0$

  - The following method applies to the polynomial $\hat{Q}(s)$

Routh Array

$$Q(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0$$

$$
\begin{array}{rr|lllll}
\text{Highest} \rightarrow & s^n & a_n & a_{n-2} & a_{n-4} & a_{n-6} & \cdots \\
\text{2nd Highest} \rightarrow & s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} & a_{n-7} & \cdots \\
 & s^{n-2} & b_1 & b_2 & b_3 & b_4 & \cdots \\
 & s^{n-3} & c_1 & c_2 & c_3 & c_4 & \cdots \\
 & \vdots & \vdots & \vdots & & & \\
 & s^2 & k_1 & k_2 & & & \\
 & s^1 & l_1 & & & & \\
 & s^0 & m_1 & & & &
\end{array}
$$

From the given polynomial

Routh Array (cont'd)

- How to compute the third row

$$
\begin{array}{c|ccccc}
s^n & a_n & a_{n-2} & a_{n-4} & a_{n-6} & \cdots \\
s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} & a_{n-7} & \cdots \\
s^{n-2} & b_1 & b_2 & b_3 & b_4 & \cdots \\
s^{n-3} & c_1 & c_2 & c_3 & c_4 & \cdots \\
\vdots & \vdots & \vdots & & & \\
s^2 & k_1 & k_2 & & & \\
s^1 & l_1 & & & & \\
s^0 & m_1 & & & &
\end{array}
$$

$$
b_1 = \frac{a_{n-2}a_{n-1} - a_n a_{n-3}}{a_{n-1}}
$$

$$
b_2 = \frac{a_{n-4}a_{n-1} - a_n a_{n-5}}{a_{n-1}}
$$

$$
\vdots
$$

Routh Array (cont'd)

- How to compute the forth row

$$
\begin{array}{l|lllll}
s^n & a_n & a_{n-2} & a_{n-4} & a_{n-6} & \cdots \\
s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} & a_{n-7} & \cdots \\
s^{n-2} & b_1 & b_2 & b_3 & b_4 & \cdots \\
s^{n-3} & c_1 & c_2 & c_3 & c_4 & \cdots \\
\vdots & \vdots & \vdots & & & \\
s^2 & k_1 & k_2 & & & \\
s^1 & l_1 & & & & \\
s^0 & m_1 & & & &
\end{array}
$$

$$
\begin{aligned}
c_1 &= \frac{a_{n-3}b_1 - a_{n-1}b_2}{b_1} \\
c_2 &= \frac{a_{n-5}b_1 - a_{n-1}b_3}{b_1} \\
&\vdots
\end{aligned}
$$

Routh-Hurwitz Criterion

$$
\begin{array}{l|lllll}
s^n & a_n & a_{n-2} & a_{n-4} & a_{n-6} & \cdots \\
s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} & a_{n-7} & \cdots \\
s^{n-2} & b_1 & b_2 & b_3 & b_4 & \cdots \\
s^{n-3} & c_1 & c_2 & c_3 & c_4 & \cdots \\
\vdots & \vdots & \vdots & & & \\
s^2 & k_1 & k_2 & & & \\
s^1 & l_1 & & & & \\
s^0 & m_1 & & & &
\end{array}
$$

The number of roots
in the open right half-plane
is equal to
the number of sign changes
in the first column of Routh array.

Example 1: Simple Polynomial

$$Q(s) = s^3 + s^2 + 2s + 8 \ (= (s+2)\underline{(s^2 - s + 4)})$$

Routh array

$$\begin{array}{c|cc} s^3 & 1 & 2 \\ s^2 & 1 & 8 \\ s^1 & -6 & \\ s^0 & 8 & \end{array}$$

$$\frac{2-8}{1}$$

$$\frac{8 \times (-6) - 0}{-6}$$

Two sign changes

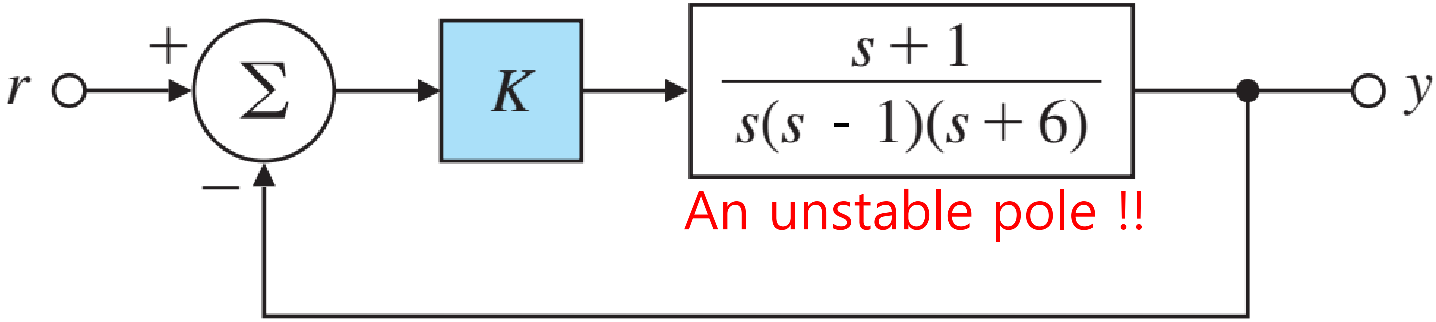
in the first column

$1 \rightarrow -6 \rightarrow 8$

Two roots in RHP

$$\frac{1}{2} \pm \frac{j\sqrt{15}}{2}$$

Example 2: In the Feedback System



- Characteristic eq. is given by

$$1 + K\frac{s+1}{s(s-1)(s+6)} = 0, \quad \Rightarrow \quad s^3 + 5s^2 + (K-6)s + K = 0.$$

- Routh array is

$$\begin{array}{lcc} s^3: & 1 & K-6 \\ s^2: & 5 & K \\ s\;: & (4K-30)/5 & \\ s^0: & K. & \end{array}$$

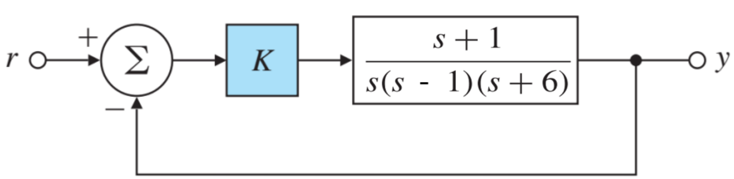
Example 2: In the Feedback System (cont'd)

- For the system to be stable, it is necessary that

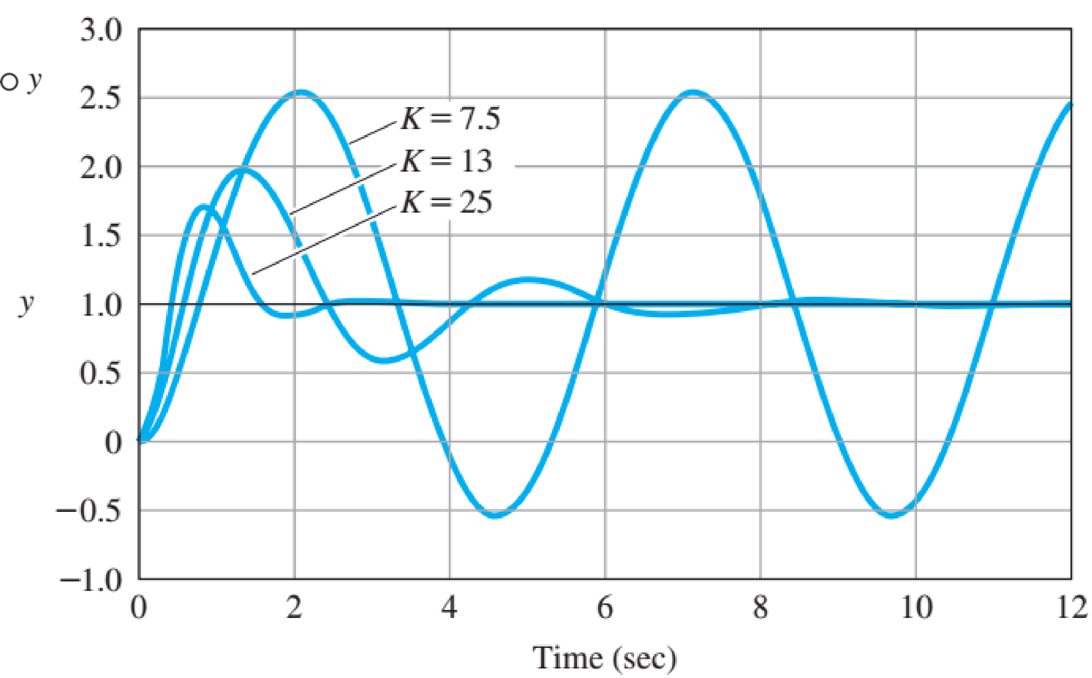
$$
\begin{array}{lll}
s^3: & 1 & K-6 \\
s^2: & 5 & K \\
s: & \boxed{} & \\
s^0: & K. &
\end{array}
$$

$$\frac{4K-30}{5} > 0 \quad \text{and} \quad K > 0,$$

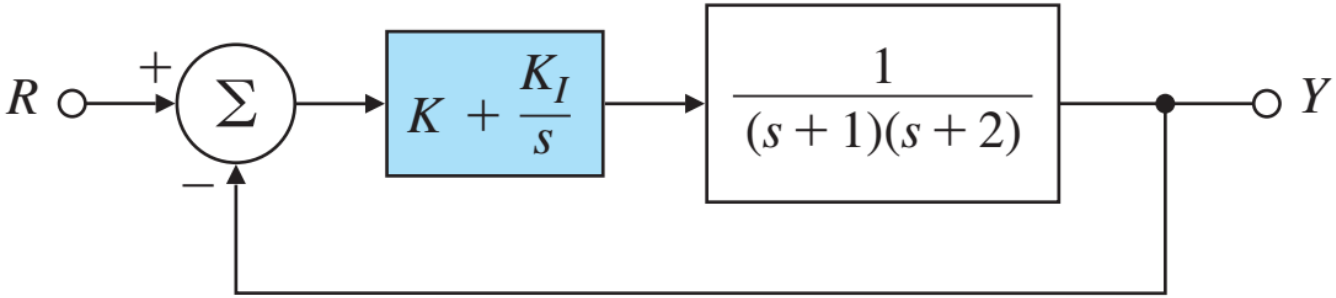
$$K > 7.5 \quad \text{and} \quad K > 0.$$



MATLAB
SIMULINK



Example 3: PI Controller Design



- Characteristic eq. is given by

$$1 + \left(K + \frac{K_I}{s}\right)\frac{1}{(s+1)(s+2)} = 0, \quad \Rightarrow \quad s^3 + 3s^2 + (2+K)s + K_I = 0.$$

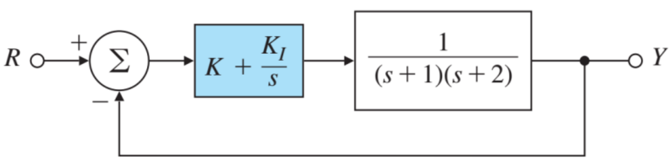
- Routh array is

$$\begin{array}{lccc} s^3 & : & 1 & 2+K \\ s^2 & : & 3 & K_I \\ s & : & (6+3K-K_I)/3 & \\ s^0 & : & K_I. & \end{array}$$

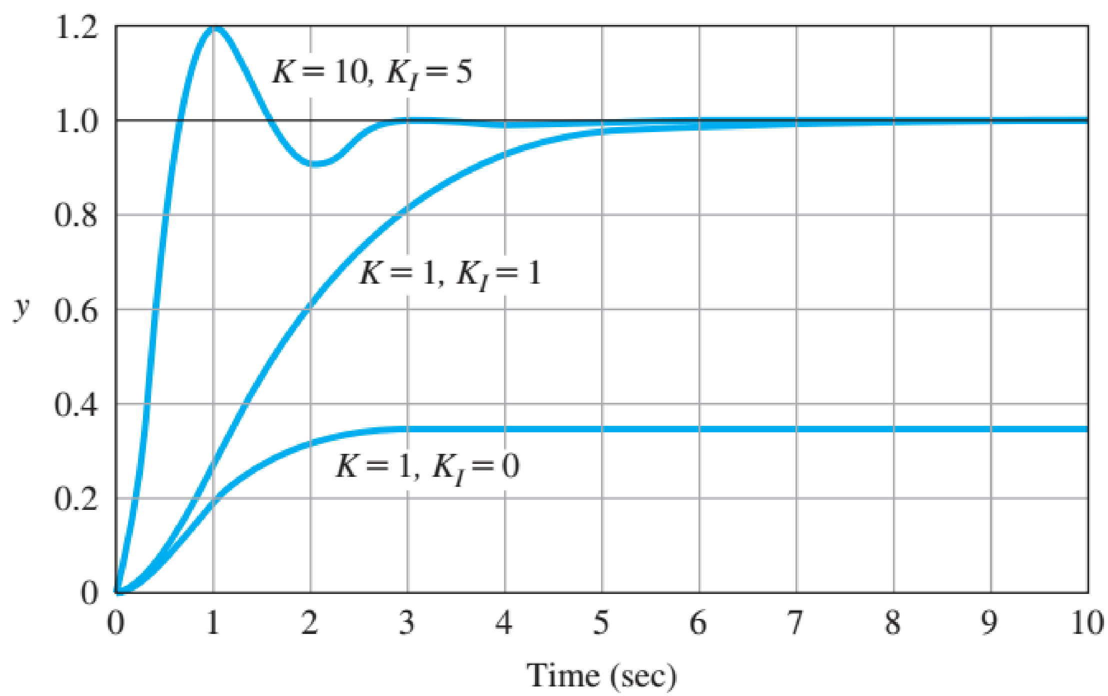
Example 3: PI Controller Design (cont'd)

- For the system to **be stable**, it is necessary that

$$\begin{array}{lcc} s^3: & 1 & 2+K \\ s^2: & 3 & K_I \\ s: & \boxed{(6+3K-K_I)/3} & \\ s^0: & K_I. & \end{array} \Rightarrow K_I > 0 \quad \text{and} \quad K > \frac{1}{3}K_I - 2.$$



MATLAB SIMULINK



Special Case I: Routh-Hurwitz Method

s^n	a_n	a_{n-2}	a_{n-4}	a_{n-6}	$\cdots$
s^{n-1}	a_{n-1}	a_{n-3}	a_{n-5}	a_{n-7}	$\cdots$
s^{n-2}	b_1	b_2	b_3	b_4	$\cdots$
s^{n-3}	c_1	c_2	c_3	c_4	$\cdots$
$\vdots$	$\vdots$	$\vdots$			
s^2	k_1	k_2			
s^1	l_1				
s^0	m_1				

$$c_1 = \frac{a_{n-3}b_1 - a_{n-1}b_2}{b_1}$$

$$c_2 = \frac{a_{n-5}b_1 - a_{n-1}b_3}{b_1}$$

$$\vdots$$

- Question: What if the b_1 (denominator) is zero?

Special Case I: Routh-Hurwitz Method (cont'd)

- Consider a polynomial of characteristic eq.

$$Q(s) = s^5 + 2s^4 + 3s^3 + 6s^2 + 5s + 2 = 0 \xrightarrow{\text{roots}} \begin{cases} \boxed{} \\ -0.5414 \pm j0.4671, \\ -1.5624 \end{cases}$$

s^5	1	3	5
s^4	2	6	2
s^3	0	4	
s^2			
s			
1			

$$b_1 = \frac{2 \cdot 3 - 6}{2} = 0$$

$$b_2 = \frac{2 \cdot 5 - 2}{2} = 4$$

Then, instead of zero, we can replace with small number ($\varepsilon \fallingdotseq 0$)

s^5	1	3	5
s^4	2	6	2
s^3	ε	4	
s^2	$-8/\varepsilon$	2	
s	4		
1	2		

$$c_1 = \frac{\varepsilon \cdot 6 - 4 \cdot 2}{\varepsilon} \approx -\frac{8}{\varepsilon}$$

$$c_2 = \frac{\varepsilon \cdot 2 - 0 \cdot 2}{\varepsilon} = 2$$

$$d_1 = \frac{4 \cdot (-8)/\varepsilon - 2 \cdot \varepsilon}{(-8)/\varepsilon} \approx 4$$

❖ **Findings:**

- Although **any value (positive or negative) of ε** is chosen, **two times of sign changes** will be done!! → **Unstable!!**

Special Case II: Routh-Hurwitz Method

$$
\begin{array}{l|lllll}
s^n & a_n & a_{n-2} & a_{n-4} & a_{n-6} & \cdots \\
s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} & a_{n-7} & \cdots \\
s^{n-2} & b_1 & b_2 & b_3 & b_4 & \\
s^{n-3} & c_1 & c_2 & c_3 & c_4 & \cdots \\
\vdots & \vdots & \vdots & & & \\
s^2 & k_1 & k_2 & & & \\
s^1 & l_1 & & & & \\
s^0 & m_1 & & & &
\end{array}
$$

All coefficients are zero??

$$c_1 = \frac{a_{n-3}b_1 - a_{n-1}b_2}{b_1}$$

zero

$$c_2 = \frac{a_{n-5}b_1 - a_{n-1}b_3}{b_1}$$

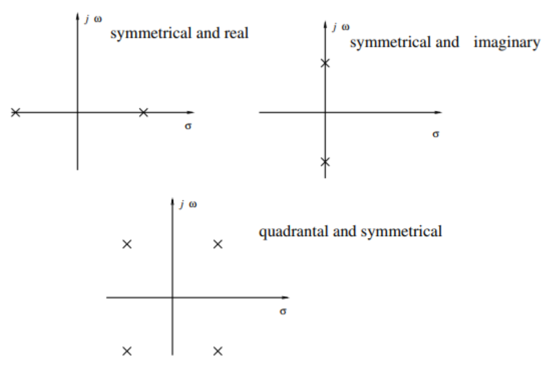
zero

$$\vdots$$

- **Question: What if all coefficients in a row are zeros?**

Special Case II: Routh-Hurwitz Method (cont'd)

- Consider a polynomial of characteristic eq.



$$Q(s) = s^5 + 2s^4 + 4s^3 + 5s^2 + 4s + 2 = 0$$

s^5	1	4	4
s^4	2	5	2
s^3	1.5	3	
s^2	1	2	
s	0	0	
1			

$A(s) = s^2 + 2$: Auxiliary polynomial equation

It indicates that Q(s) = 0 must have a pair of roots equal magnitude and opposite sign, which are also roots of A(s) = 0.

s^5	1	4	4
s^4	2	5	2
s^3	1.5	3	
s^2	1	2	
s	2	0	
1	2	0	

$$\frac{dA(s)}{ds} = 2s$$

❖ **Thus, the zero row can be calculated!! → there is no sign change!!**

Summary

❖ Summary:

- PID controller design by using Pole Placement Method
- Stability Criteria
- Routh-Hurwitz Stability Criteria and Control Design Method