Control Engineering
Lecture 11
ver. 1.3.1.3

Claudio De Persis

Last lecture

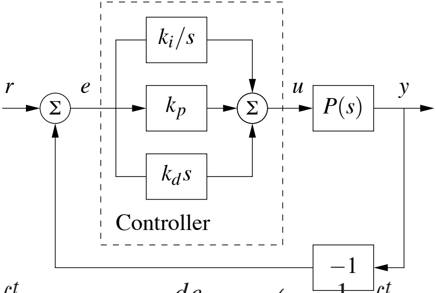
- Bode diagrams
- **Frequency domain analysis** (Chapter 9 of the textbook)
  - Nyquist plot
  - Nyquist stability theorems

Today

- **Frequency domain analysis** (Chapter 9 of the textbook)
  - Conditional stability
  - Stability margins
- **PID control** (Chapter 10 of the textbook)
  - First control design in the frequency domain

PID control

PI(D) control is the most common feedback control in engineering systems



$$u = k_p e + k_i \int_0^t e(\tau)\,d\tau + k_d \frac{de}{dt} = k_p\Big(e + \frac{1}{T_i}\int_0^t e(\tau)\,d\tau + T_d \frac{de}{dt}\Big)$$

k_p is the gain, $T_i = \frac{k_p}{k_i}$ the reset time, $T_d = \frac{k_d}{k_p}$ is the derivative time

Transfer function

$$U(s) = (k_p + \frac{k_i}{s} + k_d s)E(s) \Rightarrow C(s) = \frac{k_p s + k_i + k_d s^2}{s}$$

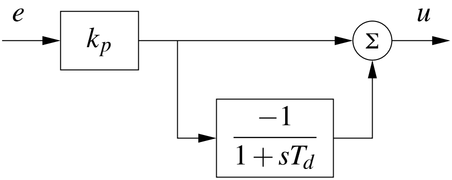
The design of a PID control amounts to the tuning of the gains k_p, k_i, k_d or k_p, T_i, T_d

PID control

PID embeds predictive action (Euler approximation $\frac{de}{dt} \approx \frac{e(t+T_d)-e(t)}{T_d}$ if T_d very small)

$$\begin{aligned} k_p e(t) + k_d \frac{de(t)}{dt} &= k_p \left(e(t) + T_d \frac{de(t)}{dt} \right) \approx k_p \left(e(t) + T_d \frac{e(t+T_d)-e(t)}{T_d} \right) \\ &= k_p e(t+T_d) \end{aligned}$$

Actual implementation of the derivative action $U(s) = k_p \overbrace{(1 - \frac{1}{1+sT_d})}^{\frac{sT_d}{1+sT_d}} E(s)$



(b) Derivative action

$$G_{ue}(s) = k_p \left(1 - \frac{1}{1+sT_d} \right) = k_p \frac{sT_d}{1+sT_d}$$

For frequencies $\ll 1/T_d$ the transfer function approximates that of a pure derivative control action (think of its Bode diagrams) – hence in practice T_d is designed such that $T_d \ll 1$

PID control

Implementing derivative control action is usually inaccurate and typically amplifies noise. Consider the noisy signal [Lewis, E6.20]

$$y(t) = A_s \sin(\omega_s t) + A_n \sin(\omega_n t + \varphi_n)$$

where $A_s \sin(\omega_s t)$ is the signal and $A_n \sin(\omega_n t + \varphi_n)$ is the noise, with $\omega_n > \omega_s$. The so-called SNR (signal-to-noise ratio) is

$$\mathrm{SNR}_y = \frac{|A_s|}{|A_n|}$$

When the signal $y(t)$ is processed via a derivative action, one obtains

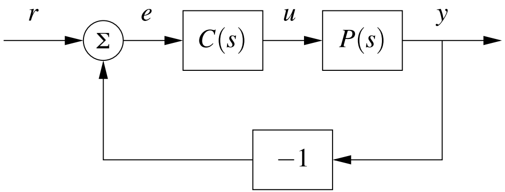
$$\dot{y}(t) = A_s\omega_s \cos(\omega_s t) + A_n\omega_n \cos(\omega_n t + \varphi_n)$$

and this signal has a much lower SNR

$$\mathrm{SNR}_{\dot{y}} = \frac{|A_s|\omega_s}{|A_n|\omega_n}$$

Such a bad effect of the derivative action on signals can also be seen using the Bode diagram of s

P control

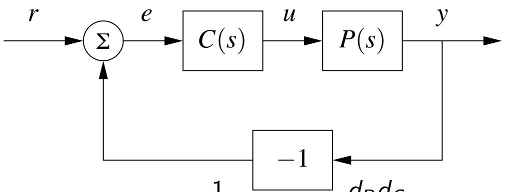


$$\boldsymbol{C(s) = k_p} \;\Rightarrow G_{er}(s) = \frac{1}{1+PC} \overset{\substack{P=\frac{n_P}{d_P} \\ C=\frac{n_C}{d_C}}}{=} \frac{d_P d_C}{d_P d_C + n_P n_C} = \frac{d_P}{d_P + k_p\, n_P}$$

If closed-loop system is stable, $P(s)$ has no pole or zero at $s = 0$, $r(t) = r$ for all $t \geq 0$ (constant reference signal), then $\lim_{t\to+\infty} e(t) = \lim_{t\to+\infty} r(t) - y(t)$ exists, is finite and can be computed as

$$\begin{aligned}
\boldsymbol{e_{steady}} &= \lim_{t\to+\infty} e(t) \\
&= \lim_{s\to 0} sE(s) = \lim_{s\to 0} sG_{er}(s)R(s) \\
&= \lim_{s\to 0} s\frac{d_P}{d_P + k_p n_P}\frac{r}{s} \\
&= \frac{d_P(0)}{d_P(0) + n_P(0)k_p} r = \frac{1}{1+P(0)k_p} r
\end{aligned}$$

P control



$$\boldsymbol{C(s) = k_p} \;\Rightarrow G_{er}(s) = \frac{1}{1+PC} = \frac{d_P d_C}{d_P d_C + n_P n_C} = \frac{d_P}{d_P + k_p\, n_P}$$

Discussion

- Asymptotic stability of the closed-loop system design k_p such that the poles of $G_{er}(s)$, i.e. roots of $d_P + k_p\, n_P = 0$, all with strictly negative real parts
- If $P(s)$ has no pole at $s = 0$ then

$$\boldsymbol{e_{steady} = \frac{1}{1+P(0)k_p} r} \overset{k_p \to +\infty}{\to} 0$$

- If $P(s)$ has a pole at $s = 0$ (integral action) then $e_{\text{steady}} = 0$
- If $P(s)$ has a zero at $s = 0$ then $e_{\text{steady}} = r$

P control

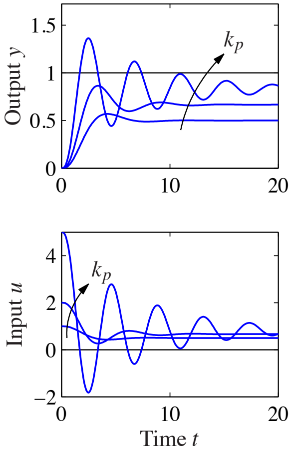
Constant steady state error; goes to zero as $k_p \to +\infty$; large k_p leads to oscillations (in the figure, the simulations are run for $P(s) = \frac{1}{(s+1)^3}$)

$$d_P + k_p\, n_P = s^3 + 3s^2 + 3s + k_p + 1$$

Routh table

$$\begin{array}{c|ll} 3 & 1 & 3 \\ 2 & 3 & k_p + 1 \\ 1 & -\frac{k_p - 8}{3} & \\ 0 & k_p + 1 & \end{array}$$

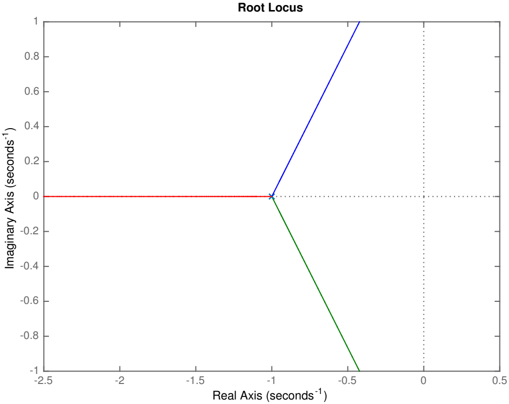
Closed-loop stability $-1 < k_p < 8$
For $k_p = 8$ two imaginary poles
For $k_p > 8$ two unstable poles



(a) Proportional control

P control

Constant steady state error; goes to zero as $k_p \to +\infty$; large k_p leads to oscillations. This can be understood by looking at how the roots of the closed-loop system characteristic polynomial evolve as k_p ranges in the interval $[0, +\infty)$.



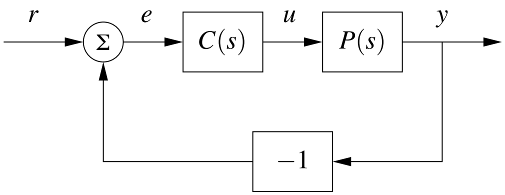
Roots are λ and $\sigma \pm i\omega$ with corresponding modes

$$\mathrm{e}^{\lambda t},\ \mathrm{e}^{\sigma t}\cos(\omega t),\ \mathrm{e}^{\sigma t}\sin(\omega t)$$

As $k_p \to 8$, $\sigma \to 0$ and the oscillations become more pronounced

PI control

PI control leads to zero steady state error without large gain

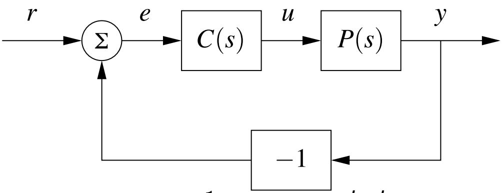


$$\boldsymbol{C(s) = \frac{k_p s + k_i}{s}} \Rightarrow G_{er}(s) = \frac{1}{1+PC} = \frac{d_P d_C}{d_P d_C + n_P n_C} = \frac{s d_P}{s d_P + n_P\left(k_p s + k_i\right)}$$

If closed-loop system is stable, $P(s)$ has no zero at $s = 0$, $r(t) = r$ for all $t \geq 0$ (constant reference signal), then $\lim_{t \to +\infty} e(t) = \lim_{t \to +\infty} r(t) - y(t)$ exists, is finite and can be computed as

$$\begin{aligned} \boldsymbol{e_{steady}} &= \lim_{s \to 0} sE(s) = \lim_{s \to 0} sG_{er}(s)R(s) \\ &= \lim_{s \to 0} s \frac{s d_P}{s d_P + n_P\left(k_p s + k_i\right)} \frac{r}{s} = \mathbf{0} \end{aligned}$$

PI control



$$\color{blue}{\boldsymbol{C(s) = \frac{k_p s + k_i}{s}}} \;\Rightarrow G_{er}(s) = \frac{1}{1+PC} = \frac{d_P d_C}{d_P d_C + n_P n_C} = \frac{s d_P}{s d_P + n_P\,(k_p s + k_i)}$$

Discussion

- Asymptotic stability of the closed-loop system design k_p, k_i such that the poles of $G_{er}(s)$, i.e. the roots of $sd_P + n_P\,(k_p s + k_i) = 0$, have all with strictly negative real parts
- The pole of $C(s)$ at $s = 0$ (integral action) guarantees $\mathrm{e}_{\mathrm{steady}} = 0$
- If $P(s)$ has a zero at $s = 0$ then the integral action is frustrated by the pole/zero cancellation

PI control

- Zero steady state error
- Increasing gain k_i leads to generally faster response but as k_i increases more it can also give rise to oscillations and eventually instability

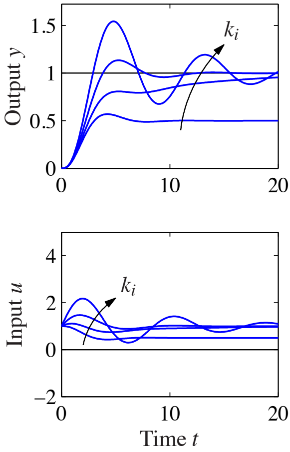
$$sd_P + (k_p s + k_i)\, n_P = s^4 + 3s^3 + 3s^2 + (k_p+1)s + k_i$$

Routh table

$$\begin{array}{c|lll} 4 & 1 & 3 & k_i \\ 3 & 3 & k_p+1 & \\ 2 & -\frac{k_p-8}{3} & k_i & \\ 1 & \frac{1}{\frac{k_p-8}{3}} \frac{9k_i + (k_p-8)(k_p+1)}{3} & & \\ 0 & k_i & & \end{array}$$

Closed-loop stability

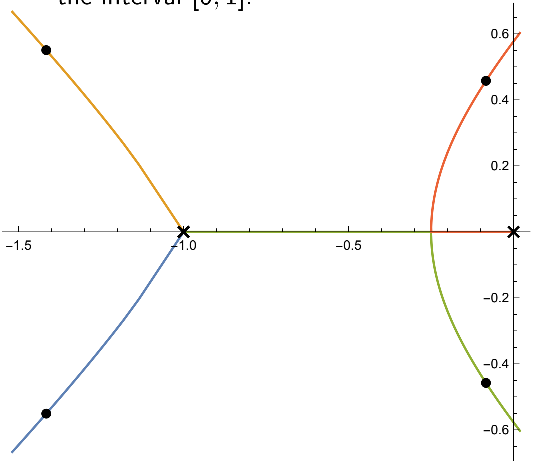
$$\begin{cases} -1 < k_p < 8 \\ 0 < k_i < -\frac{(k_p-8)(k_p+1)}{9} = -\frac{k_p^2 - 7k_p - 8}{9} \end{cases}$$



(b) PI control

PI control

Zero steady state error; large k_i leads to oscillations. The figure below represents the evolution of the poles of the closed-loop system for the process $P(s) = \frac{1}{(s+1)^3}$ controlled by $C(s) = \frac{k_i}{s}$ ($k_p = 0$) as k_i ranges in the interval $[0, 1]$.



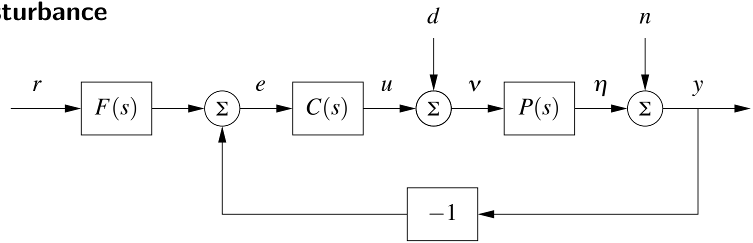
As k_i increases poles become $\sigma_1 \pm i\omega_1$ and $\sigma_2 \pm i\omega_2$ with corresponding modes

$$\mathrm{e}^{\sigma_i t} \cos(\omega_i t),\ \mathrm{e}^{\sigma_i t} \sin(\omega_i t),\ i = 1, 2$$

As $k_i \to \frac{8}{9}$, one pair of poles becomes dominant ($\sigma_1 \to 0$ with $\sigma_2 \ll \sigma_1$) giving rise to more pronounced oscillations

PI control

PI control rejects constant load disturbances ($r = n = 0$, $d \neq 0$) and **controller output settles at a value that compensate for the disturbance**



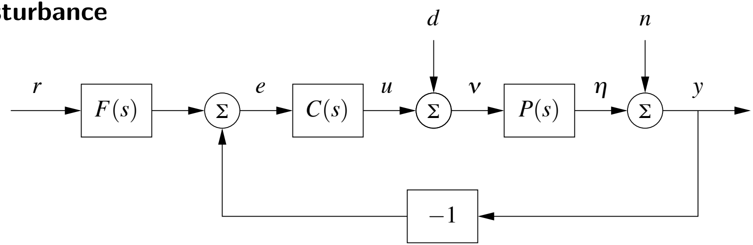
$$\boldsymbol{C(s) = \frac{k_p s + k_i}{s}} \Rightarrow G_{ud} = -\frac{PC}{1+PC} = -\frac{n_P n_C}{n_P n_C + d_P d_C}$$
$$= -\frac{n_P(k_p s + k_i)}{n_P(k_p s + k_i) + d_P s}$$

Closed-loop system is stable, $P(s)$ has no zero at $s = 0$, d constant load disturbance

$$\begin{aligned} \boldsymbol{u_{steady}} &= \lim_{s\to 0} sU(s) = \lim_{s\to 0} sG_{ud}(s)D(s) \\ &= \lim_{s\to 0} s\frac{-n_P(k_p s + k_i)}{n_P(k_p s + k_i) + d_P s}\frac{d}{s} = \boldsymbol{-d} \end{aligned}$$

PI control

PI control rejects constant load disturbances ($r = n = 0$, $d \neq 0$) and **controller output settles at a value that compensate for the disturbance**

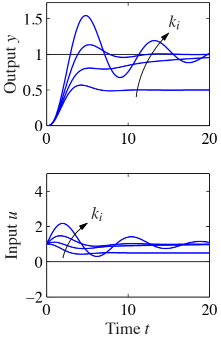


$$\boldsymbol{C(s) = \frac{k_p s + k_i}{s}} \Rightarrow G_{yd} = \frac{P}{1+PC} = \frac{n_P d_C}{n_P n_C + d_P d_C} = \frac{n_P s}{n_P(k_p s + k_i) + d_P s}$$

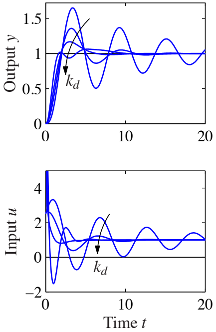
Closed-loop system is stable, $P(s)$ has no zero at $s = 0$, d constant load disturbance

$$\begin{aligned} \boldsymbol{y_{steady}} &= \lim_{s\to 0} sY(s) = \lim_{s\to 0} sG_{yd}(s)D(s) \\ &= \lim_{s\to 0} s\frac{n_P s}{n_P(k_p s + k_i) + d_P s}\frac{d}{s} = \mathbf{0} \end{aligned}$$

PID control



(b) PI control



(c) PID control

- Derivative action makes system faster if k_d not too large
- Oscillatory behavior for small k_d - damped sluggish response for large k_d
- It allows to deal with second order systems (systems with 2 poles)

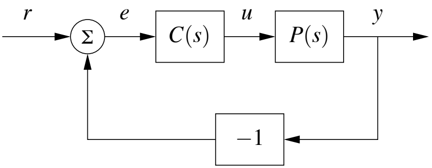
PI control for first order systems - stabilization

First order system

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

Second order system PI control

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



Closed-loop system

$$\frac{L}{1+L} = \frac{b(k_p s + k_i)}{s(s+a) + b(k_p s + k_i)} = \frac{b(k_p s + k_i)}{s^2 + (a + bk_p)s + bk_i}$$

Poles can be freely assigned via k_p, k_i

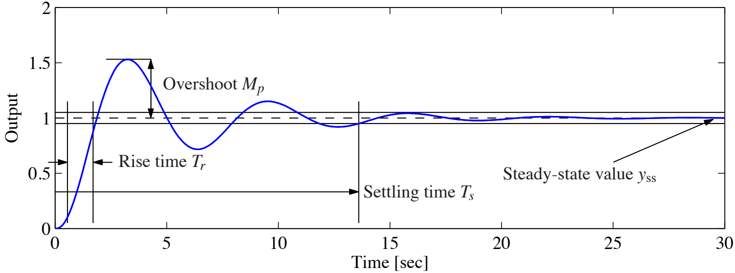
To assign closed-loop poles $s^2 + 2\zeta\omega_0 s + \omega_0^2$

$$k_p = \frac{2\zeta\omega_0 - a}{b}, \quad k_i = \frac{\omega_0^2}{b}$$

PI control for first order systems

PI control for first order systems

- Constant **reference tracking** (set point) and load **disturbance rejection** always achievable for first order systems by PI control
- Step response freely tunable



PI control for first order systems

PI control for first order systems

- Constant **reference tracking** (set point) and load **disturbance rejection** always achievable for first order systems by PI control
- Step response freely tunable

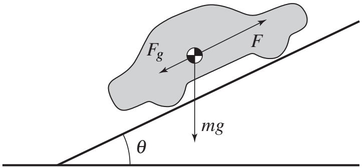
Properties of the step response for the closed-loop system (second-order)

Table 6.1: Properties of the step response for a second-order system with $0 < \zeta < 1$.

Property	Value	$\zeta = 0.5$	$\zeta = 1/\sqrt{2}$	$\zeta = 1$
Steady-state value	k	k	k	k
Rise time	$T_r \approx 1/\omega_0 \cdot e^{\varphi/\tan\varphi}$	$1.8/\omega_0$	$2.2/\omega_0$	$2.7/\omega_0$
Overshoot	$M_p = e^{-\pi\zeta/\sqrt{1-\zeta^2}}$	16%	4%	0%
Settling time (2%)	$T_s \approx 4/\zeta\omega_0$	$8.0/\omega_0$	$5.9/\omega_0$	$5.8/\omega_0$

PI control for first order systems

Example Cruise control (Section 3.1 and Example 5.11 of the textbook)



(a) Effect of gravitational forces

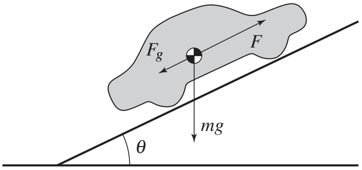
Linearized model

$$\frac{d(v-v_e)}{dt} = -a(v-v_e) + b(u-u_e) - g\theta$$

v velocity, u input from the engine, θ slope
(v_e, u_e) equilibrium pair such that $u = u_e + \frac{g\theta}{b} \Rightarrow v = v_e$

PI control for first order systems

Example Cruise control (Section 3.1 and Example 5.11 of the textbook)



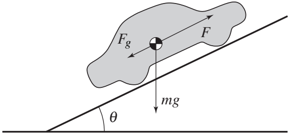
(a) Effect of gravitational forces

Linearized model - transfer function

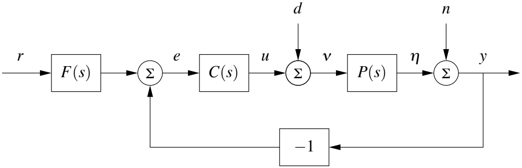
$$\begin{aligned} sV(s) = & \quad -aV(s) + bU(s) + \tfrac{1}{s}(av_e - bu_e - g\theta) \\ = & \quad -aV(s) + bU(s) + \tfrac{b}{s}\tfrac{av_e - bu_e - g\theta}{b} \\ & \quad \Downarrow \\ V(s) = & \quad \tfrac{b}{s+a}(U(s) + \underbrace{\frac{1}{s}\frac{av_e - bu_e - g\theta}{b}}_{D(s)}) \end{aligned}$$

PI control for first order systems

Example Cruise control (Section 3.1 and Example 5.11 of the textbook)



(a) Effect of gravitational forces



$$F(s) = 1,\ C(s) = \frac{k_p s + k_i}{s},\ r = \text{cruise velocity} = v_e,\ d = \frac{av_e - bu_e - g\theta}{b}$$

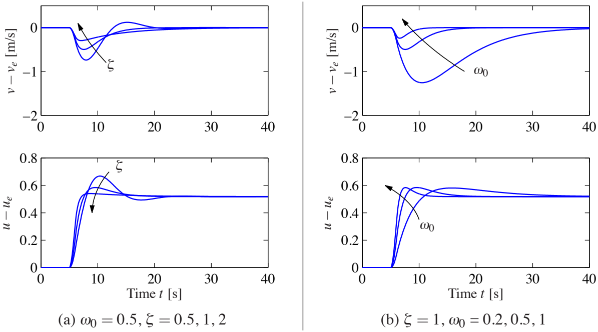
$$P(s) = \frac{b}{s+a}(U(s) + D(s)), \quad D(s) = d\frac{1}{s}$$

Cruise control guaranteed by PI controller

$$k_p = \frac{2\zeta\omega_0 - a}{b},\ k_i = \frac{\omega_0^2}{b} \Longrightarrow \frac{L(s)}{1+L(s)} = \frac{b(k_p s + k_i)}{s^2 + (a + bk_p)s + bk_i} = \frac{(2\zeta\omega_0 - a)s + \omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2}$$

PI control for first order systems

Example Cruise control (Section 3.1 and Example 5.11 of the textbook)



- Rise time, overshoot, settling time can be adjusted by designing k_p, k_i
- ω_0 compromise between response speed and control actions
- Large ω_0 gives fast response, less overshoot but requires fast actuators
- Large ζ gives less overshoot and reduced control effort
- Response with no overshoot improves comfort

PID control for second order systems

For a second order system

$$P(s) = \frac{b}{s^2 + a_1 s + a_2}$$

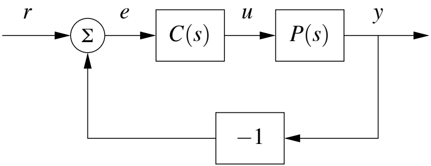
pole assignment is possible via PID control

$$C(s) = \frac{k_p s + k_i + k_d s^2}{s}$$

[Textbook, Exercise 10.2], [Tutorial 7]

Pole assignment - general case

Consider the negative feedback loop



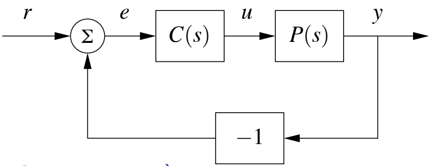
with

$$P(s) = \frac{num_p(s)}{den_p(s)} = \frac{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \ldots + b_1 s + b_0}{a_n s^n + a_{n-1}s^{n-1} + \ldots + a_1 s + a_0}, \quad a_n \neq 0 \text{ (strictly proper)}$$

and

$$C(s) = \frac{num_c(s)}{den_c(s)} = \frac{d_{n-1}s^{n-1} + d_{n-2}s^{n-2} + \ldots + d_1 s + d_0}{c_{n-1}s^{n-1} + c_{n-2}s^{n-2} + \ldots + c_1 s + c_0}$$

Pole assignment - general case



Theorem (Pole Assignment)

Let $P(s)$ and $C(s)$ be defined as before. Assume $num_p(s), den_p(s)$ are coprime, that is they have no common factors. Let

$$p_{des}(s) = g_{2n-1}s^{2n-1} + g_{2n-2}s^{2n-2} + \ldots + g_1 s + g_0$$

be an arbitrary polynomial of order $2n-1$. Then there exist polynomials $num_c(s), den_c(s)$ such that

$$num_p(s)num_c(s) + den_p(s)den_c(s) = p_{des}(s)$$

<u>Stabilization</u> If we choose the desired polynomial $p_{des}(s)$ such that the roots of $p_{des}(s) = 0$ have all strictly negative real parts, then in particular the controller $C(s)$ stabilizes the closed-loop system.

Pole assignment - general case

Proof (Sketch) Equating the coefficients of $num_p(s)num_c(s) + den_p(s)den_c(s) = p_{des}(s)$ one obtains the $2n \times 2n$ so-called *eliminant* matrix (# rows = # coefficients of p_{des}, # columns = # coefficients of den_c +# coefficients of num_c)

$$\begin{bmatrix} a_n & 0 & \dots & 0 & b_n = 0 & 0 & \dots & 0 \\ a_{n-1} & a_n & \dots & 0 & b_{n-1} & b_n = 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & \dots & a_n & b_1 & b_2 & \dots & 0 \\ a_0 & a_1 & \dots & a_{n-1} & b_0 & b_1 & \dots & b_n = 0 \\ 0 & a_0 & \dots & a_{n-2} & 0 & b_0 & \dots & b_{n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_0 & 0 & 0 & \dots & b_0 \end{bmatrix} \begin{bmatrix} c_{n-1} \\ \vdots \\ c_0 \\ d_{n-1} \\ \vdots \\ d_0 \end{bmatrix} = \begin{bmatrix} g_{2n-1} \\ \vdots \\ g_0 \end{bmatrix}$$

Pole assignment - general case

Proof (Sketch) Sylvester's theorem The polynomials

$$\begin{gathered} b_n s^n + b_{n-1} s^{n-1} + \ldots + b_1 s + b_0 \\ a_n s^n + a_{n-1} s^{n-1} + \ldots + a_1 s + a_0 \end{gathered}$$

are coprime if and only if the $(2n \times 2n)$ matrix

$$M = \begin{bmatrix} a_n & 0 & \ldots & 0 & b_n & 0 & \ldots & 0 \\ a_{n-1} & a_n & \ldots & 0 & b_{n-1} & b_n & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & \ldots & a_n & b_1 & b_2 & \ldots & 0 \\ a_0 & a_1 & \ldots & a_{n-1} & b_0 & b_1 & \ldots & b_n \\ 0 & a_0 & \ldots & a_{n-2} & 0 & b_0 & \ldots & b_{n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & a_0 & 0 & 0 & \ldots & b_0 \end{bmatrix}$$

is non-singular.

Pole assignment - general case

Proof (Sketch) Going back to the system of linear equations on Slide 28, we realize that the matrix on the left-hand side is the matrix M with $b_n = 0$. By Sylvester's theorem it is nonsingular if and only if the the polynomials $num_p(s), den_p(s)$ are coprime, which is true by assumption. Hence

$$\begin{bmatrix} c_{n-1} \\ \vdots \\ c_0 \\ d_{n-1} \\ \vdots \\ d_0 \end{bmatrix} = \begin{bmatrix} a_n & 0 & \dots & 0 & b_n = 0 & 0 & \dots & 0 \\ a_{n-1} & a_n & \dots & 0 & b_{n-1} & b_n = 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & \dots & a_n & b_1 & b_2 & \dots & 0 \\ a_0 & a_1 & \dots & a_{n-1} & b_0 & b_1 & \dots & b_n = 0 \\ 0 & a_0 & \dots & a_{n-2} & 0 & b_0 & \dots & b_{n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_0 & 0 & 0 & \dots & b_0 \end{bmatrix}^{-1} \begin{bmatrix} g_{2n-1} \\ \vdots \\ g_0 \end{bmatrix}$$

This ends the proof.

Pole assignment - general case

Example n=2

$$P(s) = \frac{b_1 s + b_0}{a_2 s^2 + a_1 s + a_0}, \quad C(s) = \frac{d_1 s + d_0}{c_1 s + c_0}$$

Then $num_p(s)num_c(s) + den_p(s)den_c(s) = p_{des}(s)$ writes as

$$(a_2 s^2 + a_1 s + a_0)(c_1 s + c_0) + (b_1 s + b_0)(d_1 s + d_0) = g_3 s^3 + g_2 s^2 + g_1 s + g_0$$
$$a_2 c_1 s^3 + (a_1 c_1 + a_2 c_0 + b_1 d_1) s^2 + (\ldots) s + (a_0 c_0 + b_0 d_0) = g_3 s^3 + g_2 s^2 + g_1 s + g_0$$

In matrix form

$$\underbrace{\begin{bmatrix} a_2 & 0 & 0 & 0 \\ a_1 & a_2 & b_1 & 0 \\ a_0 & a_1 & b_0 & b_1 \\ 0 & a_0 & 0 & b_0 \end{bmatrix}}_{\text{coefficient matrix } A} \underbrace{\begin{bmatrix} c_1 \\ c_0 \\ d_1 \\ d_0 \end{bmatrix}}_{\text{unknowns } x} = \underbrace{\begin{bmatrix} g_3 \\ g_2 \\ g_1 \\ g_0 \end{bmatrix}}_{\text{known terms } b}$$

If the polynomials $b_1 s + b_0$ and $a_2 s^2 + a_1 s + a_0$ have no common factors, then the matrix of coefficients A is nonsingular (i.e., $\det A \neq 0$) and the unknowns are obtained as $x = A^{-1} b$.

PID tuning

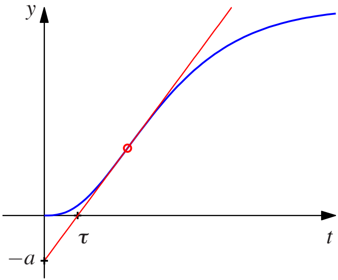
Ziegler-Nichols

Empirical rules to tune the parameters of a PI(D) controller

$$u = k_p e + k_i \int_0^t e(\tau)\, d\tau + k_d \frac{de}{dt} = k_p \left(e + \frac{1}{T_i} \int_0^t e(\tau)\, d\tau + T_d \frac{de}{dt} \right)$$

The rule implicitly assumes a step response as in the figure below from which a, τ are graphically determined (via the steepest tangent of the step response)

Step method



(a) Step response method

Type	k_p	T_i	T_d
P	1/a		
PI	0.9/a	3τ	
PID	1.2/a	2τ	0.5τ

PID tuning

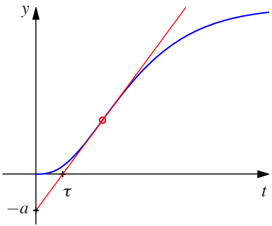
Ziegler-Nichols

Empirical rules to tune the parameters of a PI(D) controller

$$u = k_p e + k_i \int_0^t e(\tau)\,d\tau + k_d \frac{de}{dt} = k_p\left(e + \frac{1}{T_i}\int_0^t e(\tau)\,d\tau + T_d \frac{de}{dt}\right)$$

Improved step method – it assumes $P(s) = \dfrac{K}{1+sT}\mathrm{e}^{-\tau s}$. Then

$$Y(s) = \frac{K}{1+sT}\mathrm{e}^{-\tau s}\frac{1}{s} \Rightarrow y(t) = \begin{cases} 0 & 0 \le t < \tau \\ K(1(t-\tau) - \mathrm{e}^{-(t-\tau)/T}) & t \ge \tau \end{cases}$$



(a) Step response method

Parameters K, τ, T can obtained by fitting the approximate model to the measured step response: K is the steady state output value, τ is the delay and the steepest tangent ($= K/T$, because $\dot{y}(t) = \begin{cases} 0 & 0 \le t < \tau \\ \frac{K}{T}\mathrm{e}^{-(t-\tau)/T} & t \ge \tau \end{cases}$) gives

$$\frac{\Delta y}{\Delta x} = \frac{a}{\tau} \approx \frac{K}{T} \Leftrightarrow T \approx \frac{K\tau}{a}$$

PID tuning

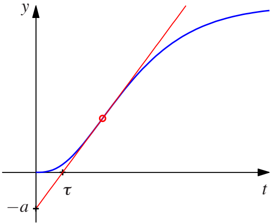
Ziegler-Nichols

Empirical rules to tune the parameters of a PI(D) controller

$$u = k_p e + k_i \int_0^t e(\tau)\, d\tau + k_d \frac{de}{dt} = k_p \left(e + \frac{1}{T_i} \int_0^t e(\tau)\, d\tau + T_d \frac{de}{dt} \right)$$

The parameters of $P(s)$ identified with this improved method are used to finely tune the PID with different parameters

Improved step method



(a) Step response method

[Textbook, Eq. (10.11) top]

Type	k_p	k_i
PI	$\frac{0.15\tau+0.35T}{K\tau}$	$\frac{0.46\tau+0.02T}{K\tau^2}$

PID tuning

Ziegler-Nichols

Empirical rules to tune the parameters of a PI(D) controller

$$u = k_p e + k_i \int_0^t e(\tau)\,d\tau + k_d \frac{de}{dt} = k_p\Big(e + \frac{1}{T_i}\int_0^t e(\tau)\,d\tau + T_d \frac{de}{dt}\Big)$$

Frequency response method

- P controller gain k_p in the closed-loop system $\frac{L(s)}{1+L(s)}$, $L(s) = k_p P(s)$, is increased until system starts oscillating for $k_p = k_c$
- Critical k_c and period T_c of oscillation are measured
- Oscillation is triggered by the occurrence of imaginary poles $s = \pm i\omega_c$, with $\omega_c = \frac{2\pi}{T_c}$, hence

$$1 + L(i\frac{2\pi}{T_c}) = 0$$

Type	k_p	T_i	T_d
P	$0.5k_c$		
PI	$0.4k_c$	$0.8T_c$	
PID	$0.6k_c$	$0.5T_c$	$0.125T_c$

PID tuning

Example [Textbook, Exercise 10.4]

$$P(s) = \frac{\mathrm{e}^{-s}}{s}$$

Determine the parameters of P, PI, PID controllers using ZN step and frequency response methods.

Step method Determine the step response of the system

$$Y(s) = P(s)\frac{1}{s} = \frac{\mathrm{e}^{-s}}{s^2}$$

Then

$$y(t) = \mathcal{L}^{-1}[\frac{\mathrm{e}^{-s}}{s^2}] = \begin{cases} 0 & 0 \le t < 1 \\ t-1 & t \ge 1 \end{cases}$$

Intercept $a = 1$ and delay $\tau = 1$.

PID tuning

Example (cont'd) Intercept $a = 1$ and delay $\tau = 1$.

$$u = k_p e + k_i \int_0^t e(\tau)\, d\tau + k_d \frac{de}{dt} = k_p \Big(e + \frac{1}{T_i} \int_0^t e(\tau)\, d\tau + T_d \frac{de}{dt} \Big)$$

Step method

Type	k_p	T_i	T_d
P	$\frac{1}{a} = 1$		
PI	$\frac{0.9}{a} = 0.9$	$3\tau = 3$	
PID	$\frac{1.2}{a} = 1.2$	$2\tau = 2$	$0.5\tau = 0.5$

PID tuning

$$P(s) = \frac{\mathrm{e}^{-s}}{s}$$

Frequency method Critical k_c and period T_c of oscillation are measured

$$1 + L(i\frac{2\pi}{T_c}) = 0$$

Set $\omega_c = \frac{2\pi}{T_c}$ and look for values of k_c, ω_c such that

$$1 + L(i\omega_c) = 0 \Leftrightarrow k_c \frac{\mathrm{e}^{-i\omega_c}}{i\omega_c} = -1$$

Note that

$$L(i\omega_c) = k_c \frac{\mathrm{e}^{-i\omega_c}}{i\omega_c} = k_c \frac{\cos\omega_c - i\sin\omega_c}{i\omega_c} = k_c \frac{\sin\omega_c + i\cos\omega_c}{-\omega_c}$$

PID tuning

Frequency method Critical k_c and period T_c of oscillation are measured

$$1 + L(i\frac{2\pi}{T_c}) = 0$$

Note that

$$L(i\omega_c) = k_c \frac{\mathrm{e}^{-i\omega_c}}{i\omega_c} = k_c \frac{\sin \omega_c + i \cos \omega_c}{-\omega_c}$$

Hence $L(i\omega_c) = -1$ if and only if

$$k_c \frac{\sin \omega_c}{-\omega_c} = -1, \quad k_c \frac{\cos \omega_c}{-\omega_c} = 0$$

from which

$$\omega_c = \frac{\pi}{2}, \quad k_c \frac{1}{-\frac{\pi}{2}} = -1$$

i.e.

$$\omega_c = \frac{\pi}{2}, \quad k_c = \frac{\pi}{2} \Leftrightarrow \omega_c = \frac{2\pi}{T_c} = \frac{\pi}{2}, \quad k_c = \frac{\pi}{2} \Leftrightarrow T_c = 4, \quad k_c = \frac{\pi}{2}$$

PID tuning

Frequency method

$$T_c = 4, \quad k_c = \frac{\pi}{2}$$

Type	k_p	T_i	T_d
P	$0.5k_c = 0.785$		
PI	$0.4k_c = 0.628$	$0.8T_c = 3.2$	
PID	$0.6k_c = 0.942$	$0.5T_c = 2$	$0.125T_c = 0.5$

Type	k_p	T_i	T_d
P	$\frac{1}{a} = 1$		
PI	$\frac{0.9}{a} = 0.9$	$3\tau = 3$	
PID	$\frac{1.2}{a} = 1.2$	$2\tau = 2$	$0.5\tau = 0.5$

Next lecture

- **Frequency domain design** (Chapter 11 of the textbook)