EE3CL4: Introduction to Linear Control Systems

Section 7: PID Control

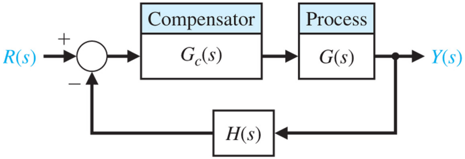
Tim Davidson

McMaster University

Winter 2019

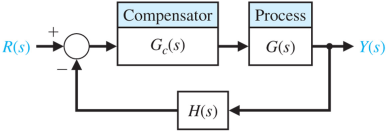
Outline

Cascade compensation



- Throughout this lecture we consider the case of $H(s) = 1$.
- We have looked at using
  - lead compensators to improve the transient performance of a closed loop
  - lag compensators to improve the steady state error responses without changing the closed loop transient response too much.
- What if we wanted to do both? What did we do?

Lead-lag compensation



- Apply lead design techniques to $G(s)$ to adjust the closed loop transient response
- Then apply lag design techniques to $G_{C,\text{lead}}(s)G(s)$ to improve steady state error response without changing the closed loop transient response too much

- Resulting compensator:

$$G_C(s) = G_{C,\text{lag}}(s)G_{C,\text{lead}}(s)$$
$$= \frac{K_{C,\text{lag}}K_{C,\text{lead}}(s+z_{\text{lag}})(s+z_{\text{lead}})}{(s+p_{\text{lag}})(s+p_{\text{lead}})}$$

- How can we gain insight into what the compensator is doing?

Lead-lag approximation

- $G_{C,\text{lag}}(s)G_{C,\text{lead}}(s) = \dfrac{K_{C,\text{lag}}K_{C,\text{lead}}(s+z_{\text{lag}})(s+z_{\text{lead}})}{(s+p_{\text{lag}})(s+p_{\text{lead}})}$
- Recall that
  - for frequencies between z_{lead} and p_{lead}, lead compensator acts like a differentiator
  - for frequencies between p_{lag} and z_{lag}, lag compensator acts like an integrator
- Rewrite:

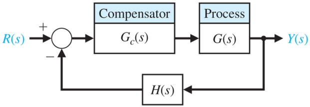
$$G_{C,\text{lag}}(s)G_{C,\text{lead}}(s) = \frac{\tilde{K}_{C,\text{ll}}(s+z_{\text{lag}})(s+z_{\text{lead}})}{(s+p_{\text{lag}})(1+s/p_{\text{lead}})}$$

- as p_{lead} gets big, and p_{lag} gets small this starts to look like

$$G_{C,\text{lag}}(s)G_{C,\text{lead}}(s) \approx \frac{\tilde{K}_{C,\text{ll}}(s+z_{\text{lag}})(s+z_{\text{lead}})}{s}$$

  for the values of s that are of greatest interest.
- Not physically realizable (more zeros than poles), but helpful approximation

Lead-lag to PID



$$G_{C,\mathrm{lag}}(s)G_{C,\mathrm{lead}}(s) \approx \frac{\tilde{K}_{C,\mathrm{ll}}(s+z_{\mathrm{lag}})(s+z_{\mathrm{lead}})}{s}$$

- Do a partial fraction on RHS and you get

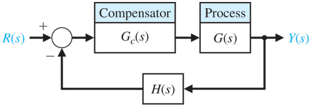
$$G_{C,\mathrm{lag}}(s)G_{C,\mathrm{lead}}(s) \approx K_P + \frac{K_I}{s} + K_D s$$

- With $H(s) = 1$, input to the compensator is $e(t) = r(t) - y(t)$
- Compensator output: $u(t) = \mathcal{L}^{-1}\{G_{C,\mathrm{lag}}(s)G_{C,\mathrm{lead}}(s)E(s)\}$

$$\Longrightarrow \qquad u(t) \approx K_P e(t) + K_I \int e(t)\,dt + K_D \frac{de(t)}{dt}$$

- That is, (approximately) PID control

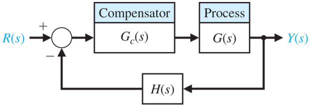
Variants of PID control



PID: $G_c(s) = K_P + K_I/s + K_D s$.

- With $K_D = 0$ we have a PI controller, $G_{PI}(s) = \hat{K}_P + \hat{K}_I/s$.
- With $K_I = 0$ we have a PD controller, $G_{PD}(s) = \bar{K}_P + \bar{K}_D s$.
- As implicit in our derivation, a PID controller can be realized as the cascade of a PI controller and a PD controller; i.e., $G_{PI}(s)G_{PD}(s)$ can be written as $K_P + K_I/s + K_D s$

PID control and root locus

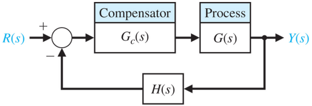


- Transfer function of idealized PID controller:

$$G_C(s) = K_P + K_I/s + K_D s = \frac{K_D(s+z_1)(s+z_2)}{s}$$

- That is, controller adds two zeros and a pole to the open loop transfer function
  - The pole is at the origin
  - The zeros can be arbitrary real numbers, or an arbitrary complex conjugate pair
- This provides considerable flexibility in re-shaping the root locus

PID Tuning



with $G_c(s) = K_P + K_I/s + K_D s$.

- How should we choose K_P, K_I and K_D?
- Can formulate as a optimization problem; e.g., *Find K_P, K_I and K_D that minimize the settling time, subject to*
  - *the damping ratio being greater than $\zeta_{\min}$,*
  - *the position and velocity error constants being greater than $K_{posn,\min}$ and $K_{v,\min}$,*
  - *the error constant for a step disturbance being greater than $K_{dist,posn,\min}$,*
  - *and the loop being stable*
- Typically difficult to find the optimal solution
- Many ad-hoc techniques that usually find “good” solutions have been proposed.

Zeigler–Nichols Tuning

- Two well established methods for finding a "good" solution in some common scenarios
- Often useful in practice because they can be applied to cases in which the model has to be measured (no analytic transfer function)
- We will look at the "ultimate gain" method
- This is based on the step response of the system
- However, the method is only suitable for a certain class of systems and a certain class of design goals
  - You need to make sure that the system you wish to control falls into an appropriate class.
  - You also need to ensure that the ZN tuning goals match your design goals. The ZN tuning scheme gives considerable weight to the response to disturbances

"Ultimate Gain" Zeigler–Nichols Tuning

1. Set K_I and K_D to zero.
2. Increase K_P until the system is marginally stable (Poles on the $j\omega$-axis)
3. The value of this gain is the "ultimate gain", K_U
4. The period of the sustained oscillations is called the "ultimate period", T_U (or P_U).
   (The position of the poles on the $j\omega$-axis is $2\pi/T_U$)
5. The gains are then chosen using the following table

"Ultimate Gain" Zeigler–Nichols Tuning

Table 7.7 Ziegler-Nichols PID Tuning Using Ultimate Gain, K_U, and Oscillation Period, P_U

Ziegler-Nichols PID Controller Gain Tuning Using Closed-loop Concepts

Controller Type	K_P	K_I	K_D
Proportional (P) $G_c(s) = K_P$	$0.5K_U$	–	–
Proportional-plus-integral (PI) $G_c(s) = K_P + \frac{K_I}{s}$	$0.45K_U$	$\frac{0.54K_U}{T_U}$	–
Proportional-plus-integral-plus-derivative (PID) $G_c(s) = K_P + \frac{K_I}{s} + K_D s$	$0.6K_U$	$\frac{1.2K_U}{T_U}$	$\frac{0.6K_U T_U}{8}$

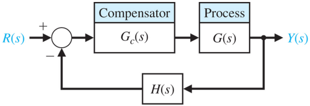
Manual refinement

- One way in which the design can be improved, is searching for “nearby” gains that improve the performance
- The following table provides guidelines for that local search. These are appropriate for a broad class of systems

Table 7.6 Effect of Increasing the PID Gains K_p, K_D, and K_I on the Step Response

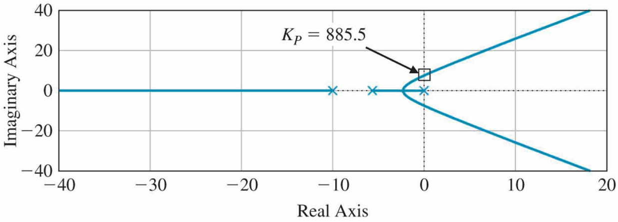
PID Gain	Percent Overshoot	Settling Time	Steady-State Error
Increasing K_P	Increases	Minimal impact	Decreases
Increasing K_I	Increases	Increases	Zero steady-state error
Increasing K_D	Decreases	Decreases	No impact

Example



$G(s) = \frac{1}{s(s+b)(s+2\zeta\omega_n)}$, with $b = 10$, $\zeta = 1/\sqrt{2}$ and $\omega_n = 4$.

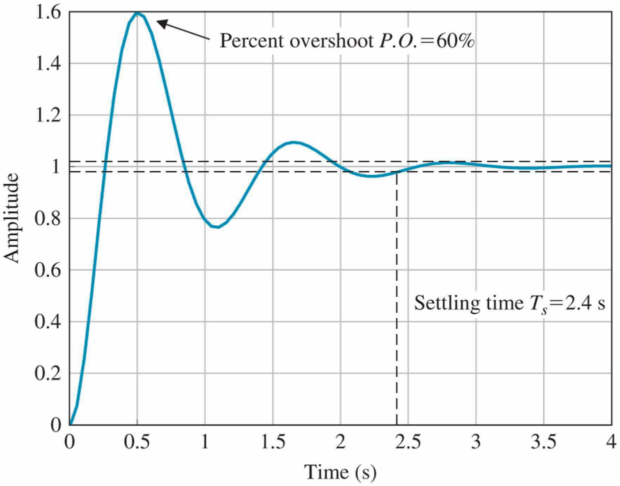
- Step 2: Plot root locus of $G(s)$ to find K_U and T_U



- Step 3: $K_U = 885.5$,
- Step 4: marginally stable poles: $\pm j7.5$; $\Rightarrow$ $T_U = 0.83$s
- Step 5: $K_P = 521.3$, $K_I = 1280.2$, $K_D = 55.1$

Example

Step response of ZN tuned closed loop,
$K_P = 521.3$, $K_I = 1280.2$, $K_D = 55.1$



Example

Step response with manually modified gains,
$K_P = 370$, $K_I = 100$, $K_D = 60$