Studio 11: Root Locus Compensator Design using PD, Lead, PID, and Lead-Lag Compensators

In this studio, first we cover design of Proportional Derivative (PD) compensators and Lead compensators that can be used to improve the transient response of a system. Then we cover PID and Lead-Lag compensators that are used to improve both the transient response and steady-state error.

1. Compensator to Improve Transient Response

1.1 Proportional-Derivative (PD) Compensators

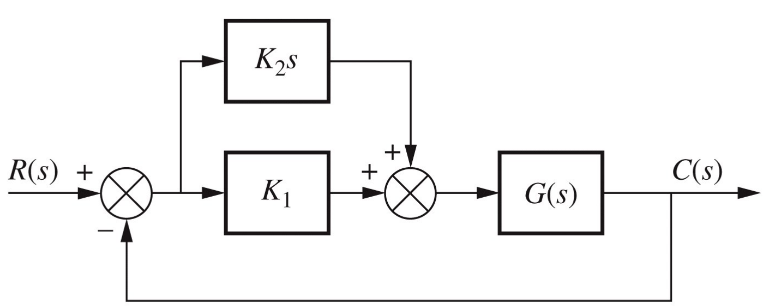
A PD compensator improves the transient response via re-shaping the original system's root locus. An additional zero is placed at a location so that the root locus angle condition can be satisfied at the target closed-loop poles. The additional zero provides a degree of freedom to adjust the root locus angle condition.

A PD compensator has two components, a proportional component and a derivative component and takes the form

$$G_c(s) = K_P + K_D s = K_D\left(s + \frac{K_P}{K_D}\right) = K_P(s + z_C),$$

where $z_C = \frac{K_P}{K_D}$ is the compensator zero.

A block diagram of a unity feedback system with a PD compensator is shown below.



A closed-loop control system with a PD compensator. Here $K_1 = K_P$ and $K_2 = K_D$.

1.2 Lead Compensators

A lead compensator improves the transient response via re-shaping the original system's root locus. An additional zero/pole pair is placed at a location so that the root locus angle condition can be satisfied at the target closed-loop poles. Since there are two degrees of freedom, i.e., the compensator zero and pole locations that are chosen to satisfy the root locus angle condition, the resulting lead compensator is not unique (see Figure 1 for a demonstration).

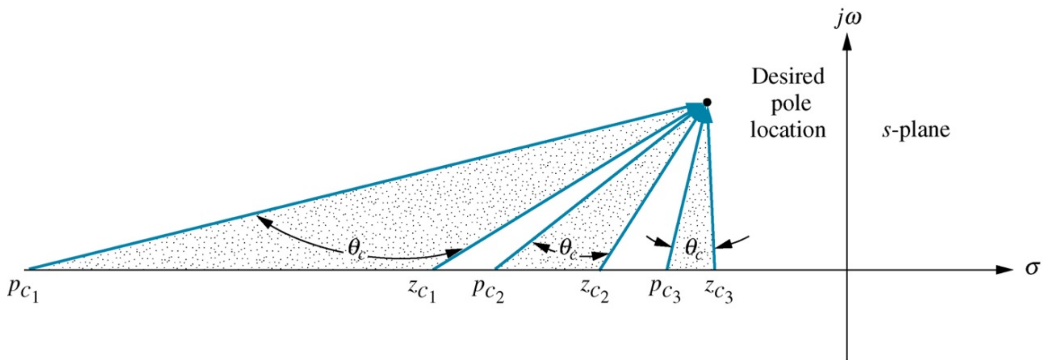


Figure 1 Multiple lead compensators exist that meet the root locus angle condition (i.e., lead compensators are not unique.)

Lead Compensator Transfer Function

A lead compensator takes the form

$$G_C(s) = K\frac{s + z_C}{s + p_C}, \quad |p_C| > |z_C|$$

Lead Compensator Design Approach

To design a lead compensator, follow these steps:

- Determine improvement needed and location of desired closed-loop (CL) system poles
- Calculate angle contribution of the uncompensated system at desired CL poles
- Determine needed compensator angle contribution using root locus angle condition

For the example shown in **Figure 2**, applying the angle criterion at the desired pole location yields:

$$\theta c = \theta 2 - \theta 1 = 180^{o} - (\theta 5 - \theta 3 - \theta 4)$$

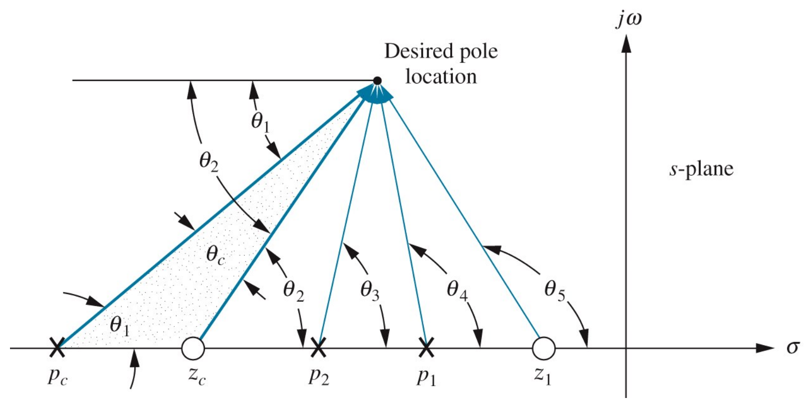
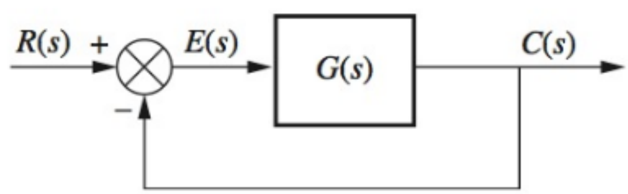


Figure 2 The lead compensator contribution angle is chosen to place the desired pole location on the compensated root locus.

Example

Consider the unity-feedback system shown below with open-loop transfer function

$$G(s) = \frac{K}{s(s+1)(s^2+10s+26)}$$



Unity feedback system for Example 2.

Perform the following:

(i) Find the settling time for the system if it is operating with 15% OS.

(ii) Develop a cascade controller so that the settling time is less than 7s (use both hand calculations and Matlab). Specifically, find the zero of the compensator and the gain K, assuming that the pole of the compensator is located at s = -15.

(iii) Plot the step response of the compensated system and determine the transient response characteristics.

2. Improving both Steady-State Error and Transient Response: PID and Lead-Lag Compensators

PID compensators combine PD and PI compensators to improve the transient response and the steady-state error, respectively. Similarly, lead-lag compensators combine lead and lag compensators to improve the transient response and the steady-state error, respectively. One approach to design a PID/lead-lag compensator is as follows:

1. Evaluate the uncompensated system's performance.

2. Design a PD (or a lead) compensator to improve transient performance (RL re-shaped) so that the design specs are met.

3. Design a PI (or a lag) compensator to improve steady-state performance (RL preserved).

4. Combine the PD and PI (or the lead and lag) compensators in cascade to produce the overall PID (or lead-lag) compensator.

Exercise

Each student shall complete the exercises below and get their work checked off by the studio instructor.

For remote students and students who do not finish within studio session:

Compile your answers and outputs of Exercise 1 along with the Matlab code you used to answer the questions and screenshots of Matlab outputs in a word file and name it LastNameFirstNameStudio11.docx. Upload your file to ELMS under the Studio11 link.

Exercise 1 Consider the unity feedback system with

$$G(s) = \frac{K}{(s+4)(s+6)(s+10)}$$

(a) Use Matlab along with hand calculations to design a PID compensator so that the closed-loop system will have $PO < 25\%$ and $T_s < 2\ sec$ for a step input and zero steady-state error for step and ramp inputs. (Hint: add a pure integrator first to increase the system type by 1 before designing your PID compensator, otherwise a PID compensator alone can't achieve zero SS error for ramp inputs)

(b) Plot the root locus of the compensated system in (a) and the step response and show that the specifications are met.