ME 8043114 Automatic Control

Lecture 7: Root Locus: Lead-Lag Compensator Design
PID Control Design based on Root Locus

Dr. Shadi M. Munshi
E-mail: smmunshi@uqu.edu.sa

Dr. Badr AlOufi
E-mail: baawfey@uqu.edu.sa

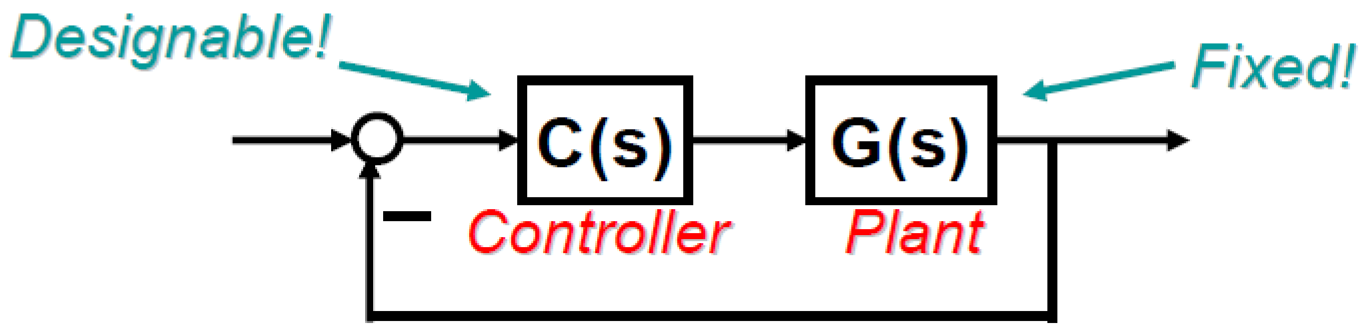
Lecture Outline

- Root Locus: Lead-Lag Compensator Design
- PID Control Design with Matlab using Root Locus

Course roadmap

Modeling	*Analysis*	*Design*
✓ Laplace transform ✓ Transfer function ✓ Models for systems ✓ electrical ✓ mechanical ✓ electromechanical ✓ Linearization, delay	✓ Time response ✓ Transient ✓ Steady state Frequency response • Bode plot ✓ Stability ✓ Routh-Hurwitz • Nyquist	Design specs Root locus Frequency domain PID & Lead-lag Design examples

Closed-loop design by root locus



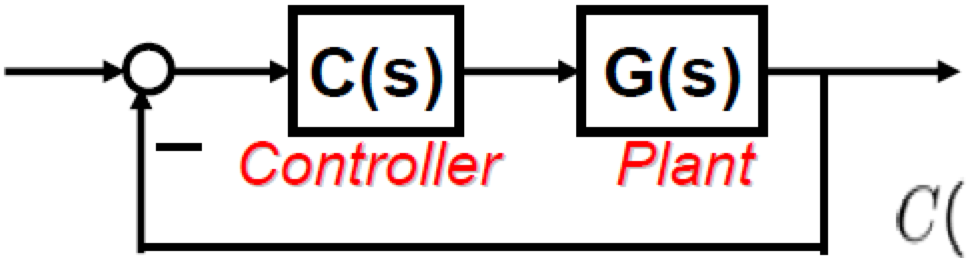
- Place closed-loop poles at desired location
  - by tuning the gain C(s)=K.

(for time domain specs)

- If root locus does not pass the desired location, then reshape the root locus
  - by adding poles/zeros to C(s). (How?)

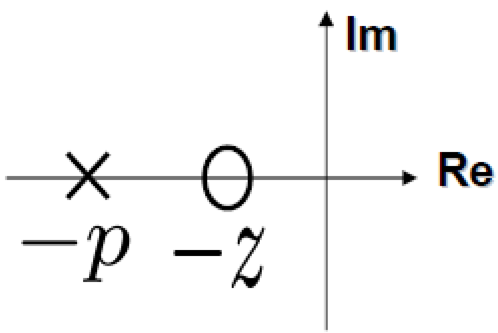
Compensation

Lead and lag compensators

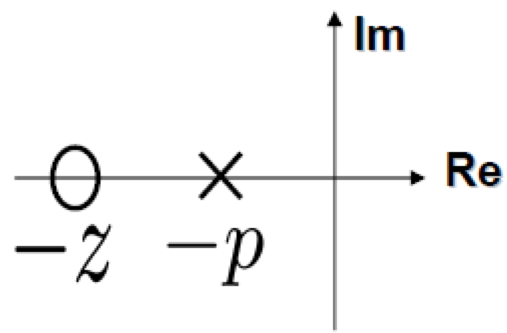


$$C(s) = K\frac{s+z}{s+p}, \quad (z > 0, p > 0)$$

- Lead compensator



- Lag compensator



The reason why these are called "lead" and "lag" will be explained in frequency response approach (later in this course).

Roles of lead & lag compensators

- Lead compensator
  - Improve transient response
  - Improve stability

$$C_{Lead}(s) = K_1 \frac{s + z_1}{s + p_1}$$

- Lag compensator
  - Reduce steady state error

$$C_{Lag}(s) = K_2 \frac{s + z_2}{s + p_2}$$

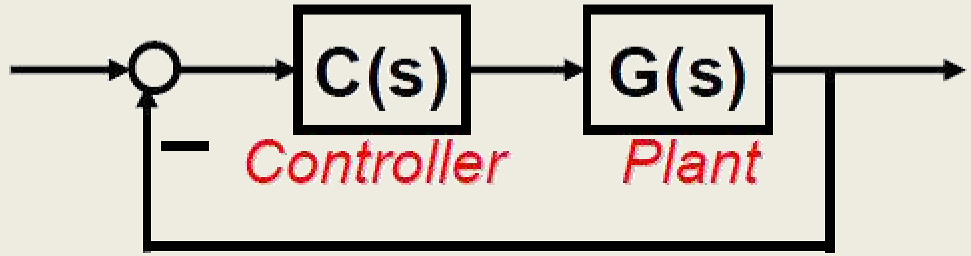
- Lead-lag compensator
  - Take into account all the above issues.

$$C_{LL}(s) = C_{Lead}(s) C_{Lag}(s)$$

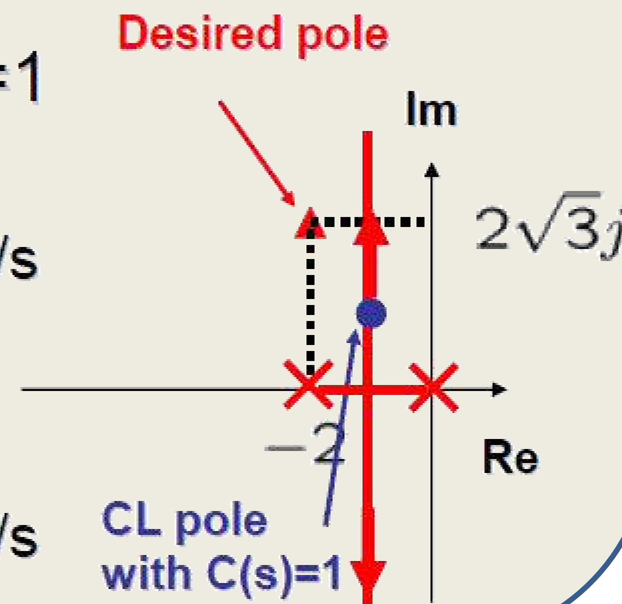
Lead compensator design

- Consider a system

$$G(s) = \frac{4}{s(s+2)}$$



- Analysis of CL system for C(s)=1
  - Damping ratio ζ=0.5
  - Undamped natural freq. ω_n=2 rad/s
- Performance specification
  - Damping ratio ζ=0.5
  - Undamped natural freq. ω_n=4 rad/s



Angle & magnitude conditions (review)

- A point s to be on root locus ←→ it satisfies
  - **Angle condition**

Odd number

$$\angle L(s) = 180^{\circ} \times (2k+1), \;\; k = 0, \pm 1, \pm 2, \ldots$$

- For a point on root locus, gain K is obtained by
  - **Magnitude condition**

$$|L(s)| = \frac{1}{K}$$

Lead compensator design (cont'd)

Evaluate G(s) at the desired pole.

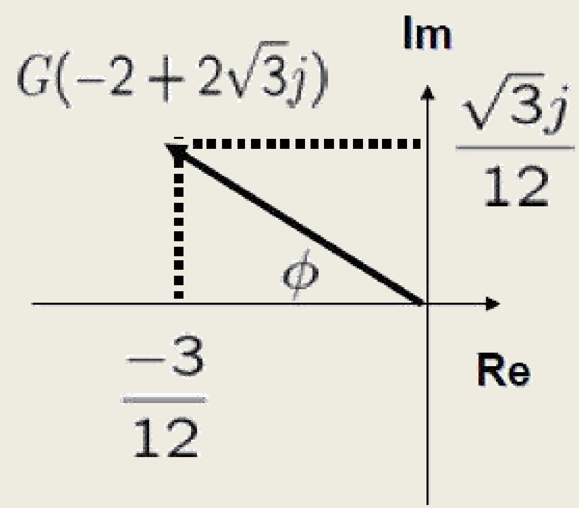
$$G(-2+2\sqrt{3}j) = \frac{4}{(-2+2\sqrt{3}j)2\sqrt{3}j} = \frac{-1}{3+\sqrt{3}j} = \frac{-3+\sqrt{3}j}{12}$$

- If *angle condition* is satisfied, compute the corresponding K.
- In this example,

  $$\angle G(-2+2\sqrt{3}j) = -210$$

  Angle condition is not satisfied.

➡ **Angle deficiency** $\phi = 30$



Lead compensator design (cont'd)

To compensate angle deficiency,

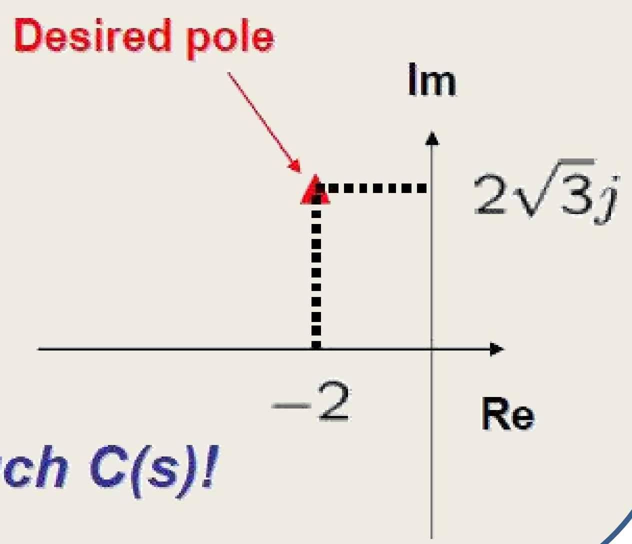
design a lead compensator C(s)

$$C(s) = K\frac{s+z}{s+p}$$

satisfying

$$\angle C(-2+2\sqrt{3}j) = 30 (=: \phi)$$

$$\left(\Rightarrow \angle GC(-2+2\sqrt{3}j) = -180 \right)$$



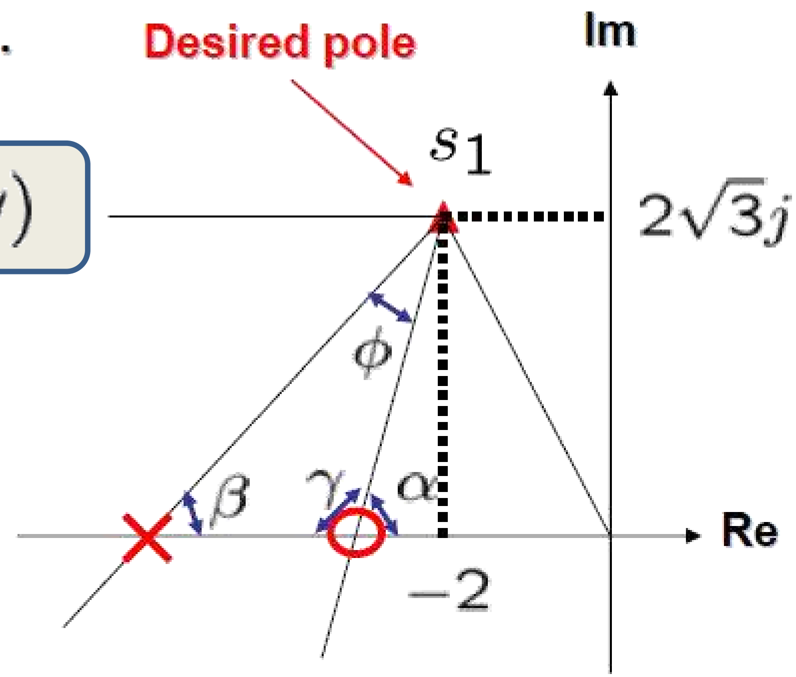
There are many ways to design such C(s)!

How to select pole and zero

- Select pole and zero s.t.

× ○

$$\phi = 30 \ \text{(angle deficiency)}$$



Why?

$$\begin{cases} \phi + \beta + \gamma = 180 \\ \alpha + \gamma = 180 \end{cases}$$

$$\Rightarrow \quad \phi + \beta = \alpha$$

$$\Rightarrow \quad \phi = \alpha - \beta = \angle(s_1 + z) - \angle(s_1 + p) = \angle \frac{s_1 + z}{s_1 + p} = \angle C(s_1)$$

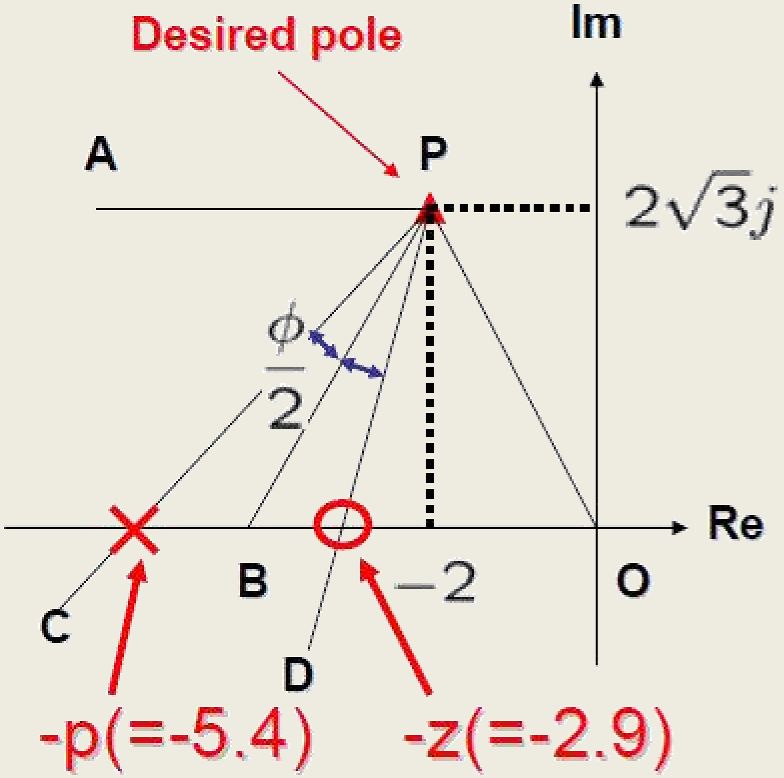
A method to select pole and zero

- Draw horizontal line PA
- Draw line PO
- Draw bisector PB

$$\angle APB = \angle BPO = \frac{1}{2}\angle APO$$

- Draw PC and PD

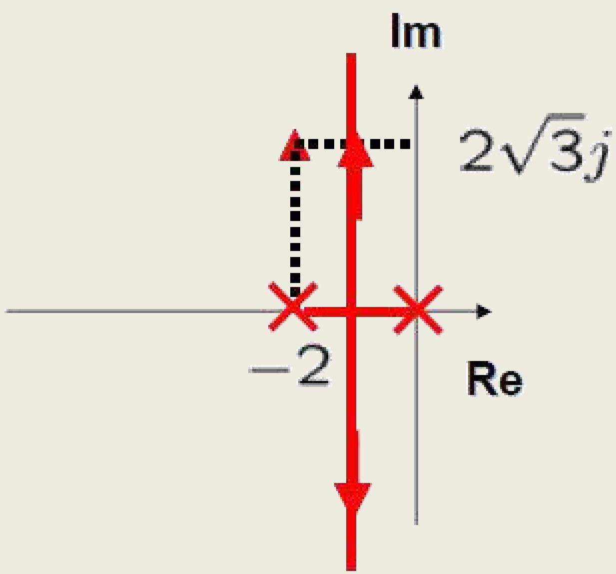
$$\angle CPB = \angle BPD = \frac{\phi}{2}$$



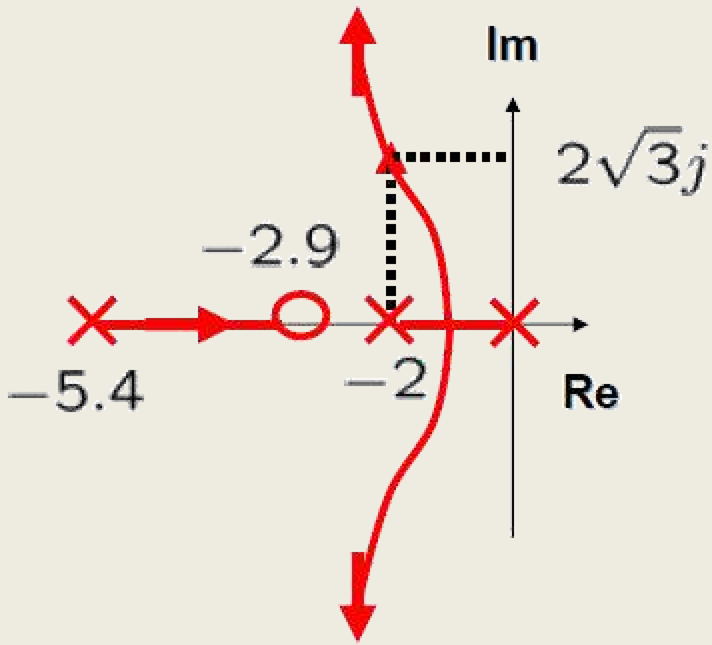
- Pole and zero of C(s) are shown in the figure.

Comparison of root locus

- G(s)



- G(s)C(s)



Improved stability!

How to design the gain K?

- Lead compensator

$$C(s) = K\frac{s+2.9}{s+5.4}$$

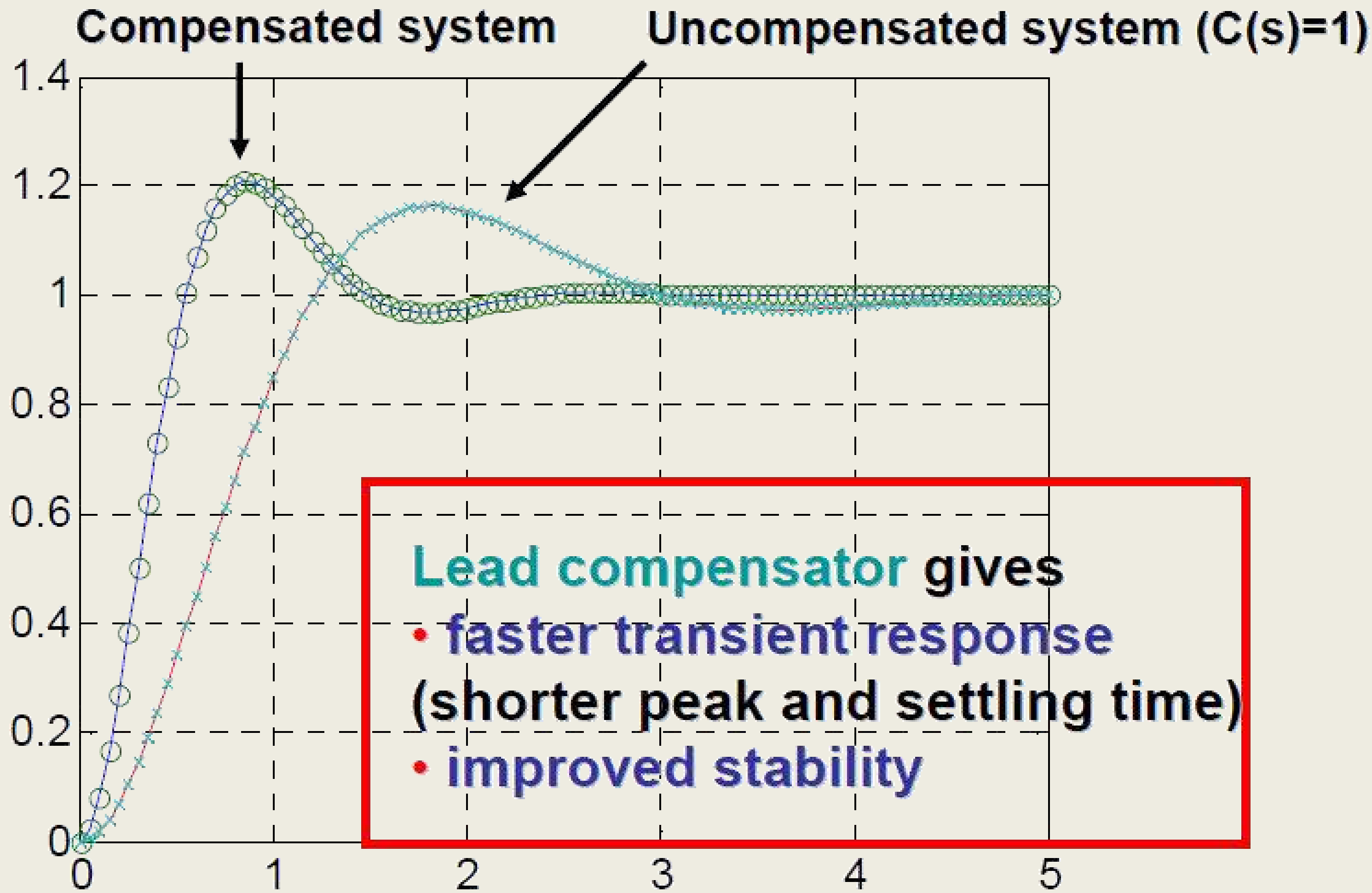
- Open loop transfer function

$$G(s)C(s) = K\frac{4(s+2.9)}{s(s+2)(s+5.4)}$$

- Magnitude condition

$$K\left|\frac{4(s+2.9)}{s(s+2)(s+5.4)}\right|_{s=-2+2\sqrt{3}j} = 1 \Rightarrow K = 4.675$$

Comparison of step responses



Lead compensator gives
- faster transient response (shorter peak and settling time)
- improved stability

Error constants

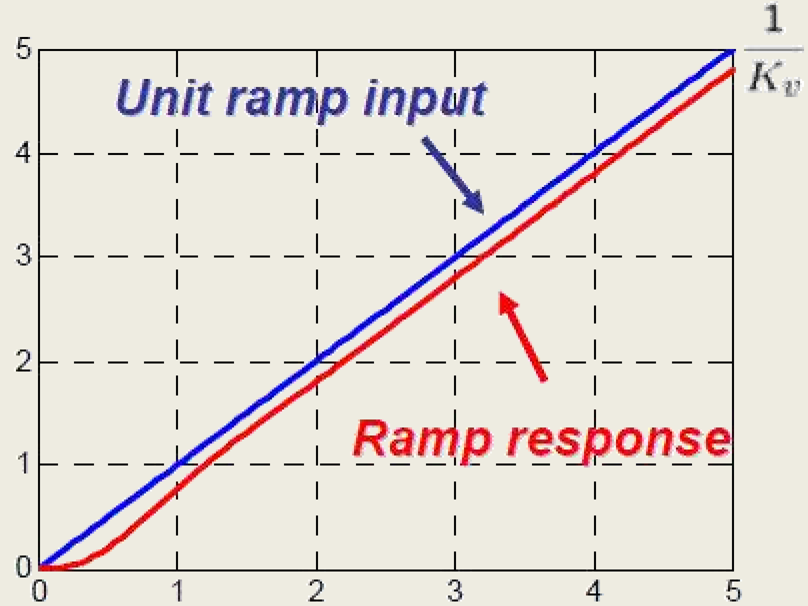
$$G(s)C(s) = \frac{4}{s(s+2)} \cdot \frac{4.675(s+2.9)}{s+5.4}$$

- Step-error constant

$$K_p := \lim_{s \to 0} G(s)C(s) = \infty$$

- Ramp-error constant

$$K_v := \lim_{s \to 0} sG(s)C(s) = 5.02$$



Lag compensator can be used to reduce steady-state error.

Roles of lead & lag compensators

- Lead compensator
  - Improve transient response
  - Improve stability

$$C_{Lead}(s) = K_1 \frac{s + z_1}{s + p_1}$$

- Lag compensator
  - Reduce steady state error

$$C_{Lag}(s) = K_2 \frac{s + z_2}{s + p_2}$$

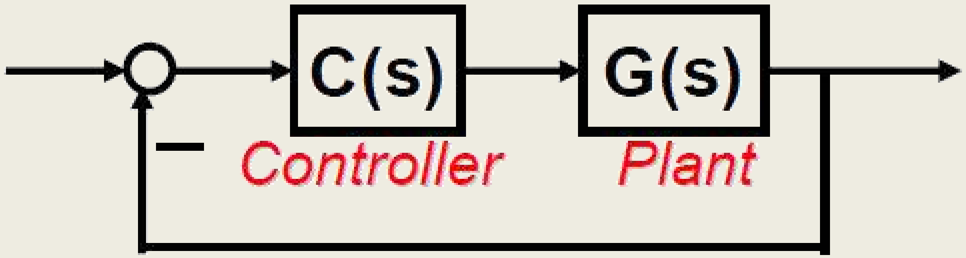
- Lead-lag compensator
  - Take into account all the above issues.

$$C_{LL}(s) = C_{Lead}(s) C_{Lag}(s)$$

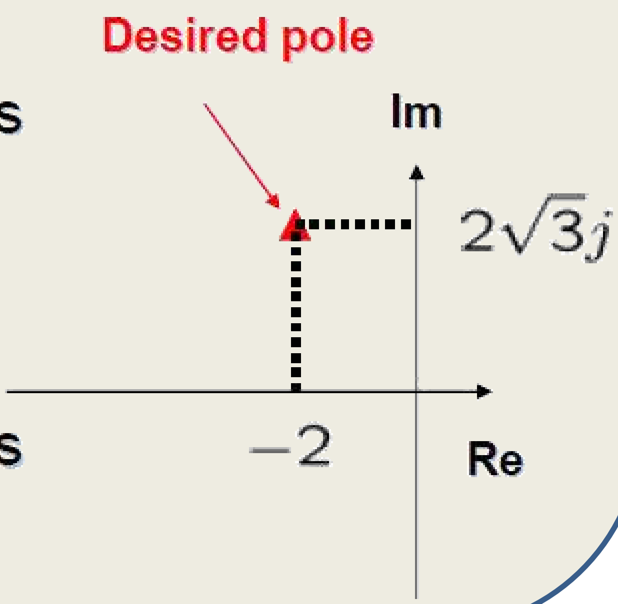
Lead-lag compensator design

- Consider a system

$$G(s) = \frac{4}{s(s+2)}$$



- Analysis of CL system for C(s)=1
  - Damping ratio ζ=0.5
  - Undamped natural freq. ω_n=2 rad/s
  - Ramp-error constant Kv=2
- Performance specification
  - Damping ratio ζ=0.5
  - Undamped natural freq. ω_n=4 rad/s
  - Ramp-error constant Kv=50



Error constants (after lead compensation)

$$G(s)C_{Lead}(s) = \frac{4}{s(s+2)} \cdot \frac{4.675(s+2.9)}{s+5.4}$$

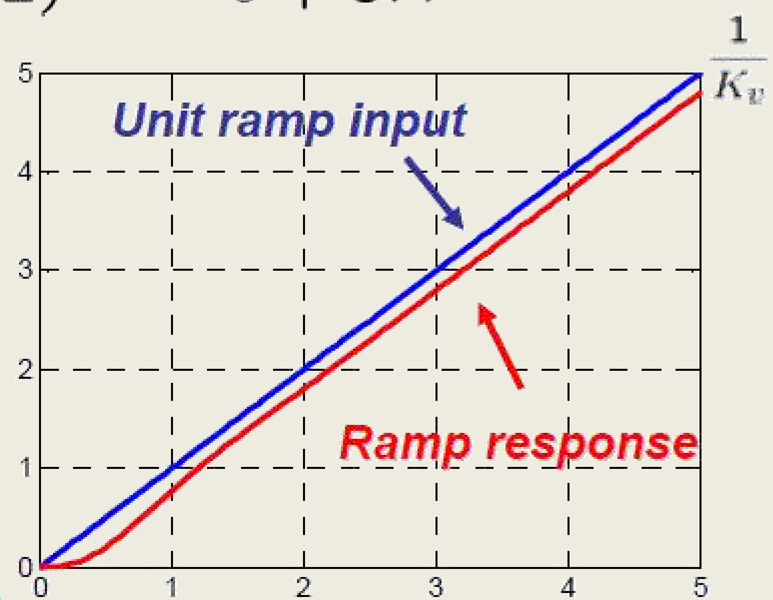
- Step-error constant

$$K_p := \lim_{s \to 0} G(s)C_{lead}(s) = \infty$$

- Ramp-error constant

$$K_v := \lim_{s \to 0} sG(s)C_{lead}(s) = 5.02$$

NOT SATISFACTORY!



Lag compensator can reduce steady-state error.

How to design lag compensator

- Lag compensator $C_{Lag}(s) = \frac{s+z}{s+p}$
- We want to increase ramp-error constant

$$K_v = \lim_{s \to 0} sG(s)C_{Lead}(s)C_{Lag}(s) = 5.02 \cdot \frac{z}{p} > 50$$

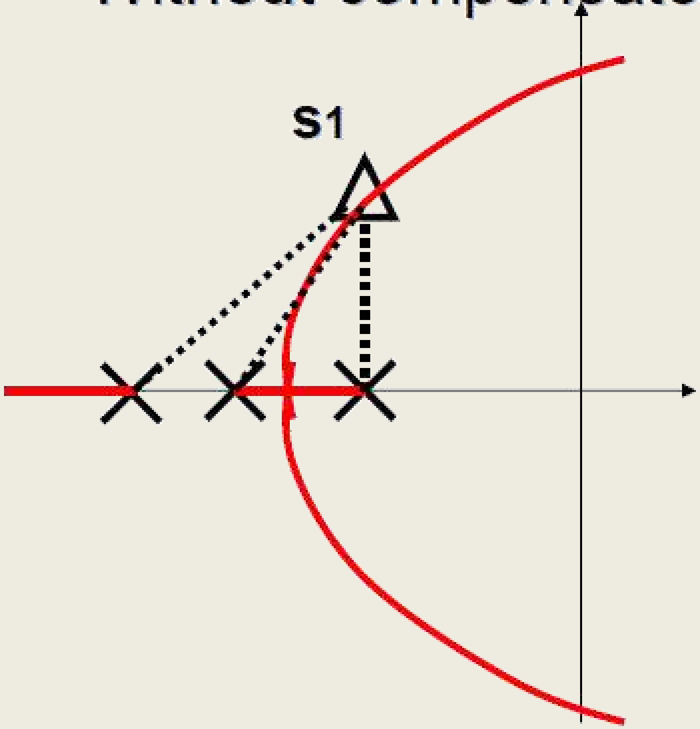
Take, for example, z=10p.

- We do not want to change CL pole location s_1 so much (already satisfactory transient).

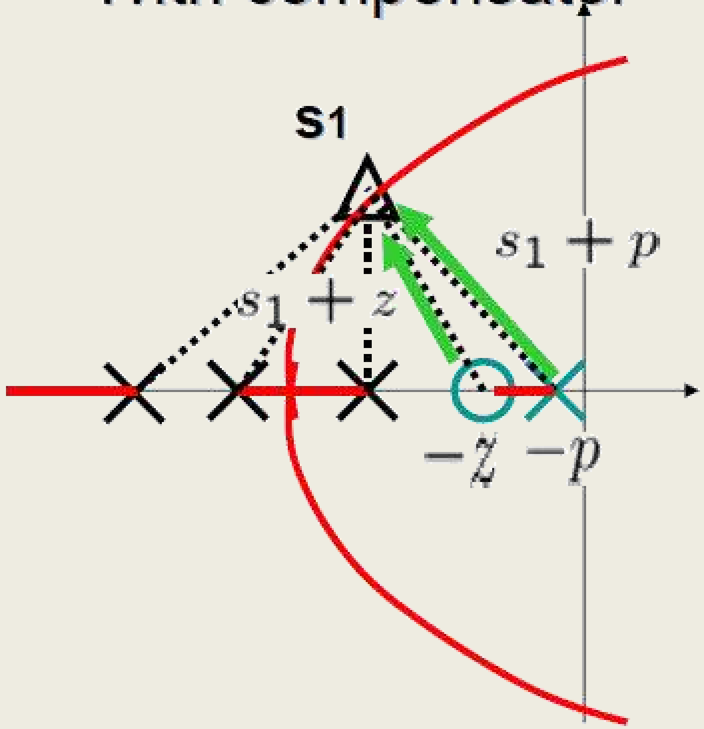
$$\left.\begin{array}{r} 1+G(s_1)C_{Lead}(s_1)=0 \\ C_{Lag}(s_1)\approx 1 \end{array}\right\} \Rightarrow 1+G(s_1)C_{Lead}(s_1)C_{Lag}(s_1)\approx 0$$

Root locus with lag compensator

- Without compensator



- With compensator



$$C_{lag}(s_1) = \frac{s_1 + z}{s_1 + p} \approx 1 \text{ if } z \approx p$$

Guidelines to choose z and p

- The zero and the pole of a lag compensator should be close to each other, for

$$C_{Lag}(s_1) \approx 1$$

- The pole of a lag compensator should be close to the origin, to have a large ratio z/p, leading to a large ramp-error constant Kv.
- However, the pole of a lag compensator too close to the origin may be problematic:
  - Slow settling (due to closed-loop pole near the origin)

How to design lag compensator

- For the desired CL pole $s_1 = -2 + 2\sqrt{3}j$

$$C_{Lag}(s_1) \approx 1 \Leftrightarrow \left|\frac{s_1 + 10p}{s_1 + p}\right| \approx 1 \quad \angle\left(\frac{s_1 + 10p}{s_1 + p}\right) \approx 0$$

- Take a small p (by trial-and-error!)

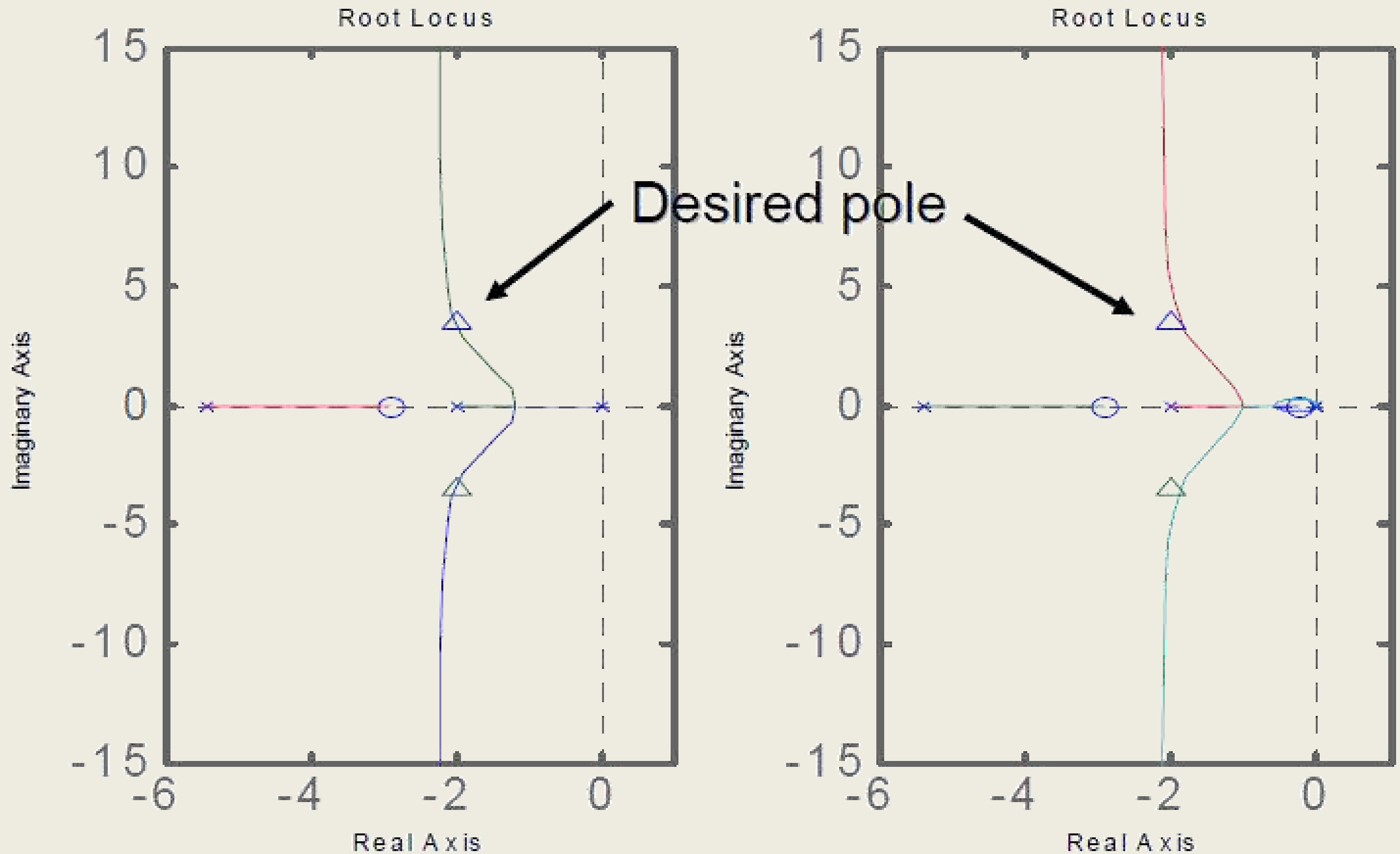
$$p = 0.025 \Rightarrow \left|\frac{s_1 + 10p}{s_1 + p}\right| = 0.97 \quad \angle\left(\frac{s_1 + 10p}{s_1 + p}\right) \approx -2.88^o$$

- Lead-lag controller

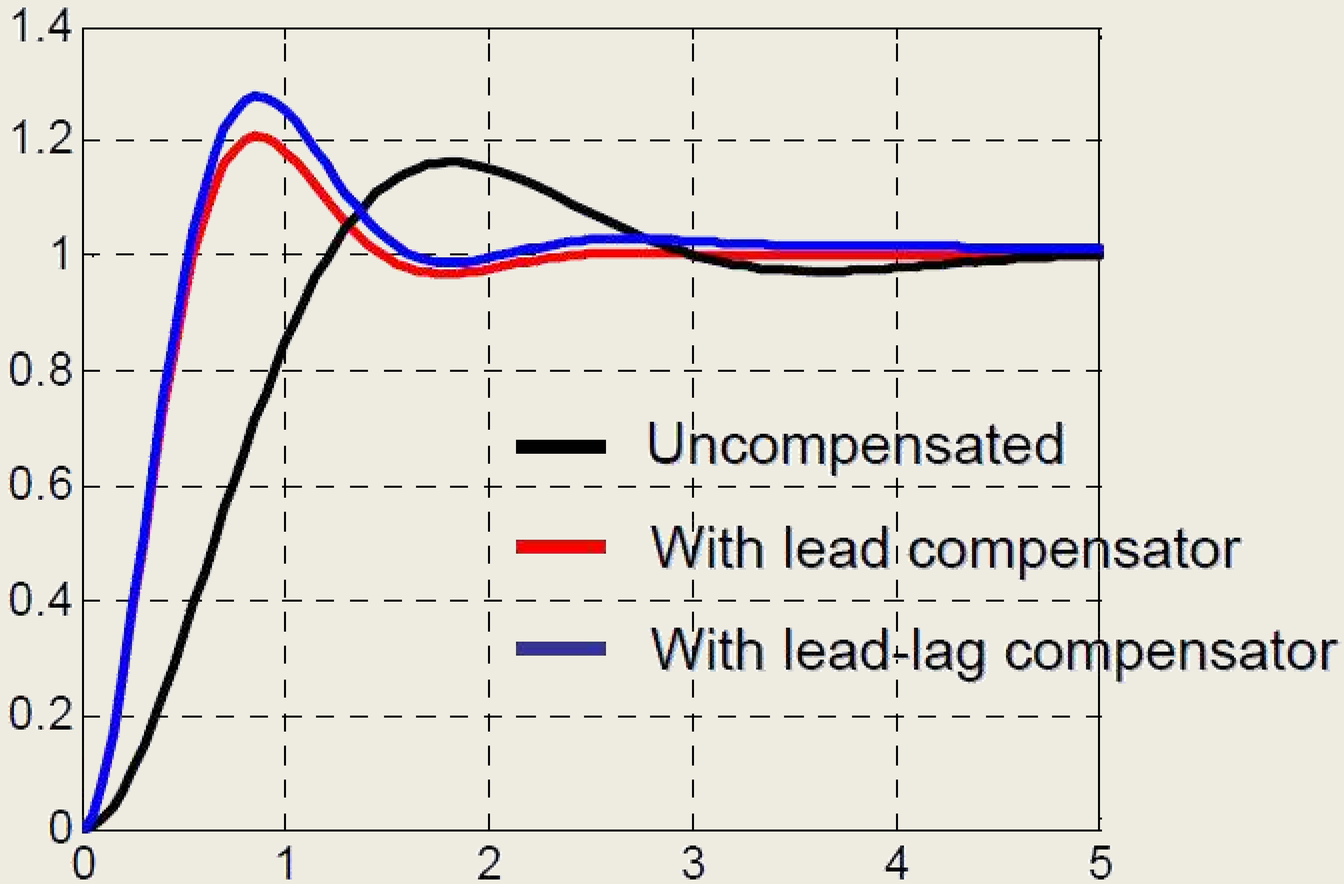
$$C_{LL}(s) = 4.675\frac{s + 2.9}{s + 5.4} \cdot \frac{s + 0.25}{s + 0.025}$$

Root locus

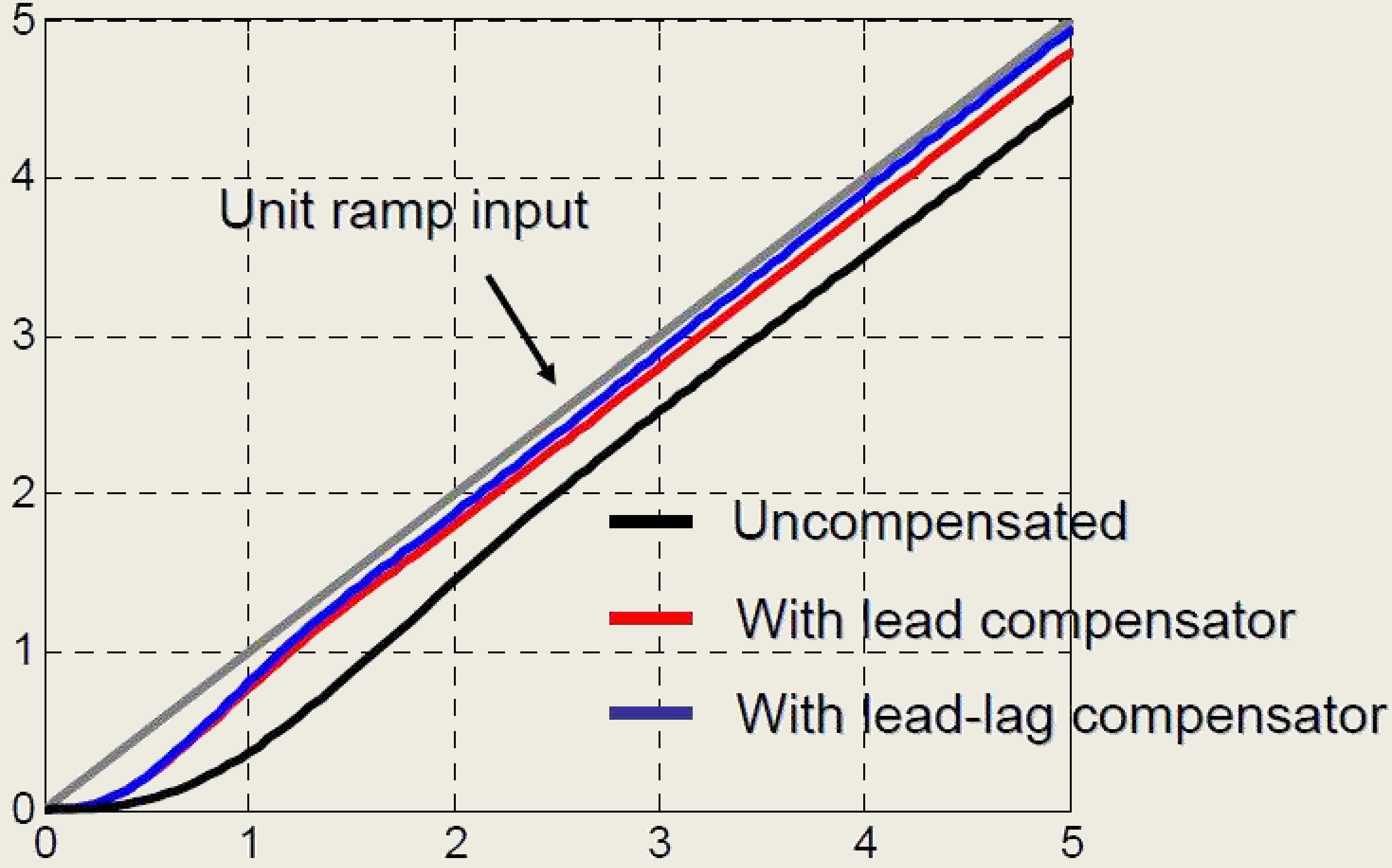
With lead compensator   With lead-lag compensator



Comparison of step responses

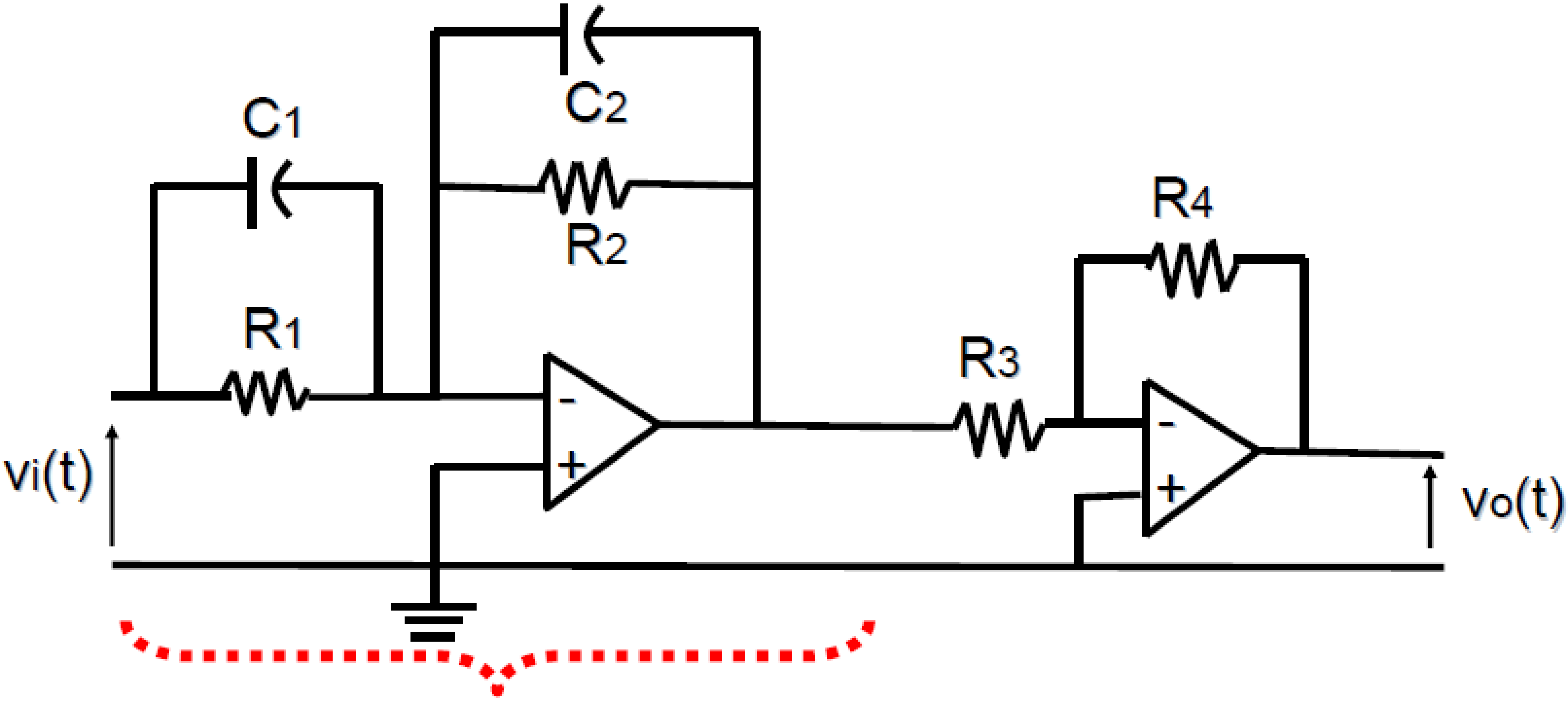


Comparison of ramp responses



Compensator realization

- One example, using operational amplifiers



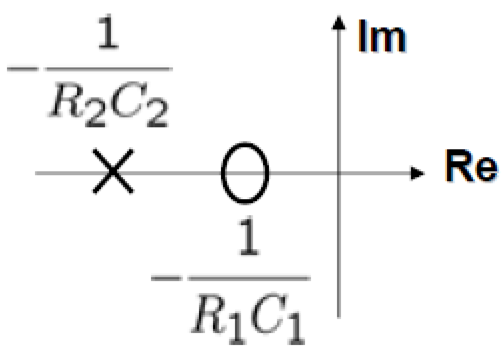
$$-\frac{R_2}{R_1}\cdot\frac{R_1C_1s+1}{R_2C_2s+1}$$

Compensator realization (cont'd)

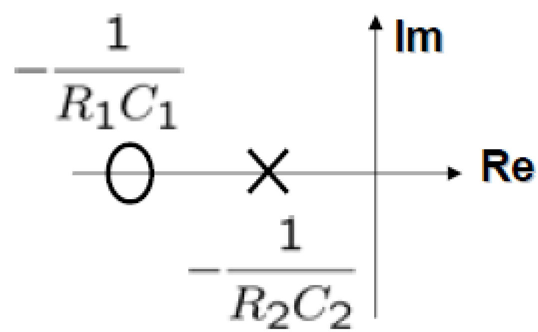
- Transfer function

$$C(s)=\frac{V_o(s)}{V_i(s)}=\frac{\dfrac{\frac{R_2}{C_2 s}}{R_2+\frac{1}{C_2 s}}}{\dfrac{\frac{R_1}{C_1 s}}{R_1+\frac{1}{C_1 s}}}\cdot\frac{R_4}{R_3}=\underbrace{\frac{R_4 C_1}{R_3 C_2}}_{K}\cdot\frac{s+\overbrace{\frac{1}{R_1 C_1}}^{z}}{s+\underbrace{\frac{1}{R_2 C_2}}_{p}}$$

- Lead compensator



- Lag compensator



Course roadmap

Modeling

- ✓ Laplace transform
- ✓ Transfer function
- ✓ Models for systems
  - ✓ electrical
  - ✓ mechanical
  - ✓ electromechanical
- ✓ Linearization

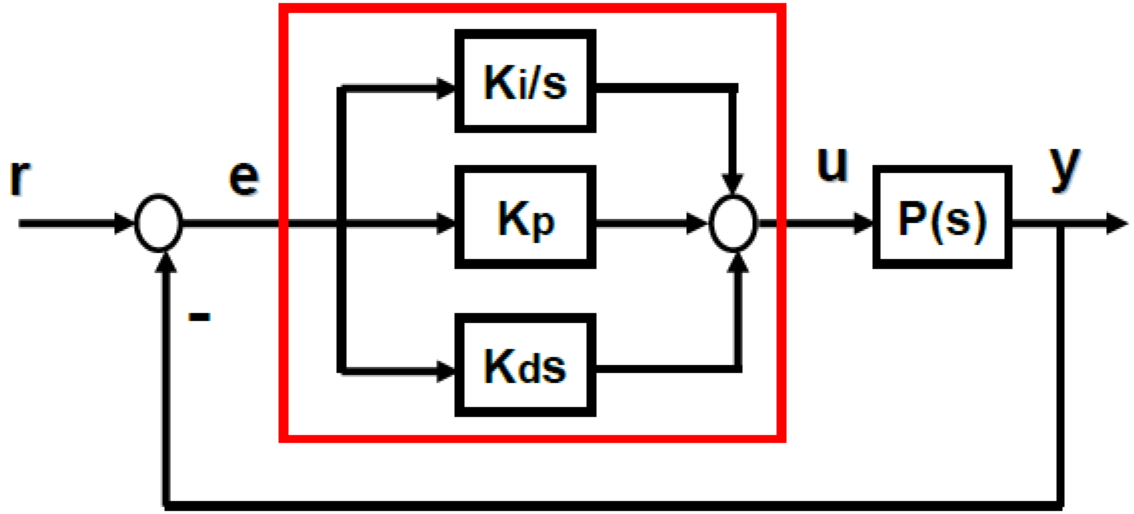
Analysis

- ✓ Time response
  - ✓ Transient
  - ✓ Steady state
- Frequency response
  - Bode plot
- ✓ Stability
  - ✓ Routh-Hurwitz
  - Nyquist

Design

- Design specs
- Root locus
- Frequency domain
- PID & Lead-lag
- Design examples

PID controller



t-domain: $$u(t) = \underbrace{K_p e(t)}_{\text{Proportional}} + \underbrace{K_i \int_0^t e(\tau)d\tau}_{\text{Integral}} + \underbrace{K_d \frac{de(t)}{dt}}_{\text{Derivative}}$$

s-domain: $$C(s) = K_p + \frac{K_i}{s} + K_d s = K_p\left(1 + \frac{1}{K_I s} + K_D s\right)$$

Notes on PID controller

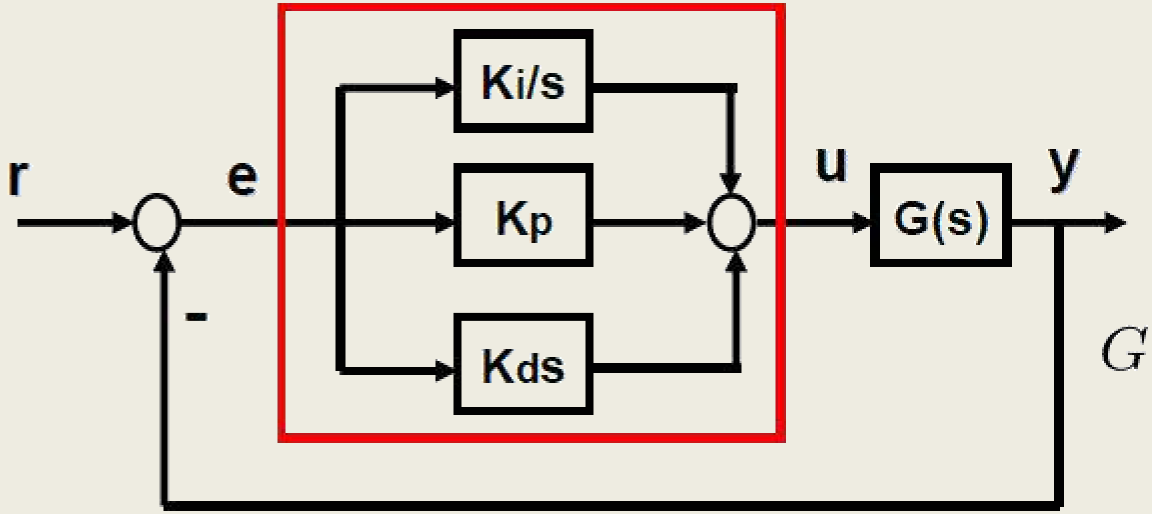
- Most popular in process and robotics industries
  - Good performance
  - Functional simplicity (Operators can easily tune.)
- To avoid high frequency noise amplification, derivative term is implemented as

$$K_d s \approx \frac{K_d s}{\tau_d s + 1}$$

with τ_d much smaller than plant time constant.

- PI controller $C(s) = K_p + K_i/s$
- PD controller $C(s) = K_p + K_d s$

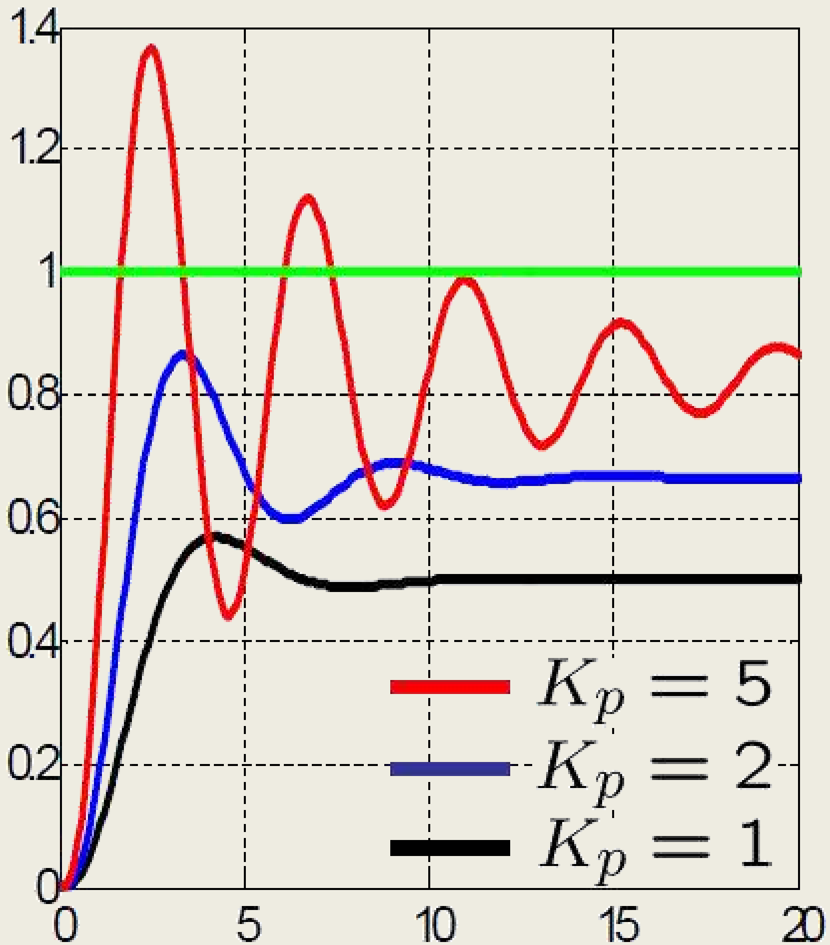
A simple example



$$G(s) = \frac{1}{(s+1)^3}$$

- We plot y(t) for step reference r(t) with
  - P controller
  - PI controller
  - PID controller

P controller

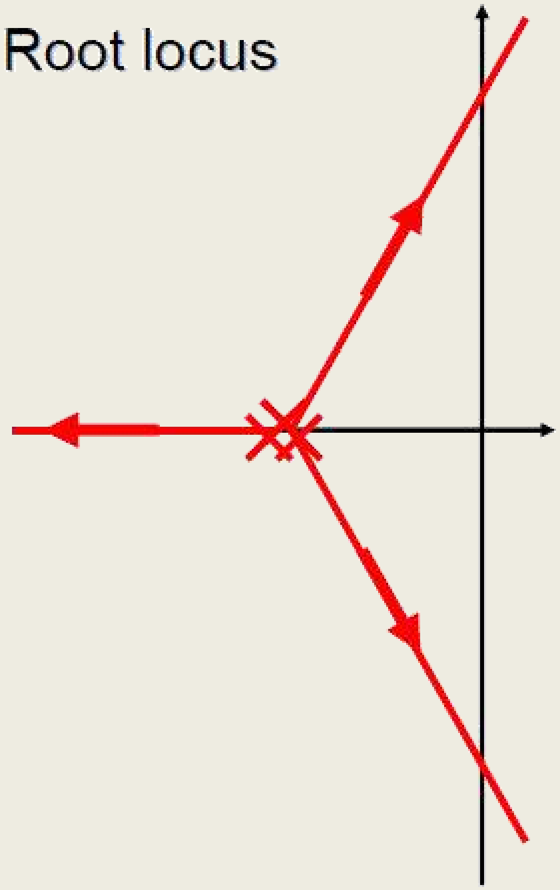


$$C(s) = K_p$$

- Simple
- Steady state error
  - Higher gain gives smaller error
- Stability
  - Higher gain gives faster and more oscillatory response

Interpretation: P controller

Root locus

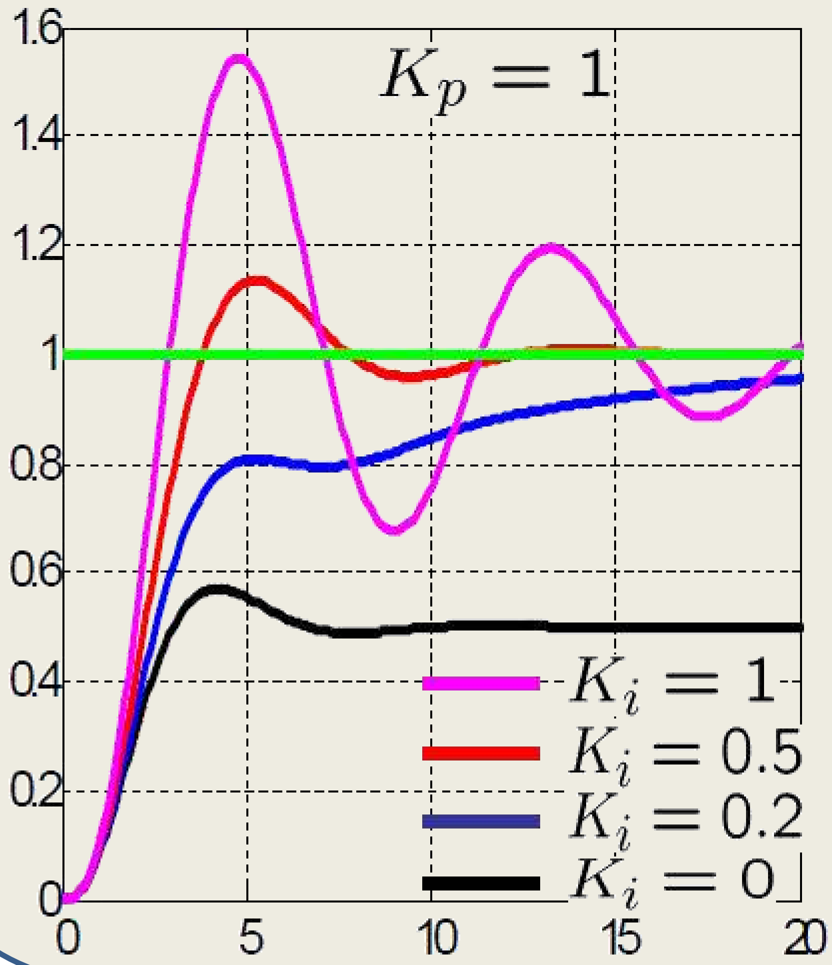


Steady state error

$$e_{ss} = \frac{1}{1+G(0)C(0)} = \frac{1}{1+K_p}$$

(This Kp is the P controller gain.)

PI controller

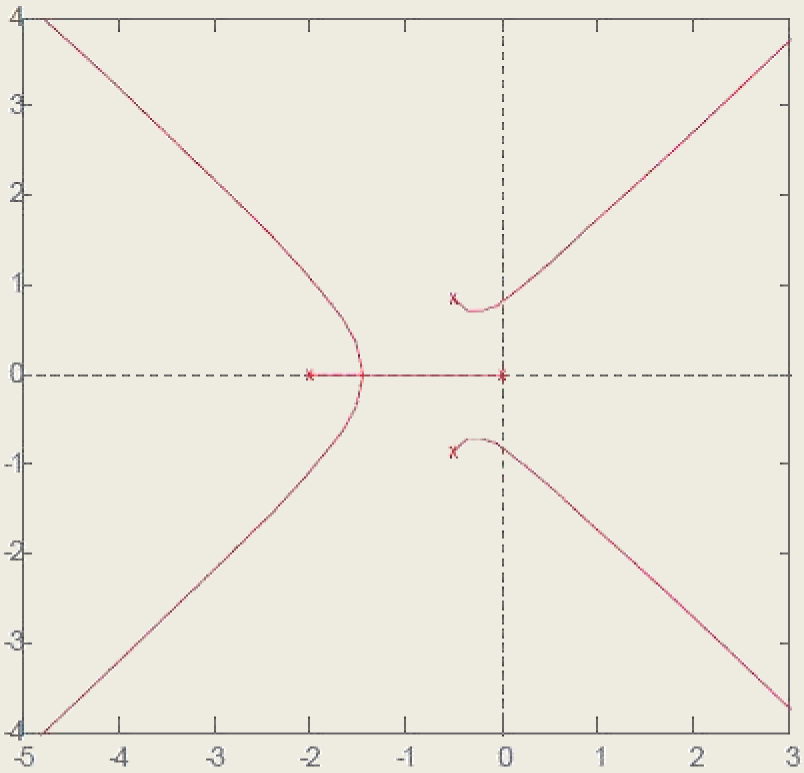


$$C(s) = K_p + \frac{K_i}{s}$$

- Zero steady state error (provided that CL is stable.)
- Stability
  - Higher gain gives faster and more oscillatory response

Interpretation: PI controller

Root locus



$$1+\frac{1}{(s+1)^3}\cdot\frac{s+K_i}{s}=0$$

$$\Rightarrow s\left\{1+(s+1)^3\right\}+K_i=0$$

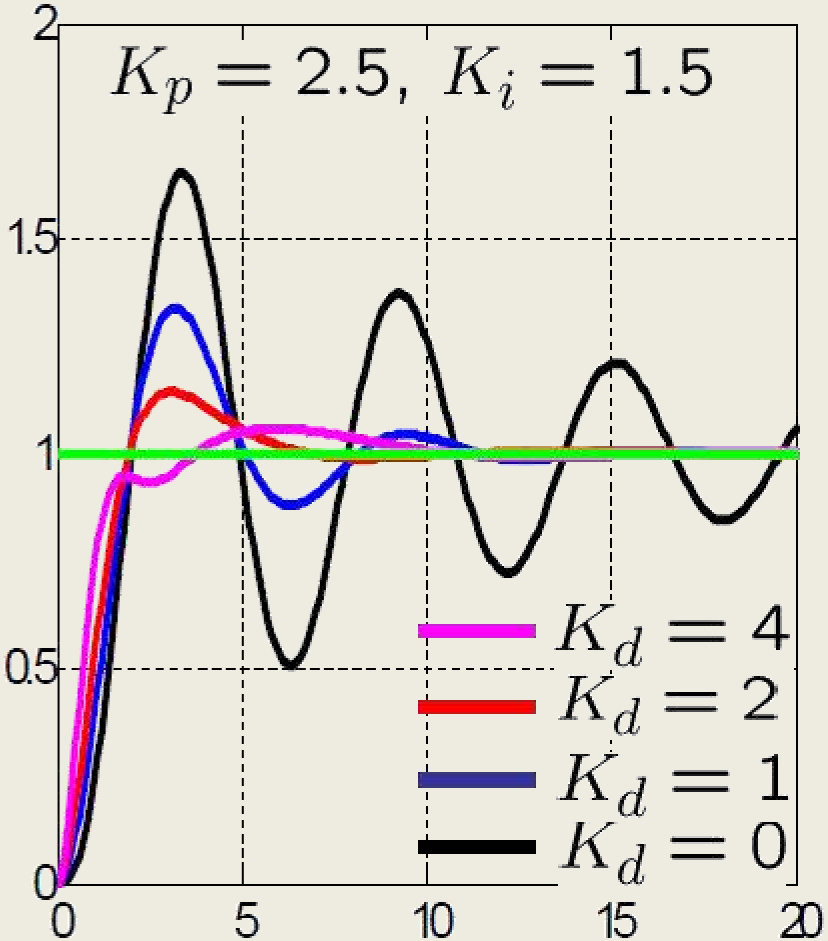
$$\Rightarrow 1+K_i\frac{1}{s(s+2)(s^2+s+1)}=0$$

Steady state error

$$e_{ss} = \frac{1}{1+G(0)C(0)}=0$$

(due to the integrator in C(s).)

PID controller

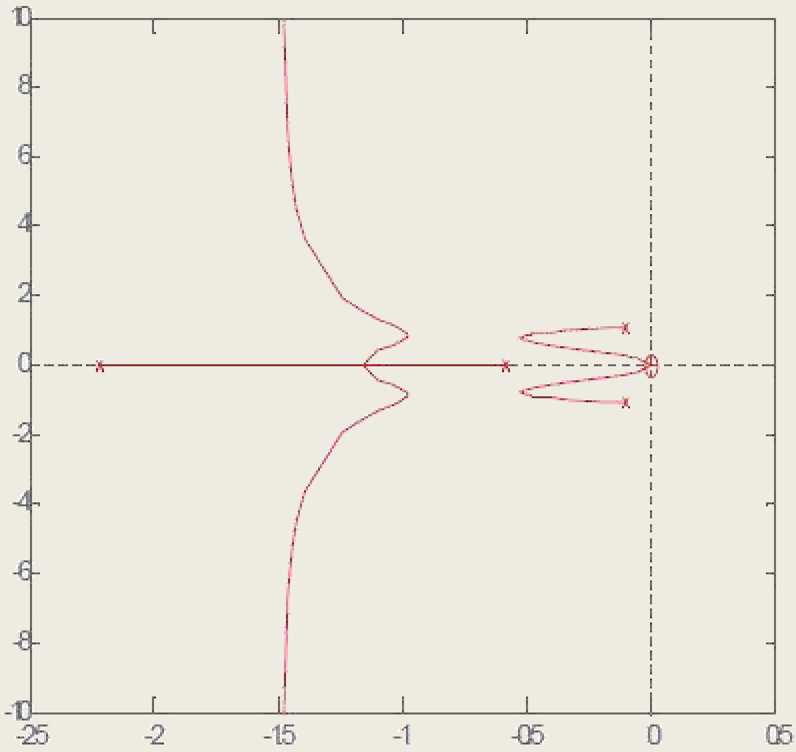


$$C(s) = K_p + \frac{K_i}{s} + K_d s$$

- Zero steady state error (due to integral control)
- Stability
  - Higher gain gives more damped response
- Too high gain worsen performance.

Interpretation: PID controller

Root locus



$$1 + \frac{1}{(s+1)^3} \cdot \frac{K_d s^2 + 2.5s + 1.5}{s} = 0$$

$$\Rightarrow \quad 1 + K_d \frac{s^2}{s(s+1)^3 + 2.5s + 1.5} = 0$$

Steady state error

$$e_{ss} = \frac{1}{1 + G(0)C(0)} = 0$$

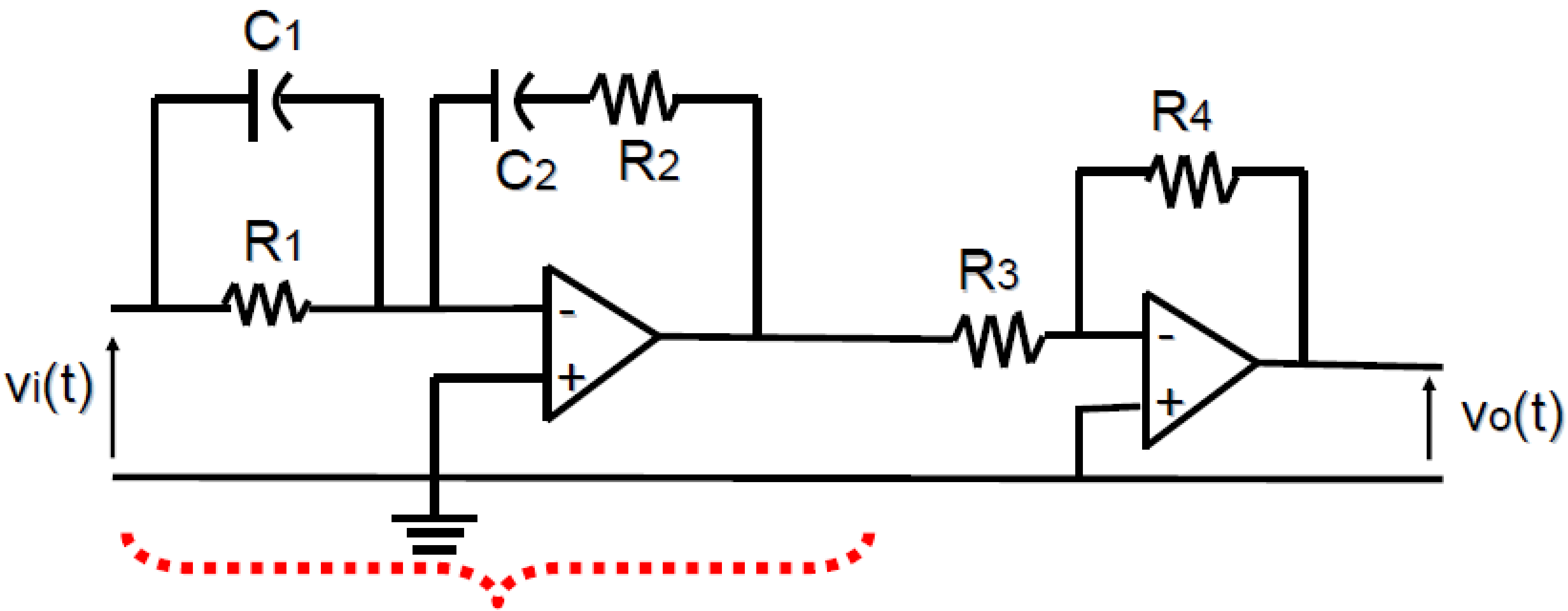
(due to the integrator in C(s).)

How to tune PID parameters

- Empirical (Model-free)
  - Trial and error
  - Ziegler-Nichols tuning rule (1942) (Appendix)
  - Useful even if a system is too complex to model
  - Useful only when trial-and-error tuning is allowed
- Model-based
  - Root locus
  - Frequency response approach
  - Useful only when a model is available
  - Necessary if a system has to work at the first trial

PID controller realization

- One example: Using OP amp



$$-\left[\left(\frac{R_2}{R_1}+\frac{C_1}{C_2}\right)+R_2C_1s+\frac{1}{R_1C_2s}\right]$$

Course roadmap

Modeling

- ✓ Laplace transform
- ✓ Transfer function
- ✓ Models for systems
  - ✓ electrical
  - ✓ mechanical
  - ✓ electromechanical
- ✓ Linearization

⇨

Analysis

- ✓ Time response
  - ✓ Transient
  - ✓ Steady state
- Frequency response
  - • Bode plot
- ✓ Stability
  - ✓ Routh-Hurwitz
  - • Nyquist

⇨

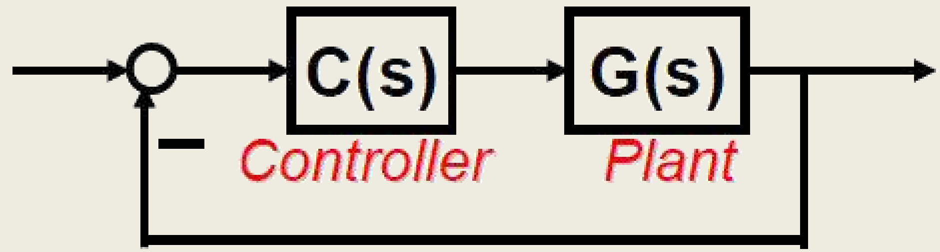
Design

- Design specs
- ➤ Root locus
- Frequency domain
- ✓ PID & Lead-lag
- Design examples

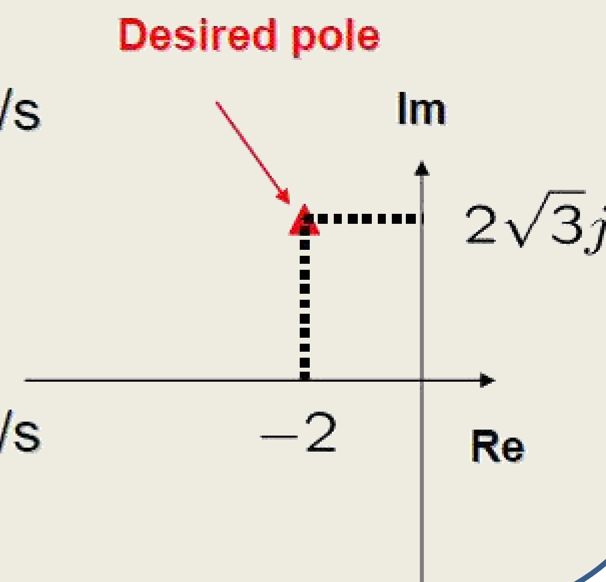
A design example (review)

- Consider a system

$$G(s) = \frac{4}{s(s+2)}$$



- Analysis of CL system for C(s)=1
  - Damping ratio ζ=0.5
  - Undamped natural freq. ω_n=2 rad/s
  - Ramp-error constant Kv=2
- Performance specification
  - Damping ratio ζ=0.5
  - Undamped natural freq. ω_n=4 rad/s
  - Ramp-error constant Kv=50



Lead & lag compensators (review)

- Lead compensator
  - Improve transient response
  - Improve stability

$$C_{Lead}(s) = K_1 \frac{s + z_1}{s + p_1}$$

- Lag compensator
  - Reduce steady state error

$$C_{Lag}(s) = K_2 \frac{s + z_2}{s + p_2}$$

- Lead-lag compensator
  - Take into account all the above issues.

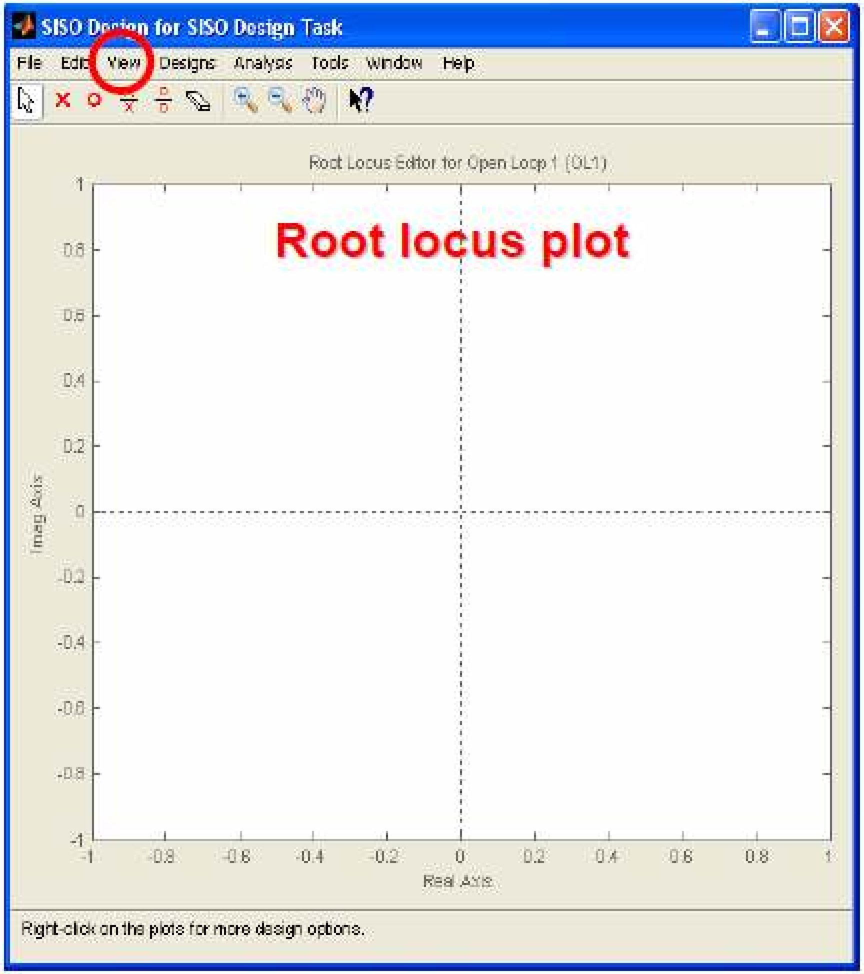
$$C_{LL}(s) = \underbrace{4.675 \frac{s + 2.9}{s + 5.4}}_{Lead} \cdot \underbrace{\frac{s + 0.25}{s + 0.025}}_{Lag}$$

SISO Design Tool in Matlab

- Graphical-user interface (GUI) that allows you to design compensators.
- Type "sisotool" in Matlab prompt.

```
>> sisotool
```

- Select "Root locus" from *View→ Design Plots configuration*.

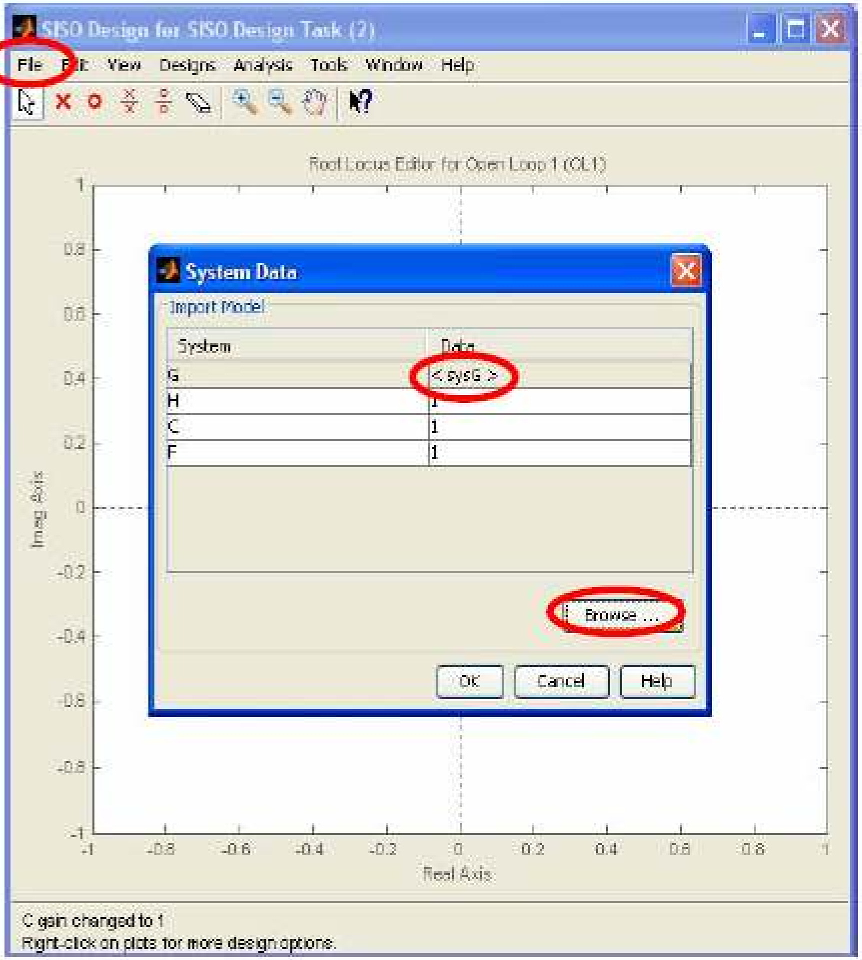


SISO Design Tool (cont'd)

- Input the plant

```
>> num=4;
>> den=[1 2 0];
>> sysG=tf(num,den);
```

- Import the plant sysG from *File → Import*

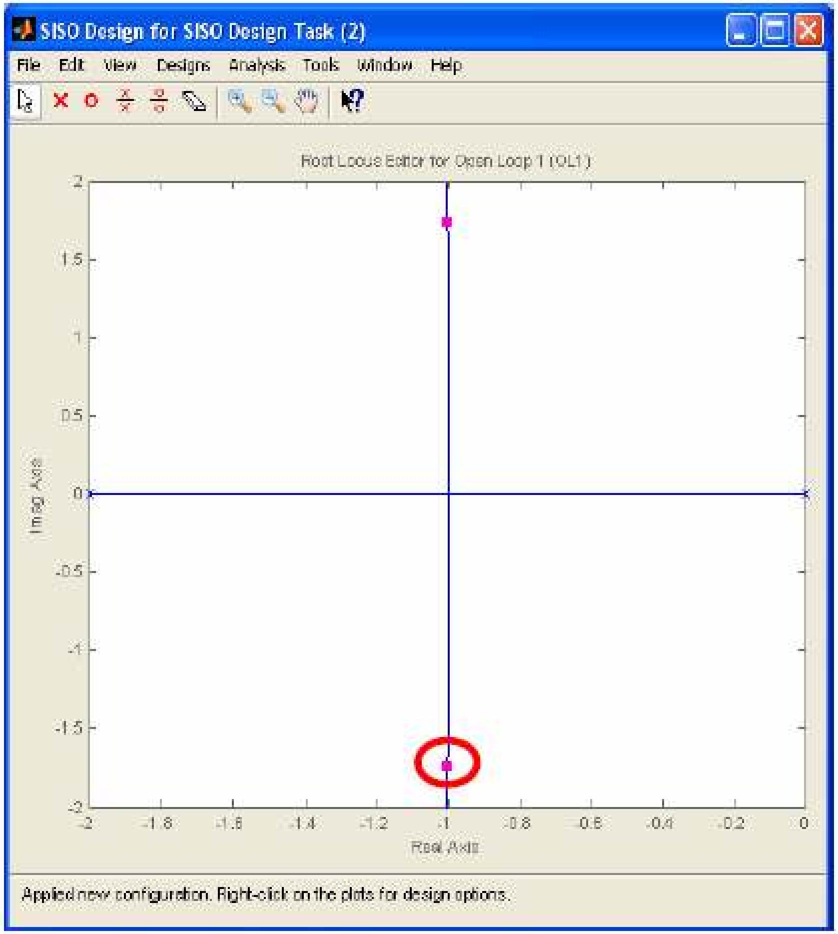


SISO Design Tool (cont'd)

- You will see the root locus plot for

$$G(s) = \frac{4}{s(s+2)}$$

- The default setting C(s)=1.
- The corresponding closed-loop poles are indicated by pink squares.

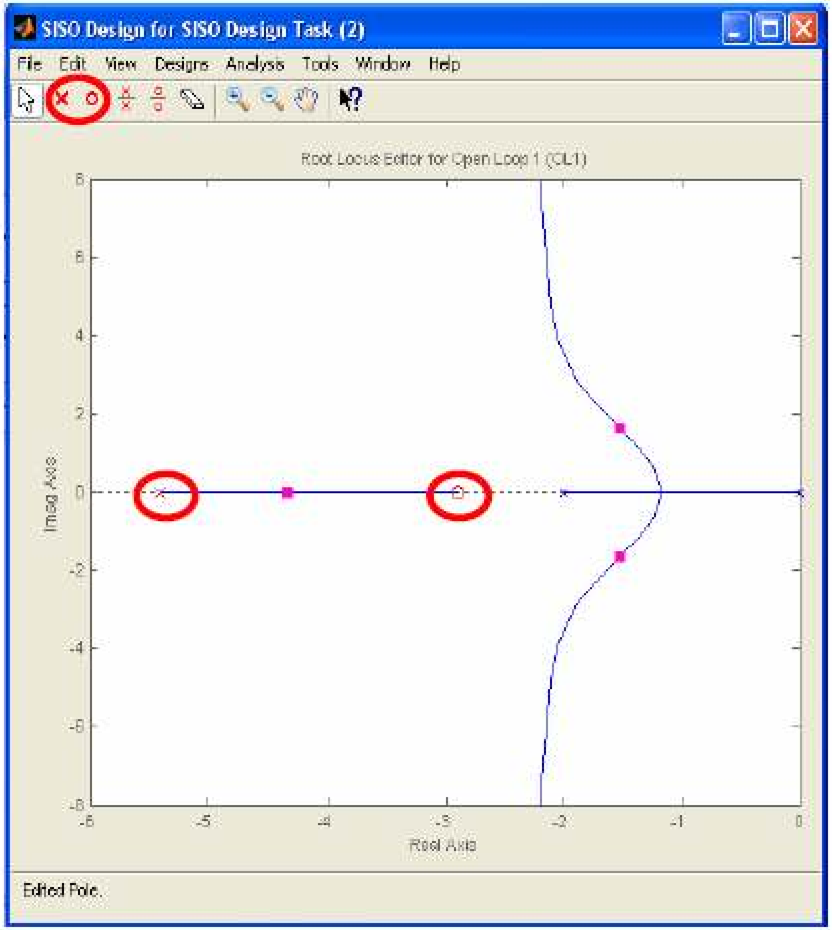


SISO Design Tool (cont'd)

- Add a pole & a zero of a compensator, and adjust its gain:

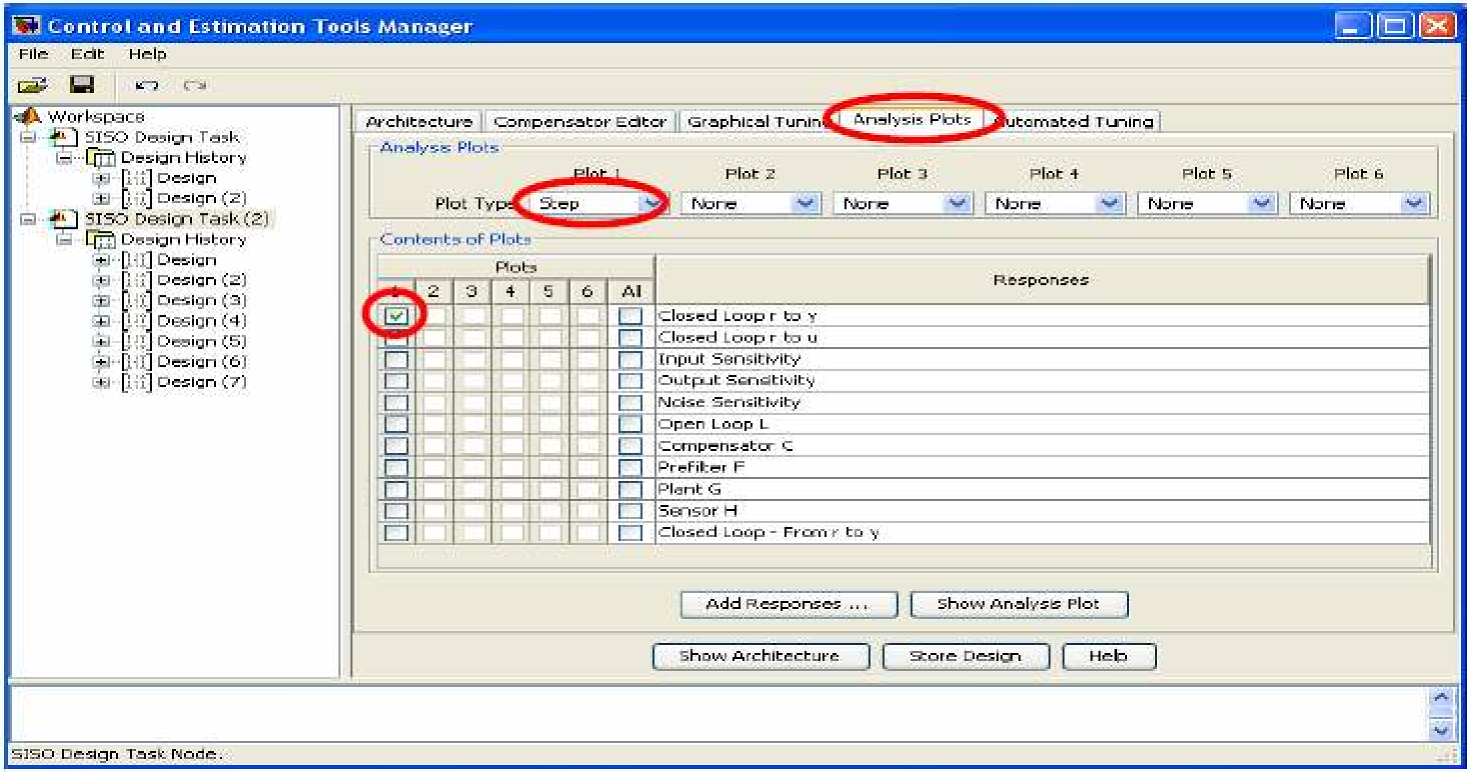
$$C_{Lead}(s) = 4.675\frac{s+2.9}{s+5.4}$$

- If necessary, move the pole and zero
  - by click-and-drag, or
  - *Design → Edit Compensator…*



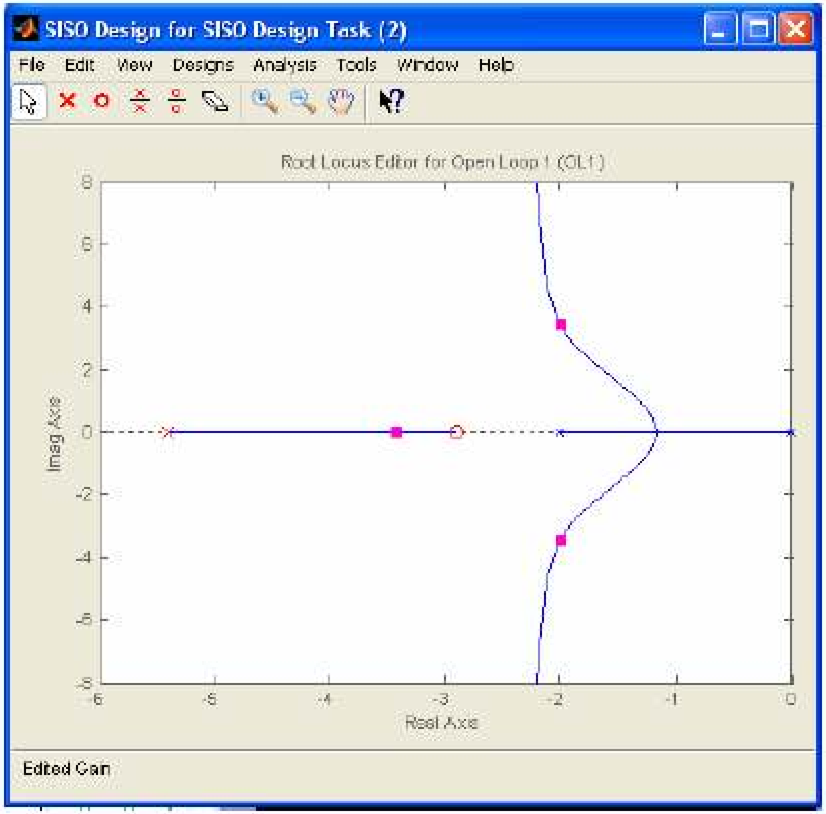
SISO Design Tool (cont'd)

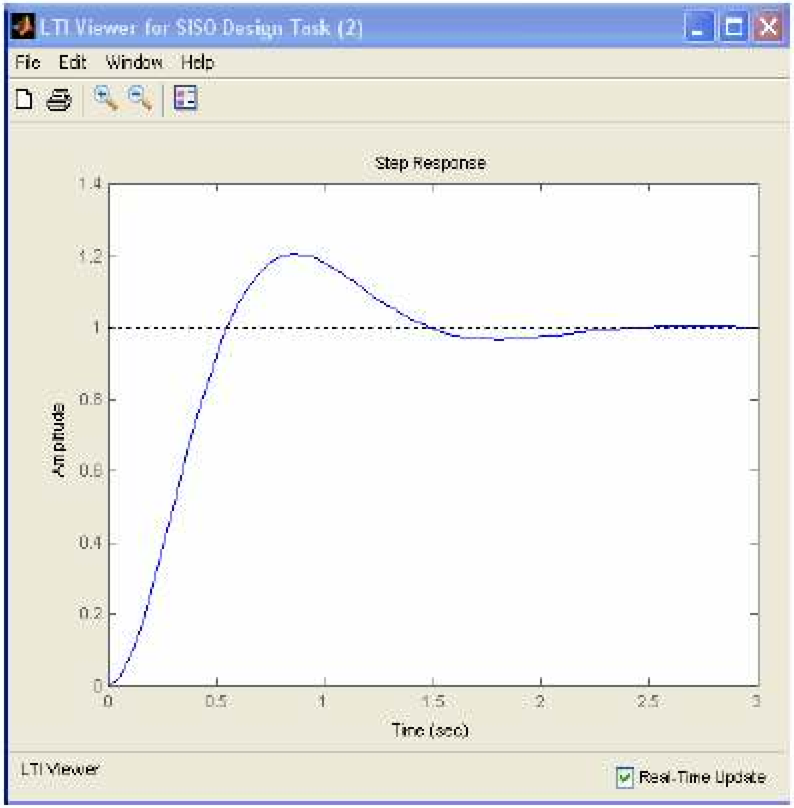
- See the step response
  - *Design → Edit Compensator… → Analysis Plots*



SISO Design Tool (cont'd)

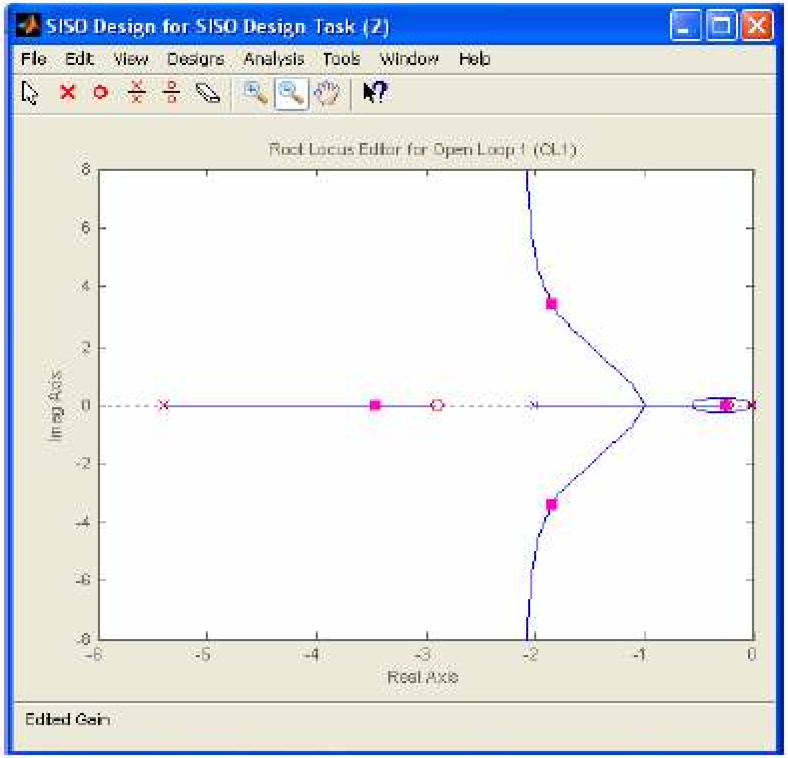
- Modify the pole/zero for "good" step response

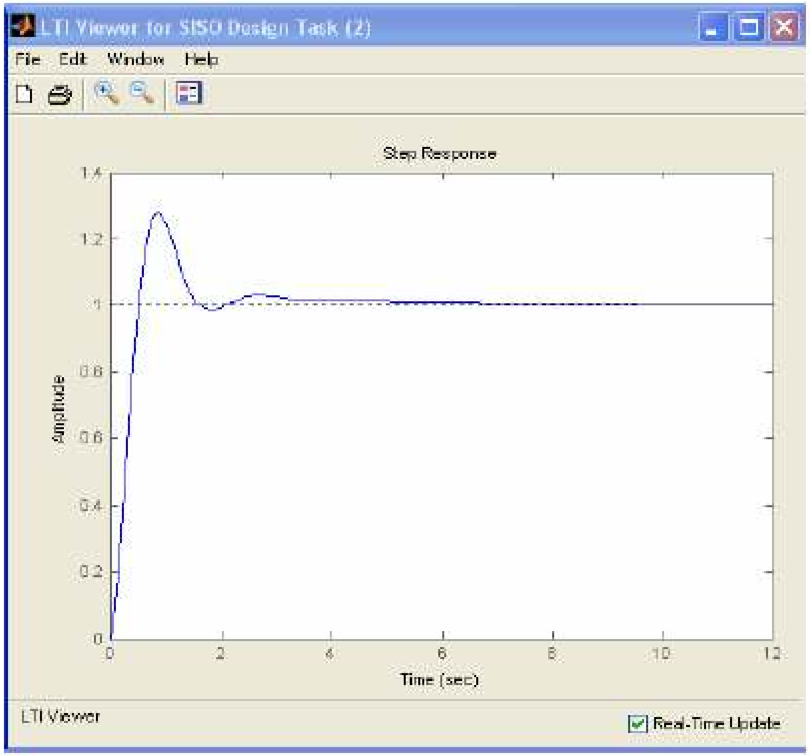




SISO Design Tool (cont'd)

- Add a pole & a zero of $C_{Lag}(s) = \dfrac{s + 0.25}{s + 0.025}$
- If necessary, adjust controller parameters.





Summary

- Controller design based on root locus
  - Lead compensator design to improve stability and transient response.
  - Lag compensator design to improve steady-state error.
  - Lead-lag compensator design to improve stability, transient and SS responses.
- PID Control
  - Most popular controller in industry
  - Simple controller structure
  - Simple controller tuning
  - Model-free methods for design are available
- Design based on root locus in Matlab

  ======================================

  ===================
- Next

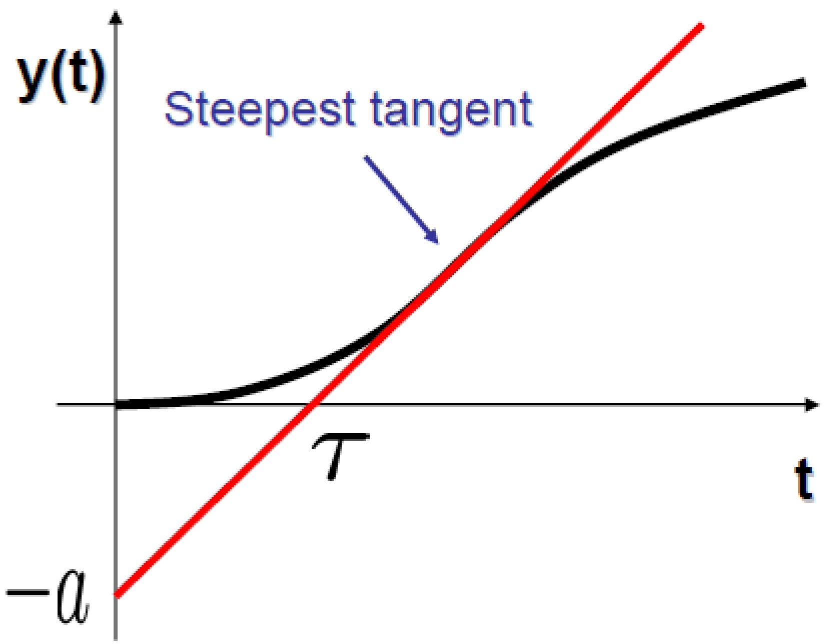
  m

  - Lecture 5 Assignment – due next Monday

Ziegler-Nichols PID tuning rules

- Step response method (for only stable systems)

Open-loop step response → **PID parameters**



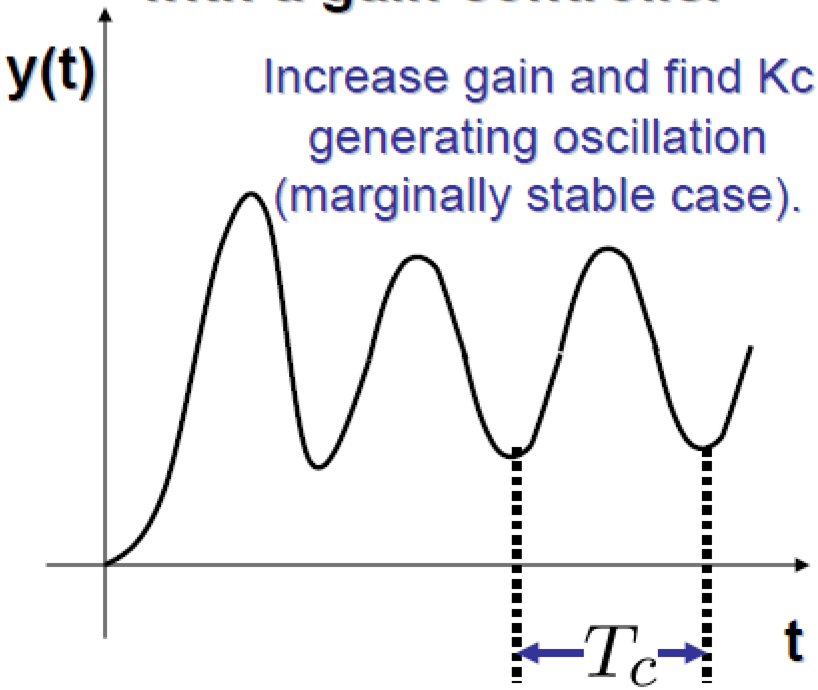
$$C(s) = K_p\left(1 + \frac{1}{T_I s} + T_D s\right)$$

Type	K_p	T_I	T_D
P	$1/a$		
PI	$0.9/a$	3τ	
PID	$1.2/a$	2τ	0.5τ

Ziegler-Nichols PID tuning rules

- Ultimate sensitivity method

Closed-loop step response with a gain controller → **PID parameters**



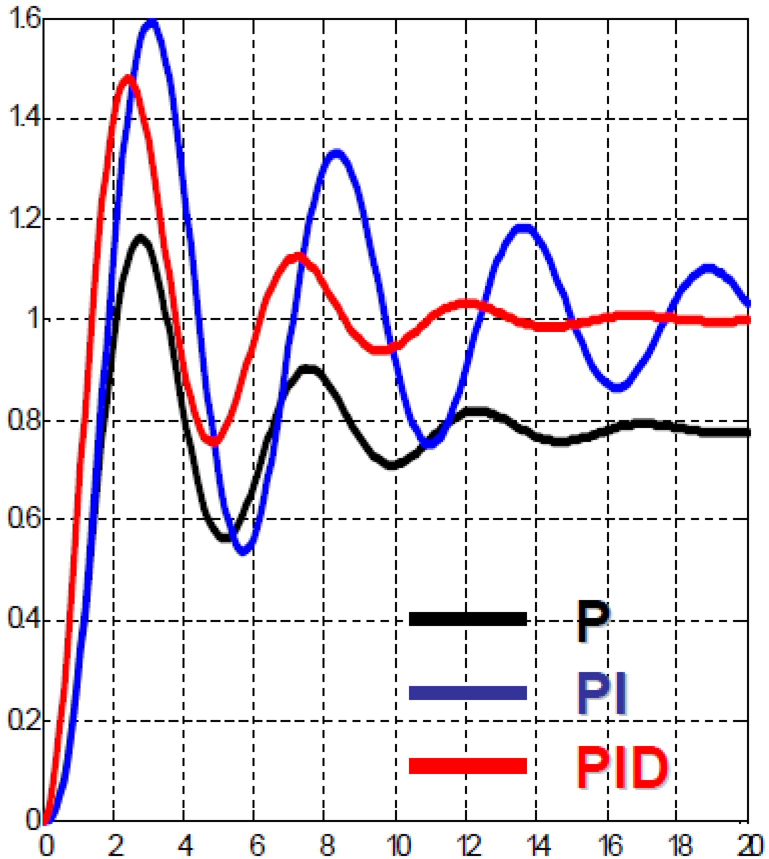
$$C(s) = K_p\left(1 + \frac{1}{T_I s} + T_D s\right)$$

Type	K_p	T_I	T_D
P	$0.5K_c$		
PI	$0.4K_c$	$0.8T_c$	
PID	$0.6K_c$	$0.5T_c$	$0.125T_c$

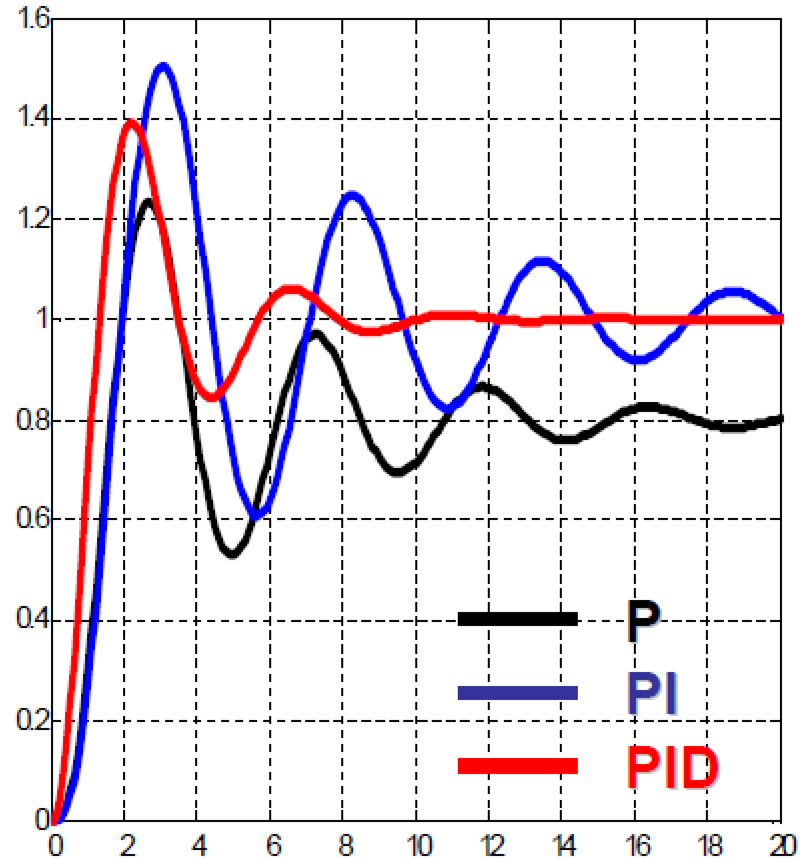
Example revisited

$$G(s) = \frac{1}{(s+1)^3}$$

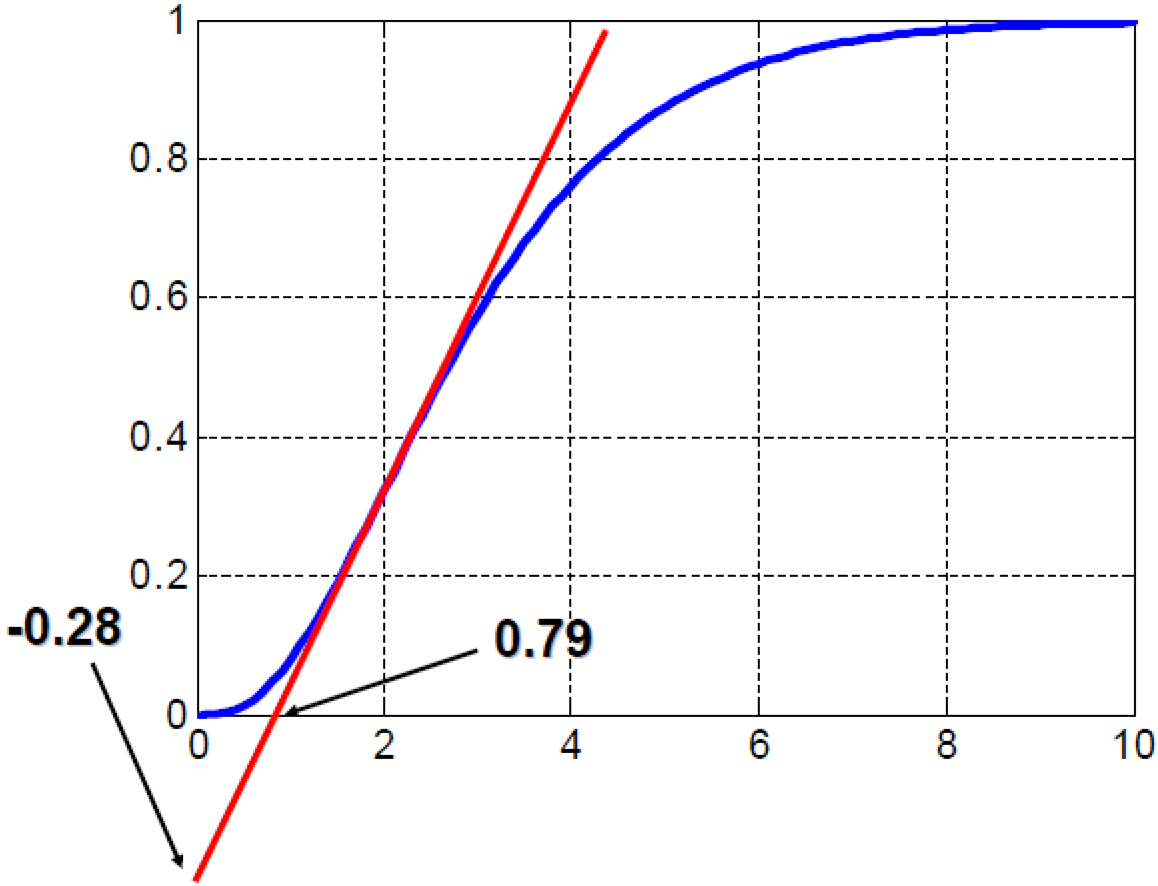
- Step response method



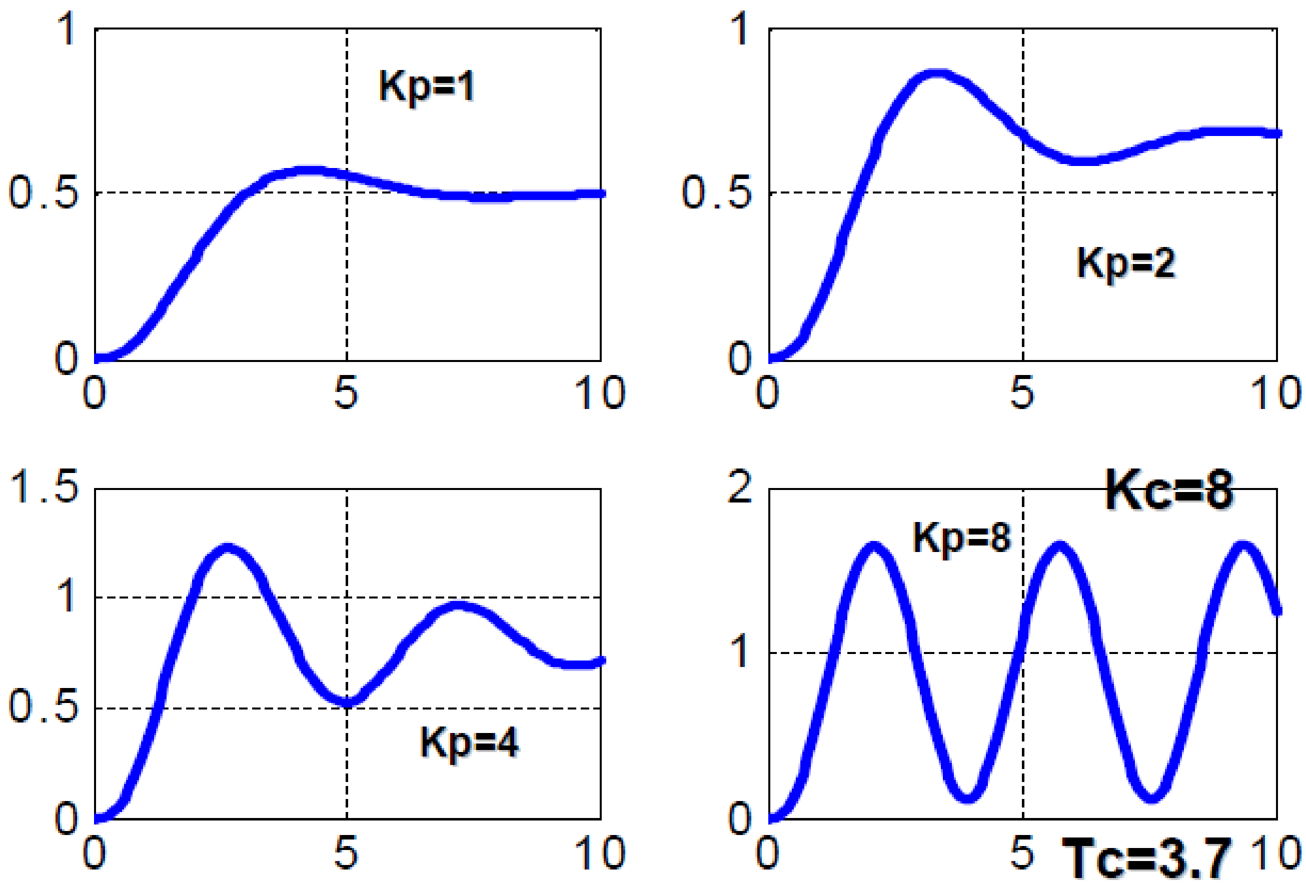
- Ultimate sensitivity



Open-loop step response for "step response method"



Closed-loop step responses for "Ultimate sensitivity method"



Assignment

1. For the system with the open loop function:

$$G(s) = \frac{1}{s(s+1)(s+3)}$$

Determine if the following points are on the root locus. (Use the Angle condition)

Angle Condition:

If $L(s) = \frac{(s+z)}{(s+p_1)(s+p_2)}$ ➔ $\angle L(s) = \angle \frac{(s+z)}{(s+p_1)(s+p_2)} == \angle(s+z) - \angle(s+p_1) - \angle(s+p_2) = \emptyset - \theta_1 - \theta_2$

For a point to be on root locus ➔ $\emptyset - \theta_1 - \theta_2 = 180\ deg$

(a) $s = -0.5$

(b) $s = \sqrt{3}j$

(c) $s = -1 + j$

End of Lecture 7