# ECE 171A: Linear Control System Theory
# Lecture 19: Root Locus and PID control

Yang Zheng

Assistant Professor, ECE, UCSD

May 15, 2024

---

Reading materials: Ch 12.5, Ch 11.1-11.2

# Outline

# Outline

# Root locus - Overview

**Motivation**: System responses are affected by the locations of the poles of its transfer function in the complex domain, e.g., *stability*, *convergence speed*, etc.

- Feedback control can move the closed-loop system poles by designing an appropriate controller – **pole placement** (not covered in this course).

**What is the root locus method? — Another graphical tool**

- The **root locus** is a graph of the roots of the characteristic polynomial as a function of a parameter — give insight into the effects of the parameter.
- i.e., the **root locus** provides all possible pole locations as a system parameter (e.g., the controller gain) varies
- **Obtain the root locus** — find the roots of the closed loop characteristic polynomial for different values of the parameter (*easy for computers*).
- The general shape of the root locus can be obtained with very **little computational effort**, and that it often gives *considerable insight*.

# Root locus: Example 1

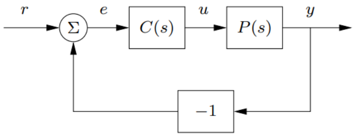



Figure: Feedback control system

- Consider a single-loop feedback control system with

$$P(s) = \frac{1}{s(s+2)}, \qquad C(s) = k$$

- The closed-loop transfer function from the reference $r$ to output $y$ is:

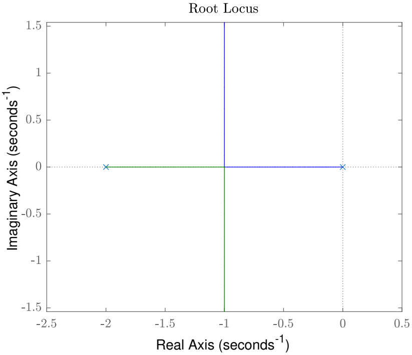
$$G_{\mathrm{yr}}(s) = \frac{kP(s)}{1 + kP(s)} = \frac{k}{s^2 + 2s + k}$$

- How do the closed-loop poles vary as a function of $k$?
  - We can actually compute the roots as $\lambda_{1,2} = -1 \pm \sqrt{1-k}$.

# Root Locus: Example 1

## Example

- Root locus for
$$P(s) = \frac{1}{s(s+2)}$$
- Matlab command: `rlocus(tf([1],[1 2 0]))`.



# Root Locus: Example 2

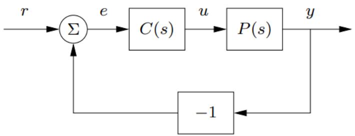



Figure: Feedback control system

## Example

- Consider a single-loop feedback control system with

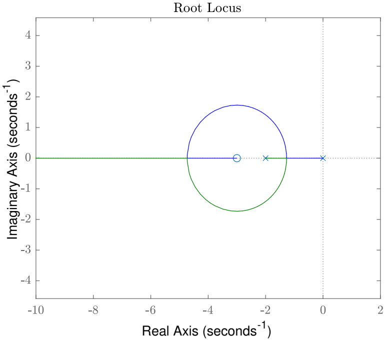
$$P(s) = \frac{(s+3)}{s(s+2)}, \qquad C(s) = k$$

- The closed-loop transfer function from $r$ to $y$ is:

$$G_{\mathrm{yr}}(s) = \frac{kP(s)}{1+kP(s)} = \frac{k(s+3)}{s^2+(2+k)s+3k}$$

# Root Locus: Example 2

- Root locus for
$$P(s) = \frac{(s+3)}{s(s+2)}$$
- Matlab command: `rlocus(tf([1 3],[1 2 0]))`.




- In this case, adding a stable zero in the open-loop system increases the relative stability of the closed-loop system by attracting the branches of the root locus.

# Root Locus: Example 3

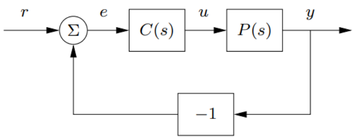



Figure: Feedback control system

## Example

- Consider a single-loop feedback control system with

$$P(s) = \frac{1}{s(s+2)(s+3)}, \qquad C(s) = k$$

- The closed-loop transfer function from $r$ to $y$ is:

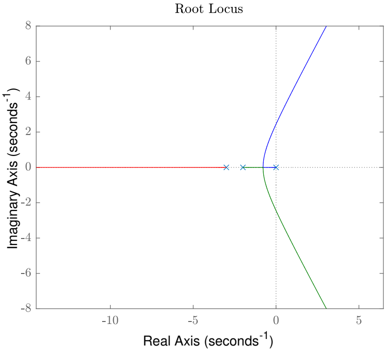
$$G_{\mathrm{yr}}(s) = \frac{k}{s^3 + 5s^2 + 6s + k}$$

# Root Locus: Example 3

- Root locus for

$$P(s) = \frac{1}{s(s+2)(s+3)}$$

- Matlab command: `rlocus(tf([1],[1 5 6 0]))`.




- In this case, adding a stable pole in the open-loop system makes the closed-loop system less stable (stable for some values of $k$);

# Root Locus: Definition

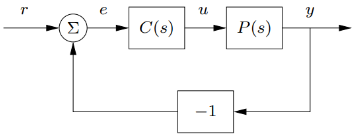



Figure: Feedback control system

- Closed-loop transfer function:

$$G_{\mathrm{yr}}(s) = \frac{kP(s)}{1 + kP(s)}$$

- The closed-loop poles satisfy:

$$1 + kP(s) = 0$$

- The **root locus** is the set of points $s$ such that $1 + kP(s) = 0$ as $k$ varies

# Root Locus: Definition

Consider the zeros and poles of $P(s)$ explicitly:

$$P(s) = \frac{b(s)}{a(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} + \cdots + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0}$$
$$= b_m \frac{(s - z_1) \cdots (s - z_m)}{(s - p_1) \cdots (s - p_n)}$$

- The closed loop characteristic polynomial is:

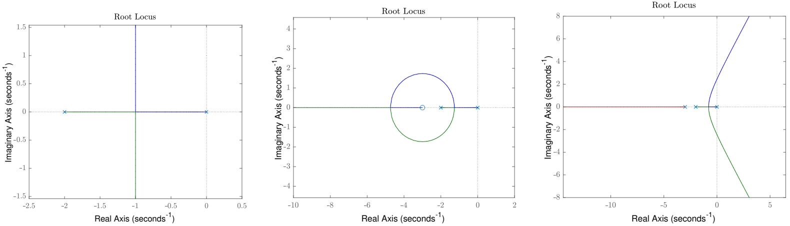
$$1 + kP(s) = 0 \qquad \Rightarrow \qquad a_{\mathrm{cl}}(s) := a(s) + kb(s) = 0$$

- The closed loop poles are the roots of $a_{\mathrm{cl}}(s)$.
- The **root locus** is a graph of the roots of $a_{\mathrm{cl}}(s)$ as the gain $k$ is varied from $0$ to $\infty$.
- Since the polynomial $a_{\mathrm{cl}}(s)$ has degree $n$, the plot will have $n$ branches.

# Starting and ending points of Root locus

- Each branch starts at a different open-loop pole.
- $m$ of the branches end at different open-loop zeros.
- The remaining $n - m$ branches go to infinity

## Example




(a) $P(s) = \frac{1}{s(s+2)}$ (b) $P(s) = \frac{s+3}{s(s+2)}$ (c) $P(s) = \frac{1}{s(s+2)(s+3)}$

# Starting and ending points of Root locus

- The closed loop characteristic polynomial is:

$$1 + kP(s) = 0 \qquad \Rightarrow \qquad a_{\mathrm{cl}}(s) := a(s) + kb(s) = 0$$

- The **root locus** is a graph of the roots of $a_{\mathrm{cl}}(s)$ as the gain $k$ is varied from $0$ to $\infty$.

**Starting points** when $k = 0$: we have $a_{\mathrm{cl}}(s) := a(s) + kb(s) = a(s)$.

- The closed-loop poles are equal to the open-loop poles.
- Open-loop poles at $s = p$ with multiplicity $l \Rightarrow$

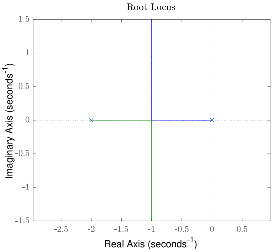
$$a(s) + kb(s) = (s - p)^l \tilde{a}(s) + kb(s) \approx (s - p)^l \tilde{a}(p) + kb(p) = 0$$

For **small value** of $k$, we have the roots are

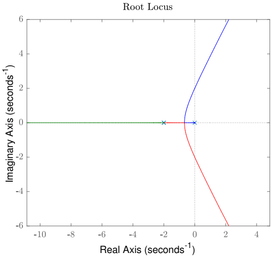
$$s = p + \sqrt[l]{-kb(p)/\tilde{a}(p)}$$

- The root locus has a **star pattern** with $l$ branches from the open-loop pole $s = p$, and the angle between two neighboring branches is $\frac{2\pi}{l}$.

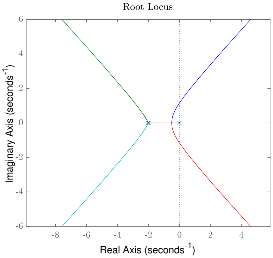
# Examples



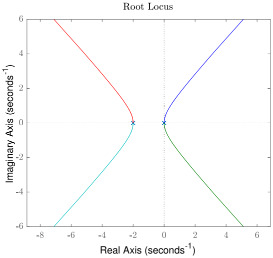

(a) $P(s) = \frac{1}{s(s+2)}$




(b) $P(s) = \frac{1}{s(s+2)^2}$




(c) $P(s) = \frac{1}{s(s+2)^3}$




(d) $P(s) = \frac{1}{s^2(s+2)^2}$

# Starting and ending points of Root locus

- The closed loop characteristic polynomial is:

$$1 + kP(s) = 0 \qquad \Rightarrow \qquad a_{\mathrm{cl}}(s) := a(s) + kb(s) = 0$$

- The **root locus** is a graph of the roots of $a_{\mathrm{cl}}(s)$ as the gain $k$ is varied from $0$ to $\infty$.

**Ending points** when $k$ goes to infinity: we have

$$a_{\mathrm{cl}}(s) := b(s)\left(\frac{a(s)}{b(s)} + k\right) \approx b(s)\left(\frac{s^{n-m}}{b_0} + k\right)$$

- For large $K$, the **closed-loop poles** are approximately the **roots (zeros of $P(s)$)** of $b(s)$ and

$$\sqrt[n-m]{-b_0 k}$$

- A better approximation of the **closed-loop poles** is

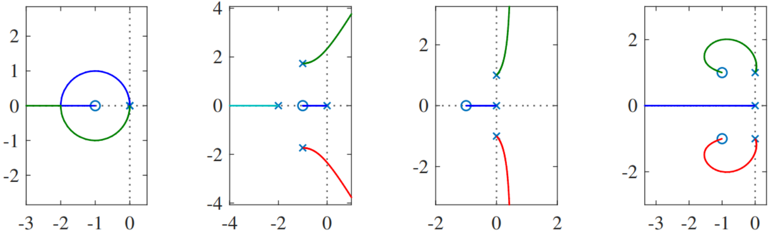
$$s = s_0 + \sqrt[n-m]{-b_0 k}, \quad s_0 = \frac{1}{n-m}\left(\sum_{k=1}^{n} p_k - \sum_{k=1}^{m} z_k\right).$$

## Example

Show the root loci for the following open-loop transfer functions

$$P_a(s) = \frac{s+1}{s^2}, \qquad P_b(s) = \frac{s+1}{s(s+2)(s^2+2s+4)},$$

$$P_c(s) = \frac{s+1}{s(s^2+1)}, \quad P_d(s) = \frac{s^2+2s+2}{s(s^2+1)}.$$




(a) $P_{\mathrm{a}}(s)$ (b) $P_{\mathrm{b}}(s)$ (c) $P_{\mathrm{c}}(s)$ (d) $P_{\mathrm{d}}(s)$

# Outline

# Overview

**Proportional-integral-derivative (PID) control** is by far the most common way of using feedback in engineering systems

- A survey of controllers for more than 100 boiler-turbine units: 94.4% of all controllers were PI, 3.7% PID, and 1.9% used advanced control.

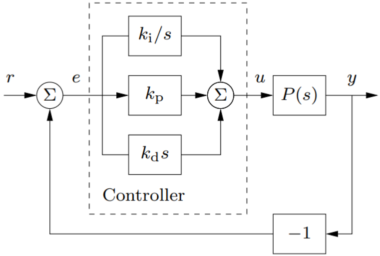



Figure: PID using error feedback

PID control

- the proportional term (P) — the **present** error;
- the integral term (I) — the **past** errors;
- the derivative term (D) — anticipated **future** errors.

- PID control appears in both **simple and complex** systems: as stand-alone controllers, as elements of hierarchical, distributed control systems, etc.

# PID controller

Input/output relation

$$u = k_\mathrm{p} e + k_\mathrm{i} \int_0^t e(\tau) d\tau + k_\mathrm{d} \frac{de}{dt} = k_p \left( e + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de}{dt} \right).$$

- **Time constant** $T_\mathrm{i} = k_\mathrm{p}/k_\mathrm{i}$ (Integral time); $T_\mathrm{d} = k_\mathrm{d}/k_\mathrm{p}$ (Derivative time)
- Also known as **three-term controllers**.

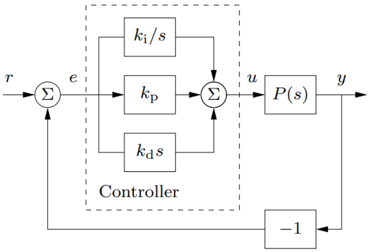



Figure: PID using error feedback

## Example

- Consider a system with dynamics

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

- Consider a controller $C(s)$
- The transfer function from reference to error is

$$G_{er}(s) = \frac{1}{1 + C(s)P(s)}.$$

# Numerical experiments

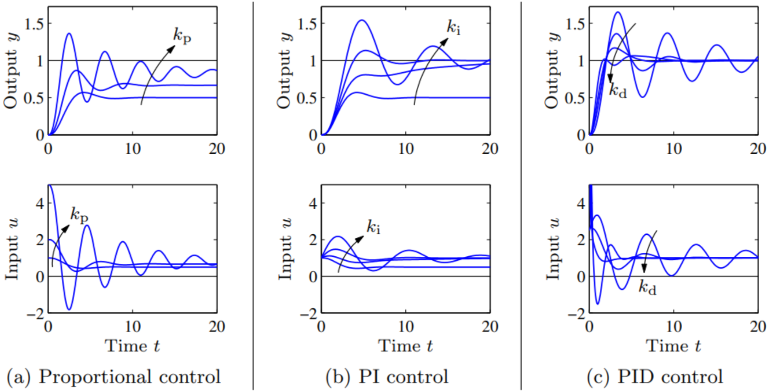



**Figure 11.2:** Responses to step changes in the reference value for a system with a proportional controller (a), PI controller (b), and PID controller (c). The process has the transfer function $P(s) = 1/(s+1)^3$, the proportional controller has parameters $k_p = 1$, 2, and 5, the PI controller has parameters $k_p = 1$, $k_i = 0$, 0.2, 0.5, and 1, and the PID controller has parameters $k_p = 2.5$, $k_i = 1.5$, and $k_d = 0$, 1, 2, and 4.

# Outline

# Summary

- **Root locus**: a graph of the closed-loop roots as $k$ is varied from $0$ to $\infty$.
  - The plot of root locus will have $n$ branches.
  - Each branch starts at a different open-loop pole.
  - $m$ of the branches end at different open-loop zeros.
  - The remaining $n - m$ branches go to infinity.

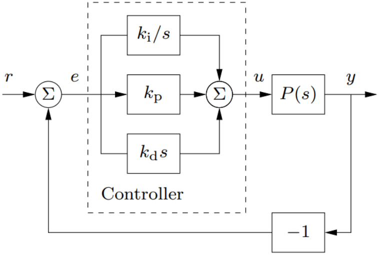



Figure: PID using error feedback

**PID control**

- the proportional term (P) — the **present** error;
- the integral term (I) — the **past** errors;
- the derivative term (D) — anticipated **future** errors.