Systems Analysis and Control

Matthew M. Peet
Arizona State University

Lecture 16: Generalized Root Locus and Controller Design

Overview

In this Lecture, you will learn:

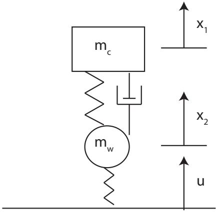
Generalized Root Locus?

- What about changing *OTHER* parameters
- T_D, T_I, et c.
- mass, damping, et c.

Compensation via Root-Locus

- Introduction
- Pole-Zero Compensation
- Lead-Lag

We may want to know the effect of changing other parameters.
Examples:

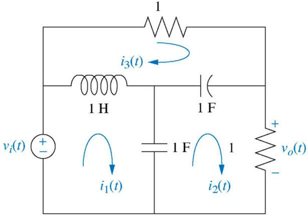


Physics (e.g. Suspension System)

- Spring Constant
- Damper
- Wheel Mass

Circuits (e.g. Toaster, Video Surveillance)

- Resisters
- Capacitors
- Inductors



Maybe there is no control at all!

Root Locus as a General Tool

What do parameters do?

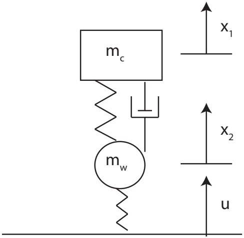
Suspension system: The full TF:

$$\frac{K_2(m_c s^2 + cs + K_1)}{m_c m_w s^4 + c(m_w + m_c)s^3 + (K_1 m_c + K_1 m_w + K_2 m_c)s^2 + cK_2 s + K_1 K_2}$$

The Effect of Damping Constant: c

- No Feedback
- Only examine c
  - Everything else is 1.

$$G(s) = \frac{s^2 + cs + 1}{s^4 + 2cs^3 + 3s^2 + cs + 1}$$



Where are the Poles?

$$s^4 + 2cs^3 + 3s^2 + cs + 1 = 0$$
$$s^4 + 3s^2 + 1 + c\left(2s^3 + s\right)$$
$$= d(s) + c\,n(s) = 0$$

Looks like the root locus!

Root Locus as a General Tool

What do parameters do?

$$G(s) = \frac{s^2 + cs + 1}{d(s) + c\,n(s)}$$

- $d(s) = s^4 + 3s^2 + 1$
- $n(s) = 2s^3 + s$

The root locus is the set of roots of

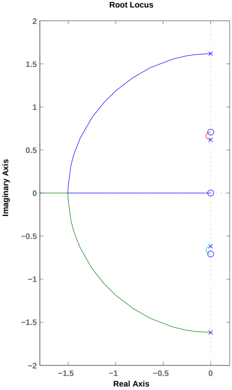
$$d(s) + kn(s)$$

We plot the root locus for

$$G_c(s) = \frac{n(s)}{d(s)} = \frac{2s^3 + s}{s^4 + 3s^2 + 1}$$

Note that G_c is totally fictional!

G_c must still be proper (n is lower degree than d).



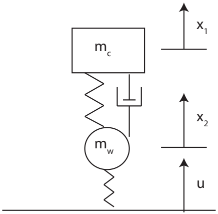
The effect of changing c is small.

Root Locus as a General Tool

Suspension Example: Damping Ratio

Root locus tells us:

- Changing c won't help with overshoot.
- We need Feedback!

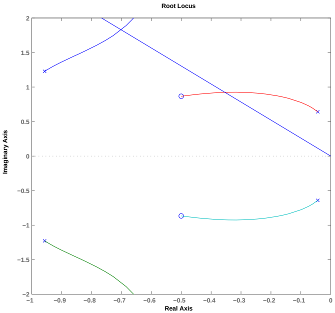


Set $c = 1$ and plot the root locus

$$G(s) = \frac{s^2 + s + 1}{s^4 + 2s^3 + 3s^2 + s + 1}$$

- Examine the gain at
  - $s_1 = -.3536 + .922\imath$
  - $s_2 = -.7 + 1.83\imath$
- Find Crossover Points
  - $k = 3.58$
  - $k = 2.61$

We'll want $k \cong 3$.



Closed Loop Transfer Function:

$$\frac{kG(s)}{1+kG(s)} = \frac{k(s^2+cs+1)}{k(s^2+cs+1)+s^4+2cs^3+3s^2+cs+1}$$

Can damping ratio get us to 30% overshoot?

- With feedback
- Set $k=3$ and plot root locus

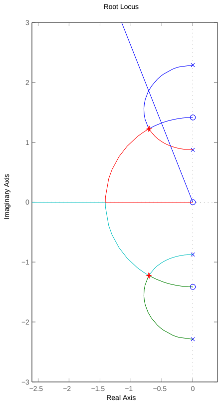
Closed Loop Transfer Function ($k=3$):

$$G_{kc}(s) = \frac{s^2+cs+1}{(s^4+6s^2+4)+c(2s^3+3s+s)}$$

Use `rlocfind` to pick off the best value of c.

Choose the point $s=-.71+1.215\imath$.

- Yields $c=\frac{|d(s)|}{|n(s)|}=1.414$



Root Locus as a General Tool

Suspension Example: Damping Ratio

Using $c = 1.414$ and $k = 3$, the closed-loop transfer function is

$$\frac{kG_c(s)}{1 + kG_c(s)} = \frac{3s^2 + 4.24s + 3}{s^4 + 2.8s^3 + 6s^2 + 5.7s + 4}$$

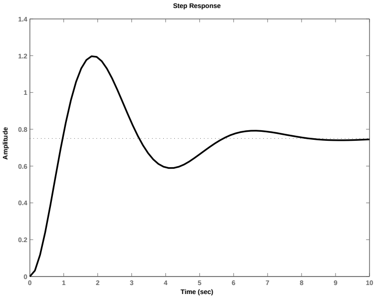
Which has repeated poles at

$$s_{1,2,3,4} = -.71 \pm 1.2\imath$$

Corresponds to an overshoot of

$$M_p = e^{\frac{\pi\sigma}{\omega}} = e^{-\frac{.71*\pi}{1.2}} = .18$$

The real overshoot is much bigger!

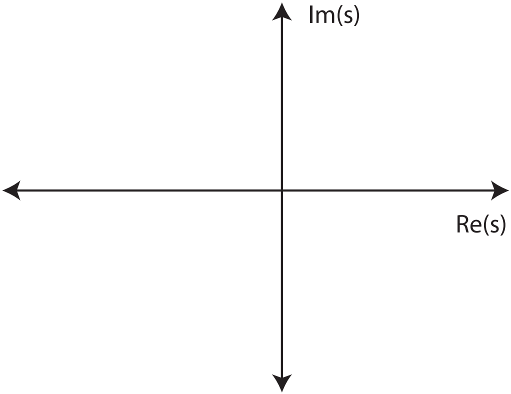


Root Locus as a General Tool

DIY Example

$$G(s) = \frac{s^2 + b^2 s + b}{s^3 + (7+b)s^2 + (12+b)s + b}$$

Find the optimal value of b.



Limitations of Root Locus

Root Locus isn't perfect

- Can only study one parameter at a time.
- What to do if root locus doesn't go where we want?

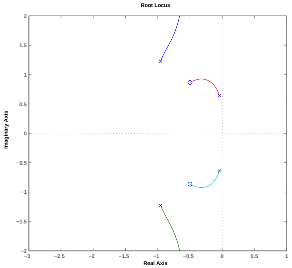
The Suspension Problem:

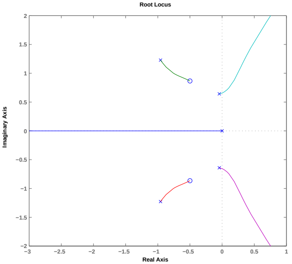
$$G(s) = \frac{s^2 + s + 1}{s^4 + 2s^3 + 3s^2 + s + 1}$$

Adding a pole at the origin has a negative effect.

Question:

Would adding a zero have a positive effect?





Limitations of Root Locus

The Inverted Pendulum:

$$G(s) = \frac{1}{s^2 + .5}$$

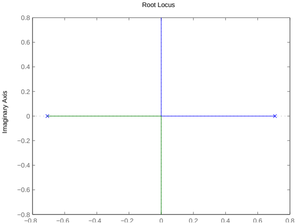
We used PD feedback $K(s) = k(1 + T_D s)$

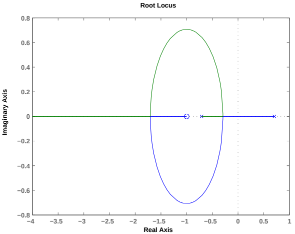
- Puts a zero at $s = \frac{1}{T_D}$

Conclusion:

- Adding a zero at $z = -1$ improves performance.

Can we generalize this?





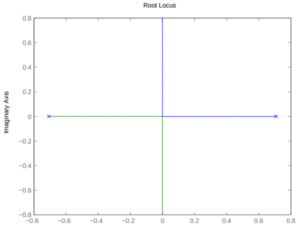
PID feedback:

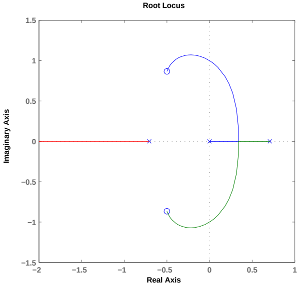
$$K(s) = k\left(1 + T_i \frac{1}{s} + T_D s\right)$$
$$= k\frac{T_D s^2 + s + T_I}{s}$$

Adds poles and zeros:

- Two zeros: $z_{1,2} = -\frac{1 \pm \sqrt{1 - 4T_D T_I}}{2T_D}$
- One pole: $p = 0$

Question: Is there a systematic way to add poles and zeros?



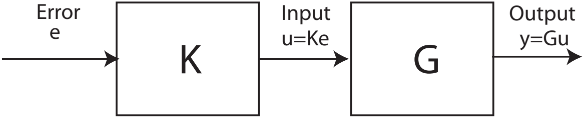


Adding Poles and Zeroes

How?

How To Add a Pole/Zero?

- What does it mean?



Constraint: The plant is fixed.

- $G(s)$ doesn't change.

The pole/zero must come from the controller. e.g.

What is a Controller?

- A system
  - Maps $e(t)$ to $u(t)$

Goal: Add a Zero

- Like PD control.

Controller:

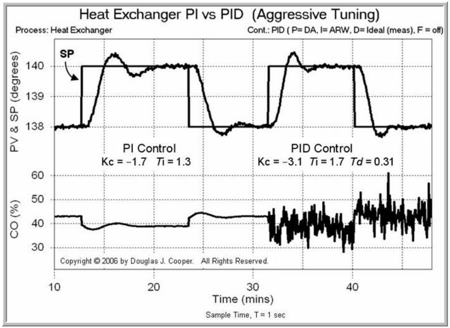
$$K(s) = k(s + z)$$

- **Input to Controller:** $\hat{e}(s)$
- **Output from Controller:**

$$\hat{u}(s) = k(s + z) \\ = ks\hat{e}(s) + kz\hat{e}(s)$$

Time-Domain:

$$u(t) = k\,e'(t) + kz\,e(t)$$



Problem: Requires differentiation.

$$e'(t) \cong \frac{e(t) - e(t - \tau)}{\tau}$$

Adding a Pole

Goal: Add a pole

Controller:

$$K(s) = k\frac{1}{s+p}$$

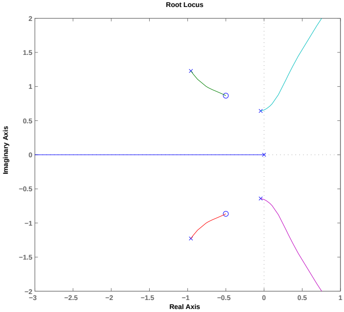
Input to Controller: $\hat{e}(s)$
Output from Controller: $\hat{u}(s) = \frac{k}{s+p}\hat{e}(s)$
Internal Variable: x.

- Frequency Domain

$$(s+p)x(s) = ke(s)$$
$$u(s) = x(s)$$

- Time-Domain

$$\dot{x}(t) = -px(t) + ke(t)$$
$$u(t) = x(t)$$



Adding a Pole:

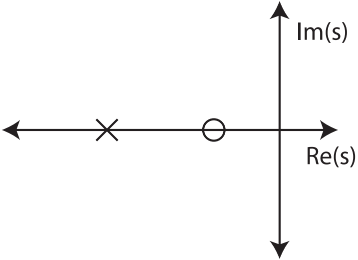
- Requires us to construct a dynamical system whose output is the control.
- Easier than adding a zero, but less useful
  - Zeros are better for attracting poles away from RHP.

Pole-Zero Compensation

The best way to modify the root locus is a using a pole-zero compensator.

- Adds a zero without differentiation

$$K(s) = k\frac{s-z}{s-p}$$



Input to Controller: $\hat{e}(s)$
Output from Controller: $\hat{u}(s) = k\frac{s-z}{s-p}\hat{e}(s)$
Doing long division:

$$\frac{s-z}{s-p} = 1 + \frac{z-p}{s-p}$$

Hence

$$\hat{u}(s) = k\hat{e}(s) + k\frac{z-p}{s-p}\hat{e}(s)$$

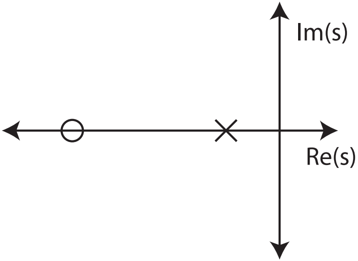
Internal Variable:

$$\hat{x}(s) = \frac{k(z-p)}{s-p}\hat{e}(s)$$

Pole-Zero Compensation

Internal Variable:

$$\hat{x}(s) = \frac{k(z-p)}{s-p}\hat{e}(s)$$



- Frequency Domain:

$$(s+p)x(s) = k(z-p)e(s)$$
$$u(s) = x(s) + k\hat{e}(s)$$

- Time-Domain:

$$\dot{x}(t) = -px(t) + k(z-p)e(t)$$
$$u(t) = x(t) + ke(t)$$

Artificial Dynamics:

- Controller State: $x(t)$
- No differentiation of $e(t)$!
- A zero should always be combined with a pole.

Lead Compensation

Definition 1.

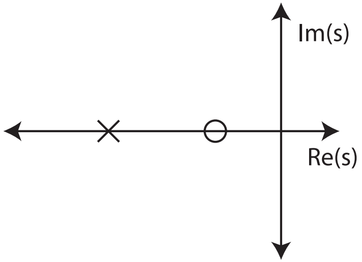
A **Lead Compensator** is a pole-zero compensator

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

where $p < z$.

Used when we really want a zero

- The pole has less effect than the zero.



Lead Compensation

Inverted Pendulum

$$G(s) = \frac{1}{s^2 - .5}$$

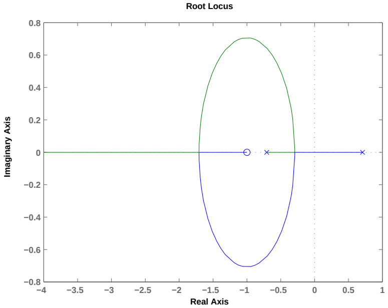


Figure : $K(s) = k(s+1)$

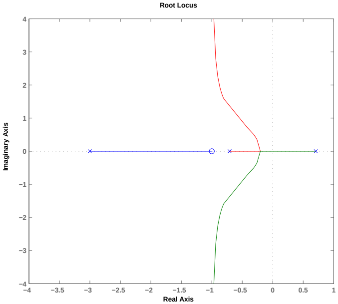


Figure : $K(s) = k\frac{s+1}{s+3}$

Lag Compensation

Definition 2.

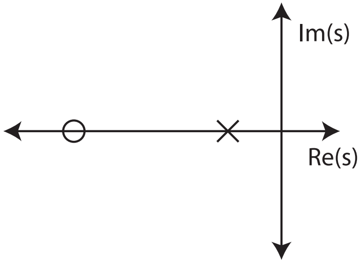
A **Lag Compensator** is a pole-zero compensator

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

where $z < p$.

Used when we really want a pole

- The zero has less effect than the pole.
- Doesn't increase the number of asymptotes.



$$G(s) = \frac{s^2 + s + 1}{s^4 + 2s^3 + 3s^2 + s + 1}$$

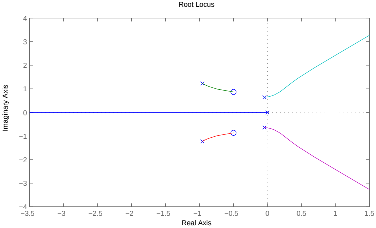


Figure : $K(s) = \frac{k}{s}$

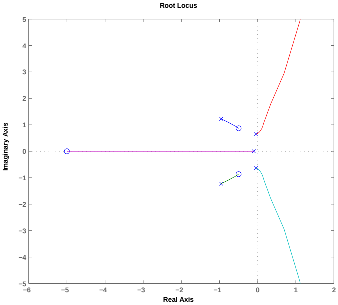


Figure : $K(s) = k\frac{s+5}{s+.1}$

Summary

What have we learned today?

Generalized Root Locus?

- What about changing *OTHER* parameters
- T_D, T_I, et c.
- mass, damping, et c.

Compensation via Root-Locus

- Introduction
- Pole-Zero Compensation
- Lead-Lag

Next Lecture: Pole-Zero Compensation and Notch Filters