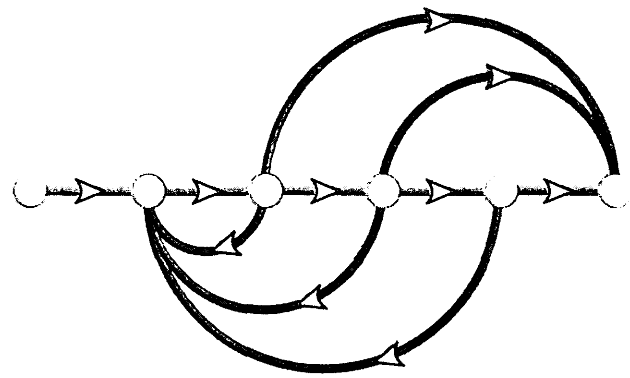
Design via Root Locus

9



Chapter Learning Outcomes

After completing this chapter the student will be able to:

- Use the root locus to design cascade compensators to improve the steady-state error (Sections 9.1–9.2)
- Use the root locus to design cascade compensators to improve the transient response (Section 9.3)
- Use the root locus to design cascade compensators to improve both the steady-state error and the transient response (Section 9.4)
- Use the root locus to design feedback compensators to improve the transient response (Section 9.5)
- Realize the designed compensators physically (Section 9.6)

Case Study Learning Outcomes

You will be able to demonstrate your knowledge of the chapter objectives with case studies as follows:

- Given the antenna azimuth position control system shown on the front endpapers, you will be able to design a cascade compensator to meet transient response and steady-state error specifications.
- Given the pitch or heading control system for the UFSS vehicle shown on the back endpapers, you will be able to design a cascade or feedback compensator to meet transient response specifications.

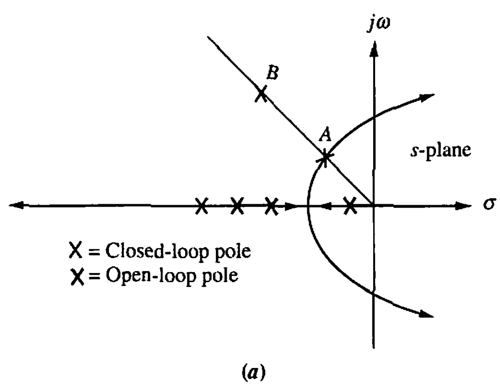
9.1 Introduction

In Chapter 8, we saw that the root locus graphically displayed both transient response and stability information. The locus can be sketched quickly to get a general idea of the changes in transient response generated by changes in gain. Specific points on the locus also can be found accurately to give quantitative design information.

The root locus typically allows us to choose the proper loop gain to meet a transient response specification. As the gain is varied, we move through different regions of response. Setting the gain at a particular value yields the transient response dictated by the poles at that point on the root locus. Thus, *we are limited to those responses that exist along the root locus.*

Improving Transient Response

Flexibility in the design of a desired transient response can be increased if we can design for transient responses that are not on the root locus. Figure 9.1(a) illustrates the concept. Assume that the desired transient response, defined by percent overshoot and settling time, is represented by point B. Unfortunately, on the current root locus at the specified percent overshoot, we only can obtain the settling time represented by point A after a simple gain adjustment. Thus, our goal is to speed up the response at A to that of B, without affecting the percent overshoot. This increase in speed cannot be accomplished by a simple gain adjustment, since point B does not lie on the root locus. Figure 9.1(b) illustrates the improvement in the transient response we seek: The faster response has the same percent overshoot as the slower response.



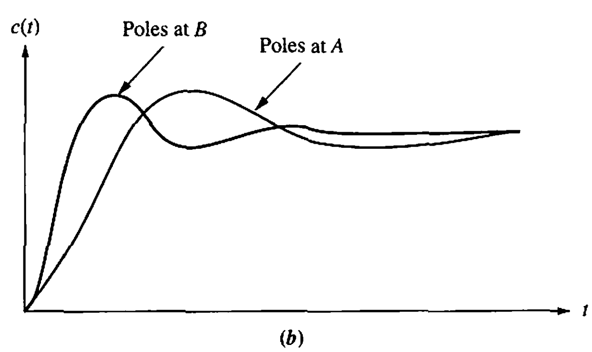


FIGURE 9.1 **a.** Sample root locus, showing possible design point via gain adjustment (A) and desired design point that cannot be met via simple gain adjustment (B); **b.** responses from poles at A and B

One way to solve our problem is to replace the existing system with a system whose root locus intersects the desired design point, *B*. Unfortunately, this replacement is expensive and counterproductive. Most systems are chosen for characteristics other than transient response. For example, an elevator cage and motor are chosen for speed and power. Components chosen for their transient response may not necessarily meet, for example, power requirements.

Rather than change the existing system, we augment, or *compensate*, the system with *additional* poles and zeros, so that the compensated system has a root locus that goes through the desired pole location for some value of gain. One of the advantages of compensating a system in this way is that additional poles and zeros can be added at the low-power end of the system before the plant. Addition of compensating poles and zeros need not interfere with the power output requirements of the system or present additional load or design problems. The compensating poles and zeros can be generated with a passive or an active network.

A possible disadvantage of compensating a system with additional open-loop poles and zeros is that the system order can increase, with a subsequent effect on the desired response. In Chapters 4 and 8, we discussed the effect of additional closed-loop poles and zeros on the transient response. At the beginning of the design process discussed in this chapter, we determine the proper location of additional *open-loop* poles and zeros to yield the desired second-order *closed-loop* poles. However, we do not know the location of the higher-order *closed-loop* poles until the end of the design. Thus, we should evaluate the transient response through simulation after the design is complete to be sure the requirements have been met.

In Chapter 12, when we discuss state-space design, the disadvantage of finding the location of higher-order closed-loop poles after the design will be eliminated by techniques that allow the designer to specify and design the location of all the closed-loop poles at the beginning of the design process.

One method of compensating for transient response that will be discussed later is to insert a differentiator in the forward path in parallel with the gain. We can visualize the operation of the differentiator with the following example. Assuming a position control with a step input, we note that the error undergoes an initial large change. Differentiating this rapid change yields a large signal that drives the plant. The output from the differentiator is much larger than the output from the pure gain. This large, initial input to the plant produces a faster response. As the error approaches its final value, its derivative approaches zero, and the output from the differentiator becomes negligible compared to the output from the gain.

Improving Steady-State Error

Compensators are not only used to improve the transient response of a system; they are also used *independently* to improve the steady-state error characteristics. Previously, when the system gain was adjusted to meet the transient response specification, steady-state error performance deteriorated, since both the transient response and the static error constant were related to the gain. The higher the gain, the smaller the steady-state error, but the larger the percent overshoot. On the other hand, reducing gain to reduce overshoot increased the steady-state error. If we use dynamic compensators, compensating networks can be designed that will allow us to meet transient and steady-state error specifications *simultaneously*.[1] We no longer

[1] The word *dynamic* describes compensators with noninstantaneous transient response. The transfer functions of such compensators are functions of the Laplace variable, *s*, rather than pure gain.

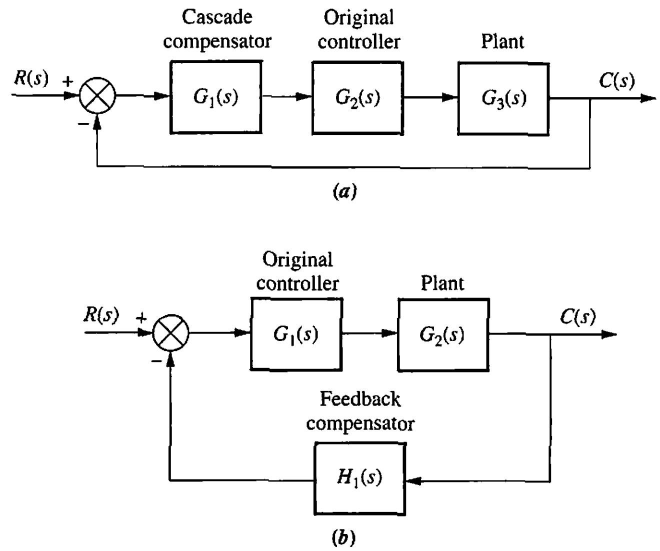


FIGURE 9.2 Compensation techniques: **a.** cascade; **b.** feedback

need to compromise between transient response and steady-state error, as long as the system operates in its linear range.

In Chapter 7, we learned that steady-state error can be improved by adding an open-loop pole at the origin in the forward path, thus increasing the system type and driving the associated steady-state error to zero. This additional pole at the origin requires an integrator for its realization.

In summary, then, transient response is improved with the addition of differentiation, and steady-state error is improved with the addition of integration in the forward path.

Configurations

Two configurations of compensation are covered in this chapter: cascade compensation and feedback compensation. These methods are modeled in Figure 9.2. With cascade compensation, the compensating network, $G_1(s)$, is placed at the low-power end of the forward path in cascade with the plant. If feedback compensation is used, the compensator, $H_1(s)$, is placed in the feedback path. Both methods change the open-loop poles and zeros, thereby creating a new root locus that goes through the desired closed-loop pole location.

Compensators

Compensators that use pure integration for improving steady-state error or pure differentiation for improving transient response are defined as *ideal compensators*. Ideal compensators must be implemented with active networks, which, in the case of electric networks, require the use of active amplifiers and possible additional power sources. An advantage of ideal integral compensators is that steady-state error is reduced to zero. Electromechanical ideal compensators, such as tachometers, are often used to improve transient response, since they can be conveniently interfaced with the plant.

Other design techniques that preclude the use of active devices for compensation can be adopted. These compensators, which can be implemented with passive elements such as resistors and capacitors, do not use pure integration and differentiation and are not ideal compensators. Advantages of passive networks are that they are less expensive and do not require additional power sources for their operation. Their disadvantage is that the steady-state error is not driven to zero in cases where ideal compensators yield zero error.

Thus, the choice between an active or a passive compensator revolves around cost, weight, desired performance, transfer function, and the interface between the compensator and other hardware. In Sections 9.2, 9.3, and 9.4, we first discuss cascade compensator design using ideal compensation and follow with cascade compensation using compensators that are not implemented with pure integration and differentiation.

9.2 Improving Steady-State Error via Cascade Compensation

In this section, we discuss two ways to improve the steady-state error of a feedback control system using cascade compensation. One objective of this design is to improve the steady-state error without appreciably affecting the transient response.

The first technique is *ideal integral compensation*, which uses a pure integrator to place an open-loop, forward-path pole at the origin, thus increasing the system type and reducing the error to zero. The second technique does not use pure integration. This compensation technique places the pole near the origin, and although it does not drive the steady-state error to zero, it does yield a measurable reduction in steady-state error.

While the first technique reduces the steady-state error to zero, the compensator must be implemented with active networks, such as amplifiers. The second technique, although it does not reduce the error to zero, does have the advantage that it can be implemented with a less expensive passive network that does not require additional power sources.

The names associated with the compensators come either from the method of implementing the compensator or from the compensator's characteristics. Systems that feed the error forward to the plant are called *proportional control systems.* Systems that feed the integral of the error to the plant are called *integral control systems.* Finally, systems that feed the derivative of the error to the plant are called *derivative control systems.* Thus, in this section we call the ideal integral compensator a *proportional-plus-integral* (*PI*) *controller*, since the implementation, as we will see, consists of feeding the error (proportional) plus the integral of the error forward to the plant. The second technique uses what we call a *lag compensator*. The name of this compensator comes from its frequency response characteristics, which will be discussed in Chapter 11. Thus, we use the name *PI controller* interchangeably with *ideal integral compensator*, and we use the name *lag compensator* when the cascade compensator does not employ pure integration.

Ideal Integral Compensation (PI)

Steady-state error can be improved by placing an open-loop pole at the origin, because this increases the system type by one. For example, a Type 0 system responding to a step input with a finite error responds with zero error if the system type is increased by one. Active circuits can be used to place poles at the origin. Later in this chapter, we show how to build an integrator with active electronic circuits.

To see how to improve the steady-state error without affecting the transient response, look at Figure 9.3(a). Here we have a system operating with a desirable

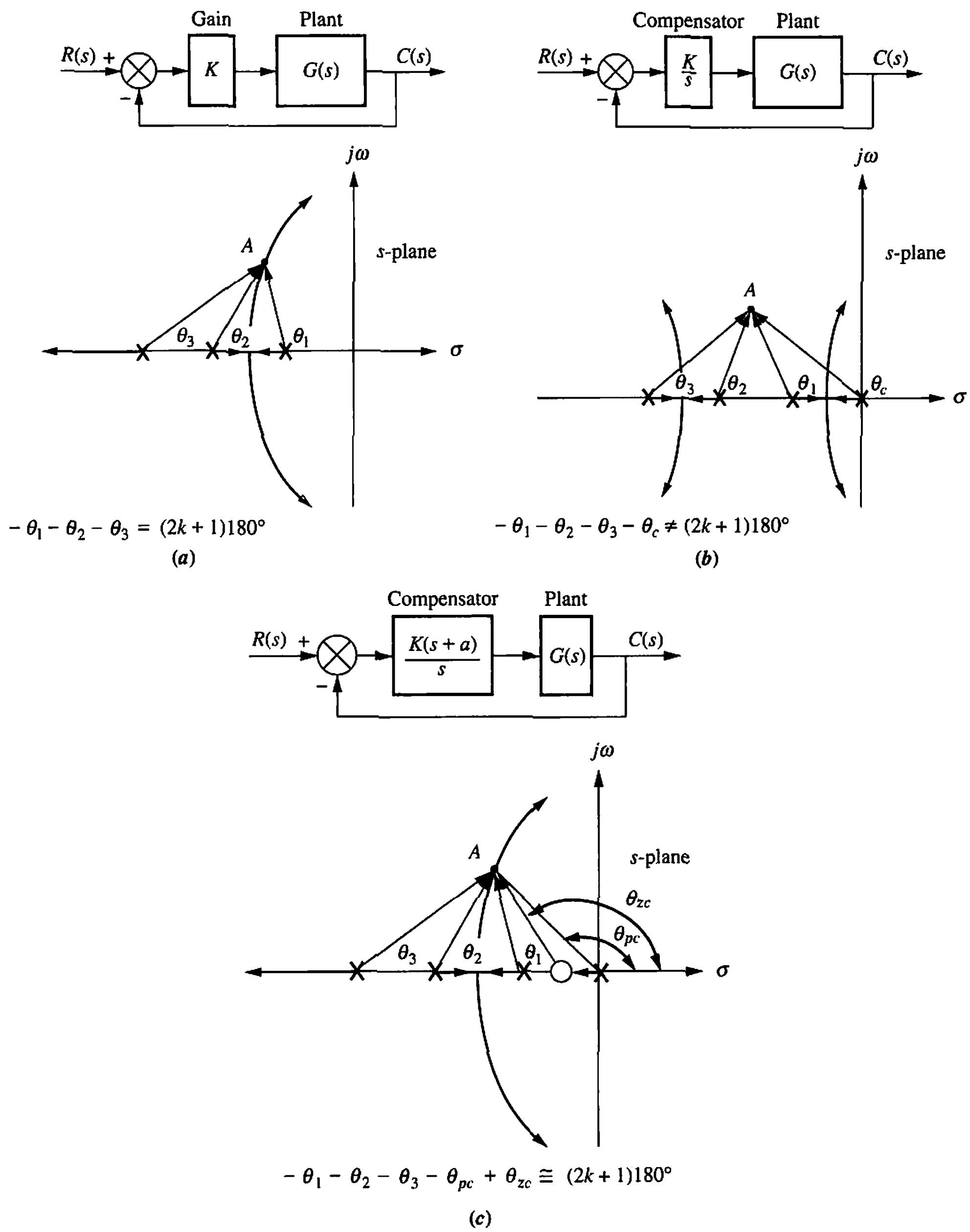


FIGURE 9.3 Pole at A is **a.** on the root locus without compensator; **b.** not on the root locus with compensator pole added; **c.** approximately on the root locus with compensator pole and zero added

transient response generated by the closed-loop poles at A. If we add a pole at the origin to increase the system type, the angular contribution of the open-loop poles at point A is no longer 180°, and the root locus no longer goes through point A, as shown in Figure 9.3(b).

To solve the problem, we also add a zero close to the pole at the origin, as shown in Figure 9.3(c). Now the angular contribution of the compensator zero and compensator pole cancel out, point A is still on the root locus, and the system type has been increased. Furthermore, the required gain at the dominant pole is about the same as before compensation, since the ratio of lengths from the compensator pole and the compensator zero is approximately unity. Thus, we have improved the steady-state error without appreciably affecting the transient response. A compensator with a pole at the origin and a zero close to the pole is called an *ideal integral compensator.*

In the example that follows, we demonstrate the effect of ideal integral compensation. An open-loop pole will be placed at the origin to increase the system type and drive the steady-state error to zero. An open-loop zero will be placed very close to the open-loop pole at the origin so that the original closed-loop poles on the original root locus still remain at approximately the same points on the compensated root locus.

Example 9.1

Effect of an Ideal Integral Compensator

PROBLEM: Given the system of Figure 9.4(*a*), operating with a damping ratio of 0.174, show that the addition of the ideal integral compensator shown in Figure 9.4(*b*) reduces the steady-state error to zero for a step input without appreciably affecting transient response. The compensating network is chosen with a pole at the origin to increase the system type and a zero at -0.1, close to the compensator pole, so that the angular contribution of the compensator evaluated at the original, dominant, second-order poles is approximately zero. Thus, the original, dominant, second-order closed-loop poles are still approximately on the new root locus.

SOLUTION: We first analyze the uncompensated system and determine the location of the dominant, second-order poles. Next we evaluate the uncompensated steady-state error for a unit step input. The root locus for the uncompensated system is shown in Figure 9.5.

A damping ratio of 0.174 is represented by a radial line drawn on the s-plane at 100.02°. Searching along this line with the root locus program discussed in Appendix H at www.wiley.com/college/nise, we find that the dominant poles are $0.694 \pm j3.926$ for a gain, K, of 164.6. Now look for the third pole on the root locus beyond -10 on the real axis. Using the root locus program and searching for the same gain as that of the dominant pair, $K = 164.6$, we find that the third pole is approximately at -11.61. This gain yields $K_p = 8.23$. Hence, the steady-state error is

$$e(\infty) = \frac{1}{1+K_p} = \frac{1}{1+8.23} = 0.108 \tag{9.1}$$

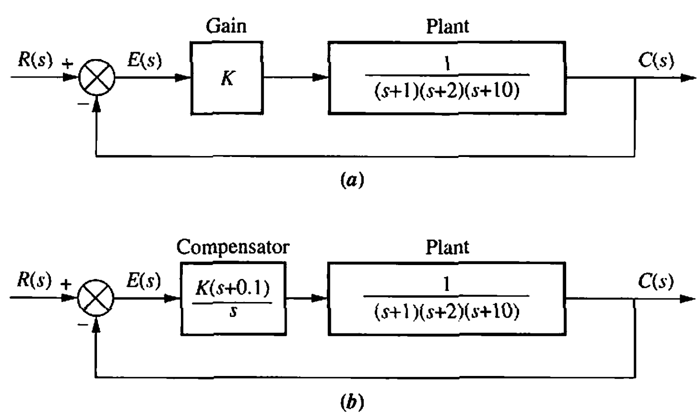


FIGURE 9.4 Closed-loop system for Example 9.1: **a.** before compensation; **b.** after ideal integral compensation

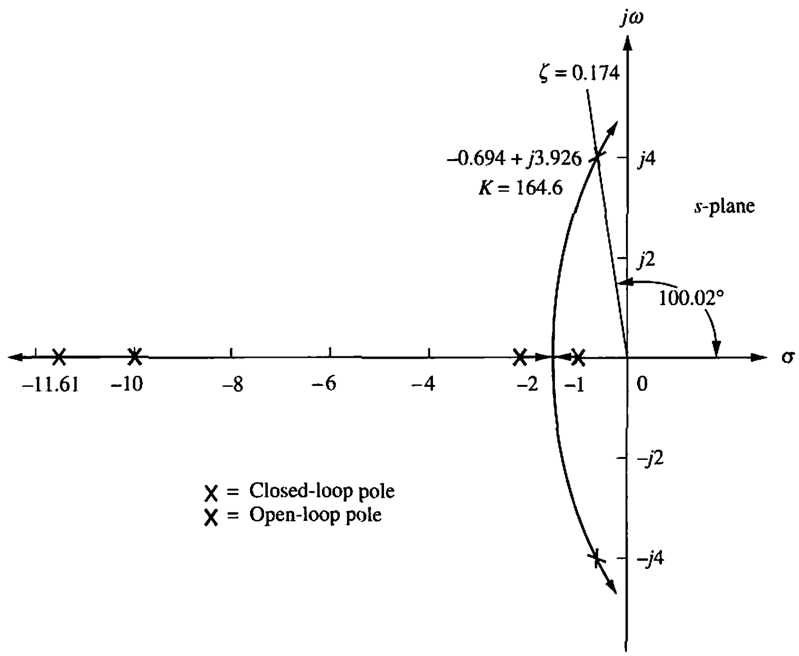


FIGURE 9.5 Root locus for uncompensated system of Figure 9.4(*a*)

Adding an ideal integral compensator with a zero at −0.1, as shown in Figure 9.4(*b*), we obtain the root locus shown in Figure 9.6. The dominant second-order poles, the third pole beyond −10, and the gain are approximately the same as for the uncompensated system. Another section of the compensated root locus is between the origin and −0.1. Searching this region for the same gain at the dominant pair, $K = 158.2$, the fourth closed-loop pole is found at −0.0902, close

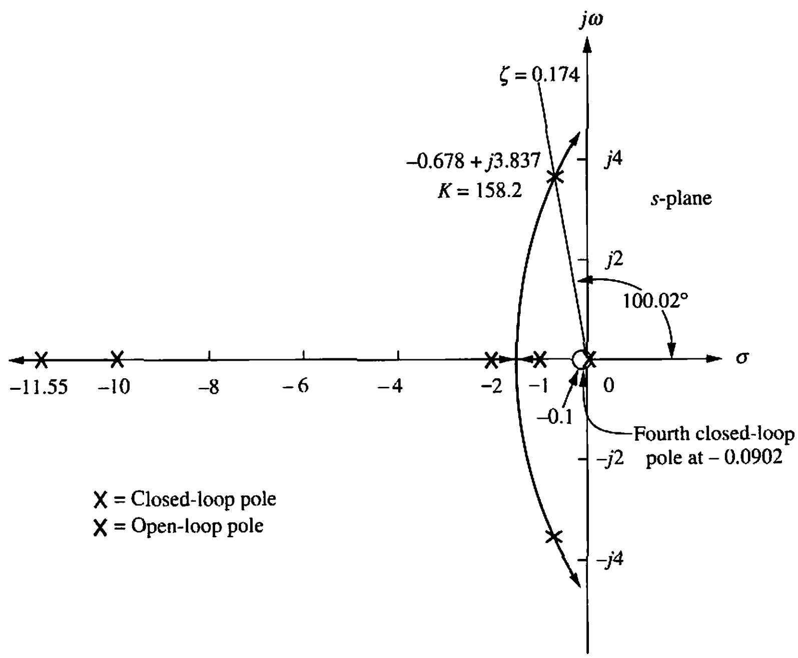


FIGURE 9.6 Root locus for compensated system of Figure 9.4(*b*)

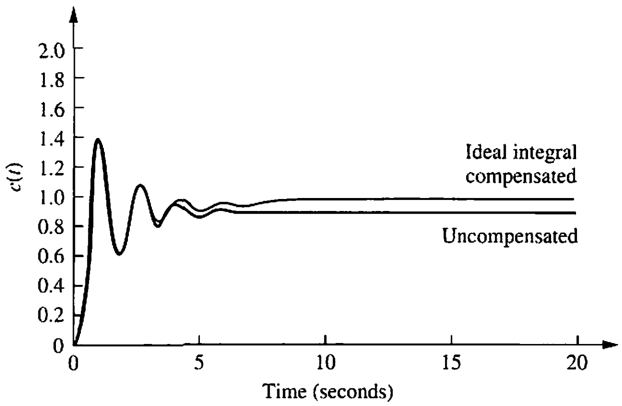


FIGURE 9.7 Ideal integral compensated system response and the uncompensated system response of Example 9.1

enough to the zero to cause pole-zero cancellation. Thus, the compensated system's closed-loop poles and gain are approximately the same as the uncompensated system's closed-loop poles and gain, which indicates that the transient response of the compensated system is about the same as the uncompensated system. However, the compensated system, with its pole at the origin, is a Type 1 system; unlike the uncompensated system, it will respond to a step input with zero error.

Figure 9.7 compares the uncompensated response with the ideal integral compensated response. The step response of the ideal integral compensated system approaches unity in the steady state, while the uncompensated system approaches 0.892. Thus, the ideal integral compensated system responds with zero steady-state error. The transient response of both the uncompensated and the ideal integral compensated systems is the same up to approximately 3 seconds. After that time the integrator in the compensator, shown in Figure 9.4(*b*), slowly compensates for the error until zero error is finally reached. The simulation shows that it takes 18 seconds for the compensated system to reach to within ±2% of the final value of unity, while the uncompensated system takes about 6 seconds to settle to within ±2% of its final value of 0.892. The compensation at first may appear to yield deterioration in the settling time. However, notice that the compensated system reaches the uncompensated system's final value in about the same time. The remaining time is used to improve the steady-state error over that of the uncompensated system.

A method of implementing an ideal integral compensator is shown in Figure 9.8. The compensating network precedes $G(s)$ and is an ideal integral compensator since

$$G_c(s) = K_1 + \frac{K_2}{s} = \frac{K_1\left(s + \frac{K_2}{K_1}\right)}{s} \tag{9.2}$$

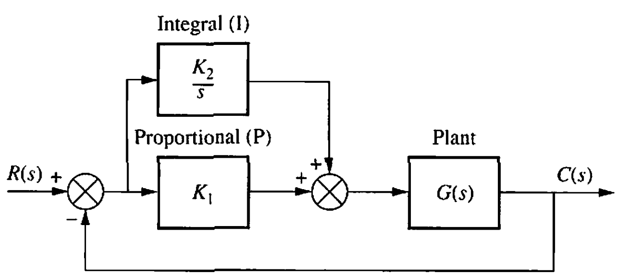


FIGURE 9.8 PI controller

The value of the zero can be adjusted by varying K_2/K_1. In this implementation, the error and the integral of the error are fed forward to the plant, $G(s)$. Since Figure 9.8 has both proportional and integral control, the ideal integral controller, or compensator, is given the alternate name *PI controller*. Later in the chapter we will see how to implement each block, K_1 and K_2/s.

Lag Compensation

Ideal integral compensation, with its pole on the origin, requires an active integrator. If we use passive networks, the pole and zero are moved to the left, close to the origin, as shown in Figure 9.9(c). One may guess that this placement of the pole, although it does not increase the system type, does yield an improvement in the static error constant over an uncompensated system. Without loss of generality, we demonstrate that this improvement is indeed realized for a Type 1 system.

Assume the uncompensated system shown in Figure 9.9(a). The static error constant, K_{v_O}, for the system is

$$K_{v_O} = \frac{K z_1 z_2 \cdots}{p_1 p_2 \cdots} \tag{9.3}$$

Assuming the lag compensator shown in Figure 9.9(b) and (c), the new static error constant is

$$K_{v_N} = \frac{(K z_1 z_2 \cdots)(z_c)}{(p_1 p_2 \cdots)(p_c)} \tag{9.4}$$

What is the effect on the transient response? Figure 9.10 shows the effect on the root locus of adding the lag compensator. The uncompensated system's root locus is shown in Figure 9.10(a), where point P is assumed to be the dominant pole. If the lag compensator pole and zero are close together, the angular contribution of the

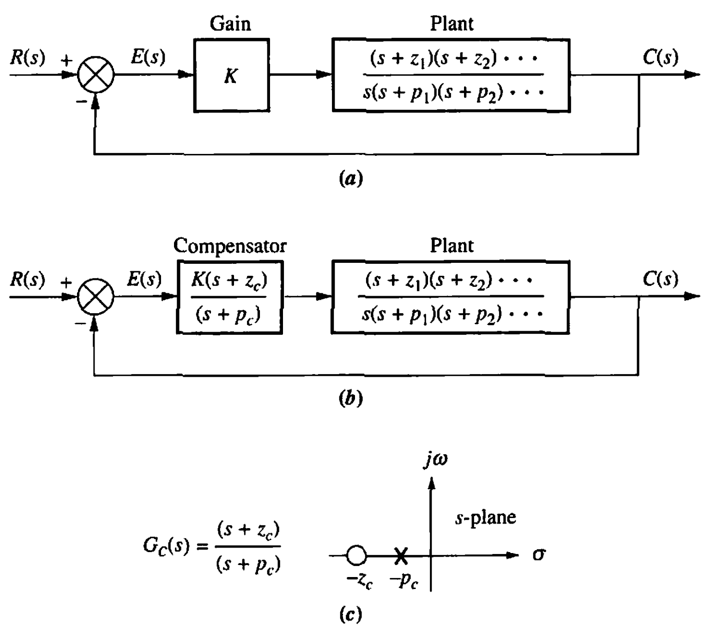


FIGURE 9.9 **a.** Type 1 uncompensated system; **b.** Type 1 compensated system; **c.** compensator pole-zero plot

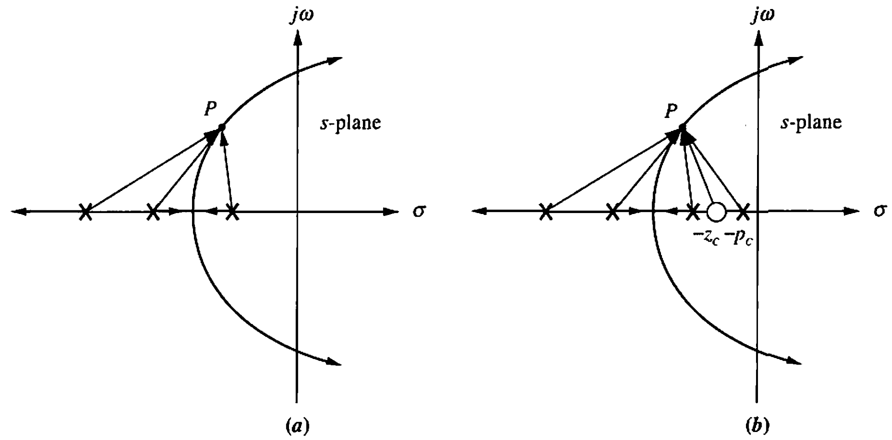


FIGURE 9.10 Root locus: **a.** before lag compensation; **b.** after lag compensation

compensator to point P is approximately zero degrees. Thus, in Figure 9.10(b), where the compensator has been added, point P is still at approximately the same location on the compensated root locus.

What is the effect on the required gain, K? After inserting the compensator, we find that K is virtually the same for the uncompensated and compensated systems, since the lengths of the vectors drawn from the lag compensator are approximately equal and all other vectors have not changed appreciably.

Now, what improvement can we expect in the steady-state error? Since we established that the gain, K, is about the same for the uncompensated and compensated systems, we can substitute Eq. (9.3) into (9.4) and obtain

$$K_{v_N} = K_{v_O}\frac{z_c}{p_c} > K_{v_O} \tag{9.5}$$

Equation (9.5) shows that the improvement in the compensated system's K_v over the uncompensated system's K_v is equal to the ratio of the magnitude of the compensator zero to the compensator pole. In order to keep the transient response unchanged, we know the compensator pole and zero must be close to each other. The only way the ratio of z_c to p_c can be large in order to yield an appreciable improvement in steady-state error and simultaneously have the compensator's pole and zero close to each other to minimize the angular contribution is to place the compensator's pole-zero pair close to the origin. For example, the ratio of z_c to p_c can be equal to 10 if the pole is at −0.001 and the zero is at −0.01. Thus, the ratio is 10, yet the pole and zero are very close, and the angular contribution of the compensator is small.

In conclusion, although the ideal compensator drives the steady-state error to zero, a lag compensator with a pole that is not at the origin will improve the static error constant by a factor equal to z_c/p_c. There also will be a minimal effect upon the transient response if the pole-zero pair of the compensator is placed close to the origin. Later in the chapter we show circuit configurations for the lag compensator. These circuit configurations can be obtained with passive networks and thus do not require the active amplifiers and possible additional power supplies that are required by the ideal integral (PI) compensator. In the following example we design a lag compensator to yield a specified improvement in steady-state error.

Example 9.2

Lag Compensator Design

PROBLEM: Compensate the system of Figure 9.4(*a*), whose root locus is shown in Figure 9.5, to improve the steady-state error by a factor of 10 if the system is operating with a damping ratio of 0.174.

SOLUTION: The uncompensated system error from Example 9.1 was 0.108 with $K_p = 8.23$. A tenfold improvement means a steady-state error of

$$e(\infty) = \frac{0.108}{10} = 0.0108 \tag{9.6}$$

Since

$$e(\infty) = \frac{1}{1+K_p} = 0.0108 \tag{9.7}$$

rearranging and solving for the required K_p yields

$$K_p = \frac{1-e(\infty)}{e(\infty)} \approx \frac{1-0.0108}{0.0108} = 91.59 \tag{9.8}$$

The improvement in K_p from the uncompensated system to the compensated system is the required ratio of the compensator zero to the compensator pole, or

$$\frac{z_c}{p_c} = \frac{K_{p_N}}{K_{p_O}} = \frac{91.59}{8.23} = 11.13 \tag{9.9}$$

Arbitrarily selecting

$$p_c = 0.01 \tag{9.10}$$

we use Eq. (9.9) and find

$$z_c = 11.13 p_c \approx 0.111 \tag{9.11}$$

Let us now compare the compensated system, shown in Figure 9.11, with the uncompensated system. First sketch the root locus of the compensated system, as shown in Figure 9.12. Next search along the $\zeta = 0.174$ line for a multiple of 180° and find that the second-order dominant poles are at $-0.678 \pm j3.836$ with a gain, K, of 158.1. The third and fourth closed-loop poles are at -11.55 and -0.101, respectively, and are found by searching the real axis for a gain equal to that of the dominant poles. All transient and steady-state results for both the uncompensated and the compensated systems are shown in Table 9.1.

The fourth pole of the compensated system cancels its zero. This leaves the remaining three closed-loop poles of the compensated system very close in value to the three closed-loop poles of the uncompensated system. Hence, the transient

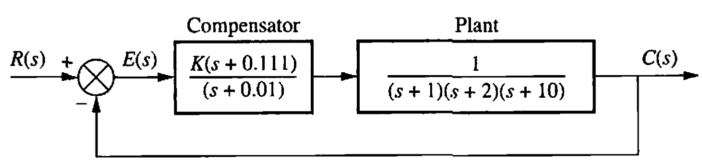


FIGURE 9.11 Compensated system for Example 9.2

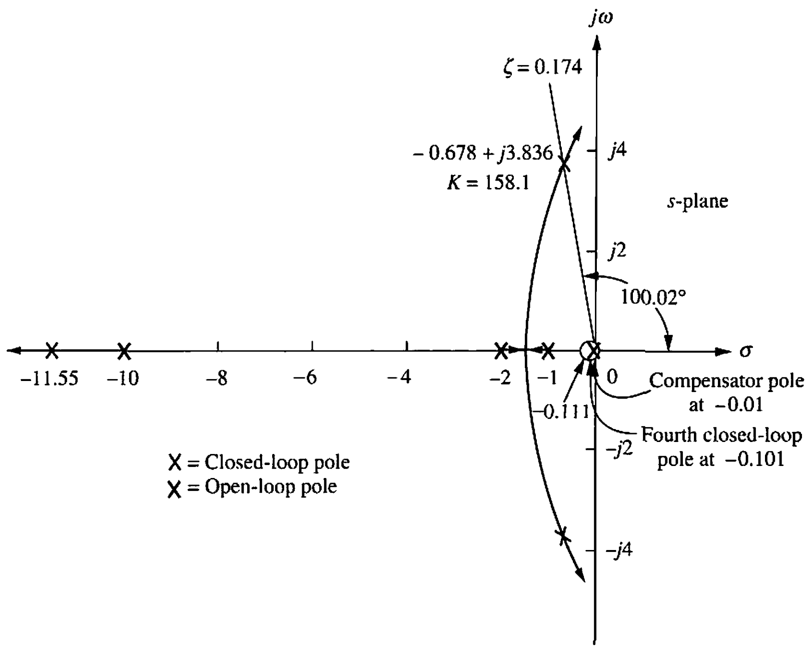


FIGURE 9.12 Root locus for compensated system of Figure 9.11

response of both systems is approximately the same, as is the system gain, but notice that the steady-state error of the compensated system is 1/9.818 that of the uncompensated system and is close to the design specification of a tenfold improvement.

Figure 9.13 shows the effect of the lag compensator in the time domain. Even though the transient responses of the uncompensated and lag-compensated systems are the same, the lag-compensated system exhibits less steady-state error by approaching unity more closely than the uncompensated system.

We now examine another design possibility for the lag compensator and compare the response to Figure 9.13. Let us assume a lag compensator whose pole and zero are 10 times as close to the origin as in the previous design. The results are compared in Figure 9.14. Even though both responses will eventually reach approximately the same steady-state value, the lag compensator previously designed, $G_c(s) = (s + 0.111)/(s + 0.01)$, approaches the final value faster than the proposed lag compensator, $G_c(s) = (s + 0.0111)/(s + 0.001)$. We can explain this phenomenon as follows. From Table 9.1, the previously designed lag compensator

TABLE 9.1 Predicted characteristics of uncompensated and lag-compensated systems for Example 9.2

Parameter	Uncompensated	Lag-compensated
Plant and compensator	$\frac{K}{(s+1)(s+2)(s+10)}$	$\frac{K(s+0.111)}{(s+1)(s+2)(s+10)(s+0.01)}$
K	164.6	158.1
K_p	8.23	87.75
$e(\infty)$	0.108	0.011
Dominant second-order poles	$-0.694 \pm j3.926$	$-0.678 \pm j3.836$
Third pole	−11.61	−11.55
Fourth pole	None	−0.101
Zero	None	−0.111

TryIt 9.1

Use the following MATLAB and Control System Toolbox statements to reproduce Figure 9.13.

```
Gu=zpk([],...
[-1 -2 -10],164.6);
Gc=zpk([-0.111],...
[-0.01],1);
Gce=Gu*Gc;
Tu=feedback(Gu,1);
Tc=feedback(Gce,1);
step(Tu)
hold
step(Tc)
```

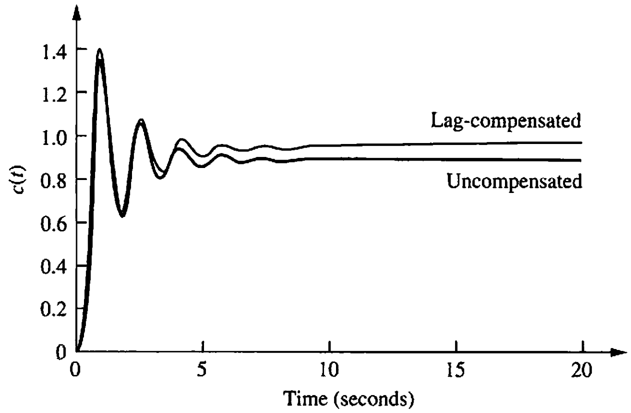


FIGURE 9.13 Step responses of uncompensated and lag-compensated systems for Example 9.2

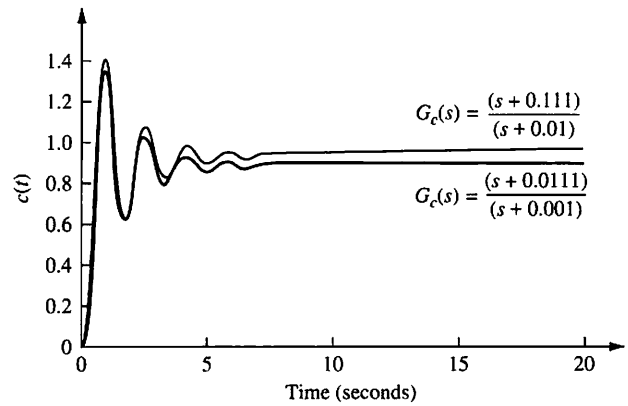


FIGURE 9.14 Step responses of the system for Example 9.2 using different lag compensators

has a fourth closed-loop pole at −0.101. Using the same analysis for the new lag compensator with its open-loop pole 10 times as close to the imaginary axis, we find its fourth closed-loop pole at −0.01. Thus, the new lag compensator has a closed-loop pole closer to the imaginary axis than the original lag compensator. This pole at −0.01 will produce a longer transient response than the original pole at −0.101, and the steady-state value will not be reached as quickly.

Skill-Assessment Exercise 9.1

WileyPLUS
WPCS
Control Solutions

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:

a. Evaluate the steady-state error for a unit ramp input.

b. Design a lag compensator to improve the steady-state error by a factor of 20.

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

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

$$G_c(s) = s + z_c \tag{9.12}$$

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

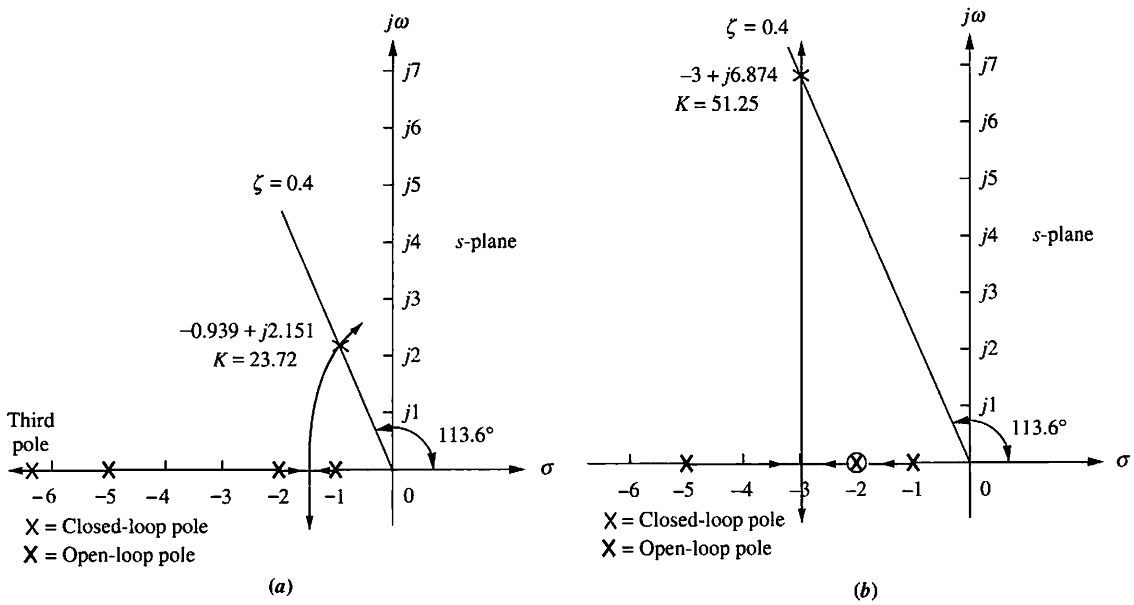


FIGURE 9.15 Using ideal derivative compensation: **a.** uncompensated; **b.** compensator zero at −2; (figure continues)

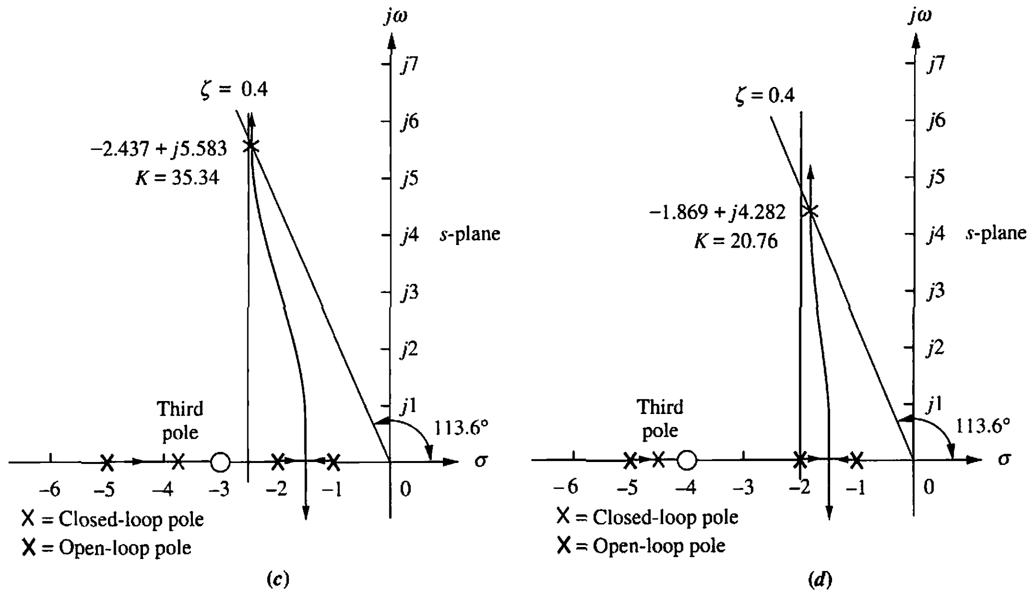


FIGURE 9.15 (*Continued*) **c.** compensator zero at −3; **d.** compensator zero at −4.

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

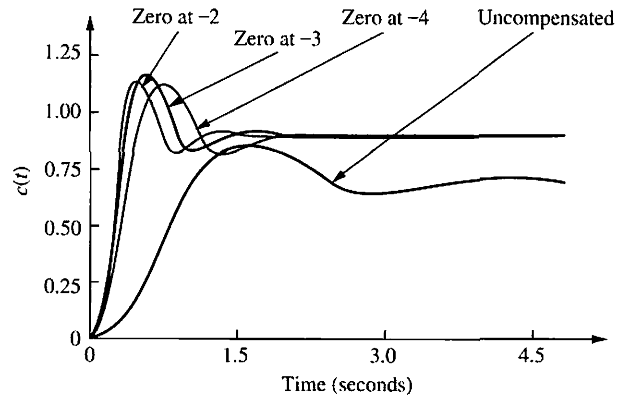


FIGURE 9.16 Uncompensated system and ideal derivative compensation solutions from Table 9.2

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.

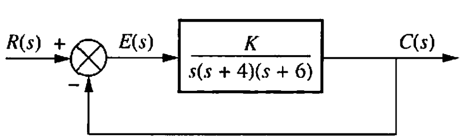


FIGURE 9.17 Feedback control system for Example 9.3

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

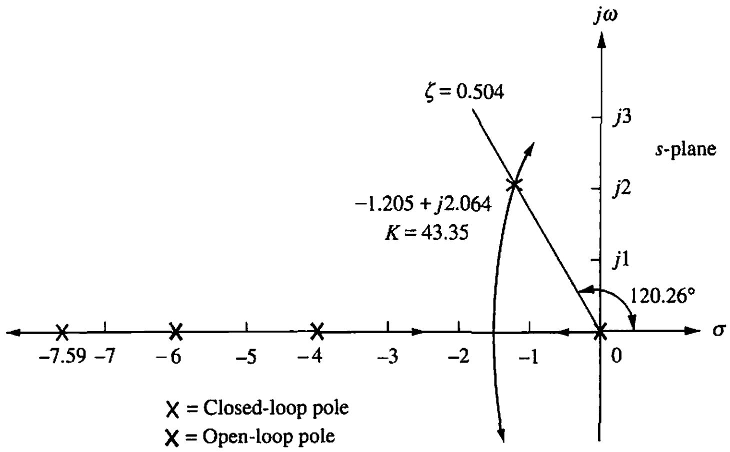


FIGURE 9.18 Root locus for uncompensated system shown in Figure 9.17

time of the uncompensated system is

$$T_s = \frac{4}{\zeta\omega_n} = \frac{4}{1.205} = 3.320 \qquad (9.13)$$

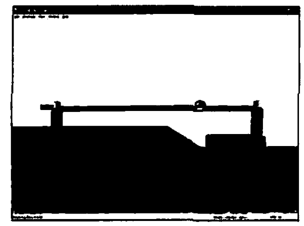
Since our evaluation of percent overshoot and settling time is based upon a second-order approximation, we must check the assumption by finding the third pole and justifying the second-order approximation. Searching beyond −6 on the real axis for a gain equal to the gain of the dominant, second-order 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.

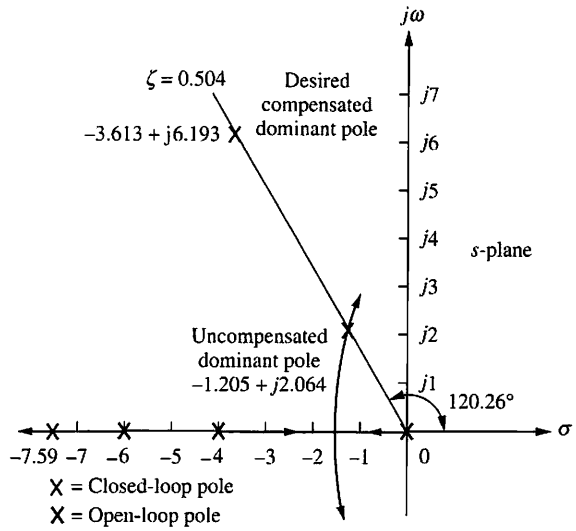


FIGURE 9.19 Compensated dominant pole superimposed over the uncompensated root locus for Example 9.3

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

$$\sigma = \frac{4}{T_s} = \frac{4}{1.107} = 3.613 \tag{9.14}$$

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

$$\omega_d = 3.613 \tan(180° - 120.26°) = 6.193 \tag{9.15}$$

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° − 180° = 95.6°. The geometry is shown in Figure 9.20, where we now must solve for $-\sigma$, the location of the compensator zero.

From the figure,

$$\frac{6.193}{3.613 - \sigma} = \tan(180° - 95.6°) \tag{9.16}$$

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

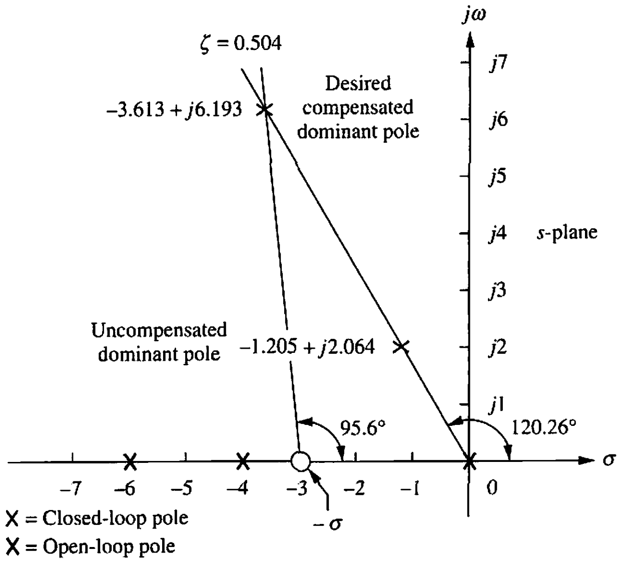


FIGURE 9.20 Evaluating the location of the compensating zero for Example 9.3

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

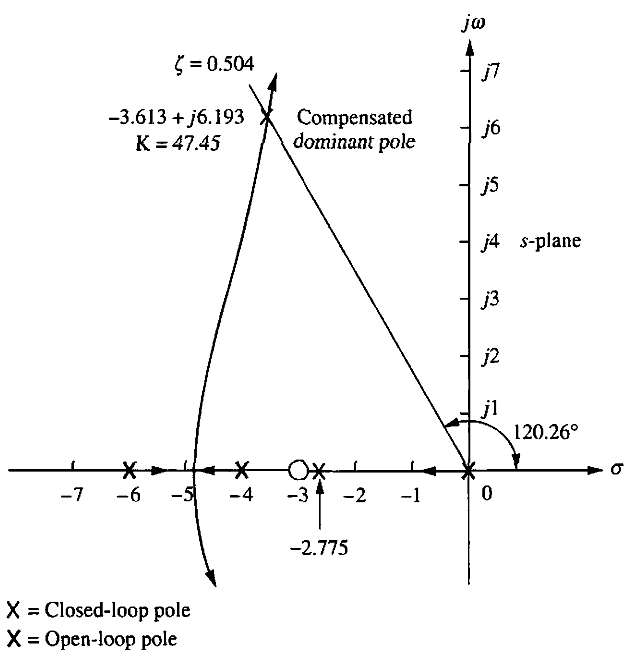


FIGURE 9.21 Root locus for the compensated system of Example 9.3

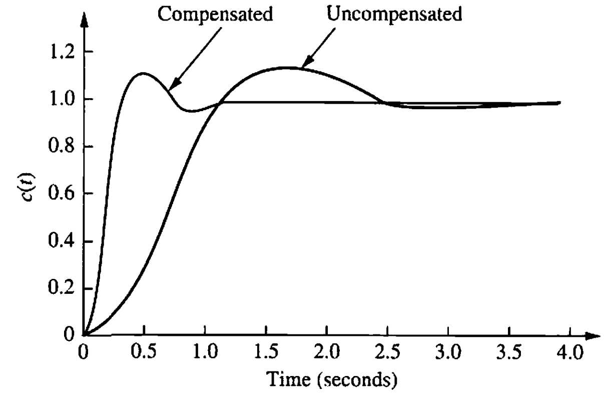


FIGURE 9.22 Uncompensated and compensated system step responses of Example 9.3

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.

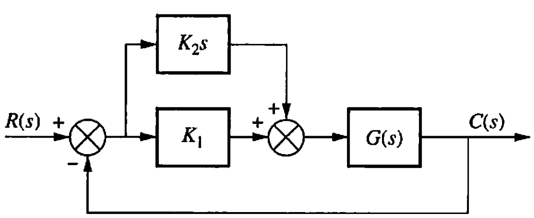


FIGURE 9.23 PD controller

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

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

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

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

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

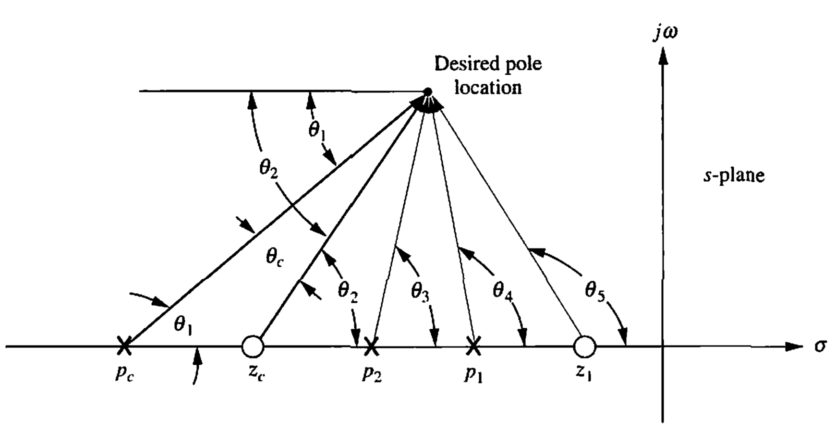


FIGURE 9.24 Geometry of lead compensation

TryIt 9.2

Use MATLAB, the Control System Toobox, 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.

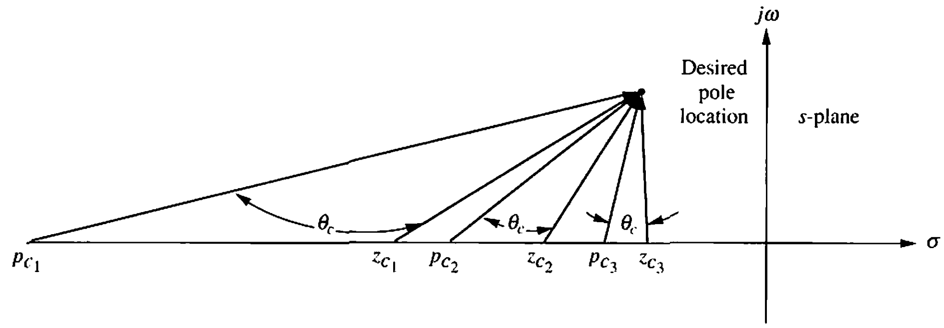


FIGURE 9.25 Three of the infinite possible lead compensator solutions

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.

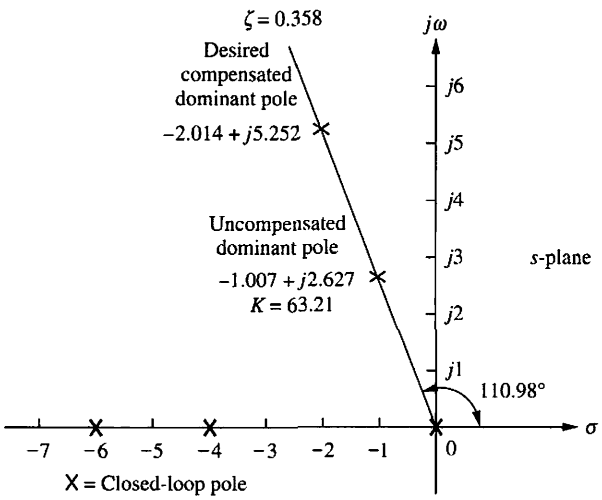


FIGURE 9.26 Lead compensator design, showing evaluation of uncompensated and compensated dominant poles for Example 9.4

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 $\omega_d = -2.014\tan(110.98°) = 5.252$.

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,

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

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.

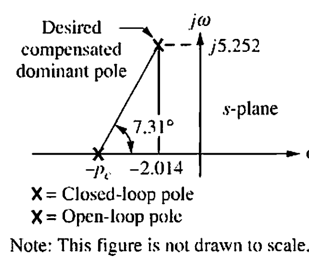


FIGURE 9.27 s-plane picture used to calculate the location of the compensator pole for Example 9.4

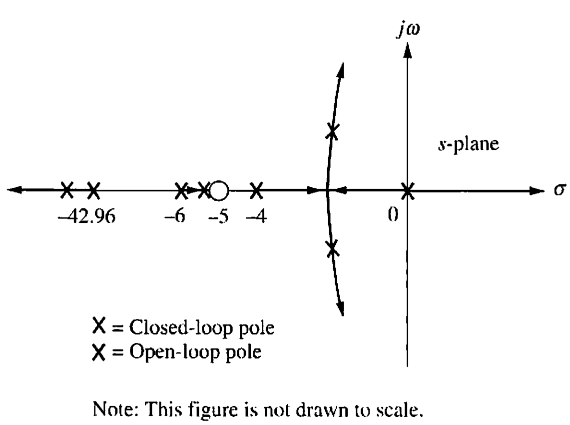


FIGURE 9.28 Compensated system root locus

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.

MATLAB ML

`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`

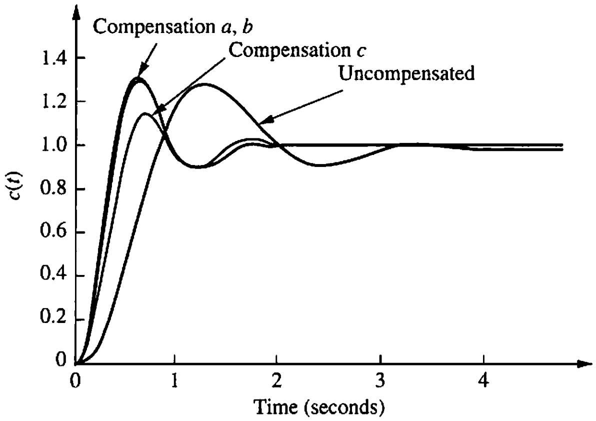


FIGURE 9.29 Uncompensated system and lead compensation responses for Example 9.4

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:

a. Evaluate the settling time.

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

ANSWERS:

a. $T_s = 1.143\text{ s}$

b. $G_{\text{lead}}(s) = \dfrac{s+10}{s+25.52}, \quad K = 476.3$

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

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

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.

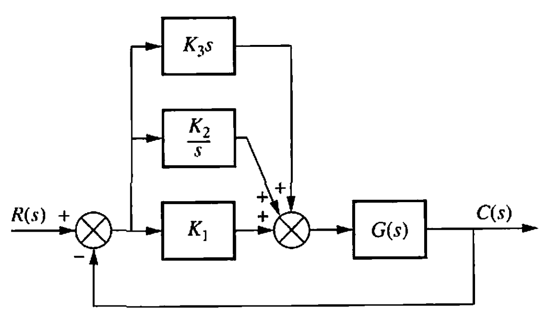


FIGURE 9.30 PID controller

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.

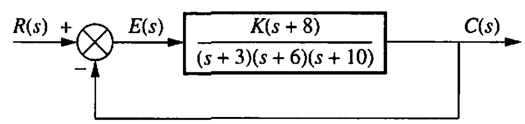


FIGURE 9.31 Uncompensated feedback control system for Example 9.5

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

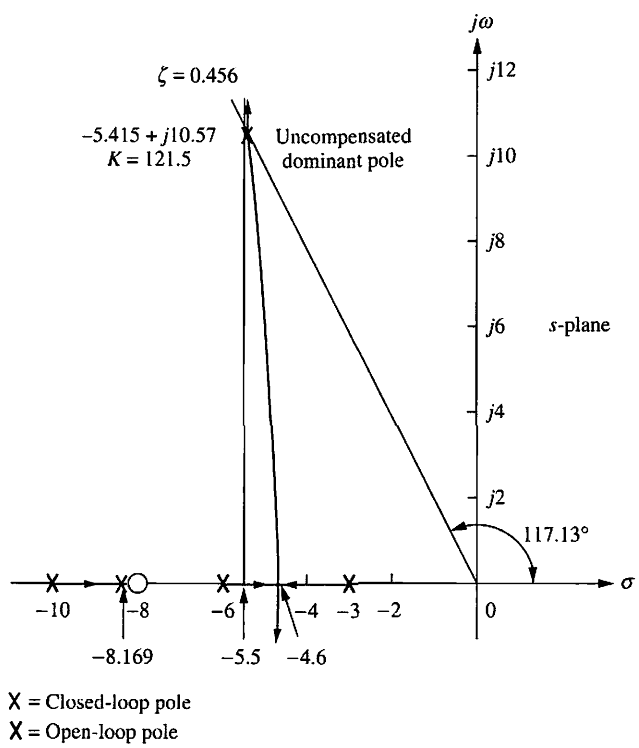


FIGURE 9.32 Root locus for the uncompensated system of Example 9.5

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

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

Thus, the real part of the compensated dominant pole is

$$\sigma = \frac{\omega_d}{\tan 117.13^\circ} = -8.13 \tag{9.23}$$

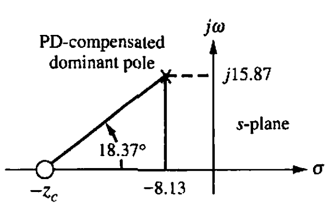


FIGURE 9.33 Calculating the PD compensator zero for Example 9.5

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° − 180° = 18.37°. Assume that the compensator zero is located at $-z_c$, as shown in Figure 9.33. Since

$$\frac{15.87}{z_c - 8.13} = \tan 18.37^\circ \tag{9.24}$$

then

$$z_c = 55.92 \tag{9.25}$$

Thus, the PD controller is

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

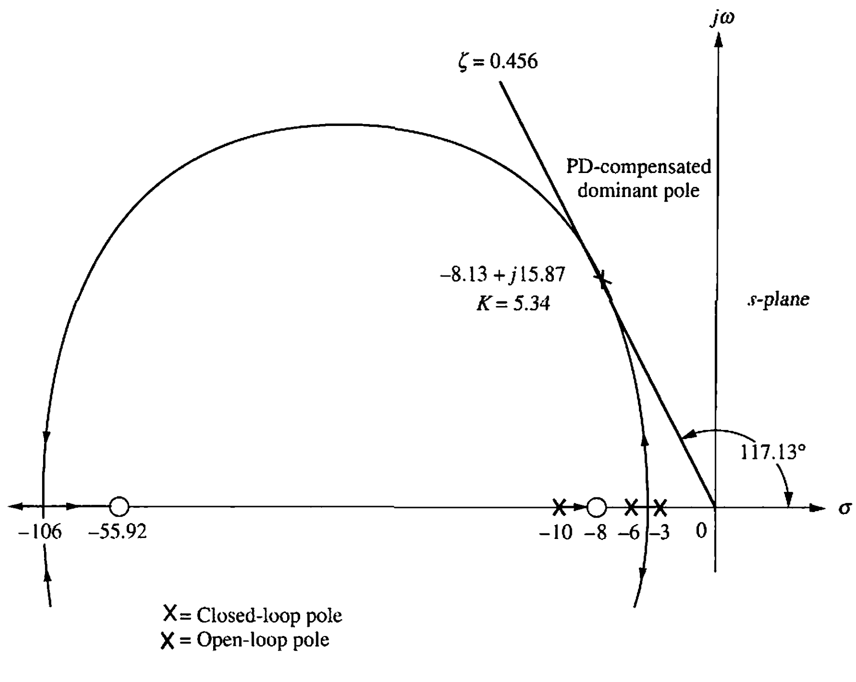


FIGURE 9.34 Root locus for PD-compensated system of Example 9.5

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

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

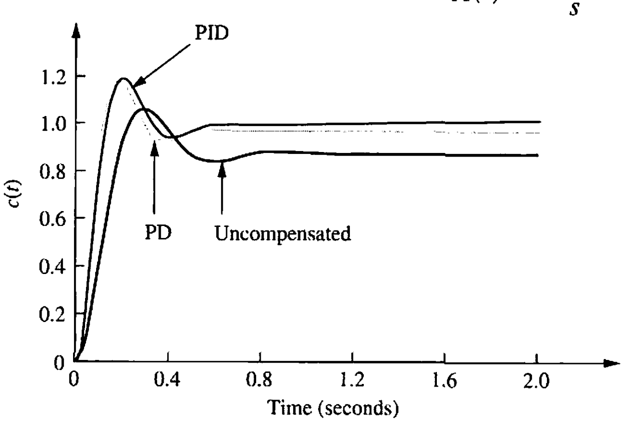


FIGURE 9.35 Step responses for uncompensated, PD-compensated, and PID-compensated systems of Example 9.5

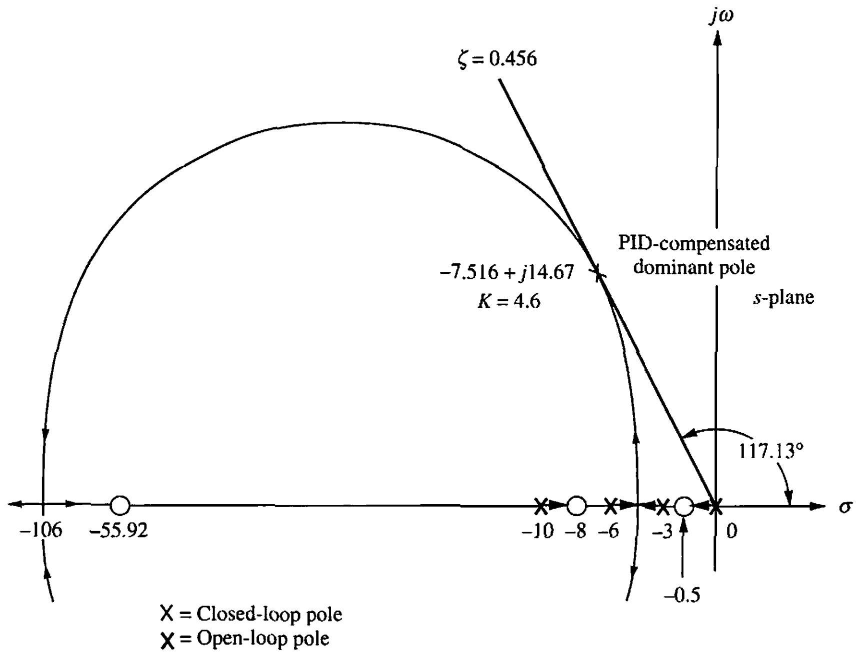


FIGURE 9.36 Root locus for PID-compensated system of Example 9.5

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.

Lag-Lead Compensator Design

In the previous example, we serially combined the concepts of ideal derivative and ideal integral compensation to arrive at the design of a PID controller that improved both the transient response and the steady-state error performance. In the next example, we improve both transient response and the steady-state error by using a lead compensator and a lag compensator rather than the ideal PID. Our compensator is called a *lag-lead compensator.*

We first design the lead compensator to improve the transient response. Next we evaluate the improvement in steady-state error still required. Finally, we design the lag compensator to meet the steady-state error requirement. Later in the chapter we show circuit designs for the passive network. The following steps summarize the design procedure:

1. Evaluate the performance of the uncompensated system to determine how much improvement in transient response is required.
2. Design the lead compensator to meet the transient response specifications. The design includes the zero location, pole location, and the loop gain.
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. Evaluate the steady-state error performance for the lead-compensated system to determine how much more improvement in steady-state error is required.
6. Design the lag compensator to yield the required steady-state error.
7. Simulate the system to be sure all requirements have been met.
8. Redesign if the simulation shows that requirements have not been met.

Example 9.6

Lag-Lead Compensator Design

PROBLEM: Design a lag-lead compensator for the system of Figure 9.37 so that the system will operate with 20% overshoot and a twofold reduction in settling time. Further, the compensated system will exhibit a tenfold improvement in steady-state error for a ramp input.

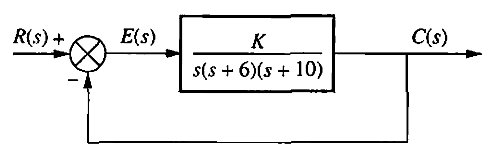


FIGURE 9.37 Uncompensated system for Example 9.6

SOLUTION: Again, our solution follows the steps just described.

Step 1 First we evaluate the performance of the uncompensated system. Searching along the 20% overshoot line ($\zeta = 0.456$) in Figure 9.38, we find the dominant poles at $-1.794 \pm j3.501$, with a gain of 192.1. The performance of the uncompensated system is summarized in Table 9.6.

Step 2 Next we begin the lead compensator design by selecting the location of the compensated system's dominant poles. In order to realize a twofold reduction in settling time, the real part of the dominant pole must be increased by a factor of 2, since the settling time is inversely proportional to the real part. Thus,

$$-\zeta\omega_n = -2(1.794) = -3.588 \tag{9.29}$$

The imaginary part of the design point is

$$\omega_d = \zeta\omega_n \tan 117.13^\circ = 3.588 \tan 117.13^\circ = 7.003 \tag{9.30}$$

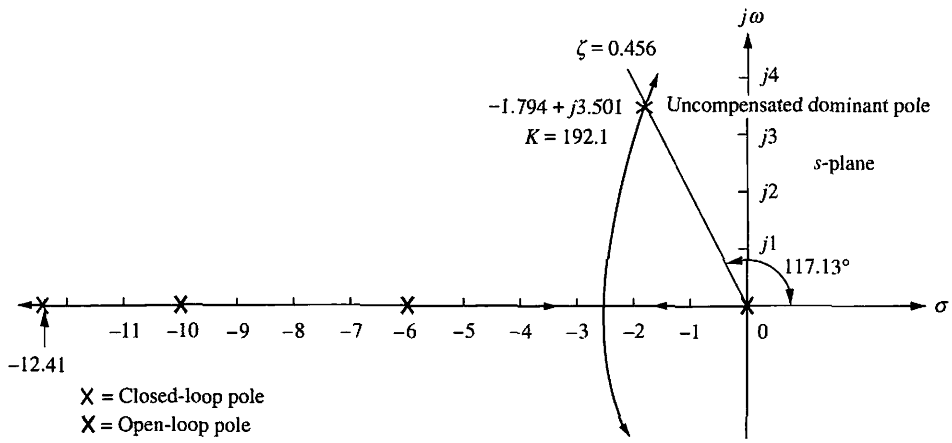


FIGURE 9.38 Root locus for uncompensated system of Example 9.6

Now we design the lead compensator. Arbitrarily select a location for the lead compensator zero. For this example, we select the location of the compensator zero coincident with the open-loop pole at −6. This choice will eliminate a zero and leave the lead-compensated system with three poles, the same number that the uncompensated system has.

We complete the design by finding the location of the compensator pole. Using the root locus program, sum the angles to the design point from the uncompensated system's poles and zeros and the compensator zero and get −164.65°. The difference between 180° and this quantity is the angular contribution required from the compensator pole, or −15.35°. Using the geometry shown in Figure 9.39,

$$\frac{7.003}{p_c - 3.588} = \tan 15.35^\circ \tag{9.31}$$

from which the location of the compensator pole, p_c, is found to be −29.1.

TABLE 9.6 Predicted characteristics of uncompensated, lead-compensated, and lag-lead-compensated systems of Example 9.6

	Uncompensated	**Lead-compensated**	**Lag-lead-compensated**
Plant and compensator	$\frac{K}{s(s+6)(s+10)}$	$\frac{K}{s(s+10)(s+29.1)}$	$\frac{K(s+0.04713)}{s(s+10)(s+29.1)(s+0.01)}$
Dominant poles	$-1.794 \pm j3.501$	$-3.588 \pm j7.003$	$-3.574 \pm j6.976$
K	192.1	1977	1971
ζ	0.456	0.456	0.456
ω_n	3.934	7.869	7.838
$\%OS$	20	20	20
T_s	2.230	1.115	1.119
T_p	0.897	0.449	0.450
K_v	3.202	6.794	31.92
$e(\infty)$	0.312	0.147	0.0313
Third pole	−12.41	−31.92	−31.91, −0.0474
Zero	None	None	−0.04713
Comments	Second-order approx. OK	Second-order approx. OK	Second-order approx. OK

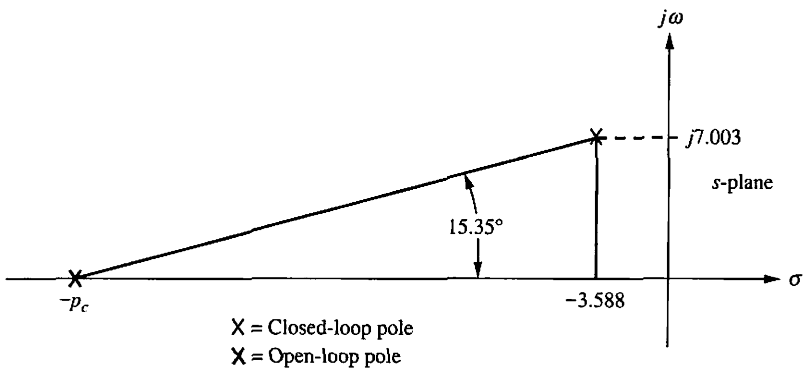


FIGURE 9.39 Evaluating the compensator pole for Example 9.6

The complete root locus for the lead-compensated system is sketched in Figure 9.40. The gain setting at the design point is found to be 1977.

Steps 3 and 4 Check the design with a simulation. (The result for the lead-compensated system is shown in Figure 9.42 and is satisfactory.)

Step 5 Continue by designing the lag compensator to improve the steady-state error. Since the uncompensated system's open-loop transfer function is

$$G(s) = \frac{192.1}{s(s+6)(s+10)} \tag{9.32}$$

the static error constant, K_v, which is inversely proportional to the steady-state error, is 3.201. Since the open-loop transfer function of the lead-compensated system is

$$G_{LC}(s) = \frac{1977}{s(s+10)(s+29.1)} \tag{9.33}$$

the static error constant, K_v, which is inversely proportional to the steady-state error, is 6.794. Thus, the addition of lead compensation has improved the steady-state error by a factor of 2.122. Since the requirements of the problem specified a tenfold improvement, the lag compensator must be designed to improve the steady-state error by a factor of 4.713 ($10/2.122 = 4.713$) over the lead-compensated system.

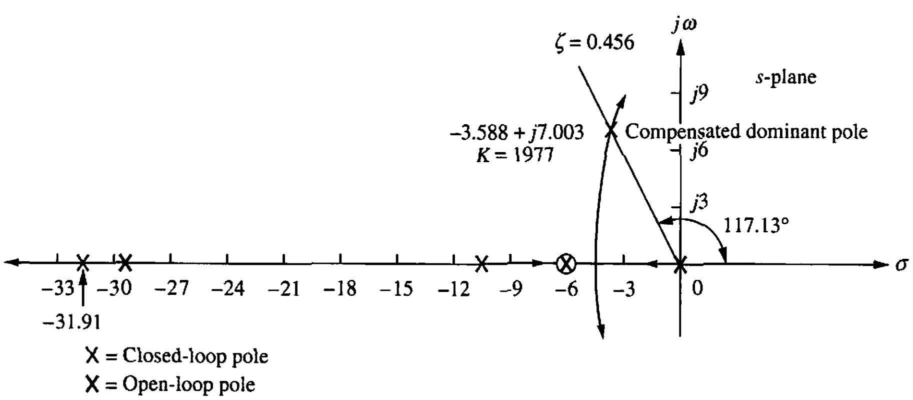
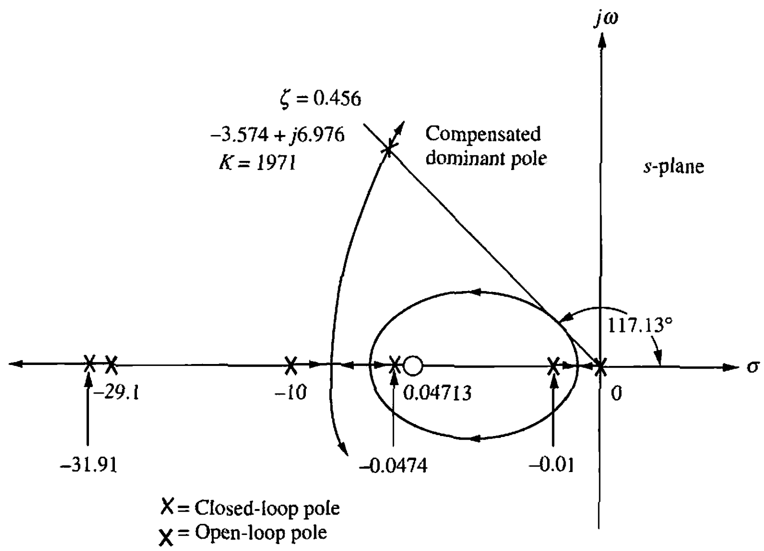


FIGURE 9.40 Root locus for lead-compensated system of Example 9.6



Note: This figure is not drawn to scale.

FIGURE 9.41 Root locus for lag-lead-compensated system of Example 9.6

Step 6 We arbitrarily choose the lag compensator pole at 0.01, which then places the lag compensator zero at 0.04713, yielding

$$G_{\text{lag}}(s) = \frac{(s + 0.04713)}{(s + 0.01)} \tag{9.34}$$

as the lag compensator. The lag-lead-compensated system's open-loop transfer function is

$$G_{\text{LLC}}(s) = \frac{K(s + 0.04713)}{s(s + 10)(s + 29.1)(s + 0.01)} \tag{9.35}$$

where the uncompensated system pole at −6 canceled the lead compensator zero at −6. By drawing the complete root locus for the lag-lead-compensated system and by searching along the 0.456 damping ratio line, we find the dominant, closed-loop poles to be at $-3.574 \pm j6.976$, with a gain of 1971. The lag-lead-compensated root locus is shown in Figure 9.41.

A summary of our design is shown in Table 9.6. Notice that the lag-lead compensation has indeed increased the speed of the system, as witnessed by the settling time or the peak time. The steady-state error for a ramp input has also decreased by about 10 times, as seen from $e(\infty)$.

Step 7 The final proof of our designs is shown by the simulations of Figures 9.42 and 9.43. The improvement in the transient response is shown in Figure 9.42, where we see the peak time occurring sooner in the lag-lead-compensated system. Improvement in the steady-state error for a ramp input is seen in Figure 9.43, where each step of our design yields more improvement. The improvement for the lead-compensated system is shown in Figure 9.43(a), and the final improvement due to the addition of the lag is shown in Figure 9.43(b).

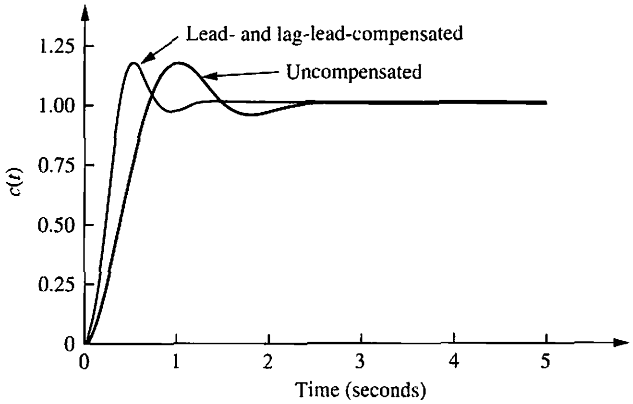


FIGURE 9.42 Improvement in step response for lag-lead-compensated system of Example 9.6

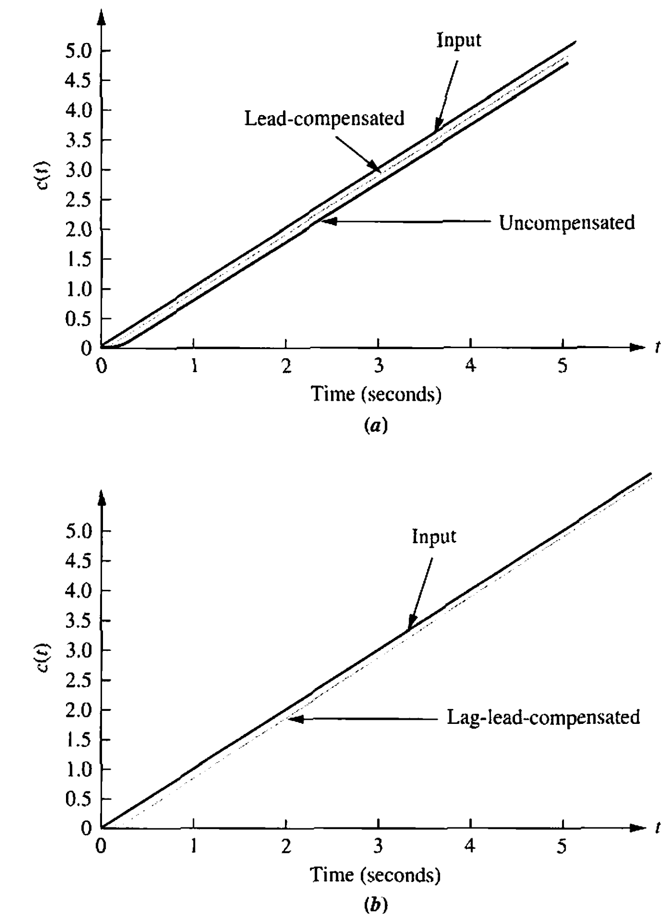


FIGURE 9.43 Improvement in ramp response error for the system of Example 9.6: **a.** lead-compensated; **b.** lag-lead-compensated

In the previous example, we canceled the system pole at –6 with the lead compensator zero. The design technique is the same if you place the lead compensator zero at a different location. Placing a zero at a different location and not canceling the open-loop pole yields a system with one more pole than the example. This increased complexity could make it more difficult to justify a second-order approximation. In any case, simulations should be used at each step to verify performance.

Notch Filter

If a plant, such as a mechanical system, has high-frequency vibration modes, then a desired closed-loop response may be difficult to obtain. These high-frequency vibration modes can be modeled as part of the plant's transfer function by pairs of complex poles near the imaginary axis. In a closed-loop configuration, these poles can move closer to the imaginary axis or even cross into the right half-plane, as shown in Figure 9.44(*a*). Instability or high-frequency oscillations superimposed over the desired response can result (see Figure 9.44(*b*)).

One way of eliminating the high-frequency oscillations is to cascade a *notch filter*[2] with the plant (*Kuo, 1995*), as shown in Figure 9.44(*c*). The notch filter has

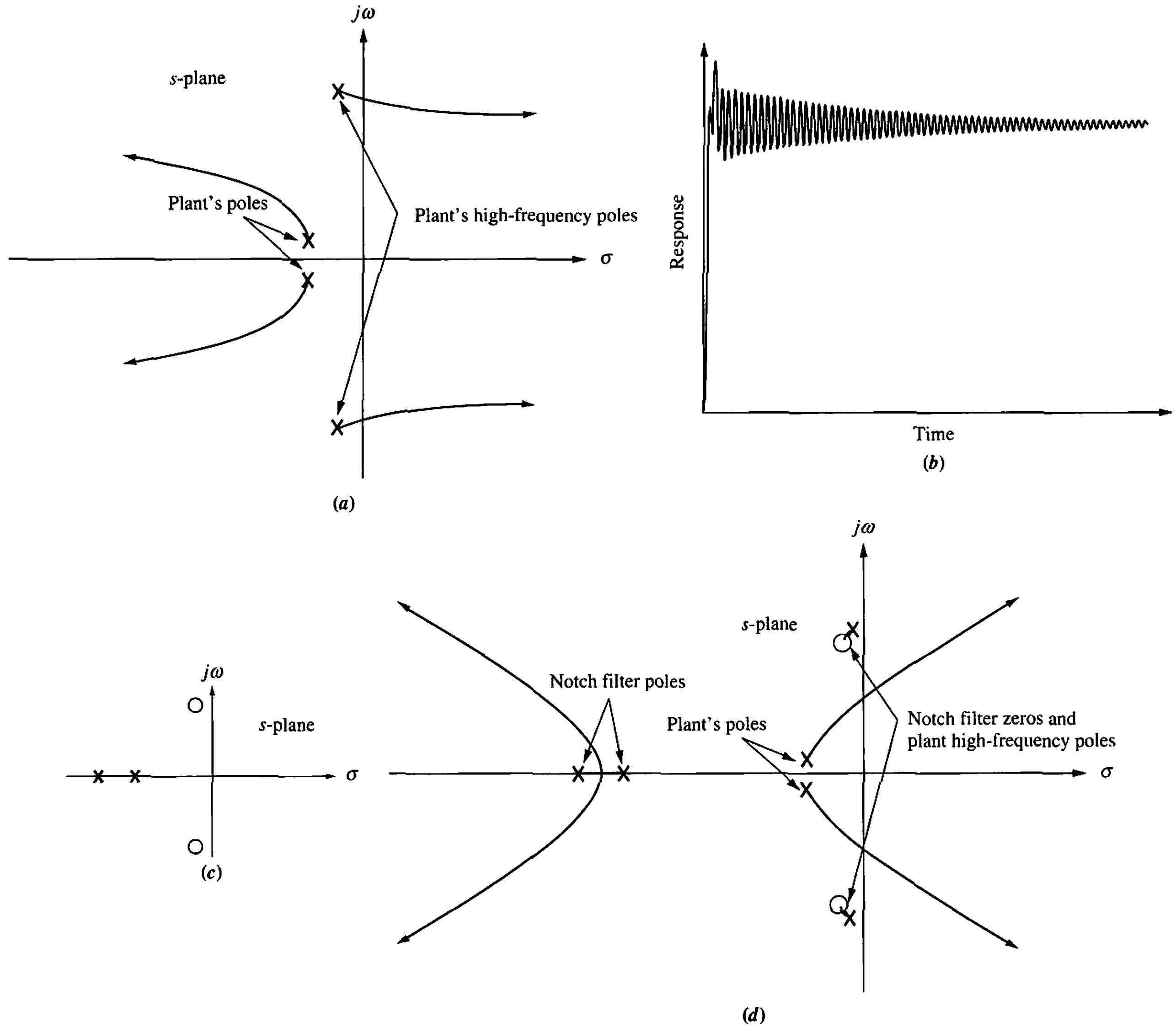


FIGURE 9.44 **a.** Root locus before cascading notch filter; **b.** typical closed-loop step response before cascading notch filter; **c.** pole-zero plot of a notch filter; **d.** root locus after cascading notch filter; (*figure continues*)

[2] The name of this filter comes from the shape of its magnitude frequency response characteristics, which shows a dip near the damped frequency of the high-frequency poles. Magnitude frequency response is discussed in Chapter 10.

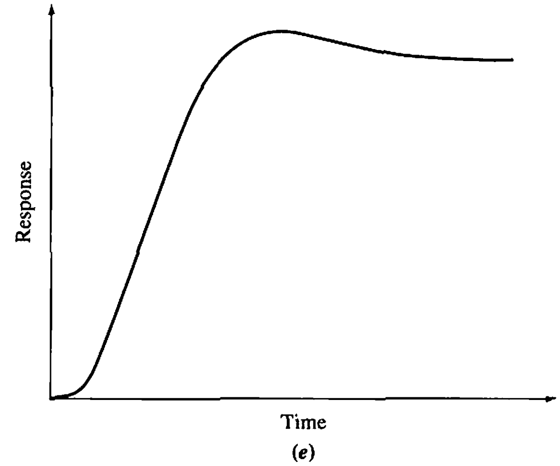


FIGURE 9.44 (*Continued*)
e. closed-loop step response after cascading notch filter

zeros close to the low-damping-ratio poles of the plant as well as two real poles. Figure 9.44(*d*) shows that the root locus branch from the high-frequency poles now goes a short distance from the high-frequency pole to the notch filter's zero. The high-frequency response will now be negligible because of the pole-zero cancellation (see Figure 9.44(*e*)). Other cascade compensators can now be designed to yield a desired response. The notch filter will be applied to Progressive Analysis and Design Problem 55 near the end of this chapter.

Skill-Assessment Exercise 9.3

WileyPLUS WPCS Control Solutions

PROBLEM: A unity feedback system with forward transfer function

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

is operating with a closed-loop step response that has 20% overshoot. Do the following:

a. Evaluate the settling time.
b. Evaluate the steady-state error for a unit ramp input.
c. Design a lag-lead compensator to decrease the settling time by 2 times and decrease the steady-state error for a unit ramp input by 10 times. Place the lead zero at −3.

ANSWERS:

a. $T_s = 1.143\,\text{s}$
b. $e_{\text{ramp}}(\infty) = 0.1189$
c. $G_c(s) = \frac{(s+3)(s+0.092)}{(s+9.61)(s+0.01)}, \quad K = 205.4$

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

Before concluding this section, let us briefly summarize our discussion of cascade compensation. In Sections 9.2, 9.3, and 9.4, we used cascade compensators to improve transient response and steady-state error. Table 9.7 itemizes the types, functions, and characteristics of these compensators.

TABLE 9.7 Types of cascade compensators

Function	Compensator	Transfer function	Characteristics
Improve steady-state error	PI	$K\frac{s+z_c}{s}$	1. Increases system type. 2. Error becomes zero. 3. Zero at $-z_c$ is small and negative. 4. Active circuits are required to implement.
Improve steady-state error	Lag	$K\frac{s+z_c}{s+p_c}$	1. Error is improved but not driven to zero. 2. Pole at $-p_c$ is small and negative. 3. Zero at $-z_c$ is close to, and to the left of, the pole at $-p_c$. 4. Active circuits are not required to implement.
Improve transient response	PD	$K(s+z_c)$	1. Zero at $-z_c$ is selected to put design point on root locus. 2. Active circuits are required to implement. 3. Can cause noise and saturation; implement with rate feedback or with a pole (lead).
Improve transient response	Lead	$K\frac{s+z_c}{s+p_c}$	1. Zero at $-z_c$ and pole at $-p_c$ are selected to put design point on root locus. 2. Pole at $-p_c$ is more negative than zero at $-z_c$. 3. Active circuits are not required to implement.
Improve steady-state error and transient response	PID	$K\frac{(s+z_{\text{lag}})(s+z_{\text{lead}})}{s}$	1. Lag zero at $-z_{\text{lag}}$ and pole at origin improve steady-state error. 2. Lead zero at $-z_{\text{lead}}$ improves transient response. 3. Lag zero at $-z_{\text{lag}}$ is close to, and to the left of, the origin. 4. Lead zero at $-z_{\text{lead}}$ is selected to put design point on root locus. 5. Active circuits required to implement. 6. Can cause noise and saturation; implement with rate feedback or with an additional pole.
Improve steady-state error and transient response	Lag-lead	$K\frac{(s+z_{\text{lag}})(s+z_{\text{lead}})}{(s+p_{\text{lag}})(s+p_{\text{lead}})}$	1. Lag pole at $-p_{\text{lag}}$ and lag zero at $-z_{\text{lag}}$ are used to improve steady-state error. 2. Lead pole at $-p_{\text{lead}}$ and lead zero at $-z_{\text{lead}}$ are used to improve transient response. 3. Lag pole at $-p_{\text{lag}}$ is small and negative. 4. Lag zero at $-z_{\text{lag}}$ is close to, and to the left of, lag pole at $-p_{\text{lag}}$. 5. Lead zero at $-z_{\text{lead}}$ and lead pole at $-p_{\text{lead}}$ are *selected to put design point on root locus.* 6. Lead pole at $-p_{\text{lead}}$ is more negative than lead zero at $-z_{\text{lead}}$. 7. Active circuits are not required to implement.

9.5 Feedback Compensation

In Section 9.4, we used cascade compensation as a way to improve transient response and steady-state response independently. Cascading a compensator with the plant is not the only way to reshape the root locus to intersect the closed-loop s-plane poles that yield a desired transient response. Transfer functions designed to be placed in a feedback path can also reshape the root locus. Figure 9.45 is a generic configuration showing a compensator, $H_c(s)$, placed in the *minor loop* of a feedback control system. Other configurations arise if we consider K unity, $G_2(s)$ unity, or both unity.

The design procedures for feedback compensation can be more complicated than for cascade compensation. On the other hand, feedback compensation can yield faster responses. Thus, the engineer has the luxury of designing faster responses into portions of a control loop in order to provide isolation. For example, the transient response of the ailerons and rudder control systems of an aircraft can be designed separately to be fast in order to reduce the effect of their dynamic response on the steering control loop. Feedback compensation can be used in cases where noise problems preclude the use of cascade compensation. Also, feedback compensation may not require additional amplification, since the signal passing through the compensator originates at the high-level output of the forward path and is delivered to a low-level input in the forward path. For example, let K and $G_2(s)$ in Figure 9.45 be unity. The input to the feedback compensator, $K_fH_c(s)$, is from the high-level output of $G_1(s)$, while the output of $K_fH_c(s)$ is one of the low-level inputs into K_1. Thus, there is a reduction in level through $K_fH_c(s)$, and amplification is usually not required.

A popular feedback compensator is a rate sensor that acts as a differentiator. In aircraft and ship applications, the rate sensor can be a rate gyro that responds with an output voltage proportional to the input angular velocity. In many other systems this rate sensor is implemented with a tachometer. A tachometer is a voltage generator that yields a voltage output proportional to input rotational speed. This compensator can easily be geared to the position output of a system. Figure 9.46 is a position

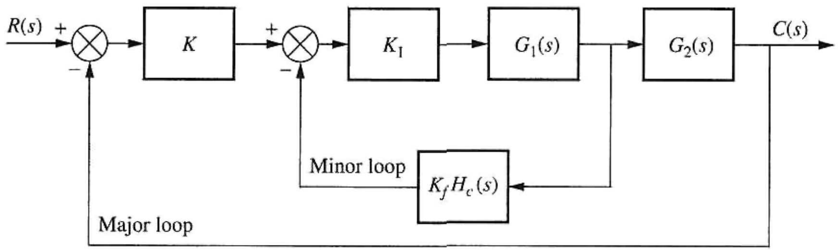


FIGURE 9.45 Generic control system with feedback compensation.

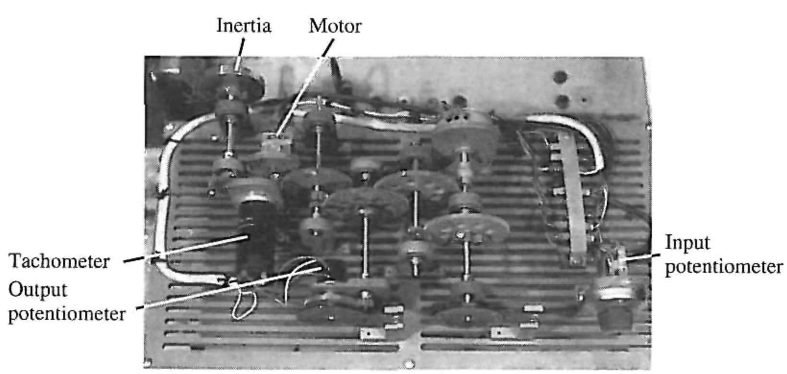


FIGURE 9.46 A position control system that uses a tachometer as a differentiator in the feedback path. Can you see the similarity between this system and the schematic on the front endpapers?

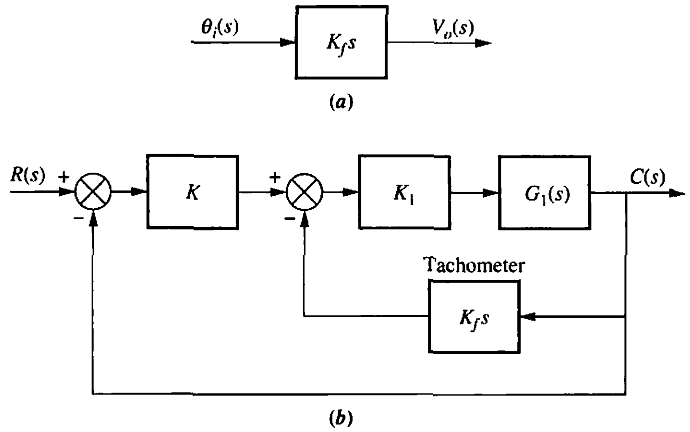


FIGURE 9.47 **a.** Transfer function of a tachometer; **b.** tachometer feed-back compensation

control system showing the gearing of the tachometer to the motor. You can see the input and output potentiometers as well as the motor and inertial load. The block diagram representation of a tachometer is shown in Figure 9.47(*a*), and its typical position within a control loop is shown in Figure 9.47(*b*).

While this section shows methods for designing systems using rate feedback, it also sets the stage for compensation techniques in Chapter 12, where not only rate but all states including position will be fed back for proper control system performance.

We now discuss design procedures. Typically, the design of feedback compensation consists of finding the gains, such as K, K_1, and K_f in Figure 9.45, after establishing a dynamic form for $H_c(s)$. There are two approaches. The first is similar to cascade compensation. Assume a typical feedback system, where $G(s)$ is the forward path and $H(s)$ is the feedback. Now consider that a root locus is plotted from $G(s)H(s)$. With cascade compensation we added poles and zeros to $G(s)$. With feedback compensation, poles and zeros are added via H(s).

With the second approach, we design a specified performance for the minor loop, shown in Figure 9.45, followed by a design of the major loop. Thus, the minor loop, such as ailerons on an aircraft, can be designed with its own performance specifications and operate within the major loop.

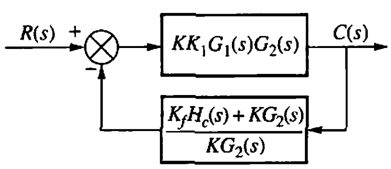


FIGURE 9.48 Equivalent block diagram of Figure 9.45

Approach 1

The first approach consists of reducing Figure 9.45 to Figure 9.48 by pushing K to the right past the summing junction, pushing $G_2(s)$ to the left past the pickoff point, and then adding the two feedback paths. Figure 9.48 shows that the loop gain, $G(s)H(s)$, is

$$G(s)H(s) = K_1G_1(s)[K_fH_c(s) + KG_2(s)] \tag{9.36}$$

Without feedback, $K_fH_c(s)$, the loop gain is

$$G(s)H(s) = KK_1G_1(s)G_2(s) \tag{9.37}$$

Thus, the effect of adding feedback is to replace the poles and zeros of $G_2(s)$ with the poles and zeros of $[K_fH_c(s) + KG_2(s)]$. Hence, this method is similar to cascade compensation in that we add new poles and zeros via $H(s)$ to reshape the root locus to go through the design point. However, one must remember that zeros of the equivalent feedback shown in Figure 9.48, $H(s) = [K_fH_c(s) + KG_2(s)]/KG_2(s)$, are not closed-loop zeros.

For example, if $G_2(s) = 1$ and the minor-loop feedback, $K_f H_c(s)$, is a rate sensor, $K_f H_c(s) = K_f s$, then from Eq. (9.36) the loop gain is

$$G(s)H(s) = K_f K_1 G_1(s)\left(s + \frac{K}{K_f}\right) \tag{9.38}$$

Thus, a zero at $-K/K_f$ is added to the existing open-loop poles and zeros. This zero reshapes the root locus to go through the desired design point. A final adjustment of the gain, K_1, yields the desired response. Again, you should verify that this zero is not a closed-loop zero. Let us look at a numerical example.

Example 9.7

Compensating Zero via Rate Feedback

PROBLEM: Given the system of Figure 9.49(*a*), design rate feedback compensation, as shown in Figure 9.49(*b*), to reduce the settling time by a factor of 4 while continuing to operate the system with 20% overshoot.

SOLUTION: First design a PD compensator. For the uncompensated system, search along the 20% overshoot line ($\zeta = 0.456$) and find that the dominant poles are at $-1.809 \pm j3.531$, as shown in Figure 9.50. The estimated specifications for the

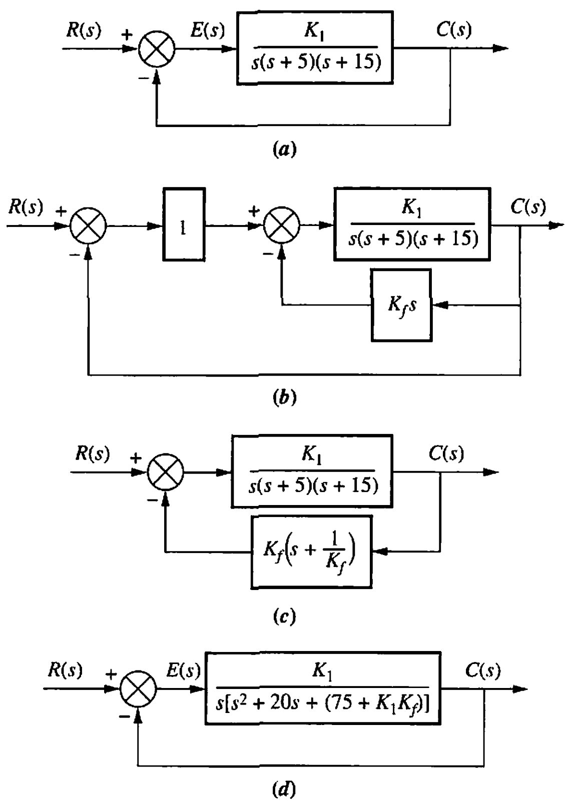


FIGURE 9.49 **a.** System for Example 9.7; **b.** system with rate feedback compensation; **c.** equivalent compensated system; **d.** equivalent compensated system showing unity feedback

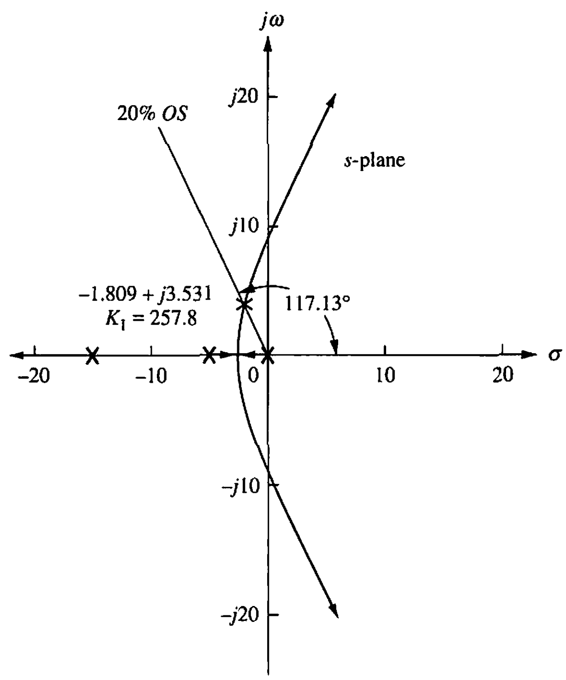


FIGURE 9.50 Root locus for uncompensated system of Example 9.7

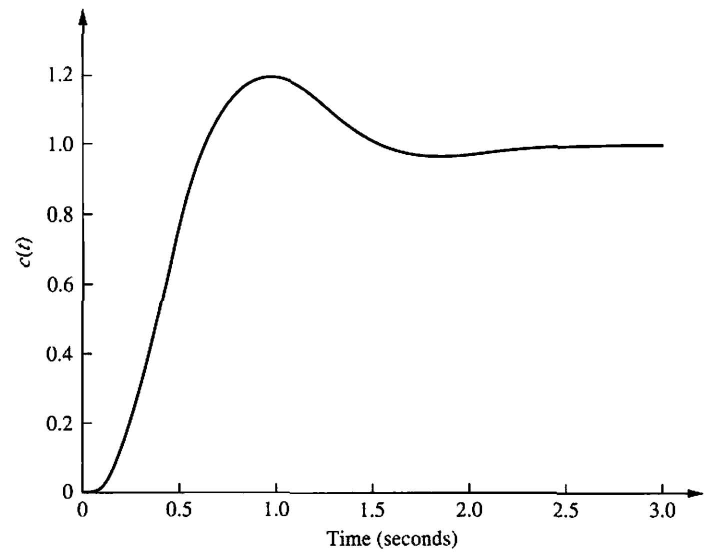


FIGURE 9.51 Step response for uncompensated system of Example 9.7

uncompensated system are shown in Table 9.8, and the step response is shown in Figure 9.51. The settling time is 2.21 seconds and must be reduced by a factor of 4 to 0.55 second.

Next determine the location of the dominant poles for the compensated system. To achieve a fourfold decrease in the settling time, the real part of the pole must be increased by a factor of 4. Thus, the compensated pole has a real part of $4(-1.809) = -7.236$. The imaginary part is then

$$\omega_d = -7.236 \tan 117.13^\circ = 14.12 \tag{9.39}$$

where 117.13° is the angle of the 20% overshoot line.

TABLE 9.8 Predicted characteristics of uncompensated and compensated systems of Example 9.7

	Uncompensated	**Compensated**
Plant and compensator	$\frac{K_1}{s(s+5)(s+15)}$	$\frac{K_1}{s(s+5)(s+15)}$
Feedback	1	$0.185(s+5.42)$
Dominant poles	$-1.809 \pm j3.531$	$-7.236 \pm j14.12$
K_1	257.8	1388
ζ	0.456	0.456
ω_n	3.97	15.87
$\%OS$	20	20
T_s	2.21	0.55
T_p	0.89	0.22
K_v	3.44	4.18
$e(\infty)$ (ramp)	0.29	0.24
Other poles	−16.4	−5.53
Zero	None	None
Comments	Second-order approx. OK	Simulate

Using the compensated dominant pole position of $-7.236 \pm j14.12$, we sum the angles from the uncompensated system's poles and obtain $-277.33°$. This angle requires a compensator zero contribution of $+97.33°$ to yield $180°$ at the design point. The geometry shown in Figure 9.52 leads to the calculation of the compensator's zero location. Hence,

$$\frac{14.12}{7.236 - z_c = \tan(180° - 97.33°)} \quad (9.40)$$

from which $z_c = 5.42$.

The root locus for the equivalent compensated system of Figure 9.49(*c*) is shown in Figure 9.53. The gain at the design point, which is K_1K_f from Figure 9.49(*c*), is found to be 256.7. Since K_f is the reciprocal of the compensator zero, $K_f = 0.185$. Thus, $K_1 = 1388$.

In order to evaluate the steady-state error characteristic, K_v is found from Figure 9.49(*d*) to be

$$K_v = \frac{K_1}{75 + K_1K_f} = 4.18 \quad (9.41)$$

Predicted performance for the compensated system is shown in Table 9.8. Notice that the higher-order pole is not far enough away from the dominant poles and thus cannot be neglected. Further, from Figure 9.49(*d*), we see that the closed-loop transfer function is

$$T(s) = \frac{G(s)}{1 + G(s)H(s)} = \frac{K_1}{s^3 + 20s^2 + (75 + K_1K_f)s + K_1} \quad (9.42)$$

Thus, as predicted, the open-loop zero is not a closed-loop zero, and there is no pole-zero cancellation. Hence, the design must be checked by simulation.

The results of the simulation are shown in Figure 9.54 and show an over-damped response with a settling time of 0.75 second, compared to the uncompensated system's settling time of approximately

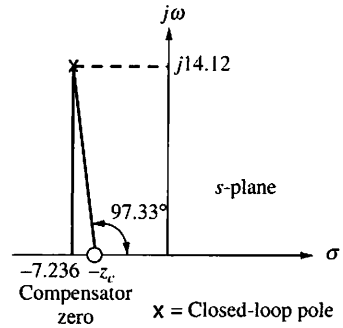


FIGURE 9.52 Finding the compensator zero in Example 9.7

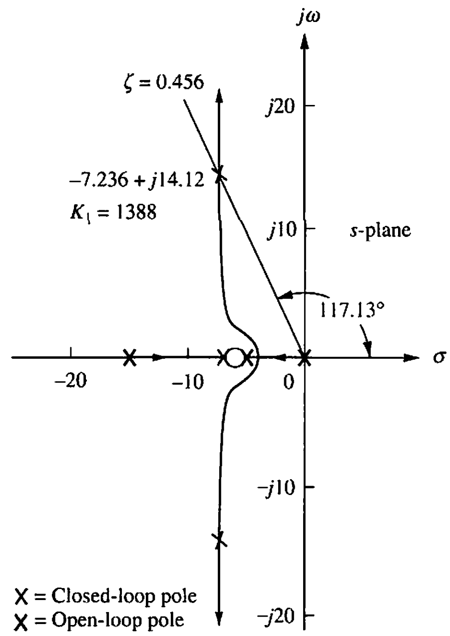


FIGURE 9.53 Root locus for the compensated system of Example 9.7

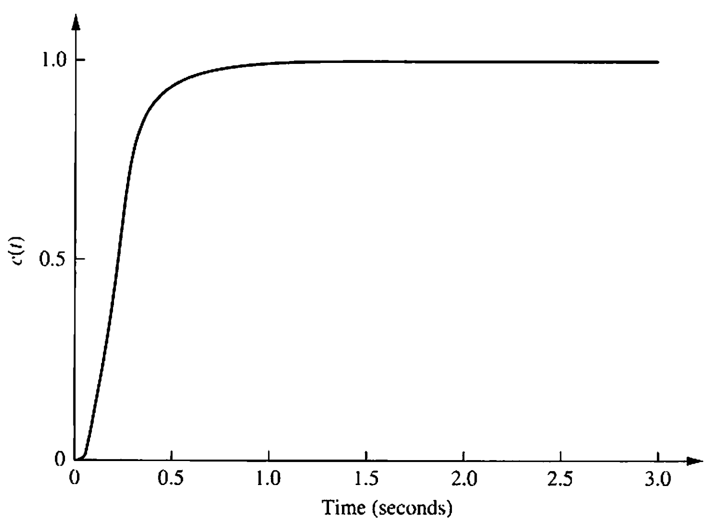


FIGURE 9.54 Step response for the compensated system of Example 9.7

2.2 seconds. Although not meeting the design requirements, the response still represents an improvement over the uncompensated system of Figure 9.51. Typically, less overshoot is acceptable. The system should be redesigned for further reduction in settling time.

You may want to do Problem 8 at the end of this chapter, where you can repeat this example using PD cascade compensation. You will see that the compensator zero for cascade compensation is a closed-loop zero, yielding the possibility of pole-zero cancellation. However, PD compensation is usually noisy and not always practical.

Approach 2

The second approach allows us to use feedback compensation to design a minor loop's transient response separately from the closed-loop system response. In the case of an aircraft, the minor loop may control the position of the aerosurfaces, while the entire closed-loop system may control the entire aircraft's pitch angle.

We will see that the minor loop of Figure 9.45 basically represents a forward-path transfer function whose poles can be adjusted with the minor-loop gain. These poles then become the open-loop poles for the entire control system. In other words, rather than reshaping the root locus with additional poles and zeros, as in cascade compensation, we can actually change the plant's poles through a gain adjustment. Finally, the closed-loop poles are set by the loop gain, as in cascade compensation.

Example 9.8

Minor-Loop Feedback Compensation

PROBLEM: For the system of Figure 9.55(*a*), design minor-loop feedback compensation, as shown in Figure 9.55(*b*), to yield a damping ratio of 0.8 for the minor loop and a damping ratio of 0.6 for the closed-loop system.

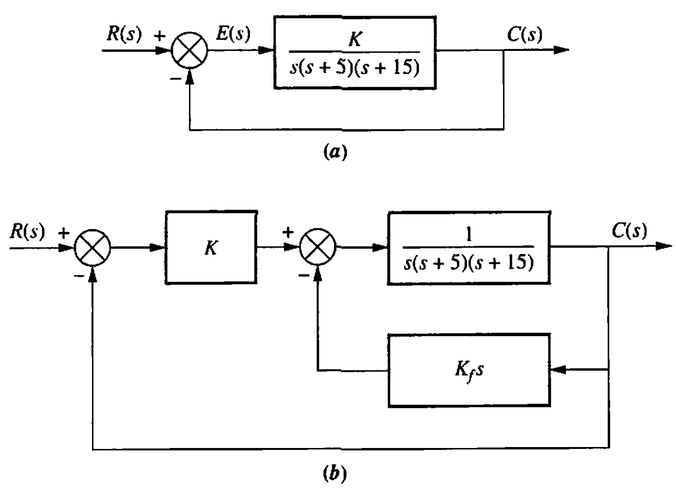


FIGURE 9.55 **a.** Uncompensated system and **b.** feedback-compensated system for Example 9.8

SOLUTION: The minor loop is defined as the loop containing the plant, $1/[s(s+5)(s+15)]$, and the feedback compensator, $K_f s$. The value of K_f will be adjusted to set the location of the minor-loop poles, and then K will be adjusted to yield the desired closed-loop response.

The transfer function of the minor loop, $G_{ML}(S)$, is

$$G_{ML}(s) = \frac{1}{s[s^2 + 20s + (75 + K_f)]} \tag{9.43}$$

The poles of $G_{ML}(s)$ can be found analytically or via the root locus. The root locus for the minor loop, where $K_f s/[s(s+5)(s+15)]$ is the open-loop transfer function, is shown in Figure 9.56. Since the zero at the origin comes from the feedback transfer function of the minor loop, this zero is not a zero of the closed-loop transfer function of the minor loop. Hence, the pole at the origin appears to remain stationary, and there is no pole-zero cancellation at the origin. Eq. (9.43) also shows this phenomenon. We see a stationary pole at the origin and two complex poles that change with gain. Notice that the compensator gain, K_f, varies the natural frequency, ω_n, of the minor-loop poles as seen from Eq. (9.43). Since the real parts of the complex poles are constant at $\zeta\omega_n = -10$, the damping ratio must also be varying to keep $2\zeta\omega_n = 20$, a constant. Drawing the $\zeta = 0.8$ line in Figure 9.56 yields the complex poles at $-10 \pm j7.5$. The gain, K_f, which equals 81.25, places the minor-loop poles in a position to meet the specifications. The poles just found, $-10 \pm j7.5$, as well as the pole at the origin (Eq. (9.43)), act as open-loop poles that generate a root locus for variations of the gain, K.

The final root locus for the system is shown in Figure 9.57. The $\zeta = 0.6$ damping ratio line is drawn and searched. The closed-loop complex poles are found to be $-4.535 \pm j6.046$, with a required gain of 624.3. A third pole is at -10.93.

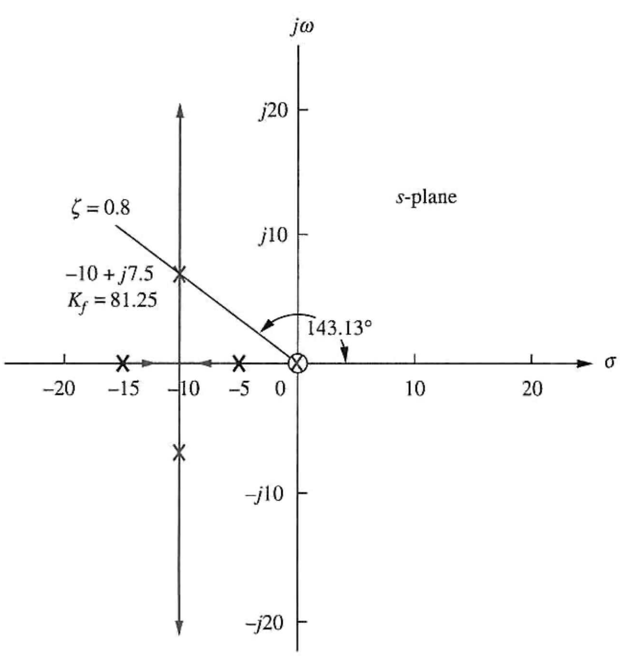
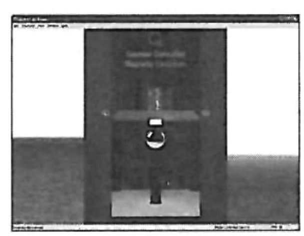


FIGURE 9.56 Root locus for minor loop of Example 9.8

Virtual Experiment 9.2 Improving Transient Response and Steady-State Error Using Rate Feedback and PD Control

Put theory into practice and design a compensator in LabVIEW that controls the ball position in the Quanser Magnetic Levitation system. Magnetic Levitation technology is used for modern transportation systems that suspend, such as the high speed Magnetic Levitation train.



Virtual experiments are found on WileyPLUS.

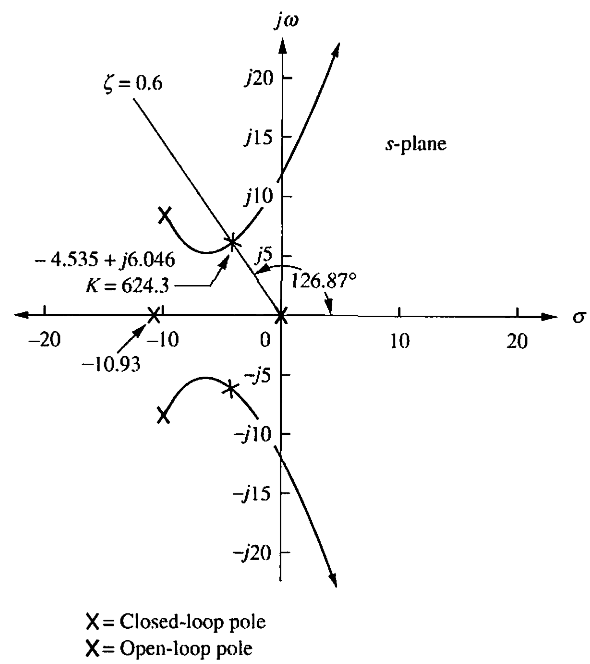


FIGURE 9.57 Root locus for closed-loop system of Example 9.8

The results are summarized in Table 9.9. We see that the compensated system, although having the same damping ratio as the uncompensated system, is much faster and also has a smaller steady-state error. The results, however, are predicted results and must be simulated to verify percent overshoot, settling time, and peak time, since the third pole is not far enough from the dominant poles. The step response is shown in Figure 9.58 and closely matches the predicted performance.

TABLE 9.9 Predicted characteristics of the uncompensated and compensated systems of Example 9.8

	Uncompensated	**Compensated**
Plant and compensator	$\frac{K_1}{s(s+5)(s+15)}$	$\frac{K}{s(s^2+20s+156.25)}$
Feedback	1	1
Dominant poles	$-1.997 \pm j2.662$	$-4.535 \pm j6.046$
K	177.3	624.3
ζ	0.6	0.6
ω_n	3.328	7.558
$\%OS$	9.48	9.48
T_s	2	0.882
T_p	1.18	0.52
K_v	2.364	3.996
$e(\infty)$(ramp)	0.423	0.25
Other poles	−16	−10.93
Zero	None	None
Comments	Second-order approx. OK	Simulate

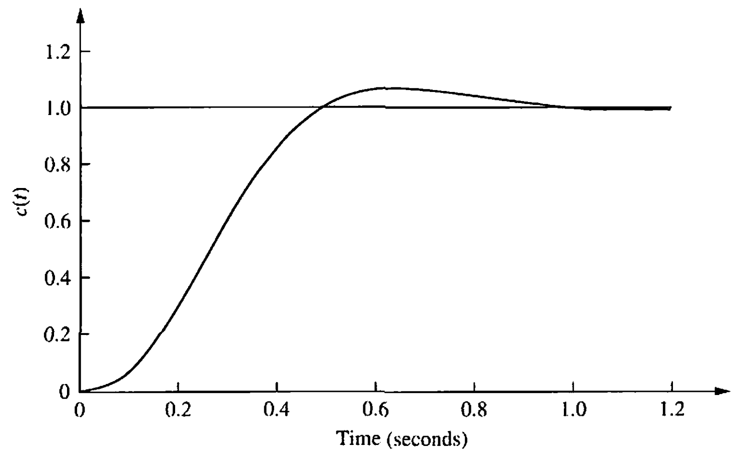


FIGURE 9.58 Step response simulation for Example 9.8

Skill-Assessment Exercise 9.4

PROBLEM: For the system of Figure 9.59, design minor-loop rate feedback compensation to yield a damping ratio of 0.7 for the minor loop's dominant poles and a damping ratio of 0.5 for the closed-loop system's dominant poles.

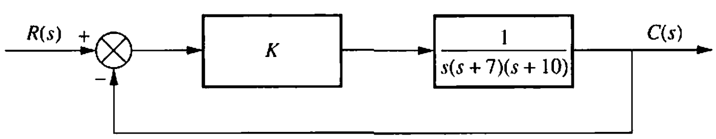


FIGURE 9.59 System for Skill-Assessment Exercise 9.4

ANSWER: The system is configured similar to Figure 9.55(*b*) with $K_f = 77.42$ and $K = 626.3$.

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

Our discussion of compensation methods is now complete. We studied both cascade and feedback compensation and compared and contrasted them. We are now ready to show how to physically realize the controllers and compensators we designed.

9.6 Physical Realization of Compensation

In this chapter, we derived compensation to improve transient response and steady-state error in feedback control systems. Transfer functions of compensators used in cascade with the plant or in the feedback path were derived. These compensators were defined by their pole-zero configurations. They were either active PI, PD, or PID controllers or passive lag, lead, or lag-lead compensators. In this section, we show how to implement the active controllers and the passive compensators.

Active-Circuit Realization

In Chapter 2, we derived

$$\boxed{\frac{V_o(s)}{V_i(s)} = -\frac{Z_2(s)}{Z_1(s)}} \tag{9.44}$$

as the transfer function of an inverting operational amplifier whose configuration is repeated here in Figure 9.60. By judicious choice of $Z_1(s)$ and $Z_2(s)$, this circuit can be used as a building block to implement the compensators and controllers, such as PID controllers, discussed in this chapter. Table 9.10 summarizes the realization of PI, PD, and PID controllers as well as lag, lead, and lag-lead compensators using operational amplifiers. You can verify the table by using the methods of Chapter 2 to find the impedances.

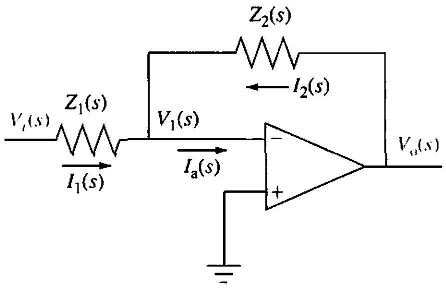


FIGURE 9.60 Operational amplifier configured for transfer function realization

TABLE 9.10 Active realization of controllers and compensators, using an operational amplifier

Function	$Z_1(s)$	$Z_2(s)$	$G_c(s) = -\frac{Z_2(s)}{Z_1(s)}$
Gain	R_1	R_2	$-\frac{R_2}{R_1}$
Integration	R	C	$-\frac{\frac{1}{RC}}{s}$
Differentiation	C	R	$-RCs$
PI controller	R_1	R_2 C	$-\frac{R_2}{R_1}\frac{\left(s+\frac{1}{R_2C}\right)}{s}$
PD controller	C, R_1	R_2	$-R_2C\left(s+\frac{1}{R_1C}\right)$
PID controller	C_1, R_1	R_2 C_2	$-\left[\left(\frac{R_2}{R_1}+\frac{C_1}{C_2}\right)+R_2C_1s+\frac{\frac{1}{R_1C_2}}{s}\right]$
Lag compensation	C_1, R_1	C_2, R_2	$-\frac{C_1}{C_2}\frac{\left(s+\frac{1}{R_1C_1}\right)}{\left(s+\frac{1}{R_2C_2}\right)}$ where $R_2C_2 > R_1C_1$
Lead compensation	C_1, R_1	C_2, R_2	$-\frac{C_1}{C_2}\frac{\left(s+\frac{1}{R_1C_1}\right)}{\left(s+\frac{1}{R_2C_2}\right)}$ where $R_1C_1 > R_2C_2$

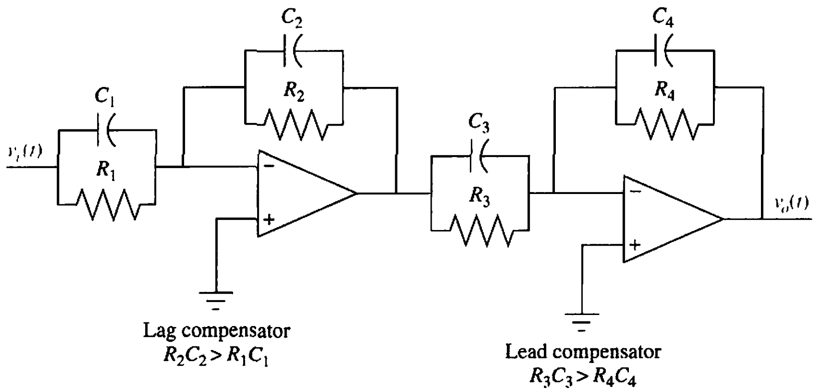


FIGURE 9.61 Lag-lead compensator implemented with operational amplifiers

Other compensators can be realized by cascading compensators shown in the table. For example, a lag-lead compensator can be formed by cascading the lag compensator with the lead compensator, as shown in Figure 9.61. As an example, let us implement one of the controllers we designed earlier in the chapter.

Example 9.9

Implementing a PID Controller

PROBLEM: Implement the PID controller of Example 9.5.

SOLUTION: The transfer function of the PID controller is

$$G_c(s) = \frac{(s+55.92)(s+0.5)}{s} \tag{9.45}$$

which can be put in the form

$$G_c(s) = s + 56.42 + \frac{27.96}{s} \tag{9.46}$$

Comparing the PID controller in Table 9.10 with Eq. (9.46), we obtain the following three relationships:

$$\frac{R_2}{R_1} + \frac{C_1}{C_2} = 56.42 \tag{9.47}$$

$$R_2 C_1 = 1 \tag{9.48}$$

and

$$\frac{1}{R_1 C_2} = 27.96 \tag{9.49}$$

Since there are four unknowns and three equations, we arbitrarily select a practical value for one of the elements. Selecting $C_2 = 0.1\ \mu F$, the remaining values are found to be $R_1 = 357.65$ kΩ, $R_2 = 178{,}891$ kΩ, and $C_1 = 5.59\ \mu F$.

The complete circuit is shown in Figure 9.62, where the circuit element values have been rounded off.

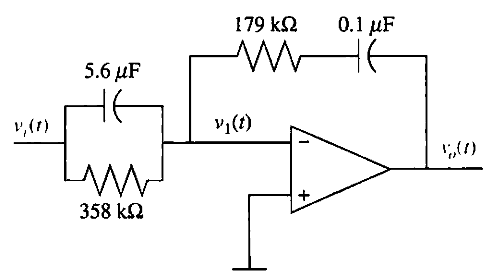


FIGURE 9.62 PID controller

Passive-Circuit Realization

Lag, lead, and lag-lead compensators can also be implemented with passive networks. Table 9.11 summarizes the networks and their transfer functions. The transfer functions can be derived with the methods of Chapter 2.

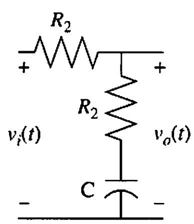
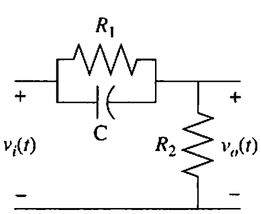
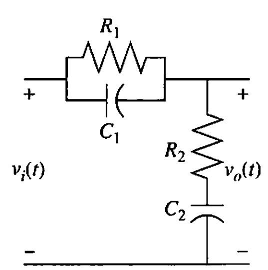
The lag-lead transfer function can be put in the following form:

$$G_c(s) = \frac{\left(s+\frac{1}{T_1}\right)\left(s+\frac{1}{T_2}\right)}{\left(s+\frac{1}{\alpha T_1}\right)\left(s+\frac{\alpha}{T_2}\right)} \tag{9.50}$$

where $\alpha < 1$. Thus, the terms with T_1 form the lead compensator, and the terms with T_2 form the lag compensator. Equation (9.50) shows a restriction inherent in using this passive realization. We see that the ratio of the lead compensator zero to the lead compensator pole must be the same as the ratio of the lag compensator pole to the lag compensator zero. In Chapter 11 we design a lag-lead compensator with this restriction.

A lag-lead compensator without this restriction can be realized with an active network as previously shown or with passive networks by cascading the lead and lag networks shown in Table 9.11. Remember, though, that the two networks must be isolated to ensure that one network does not load the other. If the networks load each other, the transfer function will not be the product of the individual transfer functions. A possible realization using the passive networks uses an operational amplifier to provide isolation. The circuit is shown in Figure 9.63. Example 9.10 demonstrates the design of a passive compensator.

TABLE 9.11 Passive realization of compensators

Function	Network	Transfer function, $\frac{V_o(s)}{V_i(s)}$
Lag compensation		$\frac{R_2}{R_1+R_2}\frac{s+\frac{1}{R_2C}}{s+\frac{1}{(R_1+R_2)C}}$
Lead compensation		$\frac{s+\frac{1}{R_1C}}{s+\frac{1}{R_1C}+\frac{1}{R_2C}}$
Lag-lead compensation		$\frac{\left(s+\frac{1}{R_1C_1}\right)\left(s+\frac{1}{R_2C_2}\right)}{s^2+\left(\frac{1}{R_1C_1}+\frac{1}{R_2C_2}+\frac{1}{R_2C_1}\right)s+\frac{1}{R_1R_2C_1C_2}}$

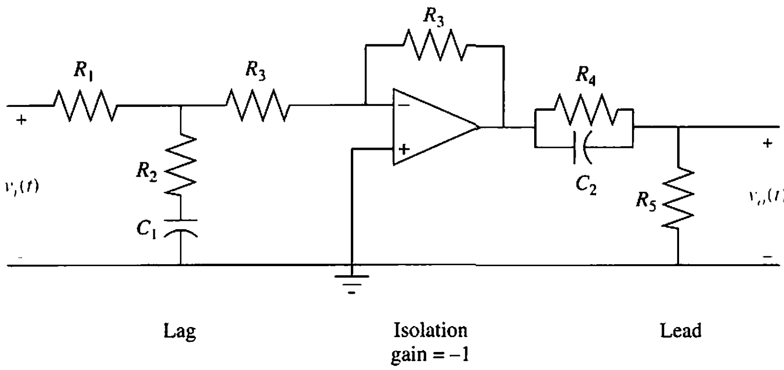


FIGURE 9.63 Lag-lead compensator implemented with cascaded lag and lead networks with isolation

Example 9.10

Realizing a Lead Compensator

PROBLEM: Realize the lead compensator designed in Example 9.4 (Compensator b).

SOLUTION: The transfer function of the lead compensator is

$$G_c(s) = \frac{s+4}{s+20.09} \tag{9.51}$$

Comparing the transfer function of a lead network shown in Table 9.11 with Eq. (9.51), we obtain the following two relationships:

$$\frac{1}{R_1C} = 4 \tag{9.52}$$

and

$$\frac{1}{R_1C} + \frac{1}{R_2C} = 20.09 \tag{9.53}$$

Hence, $R_1C = 0.25$, and $R_2C = 0.0622$. Since there are three network elements and two equations, we may select one of the element values arbitrarily. Letting $C = 1\,\mu\text{F}$, then $R_1 = 250\,\text{k}\Omega$ and $R_2 = 62.2\,\text{k}\Omega$.

Skill-Assessment Exercise 9.5

WileyPLUS
WPCS
Control Solutions

PROBLEM: Implement the compensators shown in **a.** and **b.** below. Choose a passive realization if possible.

a. $G_c(s) = \dfrac{(s+0.1)(s+5)}{s}$

b. $G_c(s) = \dfrac{(s+0.1)(s+2)}{(s+0.01)(s+20)}$

ANSWERS:

a. $G_c(s)$ is a PID controller and thus requires active realization. Use Figure 9.60 with the PID controller circuits shown in Table 9.10. One possible set of approximate component values is

$$C_1 = 10\,\mu\text{F}, \quad C_2 = 100\,\mu\text{F}, \quad R_1 = 20\,\text{k}\Omega, \quad R_2 = 100\,\text{k}\Omega$$

b. $G_c(s)$ is a lag-lead compensator that can be implemented with a passive network because the ratio of the lead pole to zero is the inverse of the ratio of the lag pole to zero. Use the lag-lead compensator circuit shown in Table 9.11. One possible set of approximate component values is

$$C_1 = 100\,\mu\text{F}, \quad C_2 = 900\,\mu\text{F}, \quad R_1 = 100\,\text{k}\Omega, \quad R_2 = 560\,\Omega$$

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

Case Studies

Antenna Control: Lag-Lead Compensation

Design

D

For the antenna azimuth position control system case study in Chapter 8, we obtained a 25% overshoot using a simple gain adjustment. Once this percent overshoot was obtained, the settling time was determined. If we try to improve the settling time by increasing the gain, the percent overshoot also increases. In this section, we continue with the antenna azimuth position control by designing a cascade compensator that yields 25% overshoot at a reduced settling time. Further, we effect an improvement in the steady-state error performance of the system.

PROBLEM: Given the antenna azimuth position control system shown on the front endpapers, Configuration 1, design cascade compensation to meet the following requirements: (1) 25% overshoot, (2) 2-second settling time, and (3) $K_v = 20$.

SOLUTION: For the case study in Chapter 8, a preamplifier gain of 64.21 yielded 25% overshoot, with the dominant, second-order poles at $-0.833 \pm j1.888$. The settling time is thus $4/\zeta\omega_n = 4/.833 = 4.8$ seconds. The open-loop function for the system as derived in the case study in Chapter 5 is $G(s) = 6.63K/[s(s+1.71)(s+100)]$. Hence $K_v = 6.63K/(1.71 \times 100) = 2.49$. Comparing these values to this example's problem statement, we want to improve the settling time by a factor of 2.4, and we want approximately an eightfold improvement in K_v.

Lead compensator design to improve transient response: First locate the dominant second-order pole. To obtain a settling time, T_s, of 2 seconds and a percent overshoot of 25%, the real part of the dominant second-order pole should be at $-4/T_s = -2$. Locating the pole on the 113.83° line ($\zeta = 0.404$, corresponding to 25% overshoot) yields an imaginary part of 4.529 (see Figure 9.64).

Second, assume a lead compensator zero and find the compensator pole. Assuming a compensator zero at -2, along with the uncompensated system's open-loop poles and zeros, use the root locus program in Appendix H.2 at www .wiley.com/college/nise to find that there is an angular contribution of $-120.14°$ at the design point of $-2 \pm j4.529$. Therefore, the compensator's pole must contribute $120.14° - 180° = -59.86°$ for the design point to be on the compensated system's root locus. The geometry is shown in Figure 9.64. To calculate the compensator pole, we use $4.529/(p_c - 2) = \tan 59.86°$ or $p_c = 4.63$.

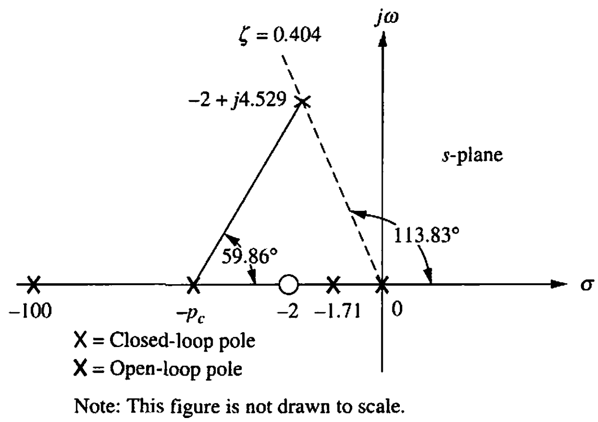


FIGURE 9.64 Locating compensator pole

Now determine the gain. Using the lead-compensated system's open-loop function,

$$G(s) = \frac{6.63K(s+2)}{s(s+1.71)(s+100)(s+4.63)} \tag{9.54}$$

and the design point $-2 + j4.529$ as the test point in the root locus program, the gain, $6.63K$, is found to be 2549.

Lag compensator design to improve the steady-state error: K_v for the lead-compensated system is found using Eq. (9.54). Hence,

$$K_v = \frac{2549(2)}{(1.71)(100)(4.63)} = 6.44 \tag{9.55}$$

Since we want $K_v = 20$, the amount of improvement required over the lead-compensated system is $20/6.44 = 3.1$. Choose $p_c = -0.01$ and calculate $z_c = 0.031$, which is 3.1 times larger.

Determine gain: The complete lag-lead-compensated open-loop function, $G_{\text{LLC}}(s)$, is

$$G_{\text{LLC}}(s) = \frac{6.63K(s+2)(s+0.031)}{s(s+.01)(s+1.71)(s+4.63)(s+100)} \tag{9.56}$$

Using the root locus program in Appendix H.2 at www.wiley.com/college/nise and the poles and zeros of Eq. (9.56), search along the 25% overshoot line (113.83°) for the design point. This point has moved slightly with the addition of the lag compensator to $-1.99 \pm j4.51$. The gain at this point equals 2533, which is $6.63K$. Solving for K yields $K = 382.1$.

Realization of the compensator: A realization of the lag-lead compensator is shown in Figure 9.63. From Table 9.11 the lag portion has the following transfer function:

$$G_{\text{lag}}(s) = \frac{R_2}{R_1+R_2}\frac{s+\dfrac{1}{R_2C}}{s+\dfrac{1}{(R_1+R_2)C}} = \frac{R_2}{R_1+R_2}\frac{(s+0.031)}{(s+0.01)} \tag{9.57}$$

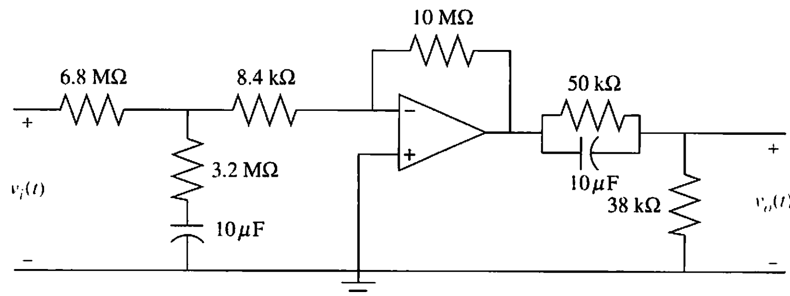


FIGURE 9.65 Realization of lag-lead compensator

Selecting $C = 10\,\mu$F, we find $R_2 = 3.2\,\text{M}\Omega$ and $R_1 = 6.8\,\text{M}\Omega$.

From Table 9.11 the lead compensator portion has the following transfer function:

$$G_{\text{lead}}(s) = \frac{s + \dfrac{1}{R_1C}}{s + \dfrac{1}{R_1C} + \dfrac{1}{R_2C}} = \frac{(s+2)}{(s+4.63)} \tag{9.58}$$

Selecting $C = 10\,\mu$F, we find $R_1 = 50\,\text{k}\Omega$ and $R_2 = 38\,\text{k}\Omega$.

The total loop gain required by the system is 2533. Hence,

$$6.63K\frac{R_2}{R_1 + R_2} = 2533 \tag{9.59}$$

where K is the gain of the preamplifier, and $R_2/(R_1 + R_2)$ is the gain of the lag portion. Using the values of R_1 and R_2 found during the realization of the lag portion, we find $K = 1194$.

The final circuit is shown in Figure 9.65, where the preamplifier is implemented with an operational amplifier whose feedback and input resistor ratio approximately equals 1194, the required preamplifier gain. The preamplifier isolates the lag and lead portions of the compensator.

Summary of the design results: Using Eq. (9.56) along with $K = 382.1$ yields the compensated value of K_v. Thus,

$$K_v = \lim_{s\to 0} sG_{\text{LLC}}(s) = \frac{2533(2)(0.031)}{(0.01)(1.71)(4.63)(100)} = 19.84 \tag{9.60}$$

which is an improvement over the gain-compensated system in the case study of Chapter 8, where $K_v = 2.49$. This value is calculated from the uncompensated $G(s)$ by letting $K = 64.21$, as found in the Case Study of Chapter 8.

Finally, checking the second-order approximation via simulation, we see in Figure 9.66 the actual transient response. Compare this to the gain-compensated system response of Figure 8.29 to see the improvement effected by cascade compensation over simple gain adjustment. The gain-compensated system yielded 25%, with a settling time of about 4 seconds. The lag-lead-compensated system yields 28% overshoot, with a settling time of about 2 seconds. If the results are not adequate for the application, the system should be redesigned to reduce the percent overshoot.

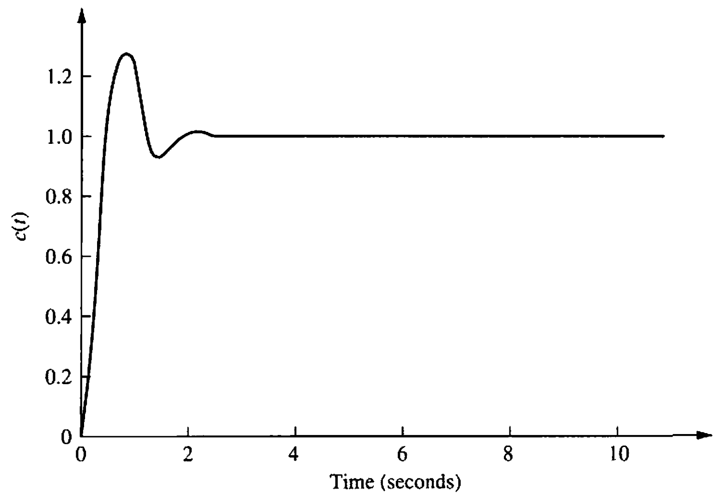


FIGURE 9.66 Step response of lag-lead-compensated antenna control

CHALLENGE: You are now given a problem to test your knowledge of this chapter's objectives. You are given the antenna azimuth position control system shown on the front endpapers, Configuration 2. In the challenge in Chapter 8, you were asked to design, via gain adjustment, an 8-second settling time.

a. For your solution to the challenge in Chapter 8, evaluate the percent overshoot and the value of the appropriate static error constant.
b. Design a cascade compensator to reduce the percent overshoot by a factor of 4 and the settling time by a factor of 2. Also, improve the appropriate static error constant by a factor of 2.
c. Repeat Part **b** using MATLAB.

MATLAB
ML

UFSS Vehicle: Lead and Feedback Compensation

As a final look at this case study, we redesign the pitch control loop for the UFSS vehicle. For the case study in Chapter 8, we saw that rate feedback improved the transient response. In this chapter's case study, we replace the rate feedback with a cascade compensator.

PROBLEM: Given the pitch control loop without rate feedback ($K_2 = 0$) for the UFSS vehicle shown on the back endpapers, design a compensator to yield 20% overshoot and a settling time of 4 seconds (*Johnson, 1980*).

SOLUTION: First determine the location of the dominant closed-loop poles. Using the required 20% overshoot and a 4-second settling time, a second-order approximation shows the dominant closed-loop poles are located at $-1 \pm j1.951$. From the uncompensated system analyzed in the Chapter 8 case study, the estimated settling time was 19.8 seconds for dominant closed-loop poles of $-0.202 \pm j0.394$. Hence, a lead compensator is required to speed up the system.

Arbitrarily assume a lead compensator zero at -1. Using the root locus program in Appendix H.2 at www.wiley.com/college/nise, we find that this compensator zero, along with the open-loop poles and zeros of the system, yields an angular contribution at the design point, $-1 + j1.951$, of $-178.92°$. The difference between this angle and 180°, or $-1.08°$, is the angular contribution required from the compensator pole.

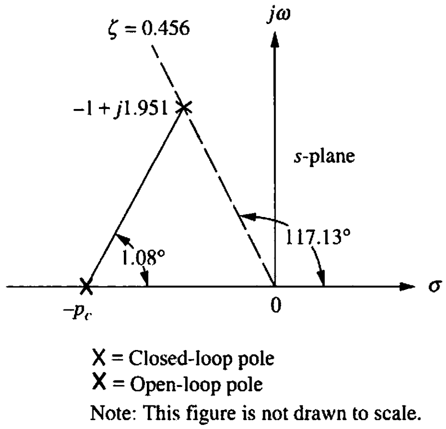


FIGURE 9.67 Locating compensator pole

Using the geometry shown in Figure 9.67, where $-p_c$ is the compensator pole location, we find that

$$\frac{1.951}{p_c - 1} = \tan 1.08^\circ \tag{9.61}$$

from which $p_c = 104.5$. The compensated open-loop transfer function is thus

$$G(s) = \frac{0.25K_1(s + 0.435)(s + 1)}{(s + 1.23)(s + 2)(s^2 + 0.226s + 0.0169)(s + 104.5)} \tag{9.62}$$

where the compensator is

$$G_c(s) = \frac{(s + 1)}{(s + 104.5)} \tag{9.63}$$

Using all poles and zeros shown in Eq. (9.62), the root locus program shows that a gain of 516.5 is required at the design point, $-1 \pm j1.951$. The root locus of the compensated system is shown in Figure 9.68.

A test of the second-order approximation shows three more closed-loop poles at -0.5, -0.9, and -104.5. Since the open-loop zeros are at -0.435 and -1, simulation is required to see if there is effectively closed-loop pole-zero cancellation with the closed-loop poles at -0.5 and -0.9, respectively. Further, the closed-loop pole at -104.5 is more than five times the real part of the dominant closed-loop pole, $-1 \pm j1.951$, and its effect on the transient response is therefore negligible.

The step response of the closed-loop system is shown in Figure 9.69, where we see a 26% overshoot and a settling time of about 4.5 seconds. Comparing this

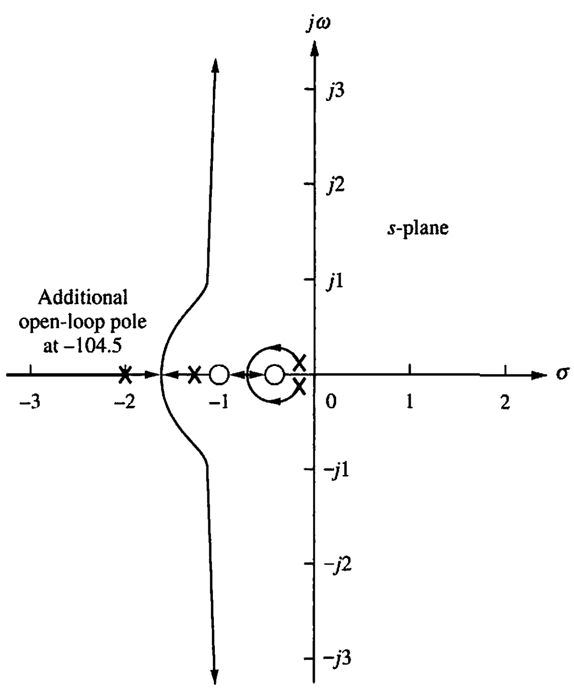


FIGURE 9.68 Root locus for lead-compensated system

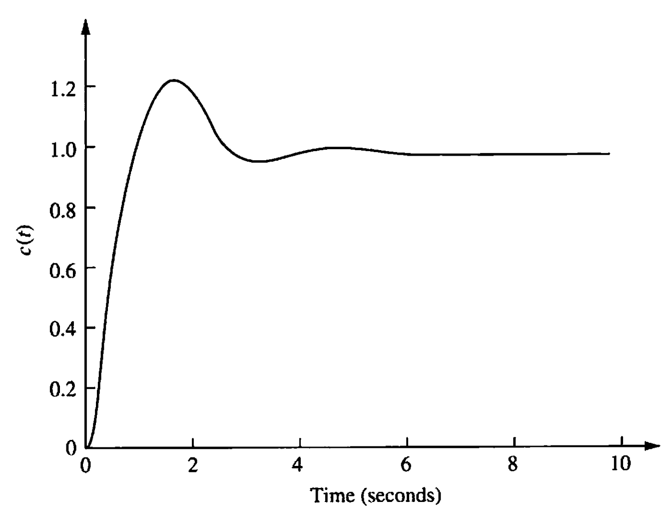


FIGURE 9.69 Step response of lead-compensated UFSS vehicle

response with Figure 8.31, the response of the uncompensated system, we see considerable improvement in the settling time and steady-state error. However, the transient response performance does not meet the design requirements. Thus, a redesign of the system to reduce the percent overshoot is suggested if required by the application.

CHALLENGE: You are now given a problem to test your knowledge of this chapter's objectives. The heading control system for the UFSS vehicle is shown on the back endpapers. The minor loop contains the rudder and vehicle dynamics, and the major loop relates output and input heading (*Johnson, 1980*).

a. Find the values of K_1 and K_2 so that the minor-loop dominant poles have a damping ratio of 0.6 and the major-loop dominant poles have a damping ratio of 0.5.

b. `Repeat, using MATLAB.` MATLAB ML

Summary

In this chapter, we learned how to design a system to meet transient and steady-state specifications. These design techniques overcame limitations in the design methodology covered in Chapter 8, whereby a transient response could be created only if the poles generating that response were on the root locus. Subsequent gain adjustment yielded the desired response. Since this value of gain dictated the amount of steady-state error in the response, a trade-off was required between the desired transient response and the desired steady-state error.

Cascade or *feedback compensation* is used to overcome the disadvantages of gain adjustment as a compensating technique. In this chapter, we saw that the transient response and the steady-state error can be designed separately from each other. No longer was a trade-off between these two specifications required. Further, we were able to design for a transient response that was not represented on the original root locus.

The transient response design technique covered in this chapter is based upon reshaping the root locus to go through a desired transient response point, followed by a gain adjustment. Typically, the resulting gain is much higher than the original if the compensated system response is faster than the uncompensated response.

The root locus is reshaped by adding additional poles and zeros via a cascade or feedback compensator. The additional poles and zeros must be checked to see that any second-order approximations used in the design are valid. All poles besides the dominant second-order pair must yield a response that is much faster than the designed response. Thus, nondominant poles must be at least five times as far from the imaginary axis as the dominant pair. Further, any zeros of the system must be close to a nondominant pole for pole-zero cancellation, or far from the dominant pole pair. The resulting system can then be approximated by two dominant poles.

The steady-state response design technique is based upon placing a pole on or near the origin in order to increase or nearly increase the system type, and then placing a zero near this pole so that the effect upon the transient response is negligible. However, final reduction of steady-state error occurs with a long-time constant. The same arguments about other poles yielding fast responses and about zeros being cancelled in order to validate a second-order approximation also hold true for this technique. If the second-order approximations cannot be justified, then a simulation is required to make sure the design is within tolerance.

Steady-state design compensators are implemented via *PI controllers* or *lag compensators.* PI controllers add a pole at the origin, thereby increasing the system type. Lag compensators, usually implemented with passive networks, place the pole off the origin but near it. Both methods add a zero very close to the pole in order not to affect the transient response.

The transient response design compensators are implemented through *PD controllers* or *lead compensators.* PD controllers add a zero to compensate the transient response; they are considered *ideal.* Lead compensators, on the other hand, are not ideal since they add a pole along with the zero. Lead compensators are usually passive networks.

We can correct both transient response and steady-state error with a *PID* or *lag-lead compensator.* Both of these are simply combinations of the previously described compensators. Table 9.7 summarized the types of cascade compensators.

Feedback compensation can also be used to improve the transient response. Here the compensator is placed in the feedback path. The feedback gain is used to change the compensator zero or the system's open-loop poles, giving the designer a wide choice of various root loci. The system gain is then varied to move along the selected root locus to the design point. An advantage of feedback compensation is the ability to design a fast response into a subsystem independently of the system's total response.

In the next chapter, we look at another method of design, frequency response, which is an alternate method to the root locus.

Review Questions

1. Briefly distinguish between the design techniques in Chapter 8 and Chapter 9.
2. Name two major advantages of the design techniques of Chapter 9 over the design techniques of Chapter 8.
3. What kind of compensation improves the steady-state error?
4. What kind of compensation improves transient response?
5. What kind of compensation improves both steady-state error and transient response?
6. Cascade compensation to improve the steady-state error is based upon what pole-zero placement of the compensator? Also, state the reasons for this placement.
7. Cascade compensation to improve the transient response is based upon what pole-zero placement of the compensator? Also, state the reasons for this placement.
8. What difference on the s-plane is noted between using a PD controller or using a lead network to improve the transient response?
9. In order to speed up a system without changing the percent overshoot, where must the compensated system's poles on the s-plane be located in comparison to the uncompensated system's poles?

10. Why is there more improvement in steady-state error if a PI controller is used instead of a lag network?
11. When compensating for steady-state error, what effect is sometimes noted in the transient response?
12. A lag compensator with the zero 25 times as far from the imaginary axis as the compensator pole will yield approximately how much improvement in steady-state error?
13. If the zero of a feedback compensator is at −3 and a closed-loop system pole is at −3.001, can you say there will be pole-zero cancellation? Why?
14. Name two advantages of feedback compensation.

Problems

1. Design a PI controller to drive the step response error to zero for the unity feedback system shown in Figure P9.1, where WileyPLUS WPCS Control Solutions

$$G(s) = \frac{K}{(s+1)(s+3)(s+10)}$$

The system operates with a damping ratio of 0.5. Compare the specifications of the uncompensated and compensated systems. [Section: 9.2]

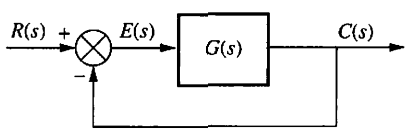


FIGURE P9.1

2. Consider the unity feedback system shown in Figure P9.1, where

$$G(s) = \frac{K}{s(s+3)(s+6)}$$

   a. Design a PI controller to drive the ramp response error to zero for any K that yields stability. [Section: 9.2]
   b. Use MATLAB to simulate your design for $K = 1$. Show both the input ramp and the output response on the same plot. MATLAB ML

3. The unity feedback system shown in Figure P9.1 with

$$G(s) = \frac{K}{(s+2)(s+3)(s+7)}$$

is operating with 10% overshoot. [Section: 9.2]

   a. What is the value of the appropriate static error constant?
   b. Find the transfer function of a lag network so that the appropriate static error constant equals 4 without appreciably changing the dominant poles of the uncompensated system.
   c. Use MATLAB or any other computer program to simulate the system to see the effect of your compensator. MATLAB ML

4. Repeat Problem 3 for $G(s) = \dfrac{K}{s(s+3)(s+7)}$. [Section: 9.2]

5. Consider the unity feedback system shown in Figure P9.1 with

$$G(s) = \frac{K}{(s+3)(s+5)(s+7)}$$

   a. Design a compensator that will yield $K_p = 20$ without appreciably changing the dominant pole location that yields a 10% overshoot for the uncompensated system. [Section: 9.2]
   b. Use MATLAB or any other computer program to simulate the uncompensated and compensated systems. MATLAB ML
   c. Use MATLAB or any other computer program to determine how much time it takes the slow response of the lag compensator to bring the output to within 2% of its final compensated value. MATLAB ML

6. The unity feedback system shown Figure P9.1 with WileyPLUS WPCS Control Solutions

$$G(s) = \frac{K(s+6)}{(s+2)(s+3)(s+5)}$$

is operating with a dominant-pole damping ratio of 0.707. Design a PD controller so that the settling time is reduced by a factor of 2. Compare the transient and steady-state performance of the uncompensated and compensated systems. Describe any problems with your design. [*Section:* 9.3]

7. Redo Problem 6 using MATLAB in the following way: (MATLAB ML)
   a. MATLAB will generate the root locus for the uncompensated system along with the 0.707 damping ratio line. You will interactively select the operating point. MATLAB will then inform you of the coordinates of the operating point, the gain at the operating point, as well as the estimated %OS, T_s, T_p, ζ, ω_n, and K_p represented by a second-order approximation at the operating point.
   b. MATLAB will display the step response of the uncompensated system.
   c. Without further input, MATLAB will calculate the compensated design point and will then ask you to input a value for the PD compensator zero from the keyboard. MATLAB will respond with a plot of the root locus showing the compensated design point. MATLAB will then allow you to keep changing the PD compensator value from the keyboard until a root locus is plotted that goes through the design point.
   d. For the compensated system, MATLAB will inform you of the coordinates of the operating point, the gain at the operating point, as well as the estimated %OS, T_s, T_p, ζ, ω_n, and K_p represented by a second-order approximation at the operating point.
   e. MATLAB will then display the step response of the compensated system.

8. Design a PD controller for the system shown in Figure P9.2 *to reduce the settling time by a factor of* 4 while continuing to operate the system with 20% overshoot. Compare your performance to that obtained in Example 9.7.

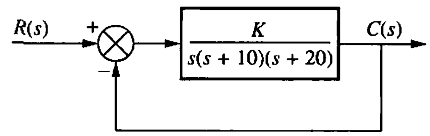


FIGURE P9.2

9. Consider the unity feedback system shown in Figure P9.1 with [Section: 9.3]

$$G(s) = \frac{K}{(s+4)^3}$$

   a. Find the location of the dominant poles to yield a 1.6 second settling time and an overshoot of 25%.
   b. If a compensator with a zero at −1 is used to achieve the conditions of Part **a**, what must the angular contribution of the compensator pole be?
   c. Find the location of the compensator pole.
   d. Find the gain required to meet the requirements stated in Part **a**.
   e. Find the location of other closed-loop poles for the compensated system.
   f. Discuss the validity of your second-order approximation.
   g. Use MATLAB or any other computer program to simulate the compensated system to check your design. (MATLAB ML)

10. The unity feedback system shown in Figure P9.1 with (WileyPLUS WPCS Control Solutions)

$$G(s) = \frac{K}{s^2}$$

is to be designed for a settling time of 1.667 seconds and a 16.3% overshoot. If the compensator zero is placed at −1, do the following: [Section: 9.3]
   a. Find the coordinates of the dominant poles.
   b. Find the compensator pole.
   c. Find the system gain.
   d. Find the location of all nondominant poles.
   e. Estimate the accuracy of your second-order approximation.
   f. Evaluate the steady-state error characteristics.
   g. Use MATLAB or any other computer *program to simulate the system* and evaluate the actual transient response characteristics for a step input. (MATLAB ML)

11. Given the unity feedback system of Figure P9.1, with

$$G(s) = \frac{K(s+6)}{(s+3)(s+4)(s+7)(s+9)}$$

do the following: [Section: 9.3]

a. Sketch the root locus.

b. Find the coordinates of the dominant poles for which $\zeta = 0.8$.

c. Find the gain for which $\zeta = 0.8$.

d. If the system is to be cascade-compensated so that $T_s = 1$ second and $\zeta = 0.8$, find the compensator pole if the compensator zero is at -4.5.

e. Discuss the validity of your second-order approximation.

f. Use MATLAB or any other computer program to simulate the compensated and uncompensated systems and compare the results to those expected. MATLAB ML

12. Redo Problem 11 using MATLAB in the following way: MATLAB ML

a. MATLAB will generate the root locus for the uncompensated system along with the 0.8 damping ratio line. You will interactively select the operating point. MATLAB will then inform you of the coordinates of the operating point, the gain at the operating point, as well as the estimated %OS, T_s, T_p, ζ, ω_n, and K_p represented by a second-order approximation at the operating point.

b. MATLAB will display the step response of the uncompensated system.

c. Without further input, MATLAB will calculate the compensated design point and will then ask you to input a value for the lead compensator pole from the keyboard. MATLAB will respond with a plot of the root locus showing the compensated design point. MATLAB will then allow you to keep changing the lead compensator pole value from the keyboard until a root locus is plotted that goes through the design point.

d. For the compensated system, MATLAB will inform you of the coordinates of the operating point, the gain at the operating point, as well as the estimated %OS, T_s, T_p, ζ, ω_n, and K_p represented by a second-order approximation at the operating point.

e. MATLAB will then display the step response of the compensated system.

f. Change the compensator's zero location a few times and collect data on the compensated system to see if any other choices of compensator zero yield advantages over the original design.

13. Consider the unity feedback system of Figure P9.1 with

$$G(s) = \frac{K}{s(s+20)(s+40)}$$

The system is operating at 20% overshoot. Design a compensator to decrease the settling time by a factor of 2 without affecting the percent overshoot and do the following: [Section: 9.3]

a. Evaluate the uncompensated system's dominant poles, gain, and settling time.

b. Evaluate the compensated system's dominant poles and settling time.

c. Evaluate the compensator's pole and zero. Find the required gain.

d. Use MATLAB or any other computer program to simulate the compensated and uncompensated systems' step response. MATLAB ML

14. The unity feedback system shown in Figure P9.1 with

$$G(s) = \frac{K}{(s+15)(s^2+6s+13)}$$

is operating with 30% overshoot. [Section: 9.3]

a. Find the transfer function of a cascade compensator, the system gain, and the dominant pole location that will cut the settling time in half if the compensator zero is at -7.

b. Find other poles and zeros and discuss your second-order approximation.

c. Use MATLAB or any other computer program to simulate both the uncompensated and compensated systems to see the effect of your compensator. MATLAB ML

15. For the unity feedback system of Figure P9.1 with

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

do the following: [Section: 9.3]

a. Find the settling time for the system if it is operating with 15% overshoot.

b. Find the zero of a compensator and the gain, K, so that the settling time is 7 seconds. Assume that the pole of the compensator is located at -15.

c. Use MATLAB or any other computer program to simulate the system's step response to test the compensator. MATLAB ML

16. A unity feedback control system has the following forward transfer function: [Section: 9.3] WileyPLUS WPCS Control Solutions

$$G(s) = \frac{K}{s^2(s+4)(s+12)}$$

a. Design a lead compensator to yield a closed-loop step response with 20.5% overshoot and a settling time of 3 seconds. Be sure to specify the value of K.

b. Is your second-order approximation valid?

c. Use MATLAB or any other computer program to simulate and compare the transient response of the compensated system to the predicted transient response. MATLAB ML

17. For the unity feedback system of Figure P9.1, with

$$G(s) = \frac{K}{(s^2+20s+101)(s+20)}$$

the damping ratio for the dominant poles is to be 0.4, and the settling time is to be 0.5 second. [Section: 9.3]

a. Find the coordinates of the dominant poles.

b. Find the location of the compensator zero if the compensator pole is at -15.

c. Find the required system gain.

d. Compare the performance of the uncompensated and compensated systems.

e. Use MATLAB or any other computer program to simulate the system to check your design. Redesign if necessary. MATLAB ML

18. Consider the unity feedback system of Figure P9.1, with

$$G(s) = \frac{K}{(s+3)(s+5)}$$

a. Show that the system cannot operate with a settling time of 2/3 second and a percent overshoot of 1.5 % with a simple gain adjustment.

b. Design a lead compensator so that the system meets the transient response characteristics of Part **a.** Specify the compensator's pole, zero, and the required gain.

19. Given the unity feedback system of Figure P9.1 with

$$G(s) = \frac{K}{(s+2)(s+4)(s+6)(s+8)}$$

Find the transfer function of a lag-lead compensator that will yield a settling time 0.5 second shorter than that of the uncompensated system, with a damping ratio of 0.5, and improve the steady-state error by a factor of 30. The compensator zero is at -5. Also, find the compensated system's gain. Justify any second-order approximations or verify the design through simulation. [Section: 9.4]

20. Redo Problem 19 using a lag-lead compensator and MATLAB in the following way: MATLAB ML

a. MATLAB will generate the root locus for the uncompensated system along with the 0.5 damping-ratio line. You will interactively select the operating point. MATLAB will then proceed to inform you of the coordinates of the operating point, the gain at the operating point, as well as the estimated %OS, T_s, T_p, ζ, ω_n, and K_p represented by a second-order approximation at the operating point.

b. MATLAB will display the step response of the uncompensated system.

c. Without further input, MATLAB will calculate the compensated design point and will then ask you to input a value for the lead compensator pole from the keyboard. MATLAB will respond with a plot of the root locus showing the compensated design point. MATLAB will then allow you to keep changing the lead compensator pole value from the keyboard until a root locus is plotted that goes through the design point.

d. For the compensated system, MATLAB will inform you of the coordinates of the operating point, the gain at the operating point, as well as the estimated $\%OS$, T_s, T_p, ζ, ω_n, and K_p represented by a second-order approximation at the operating point.

e. MATLAB will then display the step response of the compensated system.

f. Change the compensator's zero location a few times and collect data on the compensated system to see if any other choices of the compensator zero yield advantages over the original design.

g. Using the steady-state error of the uncompensated system, add a lag compensator to yield an improvement of 30 times over the uncompensated system's steady-state error, with minimal effect on the designed transient response. Have MATLAB plot the step response. Try several values for the lag compensator's pole and see the effect on the step response.

21. Given the uncompensated unity feedback system of Figure P9.1, with

$$G(s) = \frac{K}{s(s+1)(s+3)}$$

do the following: [Section: 9.4]

a. Design a compensator to yield the following specifications: settling time = 2.86 seconds; percent overshoot = 4.32%; the steady-state error is to be improved by a factor of 2 over the uncompensated system.

b. Compare the transient and steady-state error specifications of the uncompensated and compensated systems.

c. Compare the gains of the uncompensated and compensated systems.

d. Discuss the validity of your second-order approximation.

e. Use MATLAB or any other computer program to simulate the uncompensated and compensated systems and verify the specifications. MATLAB ML

22. For the unity feedback system given in Figure P9.1 with WileyPLUS WPCS Control Solutions

$$G(s) = \frac{K}{s(s+5)(s+11)}$$

do the following: [Section: 9.4]

a. Find the gain, K, for the uncompensated system to operate with 30% overshoot.

b. Find the peak time and K_v for the uncompensated system.

c. Design a lag-lead compensator to decrease the peak time by a factor of 2, decrease the percent overshoot by a factor of 2, and improve the steady-state error by a factor of 30. Specify all poles, zeros, and gains.

23. The unity feedback system shown in Figure P9.1 with

$$G(s) = \frac{K}{(s^2+4s+8)(s+10)}$$

is to be designed to meet the following specifications:

Overshoot: Less than 25%

Settling time: Less than 1 second

$K_p = 10$

Do the following: [Section: 9.4]

a. Evaluate the performance of the uncompensated system operating at 10% overshoot.

b. Design a passive compensator to meet the desired specifications.

c. Use MATLAB to simulate the compensated system. Compare the response with the desired specifications. MATLAB ML

24. Consider the unity feedback system in Figure P9.1, with

$$G(s) = \frac{K}{(s+2)(s+4)}$$

The system is operated with 4.32% overshoot. In order to improve the steady-state error, K_p is to be increased by at least a factor of 5. A lag compensator of the form

$$G_c(s) = \frac{(s+0.5)}{(s+0.1)}$$

is to be used. [Section: 9.4]

a. Find the gain required for both the compensated and the uncompensated systems.

b. Find the value of K_p for both the compensated and the uncompensated systems.

c. Estimate the percent overshoot and settling time for both the compensated and the uncompensated systems.

d. Discuss the validity of the second-order approximation used for your results in Part **c**.

e. Use MATLAB or any other computer program to simulate the step response for the uncompensated and compensated systems. What do you notice about the compensated system's response?

f. Design a lead compensator that will correct the objection you notice in Part **e**.

25. For the unity feedback system in Figure P9.1, with

$$G(s) = \frac{K}{(s+1)(s+4)}$$

design a PID controller that will yield a peak time of 1.047 seconds and a damping ratio of 0.8, with zero error for a step input. [Section: 9.4]

26. For the unity feedback system in Figure P9.1, with

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

do the following:

a. Design a controller that will yield no more than 25% overshoot and no more than a 2-second settling time for a step input and zero steady-state error for step and ramp inputs.

b. Use MATLAB and verify your design.

27. Redo Problem 26 using MATLAB in the following way:

a. MATLAB will ask for the desired percent over shoot, settling time, and PI compensator zero.

b. MATLAB will design the PD controller's zero.

c. MATLAB will display the root locus of the PID-compensated system with the desired percent overshoot line.

d. The user will interactively select the intersection of the root locus and the desired percent over shoot line.

e. MATLAB will display the gain and transient response characteristics of the PID-compensated system.

f. MATLAB will display the step response of the PID-compensated system.

g. MATLAB will display the ramp response of the PID-compensated system.

28. If the system of Figure P9.3 operates with a damping ratio of 0.517 for the dominant second-order poles, find the location of all closed-loop poles and zeros.

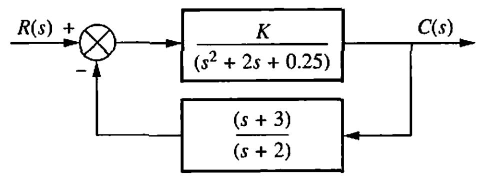


FIGURE P9.3

29. For the unity feedback system in Figure P9.1, with

$$G(s) = \frac{K}{s(s+2)(s+4)(s+6)}$$

do the following: [Section: 9.5]

a. Design rate feedback to yield a step response with no more than 15% overshoot and no more than 3 seconds settling time. Use Approach 1.

b. Use MATLAB and simulate your compensated system.

30. Given the system of Figure P9.4: [Section: 9.5]

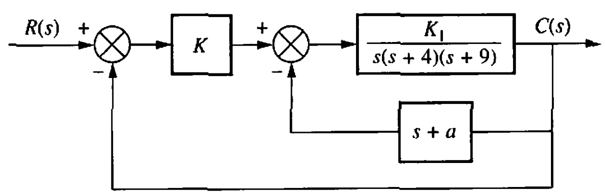


FIGURE P9.4

a. Design the value of K_1, as well as a in the feedback path of the minor loop, to yield a settling time of 1 second with 5% overshoot for the step response.

b. Design the value of K to yield a major-loop response with 10% overshoot for a step input.

c. Use MATLAB or any other computer program to simulate the step response to the entire closed-loop system. MATLAB ML

d. Add a PI compensator to reduce the major-loop steady-state error to zero and simulate the step response using MATLAB or any other computer program. MATLAB ML

31. Identify and realize the following controllers with operational amplifiers. [Section: 9.6]

a. $\dfrac{s+0.01}{s}$

b. $s+2$

32. Identify and realize the following compensators with passive networks. [Section: 9.6] WileyPLUS WPCS Control Solutions

a. $\dfrac{s+0.1}{s+0.01}$

b. $\dfrac{s+2}{s+5}$

c. $\left(\dfrac{s+0.1}{s+0.01}\right)\left(\dfrac{s+1}{s+10}\right)$

33. Repeat Problem 32 using operational amplifiers. [Section: 9.6]

DESIGN PROBLEMS

34. The room temperature of an 11 m^2 room is to be controlled by varying the power of an indoor radiator. For this specific room the open-loop transfer function from radiator power, $\dot{Q}(s)$, to temperature, $T(s)$, is (*Thomas, 2005*)

$$G(s)=\frac{T(s)}{\dot{Q}(s)}=\frac{(1\times10^{-6})s^2+(1.314\times10^{-9})s+(2.66\times10^{-13})}{s^3+0.00163s^2+(5.272\times10^{-7})s+(3.538\times10^{-11})}$$

The system is assumed to be in the closed-loop configuration shown in Figure P9.1.

a. For a unit step input, calculate the steady-state error of the system.

b. Try using the procedure of Section 9.2 to design a PI controller to obtain zero steady-state error for step inputs without appreciably changing the transient response. Then explain why it is not possible to do so.

c. Design a PI controller of the form $G_c(s)=\dfrac{K(s+z)}{s}$ that will reduce the step-response error to zero while not changing significantly the transient response. (Hint: Place the zero of the compensator in a position where the closed-loop poles of the uncompensated root locus will not be affected significantly.)

d. Use Simulink to simulate the systems of Parts **b** and **c** and to verify the correctness of your design in Part **c**. Simulink SL

35. Figure P9.5 shows a two-tank system. The liquid inflow to the upper tank can be controlled using a valve and is represented by F_0. The upper tank's outflow equals the lower tank's inflow and is represented by F_1. The outflow of the lower tank is F_2. The objective of the design is to control the liquid level, $y(t)$, in the lower tank. The open-loop transmission for this system is $\dfrac{Y(s)}{F_o(s)}=\dfrac{a_2a_3}{s^2+(a_1+a_4)s+a_1a_4}$ (*Romagnoli, 2006*). The system will be controlled in a loop analogous to that of Figure P9.1, where the lower liquid level will be measured and compared to a set point. The resulting error will be fed to a controller, which in turn will open or close the valve feeding the upper tank.

a. Assuming $a_1=0.04$, $a_2=0.0187$, $a_3=1$, and $a_4=0.227$, design a lag compensator to obtain a step-response steady-state error of 10% without affecting the system's transient response appreciably.

b. Verify your design through MATLAB simulations. MATLAB ML

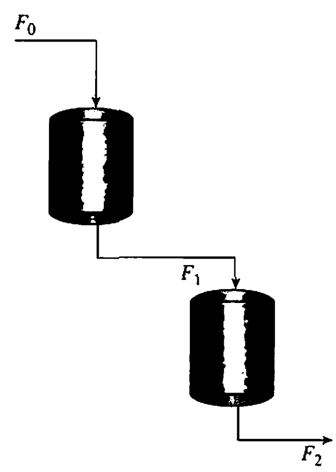


FIGURE P9.5

36. Figure P9.6(a) shows a heat-exchanger process whose purpose is to maintain the temperature of a liquid at a prescribed temperature.

The temperature is measured using a sensor and a transmitter, TT 22, that sends the measurement to a

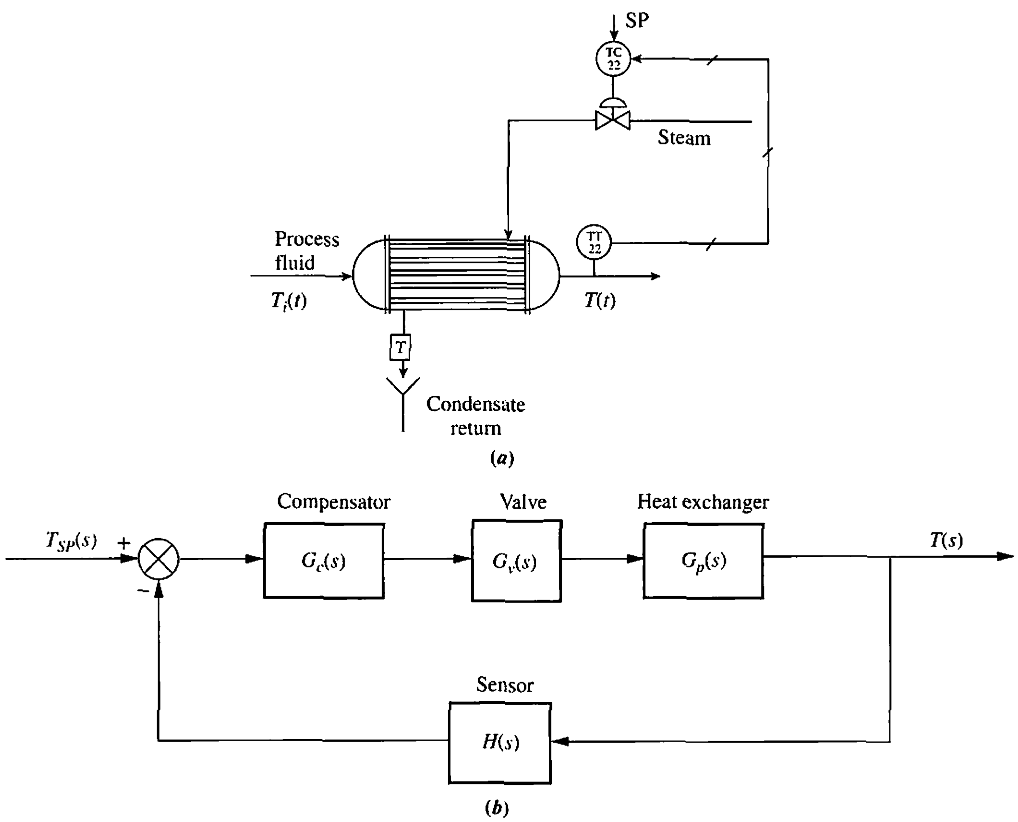


FIGURE P9.6 **a.** Heat-exchanger process (Reprinted with permission of John Wiley & Sons, Inc.); **b.** block diagram

corresponding controller, TC 22, that compares the actual temperature with a desired temperature set point, SP. The controller automatically opens or closes a valve to allow or prevent the flow of steam to change the temperature in the tank. The corresponding block diagram for this system is shown in Figure P9.6(*b*) (*Smith 2002*). Assume the following transfer functions:

$$G_v(s) = \frac{0.02}{4s+1}; \quad G_1(s) = \frac{70}{50s+1}; \quad H(s) = \frac{1}{12s+1}.$$

a. Assuming $G_c(s) = K$, find the value of K that will result in a dominant pole with $\zeta = 0.7$. Obtain the corresponding T_s.

b. Design a PD controller to obtain the same damping factor as Part **a** but with a settling time 20% smaller.

c. Verify your results through MATLAB simulation. MATLAB ML

37. Repeat Problem 36, Parts **b** and **c**, using a lead compensator.

38. a. Find the transfer function of a motor whose torque-speed curve and load are given in Figure P9.7.

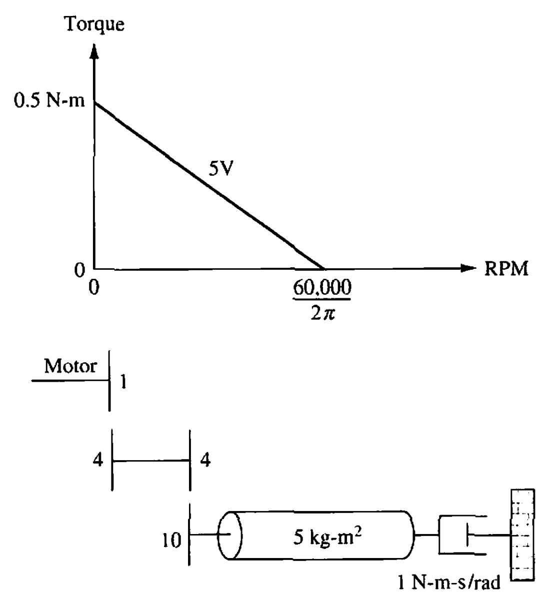


FIGURE P9.7

b. Design a tachometer compensator to yield a damping ratio of 0.5 for a position control employing a power amplifier of gain 1 and a preamplifier of gain 5000.

c. Compare the transient and steady-state characteristics of the uncompensated system and the compensated system.

39. You are given the motor whose transfer function is shown in Figure P9.8(a).

WileyPLUS WPCS Control Solutions

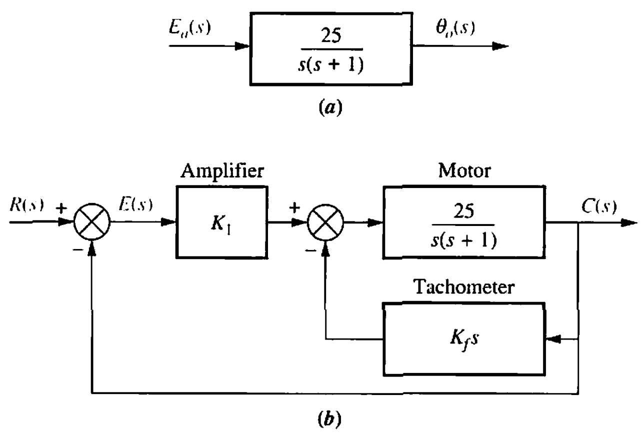


FIGURE P9.8

a. If this motor were the forward transfer function of a unity feedback system, calculate the percent overshoot and settling time that could be expected.

b. You want to improve the closed-loop response. Since the motor constants cannot be changed and you cannot use a different motor, an amplifier and tachometer are inserted into the loop as shown in Figure P9.8(b). Find the values of K_1 and K_f to yield a percent overshoot of 25% and a settling time of 0.2 second.

c. Evaluate the steady-state error specifications for both the uncompensated and the compensated systems.

40. A position control is to be designed with a 20% overshoot and a settling time of 2 seconds. You have on hand an amplifier and a power amplifier whose cascaded transfer function is $K_1/(s+20)$ with which to drive the motor. Two 10-turn pots are available to convert shaft position into voltage. A voltage of $\pm 5\pi$ volts is placed across the pots. A dc motor whose transfer function is of the form

$$\frac{\theta_o(s)}{E_a(s)} = \frac{K}{s(s+a)}$$

is also available. The transfer function of the motor is found experimentally as follows. The motor and geared load are driven open-loop by applying a large, short, rectangular pulse to the armature. An oscillo-gram of the response shows that the motor reached 63% of its final output value at 1/2 second after the application of the pulse. Further, with a constant 10 volts dc applied to the armature, the constant output speed was 100 rad/s.

a. Draw a complete block diagram of the system, specifying the transfer function of each component when the system is operating with 20% overshoot.

b. What will the steady-state error be for a unit ramp input?

c. Determine the transient response characteristics.

d. If tachometer feedback is used around the motor, as shown in Figure P9.9, find the tachometer and the amplifier gain to meet the original specifications. Summarize the transient and steady-state characteristics.

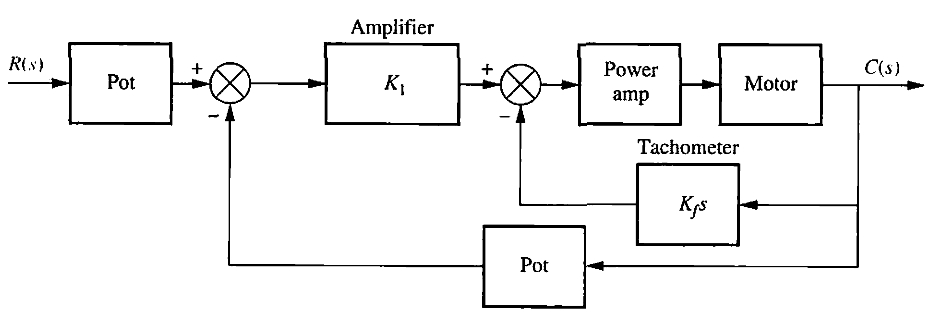


FIGURE P9.9

41. A position control is to be designed with a 10% overshoot, a settling time of 1 second, and $K_v = 1000$. You have on hand an amplifier and a power amplifier whose cascaded transfer function is $K_1/(s+40)$ with which to drive the motor. Two 10-turn pots are available to convert shaft position into voltage. A voltage of $\pm 20\pi$ volts is placed across the pots. A dc motor whose transfer function is of the form

$$\frac{\theta_o(s)}{E_a(s)} = \frac{K}{s(s+a)}$$

is also available. The following data are observed from a dynamometer test at 50 V. At 25 N-m of torque, the motor turns at 1433 rpm. At 75 N-m of torque, the motor turns at 478 rpm. The speed measured at the load is 0.1 that of the motor. The equivalent inertia, including the load, at the motor armature is 100 kg-m^2, and the equivalent viscous damping, including the load, at the motor armature is 50 N-m-s/rad.

a. Draw a complete block diagram of the system, specifying the transfer function of each component.

b. Design a passive compensator to meet the requirements in the problem statement.

c. Draw the schematic of the compensator showing all component values. Use an operational amplifier for isolation where necessary.

d. Use MATLAB or any other computer program to simulate your system and show that all requirements have been met. MATLAB ML

42. Given the system shown in Figure P9.10, find the values of K and K_f so that the closed-loop dominant poles will have a damping ratio of 0.5 and the underdamped poles of the minor loop will have a damping ratio of 0.8.

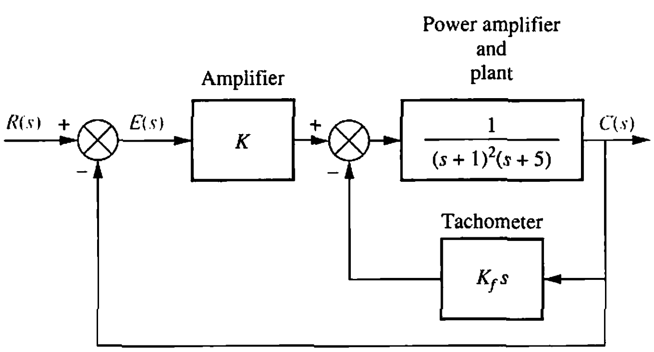


FIGURE P9.10

43. Given the system in Figure P9.11, find the values of K and K_f so that the closed-loop system will have a 4.32% overshoot and the minor loop will have a damping ratio of 0.8. Compare the expected performance of the system without tachometer compensation to the expected performance with tachometer compensation.

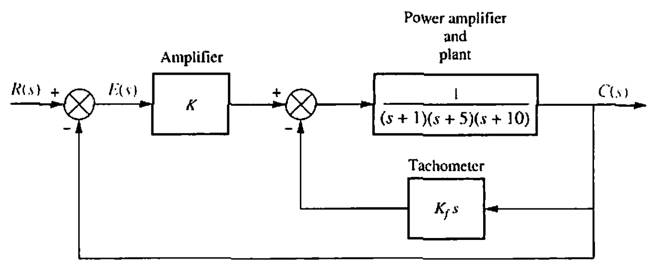


FIGURE P9.11

44. In Problem 57 of Chapter 8, a head-position control system for a floppy disk drive was designed to yield a settling time of 0.1 second through gain adjustment alone. Design a lead compensator to decrease the settling time to 0.05 second without changing the percent overshoot. Also, find the required loop gain.

45. Consider the temperature control system for a chemical process shown in Figure P9.12. The uncompensated system is operating with a rise time approximately the same as a second-order system with a peak time of 16 seconds and 5% overshoot. There is also considerable steady-state error. Design a PID controller so that the compensated system will have a rise time approximately equivalent to a second-order system with a peak time of 8 seconds and 5% overshoot, and zero steady-state error for a step input. WileyPLUS WPCS Control Solutions

46. Steam-driven power generators rotate at a constant speed via a governor that maintains constant steam pressure in the turbine. In addition, automatic generation control (AGC) or load frequency control (LFC) is added to ensure reliability and consistency despite load variations or other disturbances that can affect the distribution line frequency output. A specific turbine-governor system can be described only using the block diagram of Figure P9.1 in which $G(s) = G_c(s)G_g(s)G_t(s)G_m(s)$, where (*Khodabakhshian, 2005*)

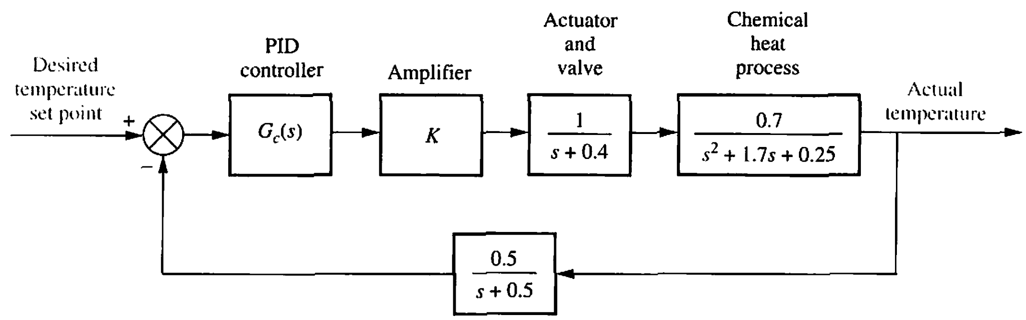


FIGURE P9.12 Chemical process temperature control system

$G_g(s) = \frac{1}{0.2s+1}$ is the governor's transfer function

$G_t(s) = \frac{1}{0.5s+1}$ is the turbine transfer function

$G_m(s) = \frac{1}{10s+0.8}$ represents the machine and load transfer functions

$G_c(s)$ is the LFC compensation to be designed

a. Assuming $G_c(s) = K$, find the value of K that will result in a dominant pole with $\zeta = 0.7$. Obtain the corresponding T_s.

b. Design a PID controller to obtain the same damping factor as in Part **a**, but with a settling time of 2 seconds and zero steady-state error to step input commands.

c. `Verify your results using a MATLAB simuation.` MATLAB ML

47. Repeat Problem 46 using a lag-lead compensator instead of a PID controller. Design for a steady-state error of 1% for a step input command.

48. Digital versatile disc (DVD) players incorporate several control systems for their operations. The control tasks include (1) keeping the laser beam focused on the disc surface, (2) fast track selection, (3) disc rotation speed control, and (4) following a track accurately. In order to follow a track, the pickup-head radial position is controlled via a voltage that operates a voice coil embedded in a magnet configuration. For a specific DVD player, the transfer function is given by

$$P(s) = \frac{X(s)}{V(s)} = \frac{0.63}{\left(1+\frac{0.36}{305.4}s+\frac{s^2}{305.4^2}\right)\left(1+\frac{0.04}{248.2}s+\frac{s^2}{248.2^2}\right)}$$

where $x(t)$ = radial pickup position and $v(t)$ = the coil input voltage (*Bittanti, 2002*).

a. Assume that the system will be controlled in a closed-loop configuration, such as the one shown in Figure P9.1. Assuming that the plant, $P(s)$, is cascaded with a proportional compensator, $G_c(s) = K$, plot the root locus of the system.

b. `Repeat Part` **a** `using MATLAB if your root locus plot was created by any other tool.` MATLAB ML

c. Find the range of K for closed-loop stability, the resulting damping factor range, and the smallest settling time.

d. Design a notch filter compensator so that the system's dominant poles have a damping factor of $\zeta = 0.7$ with a closed-loop settling time of 0.1 second.

e. `Simulate the system's step response for Part` **c** `using MATLAB.` MATLAB ML

f. Add a PI compensator to the system to achieve zero steady-state error for a step input without appreciably affecting the transient response achieved in Part **b**.

g. `Simulate the system's step response for Part` **e** `using MATLAB.` MATLAB ML

49. A coordinate measuring machine (CMM) measures coordinates on three-dimensional objects. The accuracy of CMMs is affected by temperature changes as well as by mechanical resonances due to joint elasticity. These resonances are more pronounced when the machine has to go over abrupt changes of dimension, such as sharp corners at high speed. Each of the machine links can be controlled in a closed-loop configuration, such as the one shown in

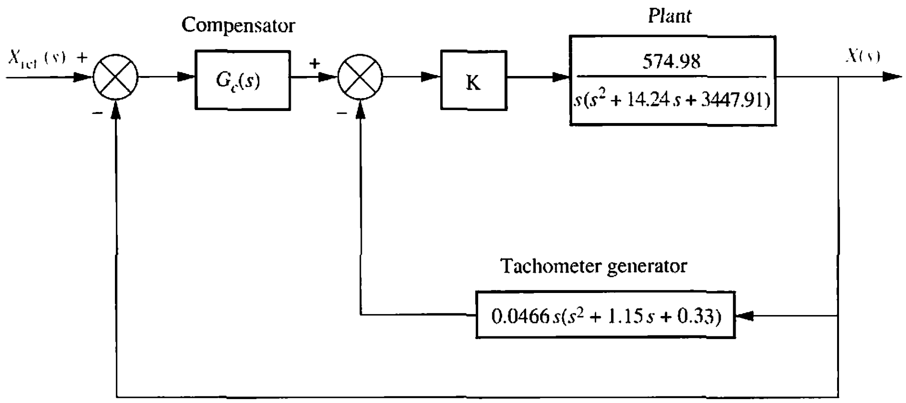


FIGURE P9.13

Figure P9.13 for a specific machine with prismatic (sliding) links. In the figure, $X_{ref}(s)$ is the commanded position and $X(s)$ is the actual position. The minor loop uses a tachometer generator to obtain the joint speed, while the main loop controls the joint's position (*Özel, 2003*).

a. Find the value of K that will result in a minor loop with $\zeta = 0.5$.

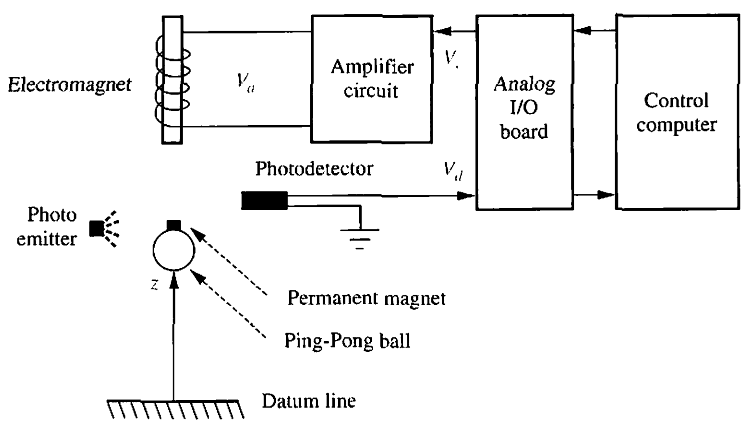
b. Use a notch filter compensator, $G_c(s)$, for the external loop so that it results in a closed-loop damping factor of $\zeta = 0.7$ with $T_s \approx 4$ seconds.

c. `Use MATLAB to simulate the compensated system's closed-loop step response.` MATLAB ML

50. Magnetic levitation systems are now used to elevate and propel trains along tracks. A diagram of a demonstration magnetic levitation system is shown in Figure P9.14(*a*). Action between a permanent magnet attached to the Ping-Pong ball, the object to be levitated, and an electromagnet provides the lift. The amount of elevation can be controlled through V_a applied to the electromagnet as shown in Figure P9.14(*a*). The elevation is controlled by using a photo-detector pair to detect the elevation of the Ping-Pong ball. Assume that the elevation control system is represented by Figure P9.14(*b*) and do the following (*Cho, 1993*):

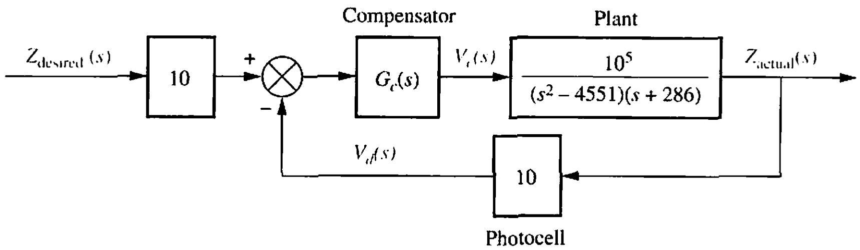
a. Design a compensator, $G_c(s)$, to yield a settling time of 0.1 second or less if the step response is to have no more than 1% overshoot. Specify the compensator's poles, zeros, and gain.

b. Cascade another compensator to minimize the steady-state error and have the total settling time



(*a*)

FIGURE P9.14 **a.** Magnetic levitation system (© 1993 IEEE); (*figure continues*)



(*b*)

FIGURE P9.14 (*Continued*) **b.** block diagram

not exceed 0.5 second. This compensator should not appreciably affect the transient response designed in Part **a**. Specify the poles and zeros of this compensator.

c. Use MATLAB or any other computer program to simulate the system to check your design. MATLAB ML

51. The transfer function for an AFTI/F-16 aircraft relating angle of attack, $\alpha(t)$, to elevator deflection, $\delta_e(t)$, is given by

$$G(s) = \frac{\alpha(s)}{\delta_e(s)} = 0.072\frac{(s+23)(s^2+0.05s+0.04)}{(s-0.7)(s+1.7)(s^2+0.08s+0.04)}$$

Assume the block diagram shown in Figure P9.15 for controlling the angle of attack, α, and do the following (*Monahemi, 1992*):

a. Find the range of K for stability.

b. Plot or sketch a root locus.

c. Design a cascade compensator to yield zero steady-state error, a settling time of about 0.05 second, and a percent overshoot not greater than 20%.

d. Use MATLAB or any other computer program to simulate the system to check your design. MATLAB ML

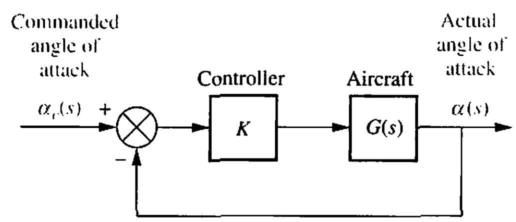


FIGURE P9.15 Simplified block diagram for angle of attack control

52. Figure P9.16 is a simplified block diagram of a self-guiding vehicle's bearing angle control. Design a lead compensator to yield a closed-loop step response with 10% overshoot and a settling time of 1.5 seconds.

53. An X-4 quadrotor flyer is designed as a small-sized unmanned autonomous vehicle (UAV) that flies mainly indoors and can help in search and recognizance missions. To minimize mechanical problems and for simplicity, this aircraft uses fixed pitch rotors with specially designed blades. Therefore, for thrust it is necessary to add a fifth propeller. A simplified design of the thrust control design can be modeled

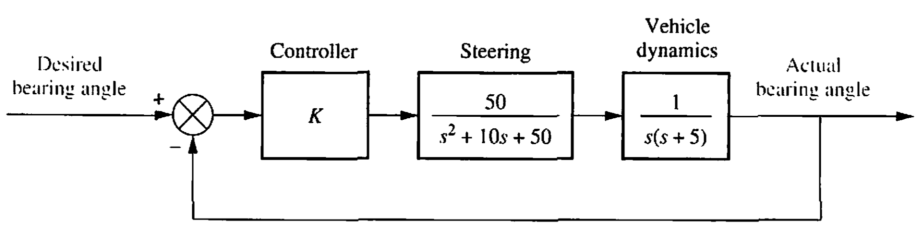


FIGURE P9.16 Simplified block diagram of a self-guiding vehicle's bearing angle control as in Figure 9.1 with $G(s) = G_c(s)P(s)$ where

$$P(s) = \frac{1.90978\left(\frac{s}{0.43}+1\right)}{\left(\frac{s}{9.6}+1\right)\left(\frac{s}{0.54}+1\right)}$$

represents the dynamics of the thruster rotor gain, the motor, and the battery dynamics. Initially, the system is designed using a proportional compensator given by $G_c(s) = 3$ (*Pounds, 2009*).

a. Calculate the resulting steady-state error for a unit step input.

b. Design a lag compensator to yield half the steady-state error of the proportional compensator, without appreciably affecting the system's transient response.

c. `Use MATLAB to simulate the original design and the lag compensated design. Verify your results.` MATLAB ML

54. Problem 8.56 described an ac/dc conversion and power distribution system for which droop control is implemented through the use of a proportional controller to stabilize the dc-bus voltage. For simplification, a system with only one source converter and one load converter was considered. The parameters and design considerations presented in that problem, along with some solution results, allow us to represent the block-diagram of that system as shown in the Figure P9.17.

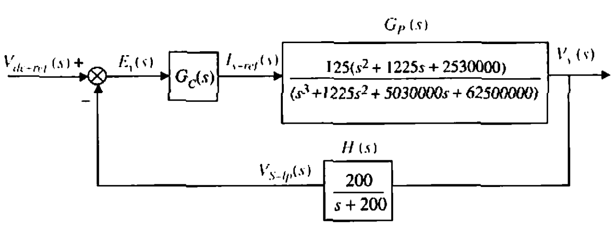


FIGURE P9.17

Here G_c(s) is the transfer function of the controller, G_p(s) represents the forward path of the controlled plant (a conversion and power distribution unit), and H(s) is the transfer function of the feedback low-pass filter (*Karlsson, 2003*).

Prepare a table, such as Table 9.5, where the first column, headed *Uncompensated*, is filled in with your results from the proportional design of Problem 8.56, assuming an input step, $v_{dc-ref}(t) = 750\ u(t)$.

Follow Steps 2–8 as described in Section 9.4 (Example 9.5), to design a proportional-plus-integral-plus-derivative (PID) controller so that the system can operate with a percent overshoot $\leq$4.4 %, a peak time 20% smaller than that of the uncompensated system, and zero steady-state error, $e_{v_{\text{step}}}(\infty) = 0$. Fill in the remaining two columns of your table, *PD-compensated* and *PID-compensated*.

PROGRESSIVE ANALYSIS AND DESIGN PROBLEMS

55. High-speed rail pantograph. Problem 21 in Chapter 1 discusses the active control of a pantograph mechanism for high-speed rail systems. In Problem 79(b), Chapter 5, you found the block diagram for the active pantograph control system. In Chapter 8, Problem 72, you designed the gain to yield a closed-loop step response with 38% overshoot. A plot of the step response should have shown a settling time greater than 0.5 second as well as a high-frequency oscillation superimposed over the step response (*O'Conner, 1997*). We want to reduce the settling time to about 0.3 second, reduce the step response steady-state error to zero, and eliminate the high-frequency oscillation. A way of eliminating the high-frequency oscillation is to cascade a notch filter with the plant. Using the notch filter,

$$G_n(s) = \frac{s^2+16s+9200}{(s+60)^2}$$

do the following:

a. Design a PD controller to yield a settling time of approximately 0.3 second with no more then 60% overshoot.

b. Add a PI controller to yield zero steady-state error for step inputs.

c. `Use MATLAB to plot the PID/ notch-compensated closed-loop step response.` MATLAB ML

56. Control of HIV/AIDS. It was shown in Chapter 6, Problem 68, that when the virus levels in an HIV/AIDS patient are controlled using RTIs the linearized plant model is

$$P(s) = \frac{Y(s)}{U_1(s)} = \frac{-520s - 10.3844}{s^3 + 2.6817s^2 + 0.11s + 0.0126}$$

Assume that the system is embedded in a configuration, such as the one shown in Figure P9.1, where $G(s) = G_c(s)\,P(s)$. Here, $G_c(s)$ is a cascade compensator. For simplicity in this problem, choose the dc gain of $G_c(s)$ less than zero to obtain a negative-feedback system (the negative signs of $G_c(s)$ and $P(s)$ cancel out) (*Craig, I. K., 2004*).

a. Consider the uncompensated system with $G_c(s) = -K$. Find the value of K that will place all closed-loop poles on the real axis.

b. Use MATLAB to simulate the unit step response of the gain-compensated system. Note the %OS and the T_s from the simulation. MATLAB ML

c. Design a PI compensator so that the steady-state error for step inputs is zero. Choose a gain value to make all poles real.

d. Use MATLAB to simulate the design in Part c for a unit step input. Compare the simulation to Part b. MATLAB ML

57. **Hybrid vehicle.** In the previous chapter, we used the root locus to design a proportional controller for the speed control of an HEV. We rearranged the block diagram to be a unity feedback system, as shown in the block diagram of Figure P7.34 (*Preitl, 2007*). The plant and compensator resulted in

$$G(s) = \frac{K(s + 0.60)}{(s + 0.5858)(s + 0.0163)}$$

and we found that $K = 0.78$ resulted in a critically damped system.

a. Use this design to itemize the performance specifications by filling in a table, similar to Table 9.5, under the column *Uncompensated.* Take advantage of the results from Chapter 8 or use MATLAB to find the entries. Plot $c(t)$ for $r(t) = 4\ u(t)$ volts.

b. Now assume that the system specifications require zero steady-state error for step inputs, a steady-state error for ramp inputs $\leq 2\%$, a %OS $\leq 4.32\%$, and a settling time ≤ 4 sec. It should be evident that this is not accomplished with a proportional controller. Thus, start by designing a PI controller to meet the requirements. If necessary add a PD mode to get a PID controller. Simulate your final design using MATLAB. Fill in the results of this design in the second column of your table with the heading *Compensated*. MATLAB ML

c. Now note the following limitations of linear control system modeling:

(1) No limit is set on system variables. For example, vehicle acceleration as well as motor and power amplifier current, torque or power do not have upper limits.

(2) It is assumed that to improve the speed of response in Part **b**, we could place the PI controller's zero on top of the pole closest to the origin. Realistically, such pole-zero cancellation is not always possible to maintain.

If you do not expand your model beyond the described limitations if required for accuracy, unrealistic response characteristics, such as rise and settling times could result. Look at your design results including response curves. Are they realistic? If not, revise your Simulink model, which you developed for Problem 5.81, as follows: Simulink SL

i. Represent the motor armature as a first-order system with a unity steady-state gain and a time constant of 50 ms, which avoids the creation of internal algebraic closed-loops and should have negligible effect on system response;

ii. Add a saturation element at the output of the motor armature and set it to an upper limit of 250 A;

iii. Use the following PI settings. The PI settings of the speed controller are P = 61 and I = 0.795. The PI settings of the torque controller are P = 10 and I = 6;

iv. Run the modified model and comment on the graphs obtained for motor current, car acceleration, and speed.

Cyber Exploration Laboratory

Experiment 9.1

Objectives To perform a trade-off study for lead compensation. To design a PI controller and see its effect upon steady-state error.

Minimum Required Software Packages MATLAB, and the Control System Toolbox

Prelab

1. How many lead compensator designs will meet the transient response specifications of a system?
2. What differences do the lead compensators of Prelab 1 make?
3. Design a lead compensator for a unity negative feedback system with a forward transfer function of $G(s) = \frac{K}{s(s+3)(s+6)}$ to meet the following specifications: percent overshoot = 20%; settling time = 2 seconds. Specify the required gain, K. Estimate the validity of the second-order approximation.
4. What is the total angular contribution of the lead compensator of Prelab 3?
5. Determine the pole and zero of two more lead compensators that will meet the requirements of Prelab 3.
6. What is the expected steady-state error for a step input for each of the lead-compensated systems?
7. What is the expected steady-state error for a ramp input for each of the lead-compensated systems?
8. Select one of the lead compensator designs and specify a PI controller that can be cascaded with the lead compensator that will produce a system with zero steady-state error for both step and ramp inputs.

Lab

1. Using the SISO Design Tool, create the design in Prelab 3 and plot the root locus, step response, and ramp response. Take data to determine the percent overshoot, settling time, and step and ramp steady-state errors. Record the gain, K.
2. Repeat Lab 1 for each of the designs in Prelab 5.
3. For the design selected in Prelab 8, use the SISO Design Tool and insert the PI controller. Plot the step response and measure the percent overshoot, settling time, and steady-state error. Also, plot the ramp response for the design and measure the steady-state error.
4. Plot the step and ramp responses for two more values of the PI controller zero.

Postlab

1. Make a table showing calculated and actual values for percent overshoot, settling time, gain, K, steady-state error for step inputs, and steady-state error for ramp inputs. Use the three systems without the PI controller and the single system with the PI controller from Lab 3.
2. Itemize the benefits of each system without the PI controller.
3. Choose a final design and discuss the reasons for your choice.

Experiment 9.2

Objective To design a PID controller via LabVIEW

Minimum Required Software Packages LabVIEW with the Control Design and Simulation Module

Prelab

1. Perform Cyber Exploration Laboratory Experiment 8.3.
2. Use the system described in Cyber Exploration Laboratory Experiment 8.3 and replace the controller described there, $G_c(s) = K_D s + K_P$, with a PID controller.
3. Design the controller to meet the following requirements: (1) shorten the settling time found in the design of Cyber Exploration Laboratory Experiment 8.3 to less than 1 sec., and (2) limit the percent overshoot to no more than 5%.
4. Design a LabVIEW VI to test your design. The front panel inputs will be the PID gains and the numerator and denominator of the plant. The indicators will be the transfer functions of the plant, PID controller, and closed-loop system. Finally, provide an indicator for the step-response graph.

Lab Run your LabVIEW VI and obtain the step response of the closed-loop system.

Postlab Compare the transient and steady-state error performance between the closed-loop step responses of Cyber Exploration Laboratory Experiment 8.3 and this experiment.

Bibliography

Bittanti, S., Dell'Orto, F., Di Carlo, A., and Savaresi, S. M. Notch Filtering and Multirate Control for Radial Tracking in High Speed DVD-Players. *IEEE Transactions on Consumer Electronics*, vol. 48, 2002, pp. 56–62.

Budak, A. *Passive and Active Network Analysis and Synthesis.* Houghton Mifflin, Boston, MA, 1974.

Cho, D., Kato, Y., and Spilman, D. Sliding Mode and Classical Controllers in Magnetic Levitation Systems. *IEEE Control Systems*, February 1993, pp. 42–48.

Craig, I. K., Xia, X., and Venter, J. W. Introducing HIV/AIDS Education into the Electrical Engineering Curriculum at the University of Pretoria. *IEEE Transactions on Education*, vol. 47, no. 1, February 2004, pp. 65–73.

Craig, J. J. *Introduction to Robotics. Mechanics and Control*, 3d ed. Prentice Hall, Upper Saddle River, NJ, 2005.

D'Azzo, J. J., and Houpis, C. H. *Feedback Control System Analysis and Synthesis*, 2d ed. McGraw-Hill, New York, 1966.

Dorf, R. C. *Modern Control Systems*, 5th ed. Addison-Wesley, Reading, MA, 1989.

Hostetter, G. H., Savant, C. J., Jr., and Stefani, R. T. *Design of Feedback Control Systems*, 2d ed. Saunders College Publishing, New York, 1989.

Johnson, H. et al. *Unmanned Free-Swimming Submersible (UFSS) System Description*. NRL Memorandum Report 4393. Naval Research Laboratory, Washington, D.C., 1980.

Karlsson, P., and Svensson, J. DC Bus Voltage Control for a Distributed Power System, *IEEE Trans. on Power Electronics*, vol. 18, no. 6, 2003, pp. 1405–1412.

Khodabakhshian, A., and Golbon, N. Design of a New Load Frequency PID Controller Using QFT. *Proceedings of the 13th Mediterranean Conference on Control and Automation*, 2005, pp. 970–975.

Kuo, B. C. *Automatic Control Systems*, 7th ed. Prentice Hall, Upper Saddle River, NJ, 1995.

Monahemi, M. M., Barlow, J. B., and O'Leary, D. P. Design of Reduced-Order Observers with Precise Loop Transfer Recovery. *Journal of Guidance, Control, and Dynamics*, vol. 15, no. 6, November–December 1992, pp. 1320–1326.

O'Connor, D. N., Eppinger, S. D., Seering, W. P., and Wormly, D. N. Active Control of a High-Speed Pantograph. *Journal of Dynamic Systems, Measurements, and Control*, vol. 119, March 1997, pp. 1–4.

Ogata, K. *Modern Control Engineering*, 2d ed. Prentice Hall, Upper Saddle River, NJ, 1990.

Özel, T. Precision Tracking Control of a Horizontal Arm Coordinate Measuring Machine. *Proceedings of the IEEE Conference on Control Applications*, 2003, pp. 103–108.

Pounds, P. E. I., Mahony, R. E., and Corke, P. I.; Design of a Static Thruster for Microair Vehicle Rotorcraft. *Journal of Aerospace Engineering*, vol. 22, no. 1, 2009, pp. 85–94.

Preitl, Z., Bauer, P., and Bokor, J. A Simple Control Solution for Traction Motor Used in Hybrid Vehicles. *Fourth International Symposium on Applied Computational Intelligence and Informatics*. IEEE. 2007.

Romagnoli, J. A., and Palazoglu, A. *Introduction to Process Control*. CRC Press, Boca Raton, FL, 2006.

Smith, C. A. *Automated Continuous Process Control*. Wiley, New York, 2002.

Thomas, B., Soleimani-Mosheni, M., and Fahlén, P. Feed-forward in Temperature Control of Buildings. *Energy and Buildings*, vol. 37, 2005, pp. 755–761.

Van de Vegte, J. *Feedback Control Systems*, 2d ed. Prentice Hall, Upper Saddle River, NJ, 1990.