

- c. Evaluate the steady-state error for a unit ramp input to your compensated system.
- d. Evaluate how much improvement in steady-state error was realized.

ANSWERS:

a. $e_{\text{ramp}}(\infty) = 0.1527$

b. $G_{\text{lag}}(s) = \frac{s + 0.2}{s + 0.01}$

c. $e_{\text{ramp}}(\infty) = 0.0078$

d. 19.58 times improvement

The complete solution is at www.wiley.com/college/nise.

9.3 Improving Transient Response via Cascade Compensation

Since we have solved the problem of improving the steady-state error without affecting the transient response, let us now improve the transient response itself. In this section, we discuss two ways to improve the transient response of a feedback control system by using cascade compensation. Typically, the objective is to design a response that has a desirable percent overshoot and a shorter settling time than the uncompensated system.

The first technique we will discuss is *ideal derivative compensation*. With ideal derivative compensation, a pure differentiator is added to the forward path of the feedback control system. We will see that the result of adding differentiation is the addition of a zero to the forward-path transfer function. This type of compensation requires an active network for its realization. Further, differentiation is a noisy process; although the level of the noise is low, the frequency of the noise is high compared to the signal. Thus, differentiating high-frequency noise yields a large, unwanted signal.

The second technique does not use pure differentiation. Instead, it approximates differentiation with a passive network by adding a zero and a more distant pole to the forward-path transfer function. The zero approximates pure differentiation as described previously.

As with compensation to improve steady-state error, we introduce names associated with the implementation of the compensators. We call an ideal derivative compensator a *proportional-plus-derivative (PD) controller*, since the implementation, as we will see, consists of feeding the error (proportional) plus the derivative of the error forward to the plant. The second technique uses a passive network called a *lead compensator*. As with the lag compensator, the name comes from its frequency response, which is discussed in Chapter 11. Thus, we use the name *PD controller* interchangeably with *ideal derivative compensator*, and we use the name *lead compensator* when the cascade compensator does not employ pure differentiation.

Ideal Derivative Compensation (PD)

The transient response of a system can be selected by choosing an appropriate closed-loop pole location on the s -plane. If this point is on the root locus, then a simple gain adjustment is all that is required in order to meet the transient response specification. If the closed-loop pole location is not on the root locus, then the root locus must be reshaped so that the compensated (new) root locus goes through the selected closed-loop pole location. In order to accomplish the latter task, poles and zeros can be added in the forward path to produce a new open-loop function whose root locus goes through the design point on the s -plane. One way to speed up the original system that generally works is to add a single zero to the forward path.

This zero can be represented by a compensator whose transfer function is

$$G_c(s) = s + z_c \quad (9.12)$$

This function, the sum of a differentiator and a pure gain, is called an *ideal derivative*, or *PD controller*. Judicious choice of the position of the compensator zero can quicken the response over the uncompensated system. In summary, transient responses unattainable by a simple gain adjustment can be obtained by augmenting the system's poles and zeros with an ideal derivative compensator.

We now show that ideal derivative compensation speeds up the response of a system. Several simple examples are shown in Figure 9.15, where the uncompensated system of Figure 9.15(a), operating with a damping ratio of 0.4, becomes a compensated system by the addition of a compensating zero at -2 , -3 , and -4 in Figures 9.15(b), (c), and (d), respectively. In each design, the zero is moved to a different position, and the root locus is shown. For each compensated case, the dominant, second-order poles are farther out along the 0.4 damping ratio line than the uncompensated system.

Each of the compensated cases has dominant poles with the same damping ratio as the uncompensated case. Thus, we predict that the percent overshoot will be the same for each case.

Also, the compensated, dominant, closed-loop poles have more negative real parts than the uncompensated, dominant, closed-loop poles. Hence, we predict that the settling times for the compensated cases will be shorter than for the

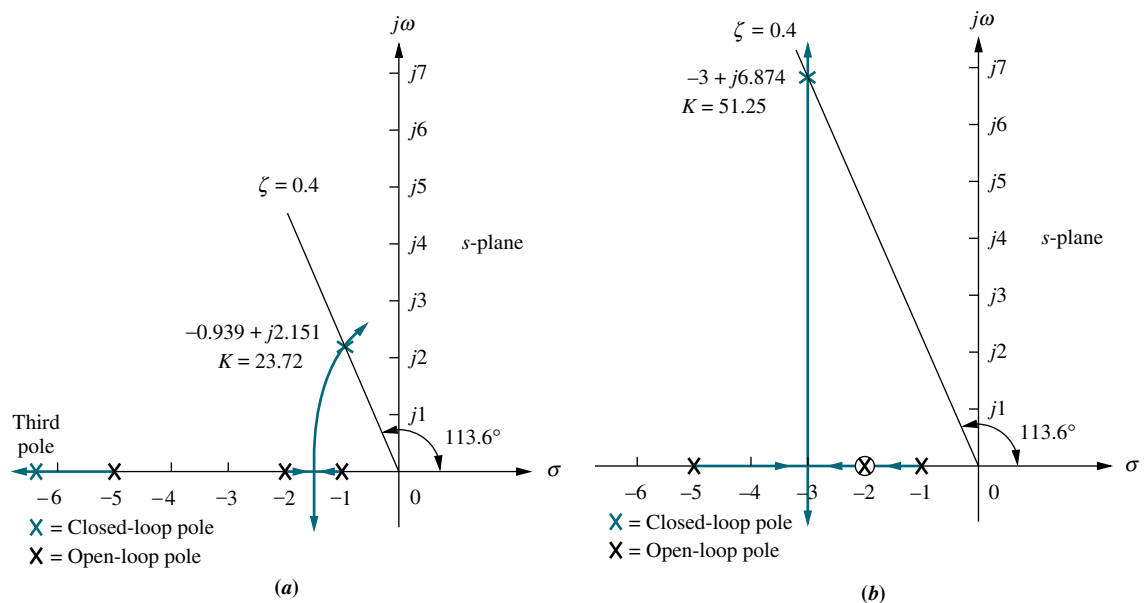


FIGURE 9.15 Using ideal derivative compensation: **a.** uncompensated; **b.** compensator zero at -2 ; (figure continues)

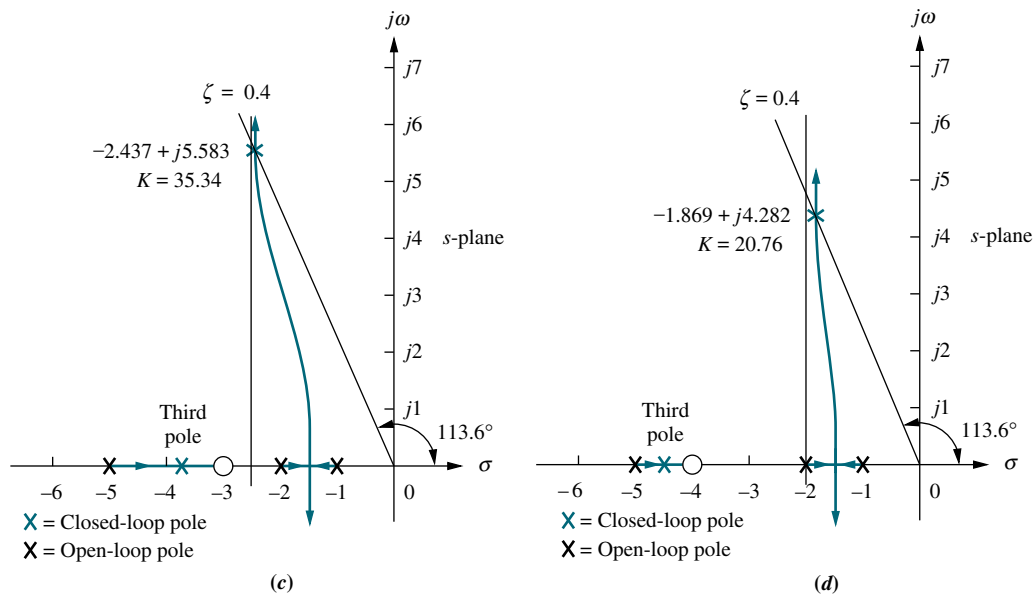


FIGURE 9.15 (Continued) **c.** compensator zero at -3 ; **d.** compensator zero at -4 .

uncompensated case. The compensated, dominant, closed-loop poles with the more negative real parts will have the shorter settling times. The system in Figure 9.15(b) will have the shortest settling time.

All of the compensated systems will have smaller peak times than the uncompensated system, since the imaginary parts of the compensated systems are larger. The system of Figure 9.15(b) will have the smallest peak time.

Also notice that as the zero is placed farther from the dominant poles, the closed-loop, compensated dominant poles move closer to the origin and to the uncompensated, dominant closed-loop poles. Table 9.2 summarizes the

TABLE 9.2 Predicted characteristics for the systems of Figure 9.15

	Uncompensated	Compensation b	Compensation c	Compensation d
Plant and compensator	$\frac{K}{(s+1)(s+2)(s+5)}$	$\frac{K(s+2)}{(s+1)(s+2)(s+5)}$	$\frac{K(s+3)}{(s+1)(s+2)(s+5)}$	$\frac{K(s+4)}{(s+1)(s+2)(s+5)}$
Dom, poles	$-0.939 \pm j2.151$	$-3 \pm j6.874$	$-2.437 \pm j5.583$	$-1.869 \pm j4.282$
K	23.72	51.25	35.34	20.76
ζ	0.4	0.4	0.4	0.4
ω_n	2.347	7.5	6.091	4.673
%OS	25.38	25.38	25.38	25.38
T_s	4.26	1.33	1.64	2.14
T_p	1.46	0.46	0.56	0.733
K_p	2.372	10.25	10.6	8.304
$e(\infty)$	0.297	0.089	0.086	0.107
Third pole	-6.123	None	-3.127	-4.262
Zero	None	None	-3	-4
Comments	Second-order approx. OK	Pure second-order	Second-order approx. OK	Second-order approx. OK

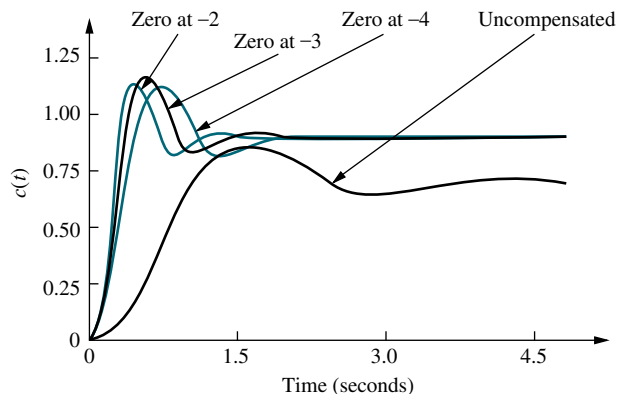


FIGURE 9.16 Uncompensated system and ideal derivative compensation solutions from Table 9.2

results obtained from the root locus of each of the design cases shown in Figure 9.15.

In summary, although compensation methods *c* and *d* yield slower responses than method *b*, the addition of ideal derivative compensation shortened the response time in each case while keeping the percent overshoot the same. This change can best be seen in the settling time and peak time, where there is at least a doubling of speed across all of the cases of compensation. An added benefit is the improvement in the steady-state error, even though lag compensation was not used. Here the steady-state error of the compensated system is at least one-third that of the uncompensated system, as seen by $e(\infty)$ and K_p . All systems in Table 9.2 are Type 0, and some steady-state error is expected. The reader must not assume that, in general, improvement in transient response always yields an improvement in steady-state error.

The time response of each case in Table 9.2 is shown in Figure 9.16. We see that the compensated responses are faster and exhibit less error than the uncompensated response.

Now that we have seen what ideal derivative compensation can do, we are ready to design our own ideal derivative compensator to meet a transient response specification. Basically, we will evaluate the sum of angles from the open-loop poles and zeros to a design point that is the closed-loop pole that yields the desired transient response. The difference between 180° and the calculated angle must be the angular contribution of the compensator zero. Trigonometry is then used to locate the position of the zero to yield the required difference in angle.

Example 9.3

Ideal Derivative Compensator Design

PROBLEM: Given the system of Figure 9.17, design an ideal derivative compensator to yield a 16% overshoot, with a threefold reduction in settling time.

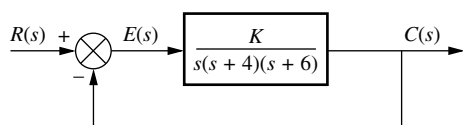


FIGURE 9.17 Feedback control system for Example 9.3

SOLUTION: Let us first evaluate the performance of the uncompensated system operating with 16% overshoot. The root locus for the uncompensated system is shown in Figure 9.18. Since 16% overshoot is equivalent to $\zeta = 0.504$, we search along that damping ratio line for an odd multiple of 180° and find that the dominant, second-order pair of poles is at $-1.205 \pm j2.064$. Thus, the settling

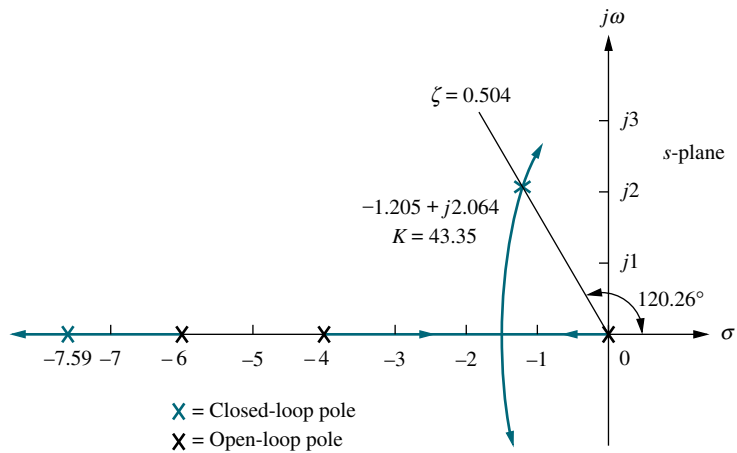


FIGURE 9.18 Root locus for uncompensated system shown in Figure 9.17

time of the uncompensated system is

$$T_s = \frac{4}{\zeta\omega_n} = \frac{4}{1.205} = 3.320 \tag{9.13}$$

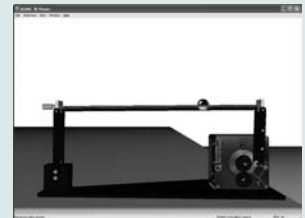
Since our evaluation of percent overshoot and settling time is based upon a second-order approximation, we must check the assumption by finding the third pole and justifying the second-order approximation. Searching beyond -6 on the real axis for a gain equal to the gain of the dominant, second-order pair, 43.35, we find a third pole at -7.59 , which is over six times as far from the $j\omega$ -axis as the dominant, second-order pair. We conclude that our approximation is valid. The transient and steady-state error characteristics of the uncompensated system are summarized in Table 9.3.

TABLE 9.3 Uncompensated and compensated system characteristic of Example 9.3

	Uncompensated	Simulation	Compensated	Simulation
Plant and compensator	$\frac{K}{s(s+4)(s+6)}$		$\frac{K(s+3.006)}{s(s+4)(s+6)}$	
Dominant poles	$-1.205 \pm j2.064$		$-3.613 \pm j6.193$	
K	43.35		47.45	
ζ	0.504		0.504	
ω_n	2.39		7.17	
%OS	16	14.8	16	11.8
T_s	3.320	3.6	1.107	1.2
T_p	1.522	1.7	0.507	0.5
K_v	1.806		5.94	
$e(\infty)$	0.554		0.168	
Third pole	-7.591		-2.775	
Zero	None		-3.006	
Comments	Second-order approx. OK		Pole-zero not canceling	

Virtual Experiment 9.1 PD Controller Design

Put theory into practice and use root-locus to design a PD controller for the Quanser Ball and Beam using LabVIEW. The Ball and Beam is an unstable system, similar to exothermic chemical processes that have to be stabilized to avoid overheating.



Virtual experiments are found on WileyPLUS.

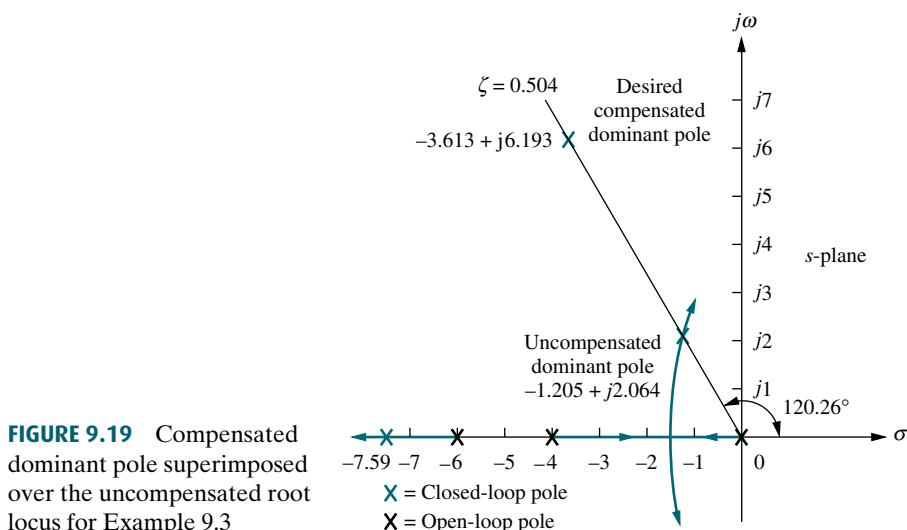


FIGURE 9.19 Compensated dominant pole superimposed over the uncompensated root locus for Example 9.3

Now we proceed to compensate the system. First we find the location of the compensated system's dominant poles. In order to have a threefold reduction in the settling time, the compensated system's settling time will be one-third of Eq. (9.13). The new settling time will be 1.107. Therefore, the real part of the compensated system's dominant, second-order pole is

$$\sigma = \frac{4}{T_s} = \frac{4}{1.107} = 3.613 \quad (9.14)$$

Figure 9.19 shows the designed dominant, second-order pole, with a real part equal to -3.613 and an imaginary part of

$$\omega_d = 3.613 \tan(180^\circ - 120.26^\circ) = 6.193 \quad (9.15)$$

Next we design the location of the compensator zero. Input the uncompensated system's poles and zeros in the root locus program as well as the design point $-3.613 \pm j6.193$ as a test point. The result is the sum of the angles to the design point of all the poles and zeros of the compensated system except for those of the compensator zero itself. The difference between the result obtained and 180° is the angular contribution required of the compensator zero. Using the open-loop poles shown in Figure 9.19 and the test point, $-3.613 + j6.193$, which is the desired dominant second-order pole, we obtain the sum of the angles as -275.6° . Hence, the angular contribution required from the compensator zero for the test point to be on the root locus is $+275.6^\circ - 180^\circ = 95.6^\circ$. The geometry is shown in Figure 9.20, where we now must solve for $-\sigma$, the location of the compensator zero.

From the figure,

$$\frac{6.193}{3.613 - \sigma} = \tan(180^\circ - 95.6^\circ) \quad (9.16)$$

Thus, $\sigma = 3.006$. The complete root locus for the compensated system is shown in Figure 9.21.

Table 9.3 summarizes the results for both the uncompensated system and the compensated system. For the uncompensated system, the estimate of the transient

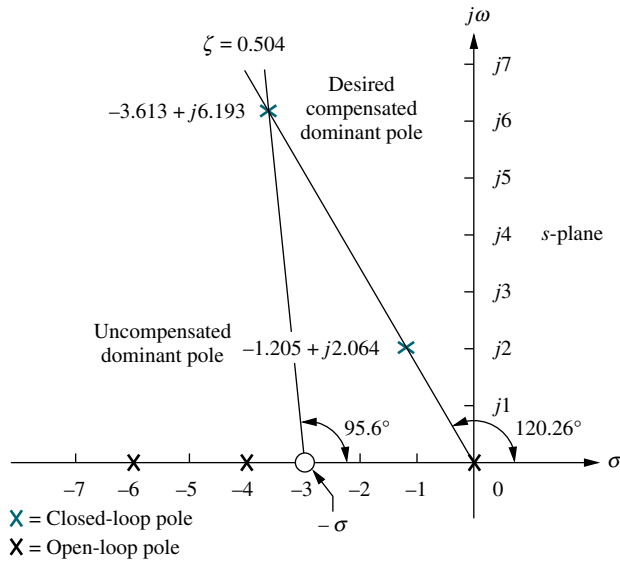


FIGURE 9.20 Evaluating the location of the compensating zero for Example 9.3

response is accurate since the third pole is at least five times the real part of the dominant, second-order pair. The second-order approximation for the compensated system, however, may be invalid because there is no approximate closed-loop third-pole and zero cancellation between the closed-loop pole at -2.775 and the closed-loop zero at -3.006 . A simulation or a partial-fraction expansion of the closed-loop response to compare the residue of the pole at -2.775 to the residues of the dominant poles at $-3.613 \pm j6.193$ is required. The results of a simulation are shown in the table's second column for the uncompensated system and the fourth column for the compensated system. The simulation results can be obtained using MATLAB (discussed at the end of this example) or a program

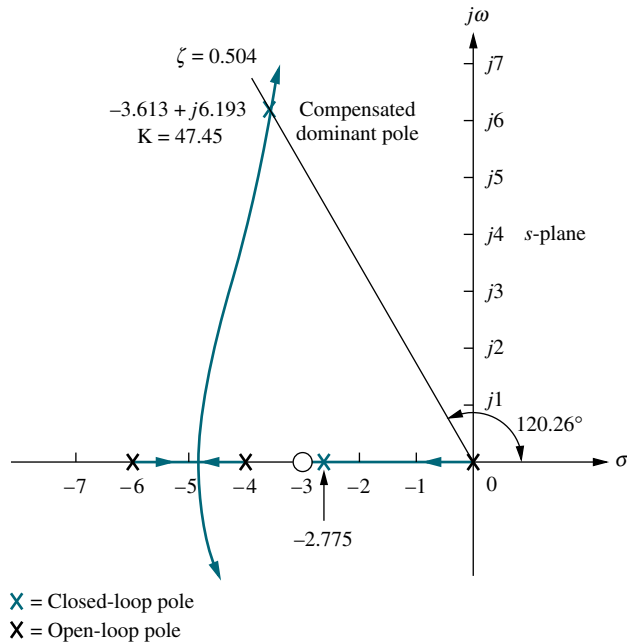


FIGURE 9.21 Root locus for the compensated system of Example 9.3

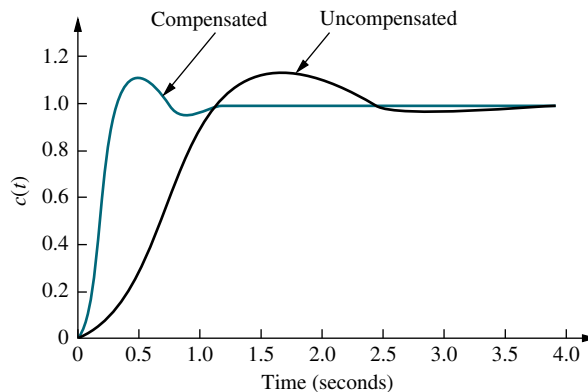


FIGURE 9.22 Uncompensated and compensated system step responses of Example 9.3

like the state-space step-response program described in Appendix H.1 at www.wiley.com/college/nise. The percent overshoot differs by 3% between the uncompensated and compensated systems, while there is approximately a threefold improvement in speed as evaluated from the settling time.

The final results are displayed in Figure 9.22, which compares the uncompensated system and the faster compensated system.

MATLAB
ML

Students who are using MATLAB should now run `ch9p1` in Appendix B. MATLAB will be used to design a PD controller. You will input the desired percent overshoot from the keyboard. MATLAB will plot the root locus of the uncompensated system and the percent overshoot line. You will interactively select the gain, after which MATLAB will display the performance characteristics of the uncompensated system and plot its step response. Using these characteristics, you will input the desired settling time. MATLAB will design the PD controller, enumerate its performance characteristics, and plot a step response. This exercise solves Example 9.3 using MATLAB.

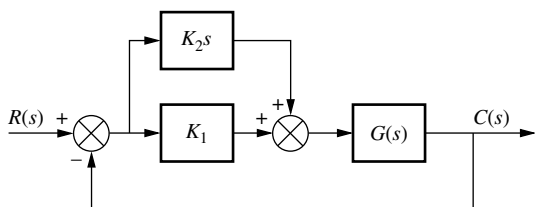


FIGURE 9.23 PD controller

Once we decide on the location of the compensating zero, how do we implement the ideal derivative, or PD controller? The ideal integral compensator that improved steady-state error was implemented with a proportional-plus-integral (PI) controller. The ideal derivative compensator used to improve the transient response is implemented with a proportional-plus-derivative (PD) controller. For example, in Figure 9.23 the transfer function of the controller is

$$G_c(s) = K_2s + K_1 = K_2 \left(s + \frac{K_1}{K_2} \right) \quad (9.17)$$

Hence, K_1/K_2 is chosen to equal the negative of the compensator zero, and K_2 is chosen to contribute to the required loop-gain value. Later in the chapter, we will study circuits that can be used to approximate differentiation and produce gain.

While the ideal derivative compensator can improve the transient response of the system, it has two drawbacks. First, it requires an active circuit to perform the

differentiation. Second, as previously mentioned, differentiation is a noisy process: The level of the noise is low, but the frequency of the noise is high compared to the signal. Differentiation of high frequencies can lead to large unwanted signals or saturation of amplifiers and other components. The lead compensator is a passive network used to overcome the disadvantages of ideal differentiation and still retain the ability to improve the transient response.

Lead Compensation

Just as the active ideal integral compensator can be approximated with a passive lag network, an active ideal derivative compensator can be approximated with a passive lead compensator. When passive networks are used, a single zero cannot be produced; rather, a compensator zero and a pole result. However, if the pole is farther from the imaginary axis than the zero, the angular contribution of the compensator is still positive and thus approximates an equivalent single zero. In other words, the angular contribution of the compensator pole subtracts from the angular contribution of the zero but does not preclude the use of the compensator to improve transient response, since the net angular contribution is positive, just as for a single PD controller zero.

The advantages of a passive lead network over an active PD controller are that (1) no additional power supplies are required and (2) noise due to differentiation is reduced. The disadvantage is that the additional pole does not reduce the number of branches of the root locus that cross the imaginary axis into the right-half-plane, while the addition of the single zero of the PD controller tends to reduce the number of branches of the root locus that cross into the right half-plane.

Let us first look at the concept behind lead compensation. If we select a desired dominant, second-order pole on the s -plane, the sum of the angles from the uncompensated system's poles and zeros to the design point can be found. The difference between 180° and the sum of the angles must be the angular contribution required of the compensator.

For example, looking at Figure 9.24, we see that

$$\theta_2 - \theta_1 - \theta_3 - \theta_4 + \theta_5 = (2k + 1)180^\circ \quad (9.18)$$

where $(\theta_2 - \theta_1) = \theta_c$ is the angular contribution of the lead compensator. From Figure 9.24 we see that θ_c is the angle of a ray extending from the design point and intersecting the real axis at the pole value and zero value of the compensator. Now visualize this ray rotating about the desired closed-loop pole location and

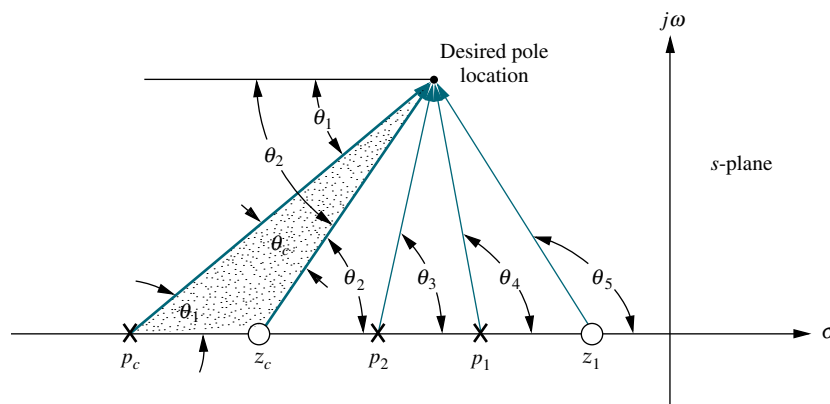


FIGURE 9.24 Geometry of lead compensation

TryIt 9.2

Use MATLAB, the Control System Toolbox, and the following steps to use SISOTOOL to perform the design of Example 9.3.

1. Type **SISOTOOL** in the MATLAB **Command Window**.
2. Select **Import** in the File menu of the **SISO Design for SISO Design Task Window**.
3. In the **Data** field for G, type `zpk([], [0, -4, -6], 1)` and hit **ENTER** on the keyboard. Click **OK**.
4. On the **Edit** menu choose **SISO Tool Preferences . . .** and select **Zero/pole/gain:** under the **Options** tab. Click **OK**.
5. Right-click on the root locus white space and choose **Design Requirements/New . . .**
6. Choose **Percent overshoot** and type in 16. Click **OK**.
7. Right-click on the root locus white space and choose **Design Requirements/New . . .**
8. Choose **Settling time** and click **OK**.
9. Drag the settling time vertical line to the intersection of the root locus and 16% overshoot radial line.
10. Read the settling time at the bottom of the window.
11. Drag the settling time vertical line to a settling time that is 1/3 of the value found in Step 9.
12. Click on a red zero icon in the menu bar. Place the zero on the root locus real axis by clicking again on the real axis.
13. Left-click on the real-axis zero and drag it along the real axis until the root locus intersects the settling time and percent overshoot lines.
14. Drag a red square along the root locus until it is at the intersection of the root locus, settling time line, and the percent overshoot line.
15. Click the **Compensator Editor** tab of the **Control and Estimation Tools Manager** window to see the resulting compensator, including the gain.

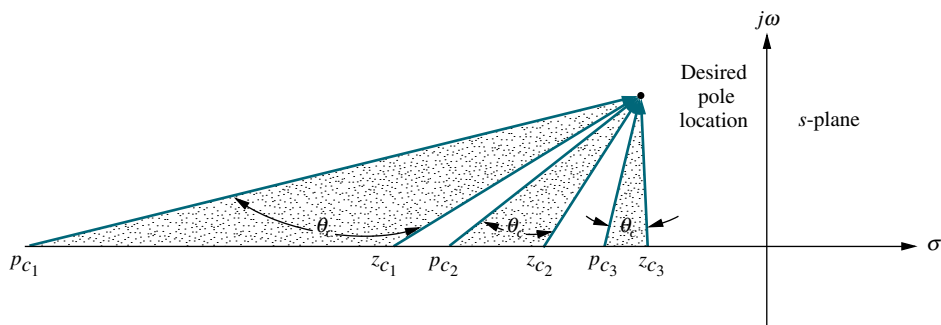


FIGURE 9.25 Three of the infinite possible lead compensator solutions

intersecting the real axis at the compensator pole and zero, as illustrated in Figure 9.25. We realize that an infinite number of lead compensators could be used to meet the transient response requirement.

How do the possible lead compensators differ? The differences are in the values of static error constants, the gain required to reach the design point on the compensated root locus, the difficulty in justifying a second-order approximation when the design is complete, and the ensuing transient response.

For design, we arbitrarily select either a lead compensator pole or zero and find the angular contribution at the design point of this pole or zero along with the system's open-loop poles and zeros. The difference between this angle and 180° is the required contribution of the remaining compensator pole or zero. Let us look at an example.

Example 9.4

Lead Compensator Design

PROBLEM: Design three lead compensators for the system of Figure 9.17 that will reduce the settling time by a factor of 2 while maintaining 30% overshoot. Compare the system characteristics between the three designs.

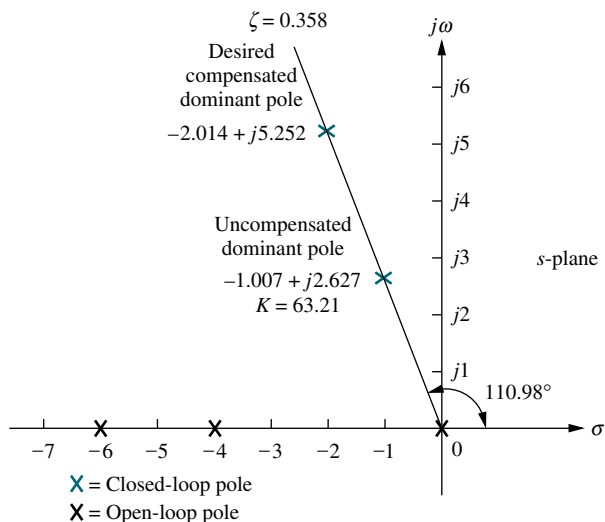


FIGURE 9.26 Lead compensator design, showing evaluation of uncompensated and compensated dominant poles for Example 9.4

SOLUTION: First determine the characteristics of the uncompensated system operating at 30% overshoot to see what the uncompensated settling time is. Since 30% overshoot is equivalent to a damping ratio of 0.358, we search along the $\zeta = 0.358$ line for the uncompensated dominant poles on the root locus, as shown in Figure 9.26. From the pole's real part, we calculate the uncompensated settling time as $T_s = 4/1.007 = 3.972$ seconds. The remaining characteristics of the uncompensated system are summarized in Table 9.4.

Next we find the design point. A twofold reduction in settling time yields $T_s = 3.972/2 = 1.986$ seconds, from which the real part of the desired pole location is $-\zeta\omega_n = -4/T_s = -2.014$. The imaginary part is $\omega_d = -2.014 \tan(110.98^\circ) = 5.252$.

We continue by designing the lead compensator. Arbitrarily assume a compensator zero at -5 on the real axis as a possible solution. Using the root locus program, sum the angles from both this zero and the

TABLE 9.4 Comparison of lead compensation designs for Example 9.4

	Uncompensated	Compensation a	Compensation b	Compensation c
Plant and compensator	$\frac{K}{s(s+4)(s+6)}$	$\frac{K(s+5)}{s(s+4)(s+6)(s+42.96)}$	$\frac{K(s+4)}{s(s+4)(s+6)(s+20.09)}$	$\frac{K(s+2)}{s(s+4)(s+6)(s+8.971)}$
Dominant poles	$-1.007 \pm j2.627$	$-2.014 \pm j5.252$	$-2.014 \pm j5.252$	$-2.014 \pm j5.252$
K	63.21	1423	698.1	345.6
ζ	0.358	0.358	0.358	0.358
ω_n	2.813	5.625	5.625	5.625
%OS*	30 (28)	30 (30.7)	30 (28.2)	30 (14.5)
T_s^*	3.972 (4)	1.986 (2)	1.986 (2)	1.986 (1.7)
T_p^*	1.196 (1.3)	0.598 (0.6)	0.598 (0.6)	0.598 (0.7)
K_v	2.634	6.9	5.791	3.21
$e(\infty)$	0.380	0.145	0.173	0.312
Other poles	-7.986	-43.8, -5.134	-22.06	-13.3, -1.642
Zero	None	-5	None	-2
Comments	Second-order approx. OK	Second-order approx. OK	Second-order approx. OK	No pole-zero cancellation

*Simulation results are shown in parentheses.

uncompensated system's poles and zeros, using the design point as a test point. The resulting angle is -172.69° . The difference between this angle and 180° is the angular contribution required from the compensator pole in order to place the design point on the root locus. Hence, an angular contribution of -7.31° is required from the compensator pole.

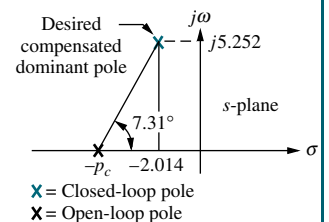
The geometry shown in Figure 9.27 is used to calculate the location of the compensator pole. From the figure,

$$\frac{5.252}{p_c - 2.014} = \tan 7.31^\circ \tag{9.19}$$

from which the compensator pole is found to be

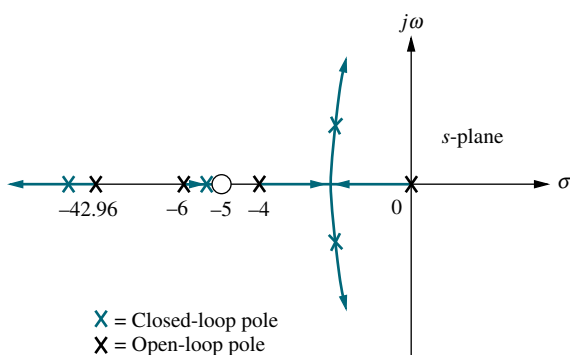
$$p_c = 42.96 \tag{9.20}$$

The compensated system root locus is sketched in Figure 9.28.



Note: This figure is not drawn to scale.

FIGURE 9.27 s -plane picture used to calculate the location of the compensator pole for Example 9.4



x = Closed-loop pole
x = Open-loop pole

Note: This figure is not drawn to scale.

FIGURE 9.28 Compensated system root locus

In order to justify our estimates of percent overshoot and settling time, we must show that the second-order approximation is valid. To perform this validity check, we search for the third and fourth closed-loop poles found beyond -42.96 and between -5 and -6 in Figure 9.28. Searching these regions for the gain equal to that of the compensated dominant pole, 1423 , we find that the third and fourth poles are at -43.8 and -5.134 , respectively. Since -43.8 is more than 20 times the real part of the dominant pole, the effect of the third closed-loop pole is negligible. Since the closed-loop pole at -5.134 is close to the zero at -5 , we have pole-zero cancellation, and the second-order approximation is valid.

All results for this design and two other designs, which place the compensator zero arbitrarily at -2 and -4 and follow similar design techniques, are summarized in Table 9.4. Each design should be verified by a simulation, which could consist of using MATLAB (discussed at the end of this example) or the state-space model and the step-response program discussed in Appendix H.1 at www.wiley.com/college/nise. We have performed a simulation for this design problem, and the results are shown by parenthetical entries next to the estimated values in the table. The only design that disagrees with the simulation is the case where the compensator zero is at -2 . For this case the closed-loop pole and zero do not cancel.

A sketch of the root locus, which you should generate, shows why the effect of the zero is pronounced, causing the response to be different from that predicted. Placing the zero to the right of the pole at -4 creates a portion of the root locus that is between the origin and the zero. In other words, there is a closed-loop pole closer to the origin than the dominant poles, with little chance of pole-zero cancellation except at high gain. Thus, a quick sketch of the root locus gives us information from which we can make better design decisions. For this example, we want to place the zero on, or to the left of, the pole at -4 , which gives a better chance for pole-zero cancellation and for a higher-order pole that is to the left of the dominant poles and subsequently faster. This is verified by the fact that our results show good second-order approximations for the cases where the zero was placed at -4 and -5 . Again, decisions about where to place the zero are based on simple rules of thumb and must be verified by simulations at the end of the design.

Let us now summarize the results shown in Table 9.4. First we notice differences in the following:

1. The position of the arbitrarily selected zero
2. The amount of improvement in the steady-state error
3. The amount of required gain, K
4. The position of the third and fourth poles and their relative effect upon the second-order approximation. This effect is measured by their distance from the dominant poles or the degree of cancellation with the closed-loop zero.

Once a simulation verifies desired performance, the choice of compensation can be based upon the amount of gain required or the improvement in steady-state error that can be obtained without a lag compensator.

The results of Table 9.4 are supported by simulations of the step response, shown in Figure 9.29 for the uncompensated system and the three lead compensation solutions.

Students who are using MATLAB should now run `ch9p2` in Appendix B. MATLAB will be used to design a lead compensator. You will input the desired percent overshoot from the keyboard. MATLAB

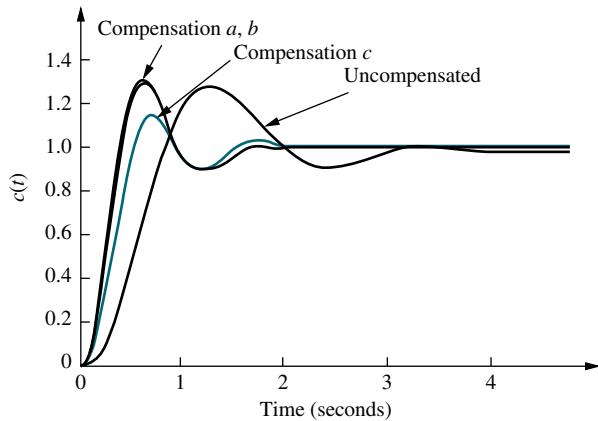


FIGURE 9.29 Uncompensated system and lead compensation responses for Example 9.4

will plot the root locus of the uncompensated system and the percent overshoot line. You will interactively select the gain, after which MATLAB will display the performance characteristics of the uncompensated system and plot its step response. Using these characteristics, you will input the desired settling time and a zero value for the lead compensator. You will then interactively select a value for the compensator pole. MATLAB will respond with a root locus. You can then continue selecting pole values until the root locus goes through the desired point. MATLAB will display the lead compensator, enumerate its performance characteristics, and plot a step response. This exercise solves Example 9.4 using MATLAB.

Skill-Assessment Exercise 9.2

PROBLEM: A unity feedback system with the forward transfer function

$$G(s) = \frac{K}{s(s+7)}$$

is operating with a closed-loop step response that has 15% overshoot. Do the following:

- Evaluate the settling time.
- Design a lead compensator to decrease the settling time by three times. Choose the compensator's zero to be at -10 .

ANSWERS:

- $T_s = 1.143$ s
- $G_{\text{lead}}(s) = \frac{s+10}{s+25.52}$, $K = 476.3$

The complete solution is at www.wiley.com/college/nise.

9.4 Improving Steady-State Error and Transient Response

We now combine the design techniques covered in Sections 9.2 and 9.3 to obtain improvement in steady-state error and transient response *independently*. Basically, we first improve the transient response by using the methods of Section 9.3. Then we improve the steady-state error of this compensated system by applying the methods of Section 9.2. A disadvantage of this approach is the slight decrease in the speed of the response when the steady-state error is improved.

As an alternative, we can improve the steady-state error first and then follow with the design to improve the transient response. A disadvantage of this approach is that the improvement in transient response in some cases yields deterioration in the improvement of the steady-state error, which was designed first. In other cases, the improvement in transient response yields further improvement in steady-state errors. Thus, a system can be overdesigned with respect to steady-state errors. Overdesign is usually not a problem unless it affects cost or produces other design problems. In this textbook, we first design for transient response and then design for steady-state error.

The design can use either active or passive compensators, as previously described. If we design an active PD controller followed by an active PI controller, the resulting compensator is called a *proportional-plus-integral-plus-derivative (PID) controller*. If we first design a passive lead compensator and then design a passive lag compensator, the resulting compensator is called a *lag-lead compensator*.

PID Controller Design

A PID controller is shown in Figure 9.30. Its transfer function is

$$G_c(s) = K_1 + \frac{K_2}{s} + K_3s = \frac{K_1s + K_2 + K_3s^2}{s} = \frac{K_3 \left(s^2 + \frac{K_1}{K_3}s + \frac{K_2}{K_3} \right)}{s} \quad (9.21)$$

which has two zeros plus a pole at the origin. One zero and the pole at the origin can be designed as the ideal integral compensator; the other zero can be designed as the ideal derivative compensator.

The design technique, which is demonstrated in Example 9.5, consists of the following steps:

1. Evaluate the performance of the uncompensated system to determine how much improvement in transient response is required.
2. Design the PD controller to meet the transient response specifications. The design includes the zero location and the loop gain.

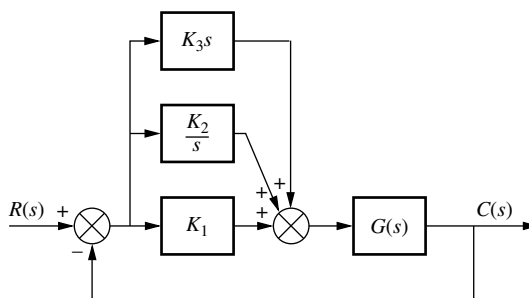


FIGURE 9.30 PID controller

3. Simulate the system to be sure all requirements have been met.
4. Redesign if the simulation shows that requirements have not been met.
5. Design the PI controller to yield the required steady-state error.
6. Determine the gains, K_1 , K_2 , and K_3 , in Figure 9.30.
7. Simulate the system to be sure all requirements have been met.
8. Redesign if simulation shows that requirements have not been met.

Example 9.5

PID Controller Design

PROBLEM: Given the system of Figure 9.31, design a PID controller so that the system can operate with a peak time that is two-thirds that of the uncompensated system at 20% overshoot and with zero steady-state error for a step input.

SOLUTION: Note that our solution follows the eight-step procedure described earlier.

Step 1 Let us first evaluate the uncompensated system operating at 20% overshoot. Searching along the 20% overshoot line ($\zeta = 0.456$) in Figure 9.32, we find the dominant poles to be $-5.415 \pm j10.57$ with a gain of 121.5. A third pole, which exists at -8.169 , is found by searching the region

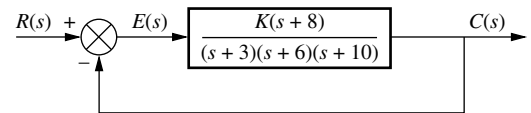


FIGURE 9.31 Uncompensated feedback control system for Example 9.5

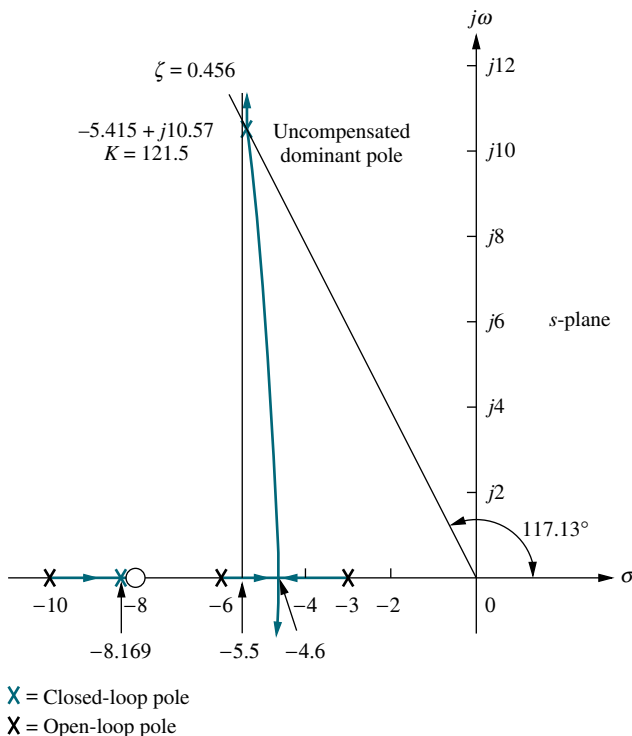


FIGURE 9.32 Root locus for the uncompensated system of Example 9.5

TABLE 9.5 Predicted characteristics of uncompensated, PD-, and PID-compensated systems of Example 9.5

	Uncompensated	PD-compensated	PID-compensated
Plant and compensator	$\frac{K(s+8)}{(s+3)(s+6)(s+10)}$	$\frac{K(s+8)(s+55.92)}{(s+3)(s+6)(s+10)}$	$\frac{K(s+8)(s+55.92)(s+0.5)}{(s+3)(s+6)(s+10)s}$
Dominant poles	$-5.415 \pm j10.57$	$-8.13 \pm j15.87$	$-7.516 \pm j14.67$
K	121.5	5.34	4.6
ζ	0.456	0.456	0.456
ω_n	11.88	17.83	16.49
%OS	20	20	20
T_s	0.739	0.492	0.532
T_p	0.297	0.198	0.214
K_p	5.4	13.27	∞
$e(\infty)$	0.156	0.070	0
Other poles	-8.169	-8.079	-8.099, -0.468
Zeros	-8	-8, -55.92	-8, -55.92, -0.5
Comments	Second-order approx. OK	Second-order approx. OK	Zeros at -55.92 and -0.5 not canceled

between -8 and -10 for a gain equivalent to that at the dominant poles. The complete performance of the uncompensated system is shown in the first column of Table 9.5, where we compare the calculated values to those obtained through simulation (Figure 9.35). We estimate that the uncompensated system has a peak time of 0.297 second at 20% overshoot.

Step 2 To compensate the system to reduce the peak time to two-thirds of that of the uncompensated system, we must first find the compensated system's dominant pole location. The imaginary part of the compensated dominant pole is

$$\omega_d = \frac{\pi}{T_p} = \frac{\pi}{(2/3)(0.297)} = 15.87 \quad (9.22)$$

Thus, the real part of the compensated dominant pole is

$$\sigma = \frac{\omega_d}{\tan 117.13^\circ} = -8.13 \quad (9.23)$$

Next we design the compensator. Using the geometry shown in Figure 9.33, we calculate the compensating zero's location. Using the root locus program, we find the sum of angles from the uncompensated system's poles and zeros to the desired compensated dominant pole to be -198.37° . Thus, the contribution required from the compensator zero is $198.37^\circ - 180^\circ = 18.37^\circ$. Assume that the compensator zero is located at $-z_c$, as shown in Figure 9.33. Since

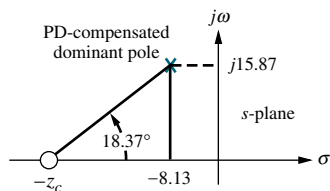
$$\frac{15.87}{z_c - 8.13} = \tan 18.37^\circ \quad (9.24)$$

then

$$z_c = 55.92 \quad (9.25)$$

Thus, the PD controller is

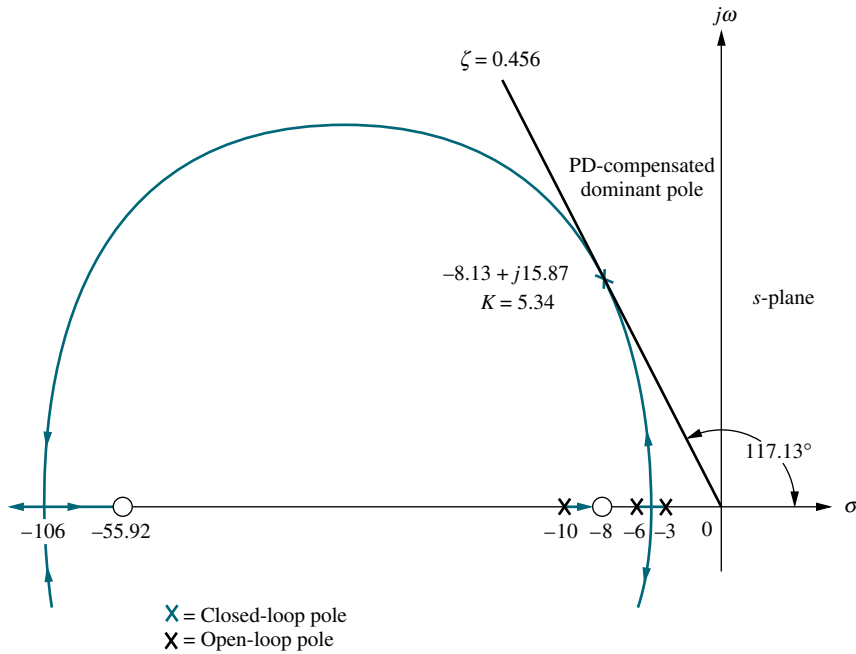
$$G_{PD}(s) = (s + 55.92) \quad (9.26)$$



X = Closed-loop pole

Note: This figure is not drawn to scale.

FIGURE 9.33 Calculating the PD compensator zero for Example 9.5



Note: This figure is not drawn to scale.

FIGURE 9.34 Root locus for PD-compensated system of Example 9.5

The complete root locus for the PD-compensated system is sketched in Figure 9.34. Using a root locus program, the gain at the design point is 5.34. Complete specifications for ideal derivative compensation are shown in the third column of Table 9.5.

Steps 3 and 4 We simulate the PD-compensated system, as shown in Figure 9.35. We see the reduction in peak time and the improvement in steady-state error over the uncompensated system.

Step 5 After we design the PD controller, we design the ideal integral compensator to reduce the steady-state error to zero for a step input. Any ideal integral compensator zero will work, as long as the zero is placed close to the origin. Choosing the ideal integral compensator to be

$$G_{PI}(s) = \frac{s + 0.5}{s} \tag{9.27}$$

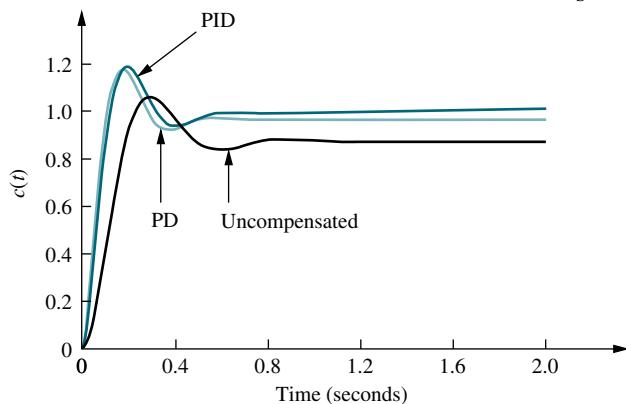
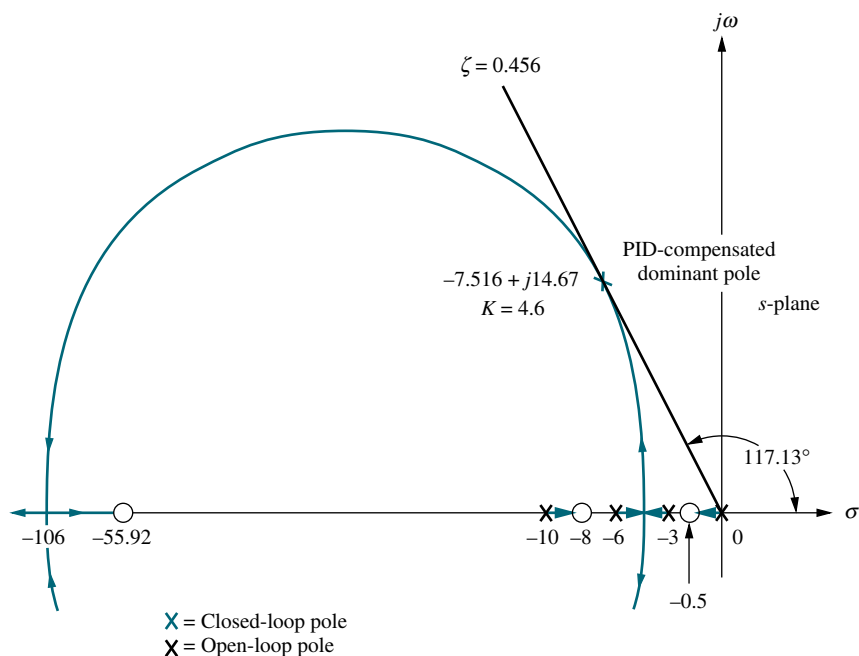


FIGURE 9.35 Step responses for uncompensated, PD-compensated, and PID-compensated systems of Example 9.5



Note: This figure is not drawn to scale.

FIGURE 9.36 Root locus for PID-compensated system of Example 9.5

we sketch the root locus for the PID-compensated system, as shown in Figure 9.36. Searching the 0.456 damping ratio line, we find the dominant, second-order poles to be $-7.516 \pm j14.67$, with an associated gain of 4.6. The remaining characteristics for the PID-compensated system are summarized in the fourth column of Table 9.5.

Step 6 Now we determine the gains, K_1, K_2 , and K_3 , in Figure 9.30. From Eqs. (9.26) and (9.27), the product of the gain and the PID controller is

$$\begin{aligned}
 G_{\text{PID}}(s) &= \frac{K(s + 55.92)(s + 0.5)}{s} = \frac{4.6(s + 55.92)(s + 0.5)}{s} \\
 &= \frac{4.6(s^2 + 56.42s + 27.96)}{s}
 \end{aligned} \tag{9.28}$$

Matching Eqs. (9.21) and (9.28), $K_1 = 259.5$, $K_2 = 128.6$, and $K_3 = 4.6$

Steps 7 and 8 Returning to Figure 9.35, we summarize the results of our design. PD compensation improved the transient response by decreasing the time required to reach the first peak as well as yielding some improvement in the steady-state error. The complete PID controller further improved the steady-state error without appreciably changing the transient response designed with the PD controller. As we have mentioned before, the PID controller exhibits a slower response, reaching the final value of unity at approximately 3 seconds. If this is undesirable, the speed of the system must be increased by redesigning the ideal derivative compensator or moving the PI controller zero farther from the origin. Simulation plays an important role in this type of design since our derived equation for settling time is not applicable for this part of the response, where there is a slow correction of the steady-state error.