10TH EDITION

AUTOMATIC CONTROL SYSTEMS

FARID GOLNARAGHI

BENJAMIN C. KUO

Mc Graw Hill Education

Automatic Control Systems

FARID GOLNARAGHI
Simon Fraser University

BENJAMIN C. KUO
University of Illinois at Urbana-Champaign

Tenth Edition

Mc Graw Hill Education

New York Chicago San Francisco
Athens London Madrid
Mexico City Milan New Delhi
Singapore Sydney Toronto

ISBN: 978-1-25-964384-2
MHID: 1-25-964384-0.

The material in this eBook also appears in the print version of this title: ISBN: 978-1-25-964383-5, MHID: 1-25-964383-2.

eBook conversion by codeMantra
Version 1.0

Information contained in this work has been obtained by McGraw-Hill Education from sources believed to be reliable. However, neither McGraw-Hill Education nor its authors guarantee the accuracy or completeness of any information published herein, and neither McGraw-Hill Education nor its authors shall be responsible for any errors, omissions, or damages arising out of use of this information. This work is published with the understanding that McGraw-Hill Education and its authors are supplying information but are not attempting to render engineering or other professional services. If such services are required, the assistance of an appropriate professional should be sought.

TERMS OF USE

THE WORK IS PROVIDED “AS IS.” McGRAW-HILL EDUCATION AND ITS LICENSORS MAKE NO GUARANTEES OR WARRANTIES AS TO THE ACCURACY, ADEQUACY OR COMPLETENESS OF OR RESULTS TO BE OBTAINED FROM USING THE WORK, INCLUDING ANY INFORMATION THAT CAN BE ACCESSED THROUGH THE WORK VIA HYPERLINK OR OTHERWISE, AND EXPRESSLY DISCLAIM ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. McGraw-Hill Education and its licensors do not warrant or guarantee that the functions contained in the work will meet your requirements or that its operation will be uninterrupted or error free. Neither McGraw-Hill Education nor its licensors shall be liable to you or anyone else for any inaccuracy, error or omission, regardless of cause, in the work or for any damages resulting therefrom. McGraw-Hill Education has no responsibility for the content of any information accessed through the work. Under no circumstances shall McGraw-Hill Education and/or its licensors be liable for any indirect, incidental, special, punitive, consequential or similar damages that result from the use of or inability to use the work, even if any of them has been advised of the possibility of such damages. This limitation of liability shall apply to any claim or cause whatsoever whether such claim or cause arises in contract, tort or otherwise.

To my wife, Mitra, and my daughters, Sophia and Carmen, the joys of my life.

FG

About the Authors

Dr. Farid Golnaraghi has been a professor and the Director of Mechatronic Systems Engineering at Simon Fraser University in Vancouver since 2006. He also holds a Burnaby Mountain Endowed Chair at SFU, and his primary research focuses on intelligent vehicle systems. Prior to joining SFU, Dr. Golnaraghi was a professor of Mechanical and Mechatronics Engineering at the University of Waterloo. His pioneering research has resulted in two textbooks, more than 150 journal and conference papers, four patents, and two start-up companies.

Dr. Benjamin C. Kuo was a member of the faculty of the Department of Electrical and Computer Engineering at the University of Illinois at Urbana-Champaign, which he joined in 1958 upon the completion of his Ph.D. and where he remained for 31 rewarding years. He was a true visionary pioneer in the field of automatic control, and the impact of his distinguished career as a researcher and educator cannot be overstated.

In Memory of Professor Benjamin C. Kuo—A Visionary Pioneer and a Brilliant Teacher

When people think of an academic institution, it is its people—their intellect, their brilliance, and their impact—that defines it in their mind. The late Professor Benjamin Kuo is one of those brilliant professors who defines the University of Illinois, and, in particular, the Department of Electrical and Computer Engineering.

Ben was both an alumnus and a member of the faculty of the Department of Electrical and Computer Engineering at the University of Illinois. Upon the completion of his Ph.D., in 1958, he joined the faculty of the department and served it for 31 years through a stellar and most-influential career.

Ben was a visionary pioneer in the field of automatic control. His book !*Automatic Control Systems*, which was first published in 1962, has been translated into many languages, and through nine editions over almost 50 years became one of the most widely read books in the field.

His symposium on Incremental Motion Control Systems and Devices, which was first held in 1972, became a pole of attraction for researchers and engineers from industry and academia from all over the world and contributed to the advancement of the field in a singular and most-influential manner.

The impact of Ben's distinguished career as a researcher and an educator is best described in terms of the ways he touched the lives of people:

- The over 120 graduate students who did their research under Ben's advising and mentorship, and who launched exciting careers in industry and academia and spread the new ideas of digital control and contributed to the growth of the field.
- The hundreds of thousands of students around the world who, through Ben's book, were introduced to the subject of automatic control and got excited about its multifaceted applications. I was one of them as an undergraduate student in Greece.
- And, finally, the billions of people all over the globe whose lives are

easier and safer today thanks to the ubiquitous control electronics that Professor Benjamin Kuo dreamed about in the fertile grounds of the University of Illinois several decades ago.

The contributions of a brilliant mind endure long after the person passes and is no longer there.

Andreas C. Cangellaris
Dean, College of Engineering, and
M. E. Van Valkenburg Professor in Electrical and Computer Engineering
University of Illinois at Urbana-Champaign

Contents

Modeling of Dynamic Systems

CHAPTER 3
Solution of Differential Equations of Dynamic Systems

CHAPTER 4

Block Diagrams and Signal-Flow Graphs

CHAPTER 6

Important Components of Feedback Control Systems

CHAPTER 7
Time-Domain Performance of Control Systems

CHAPTER 8
State-Space Analysis and Controller Design

CHAPTER 9

Root-Locus Analysis

CHAPTER 11

Design of Control Systems

The appendices can be found at www.mhprofessional.com/golnaraghi.

APPENDIX E
ACSYS 2013: Description of the Software

APPENDIX F
Properties and Construction of the Root Loci

APPENDIX G
General Nyquist Criterion

APPENDIX H
Discrete-Data Control Systems

APPENDIX I
Difference Equations

APPENDIX J
z-Transform Table

Preface

This tenth edition is with a new publisher, McGraw-Hill Education. It represents a complete overhaul of the textbook, which delivers practical coverage designed to introduce readers to the essential concepts of automatic control systems. The new edition features significant enhancements of all chapters, a greater number of solved examples, labs using both LEGO® MINDSTORMS® and MATLAB®/SIMLab, and a valuable introduction to the concept of Control Lab. The book gives students a real-world understanding of the subject and prepares them for the challenges they will one day face.

For this edition, we increased the number of examples, added more MATLAB toolboxes, and enhanced the MATLAB GUI software, **ACSYS,** to allow interface with LEGO MINDSTORMS. We also added more computer-aided tools for students and teachers. The new edition has been 5 years in the making, and was reviewed by many professors to better fine-tune the new concepts. In this edition, Chaps. 1 through 3 are organized to contain all background material, while Chaps. 4 through 11 contain material directly related to the subject of control. The Control Lab material is presented in great detail in App. D.

The following appendices for this book can be found at www.mhprofessional.com/golnaraghi:

Appendix A: Elementary Matrix Theory and Algebra

Appendix B: Mathematical Foundations

Appendix C: Laplace Transform Table

Appendix D: Control Lab

Appendix E: **ACSYS** 2013: Description of the Software

Appendix F: Properties and Construction of the Root Loci

Appendix G: General Nyquist Criterion

Appendix H: Discrete-Data Control Systems

The following paragraphs are aimed at three groups: professors who have adopted the book or who we hope will select it as their text; practicing engineers looking for answers to solve their day-to-day design problems; and, finally, students who are going to live with the book because it has been assigned for the control-systems course they are taking.

To Professors

The material assembled in this book is an outgrowth of junior- and senior-level control-systems courses taught by Professors Golnaraghi and Kuo at their respective universities throughout their teaching careers. The first nine editions have been adopted by hundreds of universities in the United States and around the world and have been translated into at least six languages.

Most undergraduate control courses have labs dealing with time response and control of dc motors—namely, speed response, speed control, position response, and position control. In many cases, because of the high cost of control lab equipment, student exposure to test equipment is limited, and as a result, many students do not gain a practical insight into the subject. In this tenth edition, recognizing these limitations, we introduce the concept of **Control Lab**, which includes two classes of experiments: **SIMLab** (**model-based** simulation) and **LEGOLab (physical experiments)**. These experiments are intended to supplement, or replace, the experimental exposure of the students in a traditional undergraduate control course.

In this edition, we have created a series of inexpensive control experiments for the **LEGO MINDSTORMS** NXT dc motor that will allow students to do their work within the **MATLAB** and **Simulink**® environment—even at home. See App. D for more details. This cost-effective approach may allow educational institutions to equip their labs with a number of LEGO test beds and maximize student access to the equipment at a fraction of the cost of currently available control-systems experiments. Alternatively, as a supplemental learning tool, students can take the equipment home after leaving a deposit and learn at their own pace. This concept has proven to be extremely successful at Simon Fraser University, Professor Golnaraghi’s home university in Vancouver, Canada.

The labs include experiments on **speed and position control** of dc motors, followed by a **controller design project** involving control of a simple robotic system conducting a pick-and-place operation and position control of an elevator system. Two other projects also appear in Chaps. 6 and 7. The specific goals of these new experiments are

- To provide an in-depth, practical discussion of the dc motor speed response, speed control, and position control concepts.
- To provide examples on how to identify the parameters of a physical system, experimentally.
- To give a better feel for controller design through realistic examples.

This text contains not only conventional MATLAB toolboxes, where students can learn MATLAB and utilize their programing skills, but also a graphical MATLAB-based software, **ACSYS**. The **ACSYS** software added to this edition is very different from the software accompanying any other control book. Here, through extensive use of MATLAB GUI programming, we have created software that is easy to use. As a result, students need only to focus on learning control problems, not programming!

To Practicing Engineers

This book was written with the readers in mind and is very suitable for self-study. Our objective was to treat subjects clearly and thoroughly. The book does not use the theorem–proof–Q.E.D. style and is without heavy mathematics. We have consulted extensively for wide sectors of the industry for many years and have participated in solving numerous control-systems problems, from aerospace systems to industrial controls, automotive controls, and control of computer peripherals. Although it is difficult to adopt all the details and realism of practical problems in a textbook at this level, some examples and problems reflect simplified versions of real-life systems.

To Students

You have had it now that you have signed up for this course and your professor has assigned this book! You had no say about the choice, though you can form and express your opinion on the book after reading it. Worse yet, one of the reasons that your professor made the selection is because he or she intends to make you work hard. But please don’t misunderstand us: What we really mean is that, though this is an easy book to study (in our opinion), it is a no-nonsense book. It doesn't have cartoons or nice-looking photographs to amuse you. From here on, it is all business and hard work. You should have had the prerequisites of subjects found in a typical linear-systems course, such as how to solve linear ordinary differential equations, Laplace transforms and applications, and time-response and frequency-domain analysis of linear systems. In this book, you will not find too much new mathematics to which you have not been exposed before. What is interesting and challenging is that you are going to learn how to apply some of the mathematics that you have acquired during the past 2 or 3 years of study in college. In case you need to review some of the mathematical foundations, you can find them in the appendices, at www.mhprofessional.com/golnaraghi. You will also find the Simulink-based SIMLab and LEGOLab, which will help you to gain understanding of real-world control systems.

This book has numerous illustrative examples. Some of these are deliberately simple for the purpose of showing new ideas and subject matter. Some examples are more elaborate, in order to bring the practical world closer to you. Furthermore, the objective of this book is to present a complex subject in a clear and thorough way. One of the important learning strategies for you as a student is not to rely strictly on the textbook assigned. When studying a certain subject, go to the library and check out a few similar texts to see how other authors treat the same subject. You may gain new perspectives on the subject and discover that one author treats the material with more care and thoroughness than the others. Do not be distracted by written-down coverage with oversimplified examples. The minute you step into the real world, you will face the design of control systems with nonlinearities and/or time-varying elements as well as orders that can boggle your mind. You may find it discouraging to be told now that strictly linear and first-order systems do not exist in the real world.

Special Acknowledgments

Farid Golnaraghi wishes to thank Robert L. Argentieri, Editorial Director, Engineering, International & Professional Group, McGraw-Hill Education, for his efforts and support in publishing this new edition. Special thanks go to the reviewers for their invaluable comments and suggestions; the prepublication reviews have had a great impact on this revision. More special

thanks go to the many School of Mechatronic Systems Engineering at Simon Fraser University faculty, students, and research associates, and all contributing undergraduate co-op students.

Professor Golnaraghi is grateful to and recognizes the Kuo estate, particularly Lori Dillon, for their support of this project.

Finally, he wishes to thank the late Professor Benjamin Kuo for sharing the pleasure of writing this wonderful book, and for his teachings, patience, and support throughout this experience.

Farid Golnaraghi
Vancouver, British Columbia, Canada

CHAPTER 1

Introduction to Control Systems

The main objectives of this chapter are

1. To define a control system.
2. To explain why control systems are important.
3. To introduce the basic components of a control system.
4. To give some examples of control-system applications.
5. To explain why feedback is incorporated into most control systems.
6. To introduce types of control systems.

Over the past five decades, control systems have assumed an increasingly important role in the development and advancement of modern civilization and technology. Practically every aspect of our day-to-day activities is affected by some type of control system. For instance, in the domestic domain, we need to regulate the temperature and humidity of homes and buildings for comfortable living. For transportation, various functionalities of the modern automobiles and airplanes involve control systems. Industrially, manufacturing processes contain numerous objectives for products that will satisfy the precision and cost-effectiveness requirements. A human being is capable of performing a wide range of tasks, including decision making. Some of these tasks, such as picking up objects and walking from one point to another, are commonly carried out in a routine fashion. Under certain conditions, some of these tasks are to be performed in the best possible way. For instance, an athlete running a 100-yd dash has the objective of running that distance in the shortest possible time. A marathon runner, on the other hand, not only must run the distance as quickly as possible, but, in doing so, he or she must also control the consumption of energy and devise the best strategy for the race. The means of achieving these “objectives” usually involve the use of control systems that implement certain control strategies.

Learning Outcomes

After successful completion of this chapter, you will be able to

1. Appreciate the role and importance of control systems in our daily lives.
2. Understand the basic components of a control system.
3. Understand the difference between the open-loop and closed-loop systems, and the role of feedback in a closed-loop control system.
4. Gain a practical sense of real life control problems, through the use of **LEGO® MINDSTORMS®**, **MATLAB®**, and **Simulink®**.

Control systems are found in abundance in all sectors of industry, such as quality control of manufactured products, automatic assembly lines, machine-tool control, space technology, computer control, transportation systems, power systems, robotics, microelectromechanical systems (MEMS), nanotechnology, and many others. Even the control of inventory and social and economic systems may be approached from the control system theory. More specifically, applications of control systems benefit many areas, including

Control systems abound in modern civilization.

- ***Process control.*** Enable automation and mass production in industrial setting.
- ***Machine tools.*** Improve precision and increase productivity.
- ***Robotic systems.*** Enable motion and speed control.
- ***Transportation systems.*** Various functionalities of the modern automobiles and airplanes involve control systems.
- ***MEMS.*** Enable the manufacturing of very small electromechanical devices such as microsensors and microactuators.
- ***Lab-on-a-chip.*** Enable functionality of several laboratory tasks on a single chip of only millimeters to a few square centimeters in size for medical diagnostics or environmental monitoring.
- ***Biomechanical and biomedical.*** Artificial muscles, drug delivery systems, and other assistive technologies.

1-1 BASIC COMPONENTS OF A CONTROL SYSTEM

The basic ingredients of a control system can be described by

- Objectives of control.
- Control-system components.
- Results or outputs.

The basic relationship among these three components is illustrated in a block diagram representation, as shown Fig. 1-1. The **block diagram** representation, as later discussed in Chap. 4, provides a graphical approach to describe how components of a control system interact. In this case, the **objectives** can be identified with **inputs**, or **actuating signals, *u*,** and the results are also called **outputs**, or **controlled variables, *y***. In general, the objective of the control system is to control the outputs in some prescribed manner by the inputs through the elements of the control system.

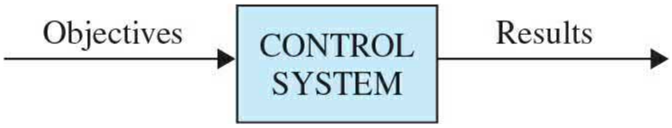


Figure 1-1 Basic components of a control system.

1-2 EXAMPLES OF CONTROL-SYSTEM APPLICATIONS

Applications of control systems have significantly increased through advances in computer technology and development of new materials, which provide unique opportunities for highly efficient actuation and sensing, thereby reducing energy losses and environmental impacts. State-of-the-art actuators and sensors may be implemented in virtually any system, including biological propulsion; locomotion; robotics; material handling; biomedical, surgical, and endoscopic; aeronautics; marine; and the defense and space industries.

The following represent some of the applications of control that have become part of our daily lives.

1-2-1 Intelligent Transportation Systems

The automobile and its evolution in the past two centuries is arguably the most transformative invention of man. Over the years, many innovations have made cars faster, stronger, and aesthetically appealing. We have grown to desire cars that are “intelligent” and provide maximum levels of comfort, safety, and fuel efficiency. Examples of intelligent systems in cars include climate control, cruise control, antilock brake systems (ABSs), active suspensions that reduce vehicle vibration over rough terrain, air springs that self-level the vehicle in high-G turns (in addition to providing a better ride), integrated vehicle dynamics that provide yaw control when the vehicle is either over- or understeering (by selectively activating the brakes to regain vehicle control), traction control systems to prevent spinning of wheels during acceleration, and active sway bars to provide “controlled” rolling of the vehicle. The following are a few examples.

Drive-by-Wire and Driver-Assist Systems

The new generations of intelligent vehicles are able to understand the driving environment, know their whereabouts, monitor their health, understand the road signs, and monitor driver performance, even overriding drivers to avoid catastrophic accidents. These tasks require significant overhaul of past designs. Drive-by-wire technology is replacing the traditional mechanical and hydraulic systems with electronics and control systems, using electromechanical actuators and human–machine interfaces such as pedal and steering feel emulators—otherwise known as haptic systems. Hence, the traditional components—such as the steering column, intermediate shafts, pumps, hoses, fluids, belts, coolers, brake boosters, and master cylinders—are eliminated from the vehicle. Haptic interfaces can offer adequate transparency to the driver while maintaining safety and stability of the system. Removing the bulky mechanical steering wheel column and the rest of the steering system has clear advantages in terms of mass reduction and safety in modern vehicles, along with improved ergonomics as a result of creating more driver space. Replacing the steering wheel with a haptic device that the driver controls through the sense of touch would be useful in this regard. The haptic device would produce the same sense to the driver as the mechanical steering wheel but with improvements in cost, safety, and fuel consumption as a result of eliminating the bulky mechanical system.

Driver-assist systems help drivers avoid or mitigate an accident by sensing the nature and significance of the danger. Depending on the significance and timing of the threat, these on-board safety systems will initially alert the driver as early as possible to an impending danger. Then, it will actively assist or, ultimately, intervene in order to avert the accident or mitigate its consequences. Provisions for automatic override features, when the driver loses control due to fatigue or lack of attention, will be an important part of the system. In such systems, the so-called advanced vehicle control system monitors the longitudinal and lateral control, and by interacting with a central management unit, it will be ready to take control of the vehicle whenever the need arises. The system can be readily integrated with sensor networks that monitor every aspect of the conditions on the road and are prepared to take appropriate action in a safe manner.

Integration and Utilization of Advanced Hybrid Powertrains

Hybrid technologies offer improved fuel consumption while enhancing driving experience. Utilizing new energy storage and conversion technologies and integrating them with powertrains are prime objectives in hybrid technologies. Such technologies must be compatible with combustion engine platforms and must enhance, rather than compromise, vehicle function. Sample applications include plug-in hybrid technology, which would enhance the vehicle cruising distance based on using battery power alone, and utilizing fuel cells, energy harvesting (e.g., by converting the vibration energy in the suspension or the energy in the brakes into electrical energy) or sustainable energy resources, such as solar and wind power, to charge the batteries. The smart plug-in vehicle can be a part of an integrated smart home and grid energy system of the future, which would utilize smart energy metering devices for optimal use of grid energy by avoiding peak energy consumption hours.

High-Performance Real-Time Control, Health Monitoring, and Diagnosis

Modern vehicles utilize an increasing number of sensors, actuators, and networked embedded computers. The need for high-performance computing would increase with the introduction of such revolutionary features as drive-by-wire systems into modern vehicles. The tremendous computational burden of processing sensory data into appropriate control and monitoring signals and diagnostic information creates challenges in the design of embedded computing technology. Toward this end, a related challenge is to incorporate sophisticated computational techniques that control, monitor, and diagnose complex automotive systems while meeting requirements such as low power consumption and cost-effectiveness.

1-2-2 Steering Control of an Automobile

As a simple example of the control system, as shown in Fig. 1-1, consider the steering control of an automobile. The direction of the two front wheels can be regarded as the controlled variable, or the output, y; the direction of the steering wheel is the actuating signal, or the input, u. The control system, or process in this case, is composed of the steering mechanism and the dynamics of the entire automobile. However, if the objective is to control the speed of the automobile, then the amount of pressure exerted on the accelerator is the actuating signal, and the vehicle speed is the controlled variable. As a whole, we can regard the simplified automobile control system as one with two inputs (steering and accelerator) and two outputs (heading and speed). In this case, the two controls and two outputs are independent of each other, but there are systems for which the controls are coupled. Systems with more than one input and one output are called **multivariable systems**.

1-2-3 Idle-Speed Control of an Automobile

As another example of a control system, we consider the idle-speed control of an automobile engine. The objective of such a control system is to maintain the engine idle speed at a relatively low value (for fuel economy) regardless of the applied engine loads (e.g., transmission, power steering, air conditioning). Without the idle-speed control, any sudden engine-load application would cause a drop in engine speed that might cause the engine to stall. Thus the main objectives of the idle-speed control system are (1) to eliminate or minimize the speed droop when engine loading is applied and (2) to maintain the engine idle speed at a desired value. Figure 1-2 shows the block diagram of the idle-speed control system from the standpoint of inputs–system–outputs. In this case, the throttle angle α and the load torque T_L (due to the application of air conditioning, power steering, transmission, or power brakes, etc.) are the inputs, and the engine speed ω is the output. The engine is the controlled process of the system.

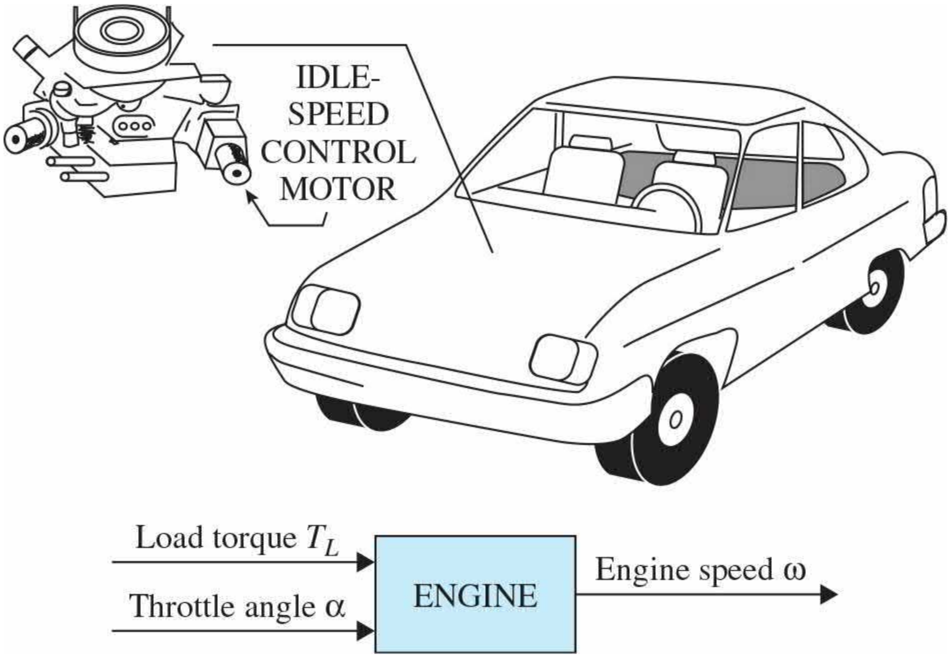


Figure 1-2 Idle-speed control system.

1-2-4 Sun-Tracking Control of Solar Collectors

To achieve the goal of developing economically feasible non-fossil-fuel electrical power, a great deal of effort has been placed on alternative energy including research and development of solar power conversion methods, including the solar-cell conversion techniques. In most of these systems, the need for high efficiencies dictates the use of devices for sun tracking. Figure 1-3 shows a solar collector field. Figure 1-4 shows a conceptual method of efficient water extraction using solar power. During the hours of daylight, the solar collector would produce electricity to pump water from the underground water table to a reservoir (perhaps on a nearby mountain or hill), and in the early morning hours, the water would be released into the irrigation system.

Figure 1-3 Solar collector field.[1]

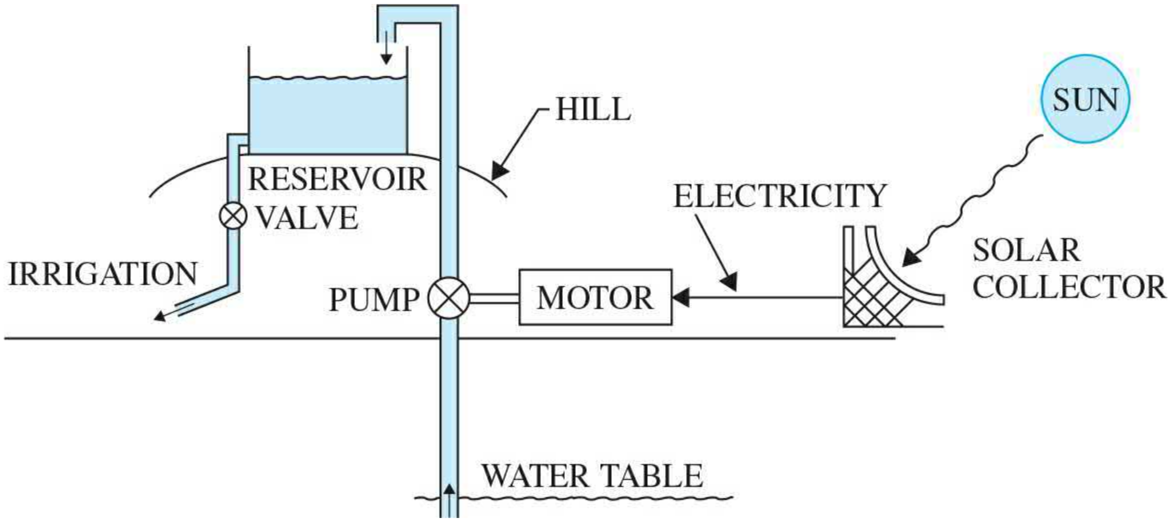


Figure 1-4 Conceptual method of efficient water extraction using solar power.

One of the most important features of the solar collector is that the collector dish must track the sun accurately. Therefore, the movement of the collector dish must be controlled by sophisticated control systems. The block diagram of Fig. 1-5 describes the general philosophy of the sun-tracking system together with some of the most important components. The controller ensures that the tracking collector is pointed toward the sun in the morning and sends a “start track” command. The controller constantly calculates the sun’s rate for the two axes (azimuth and elevation) of control during the day. The controller uses the sun rate and sun sensor information as inputs to generate proper motor commands to slew the collector.

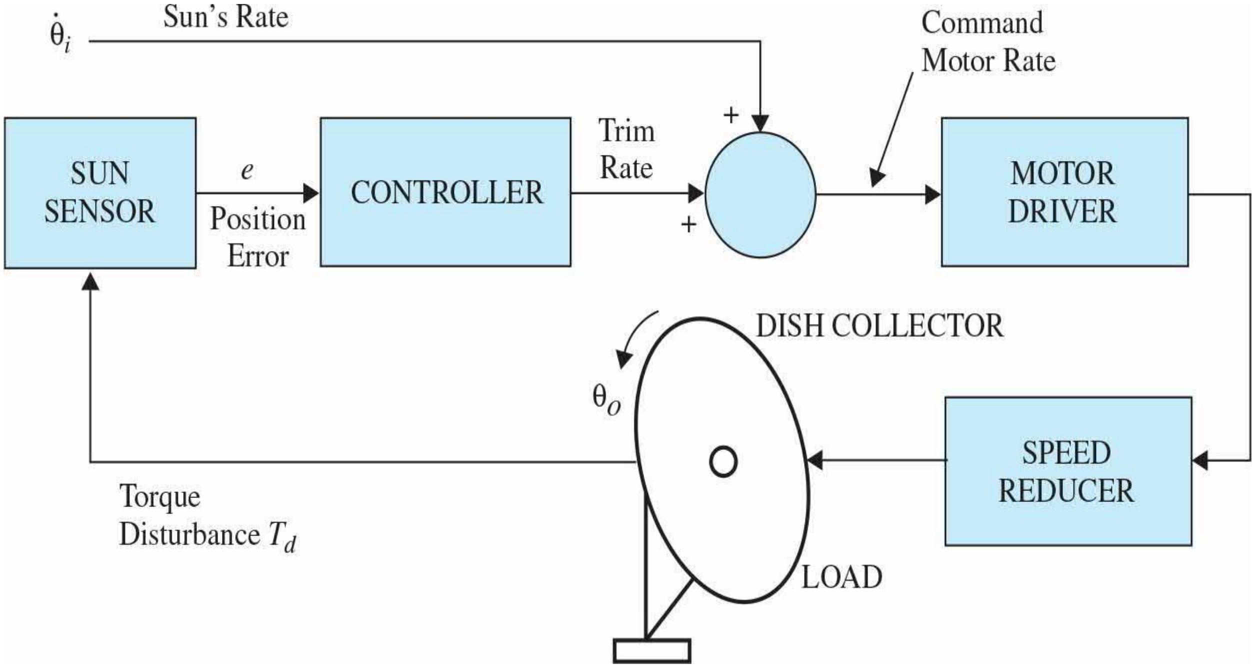


Figure 1-5 Important components of the sun-tracking control system.

1-3 OPEN-LOOP CONTROL SYSTEMS (NONFEEDBACK SYSTEMS)

The idle-speed control system illustrated in Fig. 1-2, shown previously, is rather unsophisticated and is called an **open-loop control system**. It is not difficult to see that the system as shown would not satisfactorily fulfill critical performance requirements. For instance, if the throttle angle α is set at a certain initial value that corresponds to a certain engine speed, then when a load torque T_L is applied, there is no way to prevent a drop in the engine speed. The only way to make the system work is to have a means of adjusting α in response to a change in the load torque in order to maintain ω at the desired level. The conventional electric washing machine is another example of an open-loop control system because, typically, the amount of machine wash time is entirely determined by the judgment and estimation of the human operator.

Open-loop systems are economical but usually inaccurate.

The elements of an open-loop control system can usually be divided into two parts: the **controller** and the **controlled process**, as shown by the block diagram of Fig. 1-6. An input signal, or command, *r*, is applied to the controller, whose output acts as the actuating signal *u*; the actuating signal then controls the controlled process so that the controlled variable *y* will perform according to some prescribed standards. In simple cases, the controller can be an amplifier, a mechanical linkage, a filter, or other control elements, depending on the nature of the system. In more sophisticated cases, the controller can be a computer such as a microprocessor. Because of the simplicity and economy of open-loop control systems, we find this type of system in many noncritical applications.

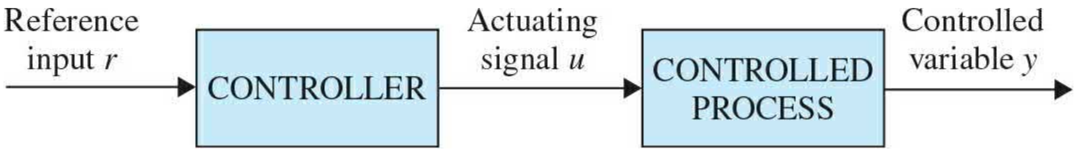


Figure 1-6 Elements of an open-loop control system.

1-4 CLOSED-LOOP CONTROL SYSTEMS (FEEDBACK CONTROL SYSTEMS)

What is missing in the open-loop control system for more accurate and more adaptive control is a link or feedback from the output to the input of the system. To obtain more accurate control, the controlled signal *y* should be fed back and compared with the reference input, and an actuating signal proportional to the difference of the input and the output must be sent through the system to correct the error. A system with one or more feedback paths such as that just described is called a **closed-loop system**.

Closed-loop systems have many advantages over open-loop systems.

A closed-loop idle-speed control system is shown in Fig. 1-7. The reference input ω_r sets the desired idling speed. The engine speed at idle should agree with the reference value ω_r, and any difference such as the load torque T_L is sensed by the speed transducer and the error detector. The controller will operate on the difference and provide a signal to adjust the throttle angle α to correct the error. Figure 1-8 compares the typical performances of open-loop and closed-loop idle-speed control systems. In Fig. 1-8*a*, the idle speed of the open-loop system will drop and settle at a lower value after a load torque is applied. In Fig. 1-8*b*, the idle speed of the closed-loop system is shown to recover quickly to the preset value after the application of T_L.

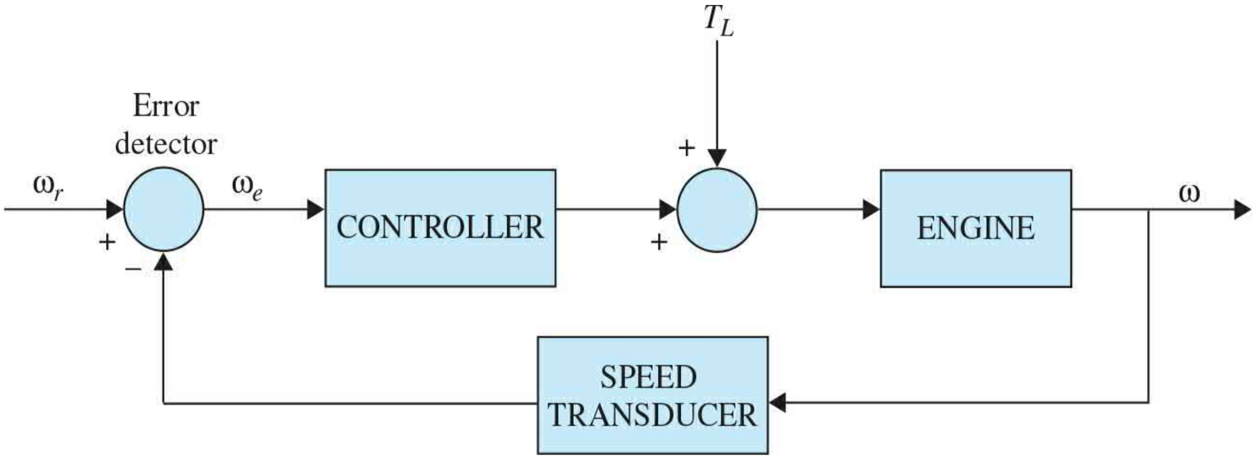


Figure 1-7 Block diagram of a closed-loop idle-speed control system.

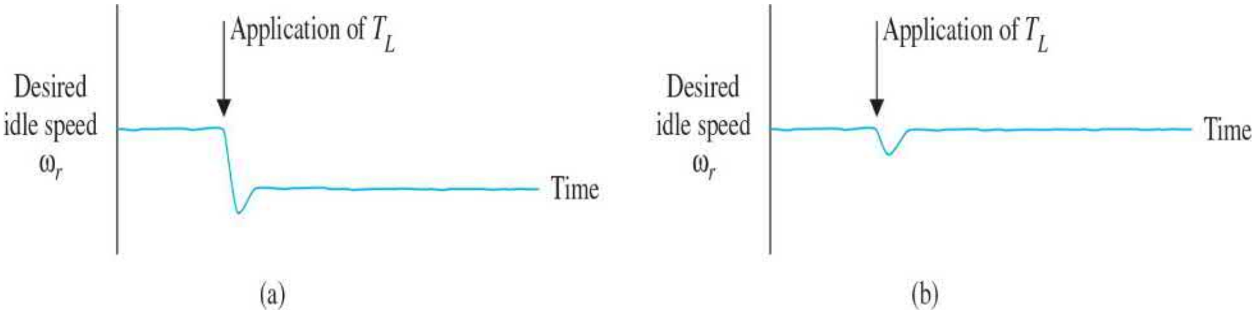


Figure 1-8 (*a*) Typical response of the open-loop idle-speed control system. (*b*) Typical response of the closed-loop idle-speed control system.

The objective of the idle-speed control system illustrated, also known as a **regulator system**, is to maintain the system output at a prescribed level.

1-5 WHAT IS FEEDBACK, AND WHAT ARE ITS EFFECTS?

The motivation for using feedback, as illustrated by the examples in Sec. 1-1, is somewhat oversimplified. In these examples, feedback is used to reduce the error between the reference input and the system output. However, the significance of the effects of feedback in control systems is more complex than is demonstrated by these simple examples. The reduction of system error is merely one of the many important effects that feedback may have upon a system. We show in the following sections that feedback also has effects on such system performance characteristics as **stability**, **bandwidth**, **overall gain**, **impedance**, and **sensitivity**.

> Feedback exists whenever there is a closed sequence of cause-and-effect relationships.

To understand the effects of feedback on a control system, it is essential to examine this phenomenon in a broad sense. When feedback is deliberately introduced for the purpose of control, its existence is easily identified. However, there are numerous situations where a physical system that we recognize as an inherently nonfeedback system turns out to have feedback when it is observed in a certain manner. In general, we can state that whenever a closed sequence of **cause-and-effect relationships** exists among the variables of a system, feedback is said to exist. This viewpoint will inevitably admit feedback in a large number of systems that ordinarily would be identified as nonfeedback systems. However, control-system theory allows numerous systems, with or without physical feedback, to be studied in a systematic way once the existence of feedback in the sense mentioned previously is established.

We shall now investigate the effects of feedback on the various aspects of system performance. Without the necessary mathematical foundation of linear-system theory, at this point we can rely only on simple static-system notation for our discussion. Let us consider the simple feedback system configuration shown in Fig. 1-9, where *r* is the input signal; *y*, the output signal; *e*, the error; and *b*, the feedback signal. The parameters *G* and *H* may be considered as constant gains. By simple algebraic manipulations, it is simple to show that the input–output relation of the system is

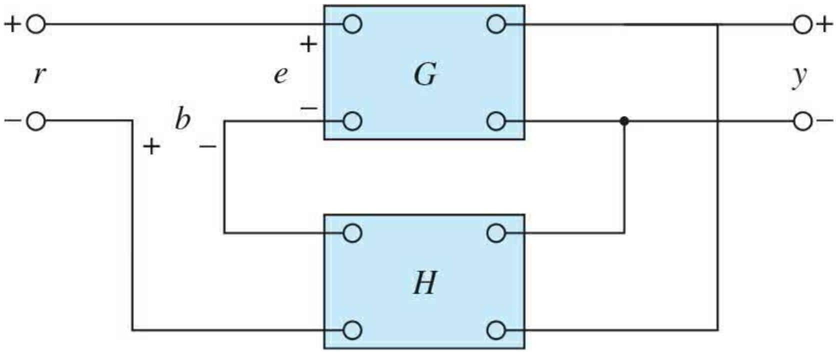


Figure 1-9 Feedback system.

$$M = \frac{y}{r} = \frac{G}{1+GH} \tag{1-1}$$

Using this basic relationship of the feedback system structure, we can uncover some of the significant effects of feedback.

1-5-1 Effect of Feedback on Overall Gain

As seen from Eq. (1-1), feedback affects the gain *G* of a nonfeedback system by a factor of 1 + *GH*. The system of Fig. 1-9 is said to have **negative feedback** because a minus sign is assigned to the feedback signal. The quantity *GH* may itself include a minus sign, so the *general effect of feedback is that it may increase or decrease the gain* G. In a practical control system, *G* and *H* are functions of frequency, so the magnitude of 1 + *GH* may be greater than 1 in one frequency range but less than 1 in another. Therefore, *feedback could increase the gain of system in one frequency range but decrease it in another*.

Feedback may increase the gain of a system in one frequency range but decrease it in another.

1-5-2 Effect of Feedback on Stability

Stability is a notion that describes whether the system will be able to follow the input command, that is, be useful in general. In a nonrigorous manner, *a system is said to be unstable if its output is out of control*. To investigate the effect of feedback on stability, we can again refer to the expression in Eq. (1-1). If $GH = -1$, the output of the system is infinite for any finite input, and the system is said to be unstable. Therefore, we may state that *feedback can cause a system that is originally stable to become unstable*. Certainly, feedback is a double-edged sword; when it is improperly used, it can be harmful. It should be pointed out, however, that we are only dealing with the static case here, and, in general, $GH = -1$ is not the only condition for instability. The subject of system stability will be treated formally in Chap. 5.

A system is unstable if its output is out of control.

It can be demonstrated that one of the advantages of incorporating feedback is that it can stabilize an unstable system. Let us assume that the feedback system in Fig. 1-9 is unstable because $GH = -1$. If we introduce another feedback loop through a negative feedback gain of F, as shown in Fig. 1-10, the input–output relation of the overall system is

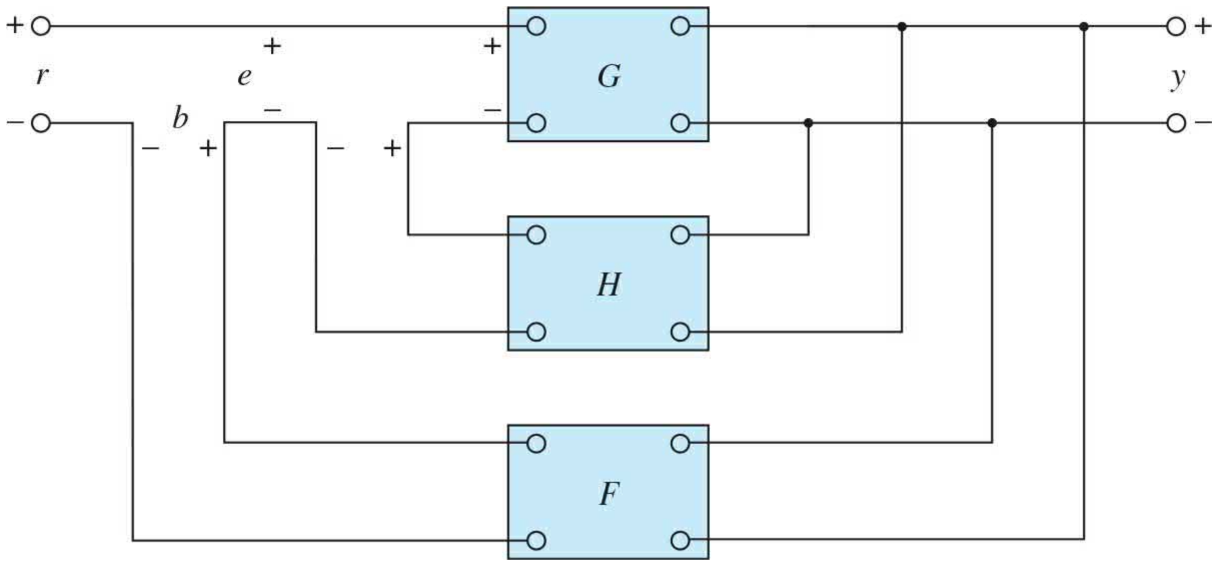


Figure 1-10 Feedback system with two feedback loops.

$$\frac{y}{r}=\frac{G}{1+GH+GF} \tag{1-2}$$

It is apparent that although the properties of G and H are such that the inner-loop feedback system is unstable because $GH = -1$, the overall system can be stable by properly selecting the outer-loop feedback gain F. In practice, GH is a function of frequency, and the stability condition of the closed-loop system depends on the **magnitude** and **phase** of GH. The bottom line is that *feedback can improve stability or be harmful to stability if it is not properly applied*.

Feedback can improve stability or be harmful to stability.

Sensitivity considerations often are important in the design of control systems. Because all physical elements have properties that change with environment and age, we cannot always consider the parameters of a control system to be completely stationary over the entire operating life of the system. For instance, the winding resistance of an electric motor changes as the temperature of the motor rises during operation. Control systems with electric components may not operate normally when first turned on because of the still-changing system parameters during warm-up. This phenomenon is sometimes called **morning sickness**. Most duplicating machines have a warm-up period during which time operation is blocked out when first turned on.

Note: Feedback can increase or decrease the sensitivity of a system.

In general, a good control system should be very insensitive to parameter variations but sensitive to the input commands. We shall investigate what effect feedback has on sensitivity to parameter variations. Referring to the system in Fig. 1-9, we consider G to be a gain parameter that may vary. The sensitivity of the gain of the overall system M to the variation in G is defined as

$$S_G^M = \frac{\partial M/G}{\partial G/M} = \frac{\text{percentage change in } M}{\text{percentage change in } G} \tag{1-3}$$

where ∂M denotes the incremental change in M due to the incremental change in G, or ∂G. By using Eq. (1-1), the sensitivity function is written

$$S_G^M = \frac{\partial M}{\partial G}\frac{G}{M} = \frac{1}{1+GH} \tag{1-4}$$

This relation shows that if GH is a positive constant, the magnitude of the sensitivity function can be made arbitrarily small by increasing GH, provided that the system remains stable. It is apparent that, in an open-loop system, the gain of the system will respond in a one-to-one fashion to the variation in G (i.e., $S_G^M = 1$). Again, in practice, GH is a function of frequency; the magnitude of $1 + GH$ may be less than unity over some frequency ranges, so feedback could be harmful to the sensitivity to parameter variations in certain cases. In general, the sensitivity of the system gain of a feedback system to parameter variations depends on where the parameter is located. The reader can derive the sensitivity of the system in Fig. 1-9 due to the variation of H.

1-5-3 Effect of Feedback on External Disturbance or Noise

All physical systems are subject to some types of extraneous signals or noise during operation. Examples of these signals are thermal-noise voltage in electronic circuits and brush or commutator noise in electric motors. External disturbances, such as wind gusts acting on an antenna, are also quite common in control systems. Therefore, control systems should be designed so that they are insensitive to noise and disturbances and sensitive to input commands.

Feedback can reduce the effect of noise.

The effect of feedback on noise and disturbance depends greatly on where these extraneous signals occur in the system. No general conclusions can be reached, but in many situations, *feedback can reduce the effect of noise and disturbance on system performance*. Let us refer to the system shown in Fig. 1-11, in which r denotes the command signal and n is the noise signal. In the absence of feedback, that is, $H = 0$, the output y due to n acting alone is

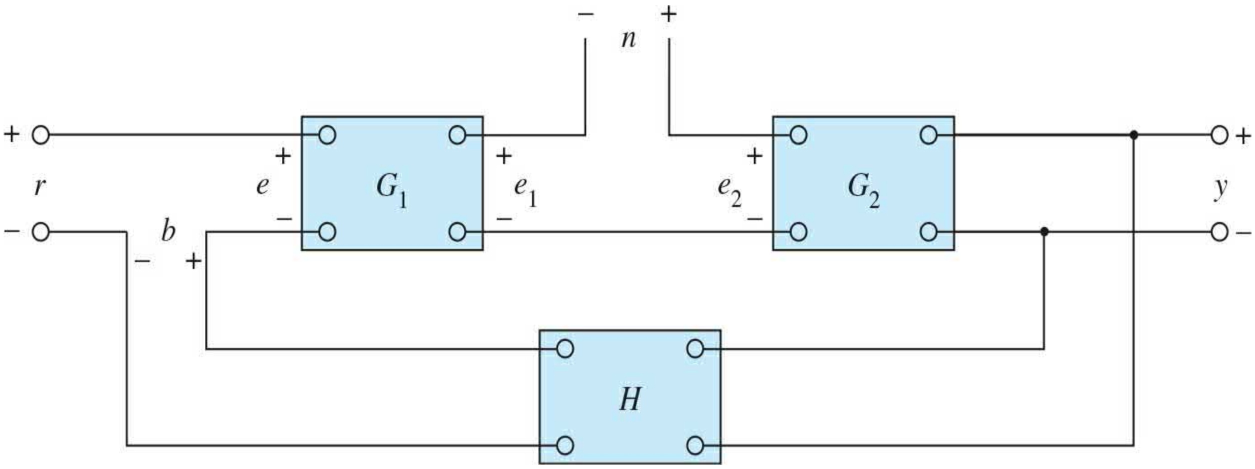


Figure 1-11 Feedback system with a noise signal.

$$y = G_2 n \tag{1-5}$$

With the presence of feedback, the system output due to n acting alone is

$$y = \frac{G_2}{1 + G_1 G_2 H} n \tag{1-6}$$

Comparing Eq. (1-6) with Eq. (1-5) shows that the noise component in the output of Eq. (1-6) is reduced by the factor $1 + G_1 G_2 H$ if the latter is greater than unity and the system is kept stable.

In Chap. 11, the feedforward and forward controller configurations are used along with feedback to reduce the effects of disturbance and noise inputs. In general, feedback also has effects on such performance characteristics as bandwidth, impedance, transient response, and frequency response. These effects will be explained as we continue.

Feedback also can affect bandwidth, impedance, transient responses, and frequency responses.

1-6 TYPES OF FEEDBACK CONTROL SYSTEMS

Feedback control systems may be classified in a number of ways, depending upon the purpose of the classification. For instance, according to the method of analysis and design, control systems are classified as **linear** or **nonlinear**, and **time-varying** or **time-invariant**. According to the types of signal found in the system, reference is often made to **continuous-data** or **discrete-data** systems, and **modulated** or **unmodulated** systems. Control systems are often classified according to the main purpose of the system. For instance, a **position-control system** and a **velocity-control system** control the output variables just as the names imply. In Chap. 11, the **type** of control system is defined according to the form of the open-loop transfer function. In general, there are many other ways of identifying control systems according to some special features of the system. It is important to know some of the more common ways of classifying control systems before embarking on the analysis and design of these systems.

Most real-life control systems have nonlinear characteristics to some extent.

1-7 LINEAR VERSUS NONLINEAR CONTROL SYSTEMS

This classification is made according to the methods of analysis and design. Strictly speaking, linear systems do not exist in practice because all physical systems are nonlinear to some extent. Linear feedback control systems are idealized models fabricated by the analyst purely for the simplicity of analysis and design. When the magnitudes of signals in a control system are limited to ranges in which system components exhibit linear characteristics (i.e., the principle of superposition applies), the system is essentially linear. But when the magnitudes of signals are extended beyond the range of the linear operation, depending on the severity of the nonlinearity, the system should no longer be considered linear. For instance, amplifiers used in control systems often exhibit a saturation effect when their input signals become large; the magnetic field of a motor usually has saturation properties. Other common nonlinear effects found in control systems are the backlash or dead play between coupled gear members, nonlinear spring characteristics, nonlinear friction force or torque between moving members, and so on. Quite often, nonlinear characteristics are intentionally introduced in a control system to improve its performance or provide more effective control. For instance, to achieve minimum-time control, an on-off–type (bang-bang or relay) controller is used in many missile or spacecraft control systems. Typically in these systems, jets are mounted on the sides of the vehicle to provide reaction torque for attitude control. These jets are often controlled in a full-on or full-off fashion, so a fixed amount of air is applied from a given jet for a certain time period to control the attitude of the space vehicle.

> There are no general methods for solving a wide class of nonlinear systems.

For linear systems, a wealth of analytical and graphical techniques is available for design and analysis purposes. A majority of the material in this text is devoted to the analysis and design of linear systems. Nonlinear systems, on the other hand, are usually difficult to treat mathematically, and there are no general methods available for solving a wide class of nonlinear systems. It is practical to first design the controller based on the linear-system model by neglecting the nonlinearities of the system. The designed controller is then applied to the nonlinear system model for evaluation or redesign by computer simulation. The Control Lab introduced in Chap. 8 may be used to model the characteristics of practical systems with realistic physical components.

1-8 TIME-INVARIANT VERSUS TIME-VARYING SYSTEMS

When the parameters of a control system are stationary with respect to time during the operation of the system, the system is called a time-invariant system. In practice, most physical systems contain elements that drift or vary with time. For example, the winding resistance of an electric motor will vary when the motor is first being excited and its temperature is rising. Another example of a time-varying system is a guided-missile control system in which the mass of the missile decreases as the fuel on board is being consumed during flight. Although a time-varying system without nonlinearity is still a linear system, the analysis and design of this class of systems are usually much more complex than that of the linear time-invariant systems.

1-9 CONTINUOUS-DATA CONTROL SYSTEMS

A continuous-data system is one in which the signals at various parts of the system are all functions of the continuous time variable *t*. The signals in continuous-data systems may be further classified as ac or dc. Unlike the general definitions of ac and dc signals used in electrical engineering, ac and dc control systems carry special significance in control systems terminology. When one refers to an **ac control system**, it usually means that the signals in the system are *modulated* by some form of modulation scheme. A **dc control system**, on the other hand, simply implies that the signals are *unmodulated*, but they are still ac signals according to the conventional definition. The schematic diagram of a closed-loop dc control system is shown in Fig. 1-12. Typical waveforms of the signals in response to a step-function input are shown in the figure. Typical components of a dc control system are potentiometers, dc amplifiers, dc motors, dc tachometers, and so on.

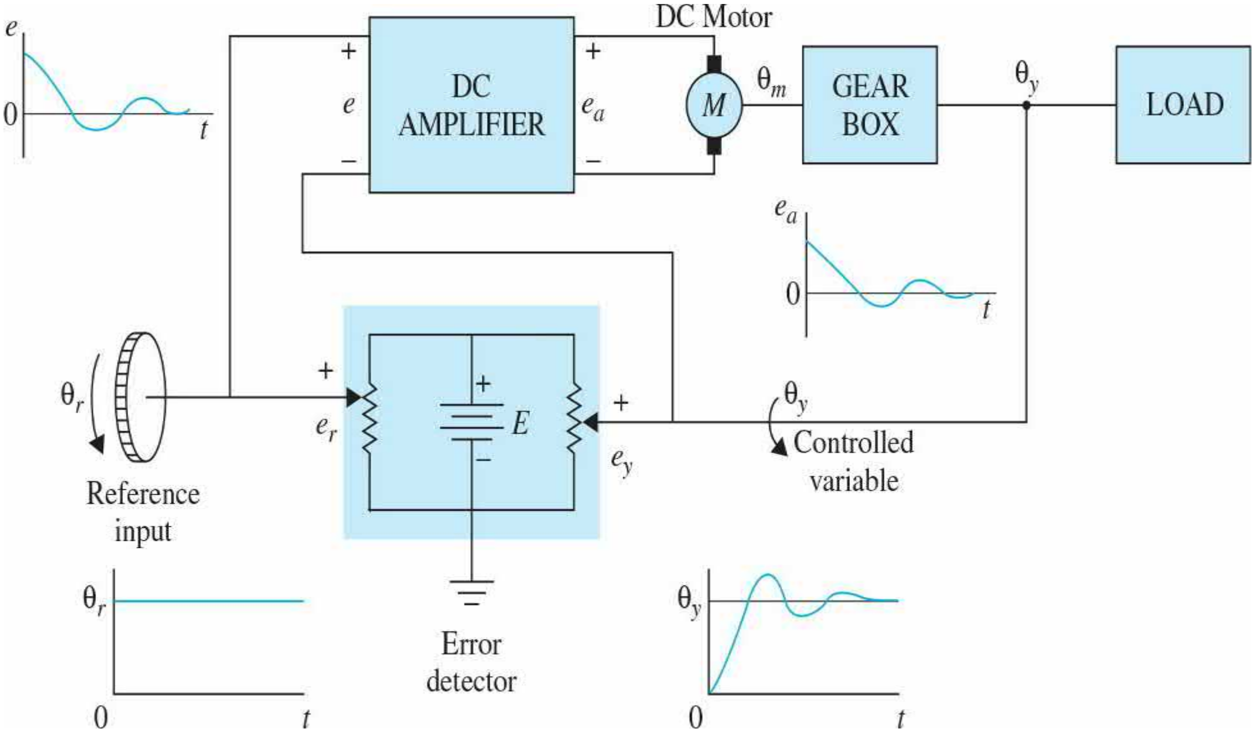


Figure 1-12 Schematic diagram of a typical dc closed-loop system.

Figure 1-13 shows the schematic diagram of a typical ac control system that performs essentially the same task as the dc system in Fig. 1-12. In this case, the signals in the system are modulated; that is, the information is transmitted by an ac carrier signal. Notice that the output controlled variable still behaves similarly to that of the dc system. In this case, the modulated signals are demodulated by the low-pass characteristics of the ac motor. Ac control systems are used extensively in aircraft and missile control systems in which noise and disturbance often create problems. By using modulated ac control systems with carrier frequencies of 400 Hz or higher, the system will be less susceptible to low-frequency noise. Typical components of an ac control system are synchros, ac amplifiers, ac motors, gyroscopes, accelerometers, and so on.

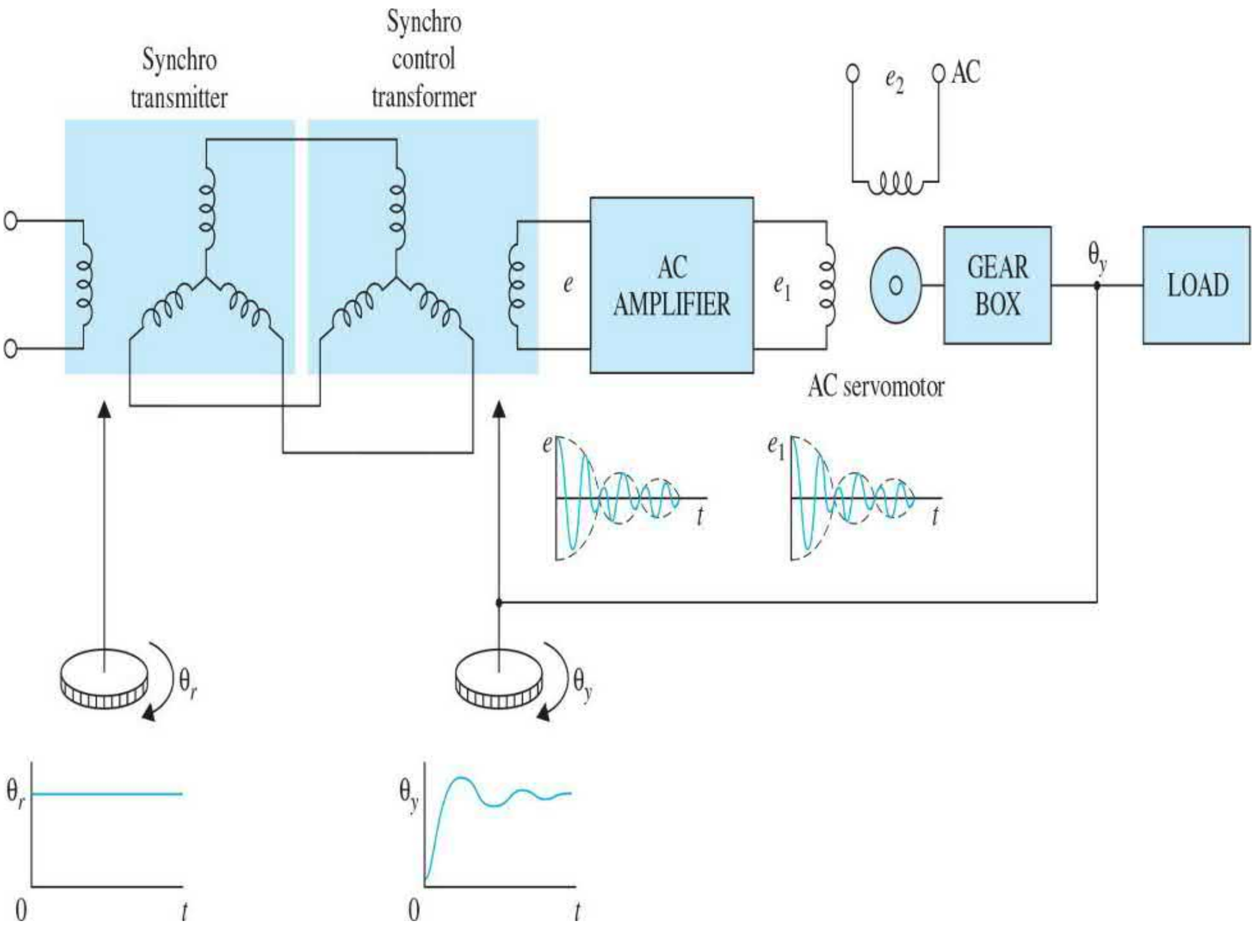


Figure 1-13 Schematic diagram of a typical ac closed-loop control system.

In practice, not all control systems are strictly of the ac or dc type. A system may incorporate a mixture of ac and dc components, using modulators and demodulators to match the signals at various points in the system.

1-10 DISCRETE-DATA CONTROL SYSTEMS

Discrete-data control systems differ from the continuous-data systems in that the signals at one or more points of the system are in the form of either a pulse train or a digital code. Usually, discrete-data control systems are subdivided into **sampled-data** and **digital control systems**. Sampled-data control systems refer to a more general class of discrete-data systems in which the signals are in the form of pulse data. A digital control system refers to the use of a digital computer or controller in the system so that the signals are digitally coded, such as in binary code.

Digital control systems are usually less susceptible to noise.

In general, a sampled-data system receives data or information only intermittently at specific instants of time. For example, the error signal in a control system can be supplied only in the form of pulses, in which case the control system receives no information about the error signal during the periods between two consecutive pulses. Strictly, a sampled-data system can also be classified as an ac system because the signal of the system is pulse modulated.

Figure 1-14 illustrates how a typical sampled-data system operates. A continuous-data input signal $r(t)$ is applied to the system. The error signal $e(t)$ is sampled by a sampling device, the **sampler**, and the output of the sampler is a sequence of pulses. The sampling rate of the sampler may or may not be uniform. There are many advantages to incorporating sampling into a control system. One important advantage is that expensive equipment used in the system may be time-shared among several control channels. Another advantage is that pulse data are usually less susceptible to noise.

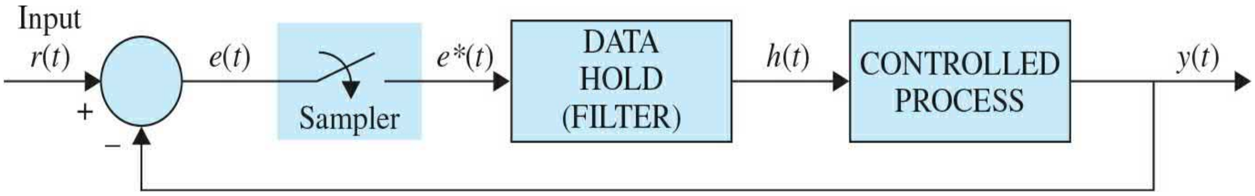


Figure 1-14 Block diagram of a sampled-data control system.

Because digital computers provide many advantages in size and flexibility, computer control has become increasingly popular in recent years. Many airborne systems contain digital controllers that can pack thousands of discrete elements into a space no larger than the size of this book. Figure 1-15 shows the basic elements of a digital autopilot for aircraft attitude control.

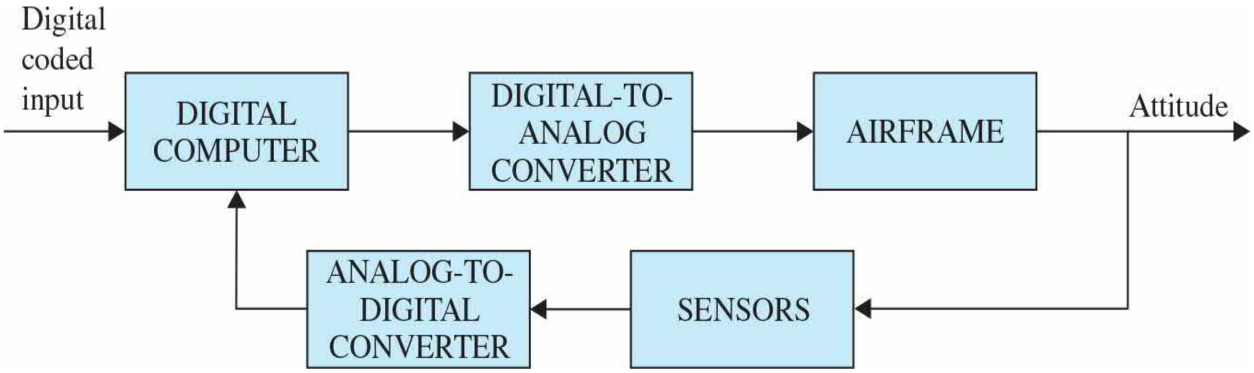


Figure 1-15 Digital autopilot system for aircraft attitude control.

1-11 CASE STUDY: INTELLIGENT VEHICLE OBSTACLE AVOIDANCE—LEGO MINDSTORMS

The goal of this section is to provide you with a better understanding of the controller design process for a practical system—in this case a **LEGO® MINDSTORMS®** NXT programmable robotic system. Note the example used here may appear too difficult at this stage but it can demonstrate the steps you need to take for successful implementation of a control system. You may revisit this example after successful completion of App. D.

Description of the Project[2]

The system setup, shown in Fig. 1-16, is a LEGO MINDSTORMS car that is controlled using MATLAB® and Simulink®. The LEGO car, shown in Figs. 1-17 and 1-18, is equipped with an ultrasonic sensor, a light sensor, an indicating light, an NXT motor gearbox and the NXT brick. An encoder (sensor) is used to read the angular position of the motor gearbox. The NXT brick can take input from up to four sensors and control up to three motors, via RJ12 cables—see Chap. 8 for more details. The ultrasonic sensor is placed in the front to detect the distance from the obstacle. The light sensor is facing downward to detect the color of the running surface—in this case white means go! The system interfaces with the host computer using a USB connection, while the host computer logs encoder data in real-time using a Bluetooth connection.

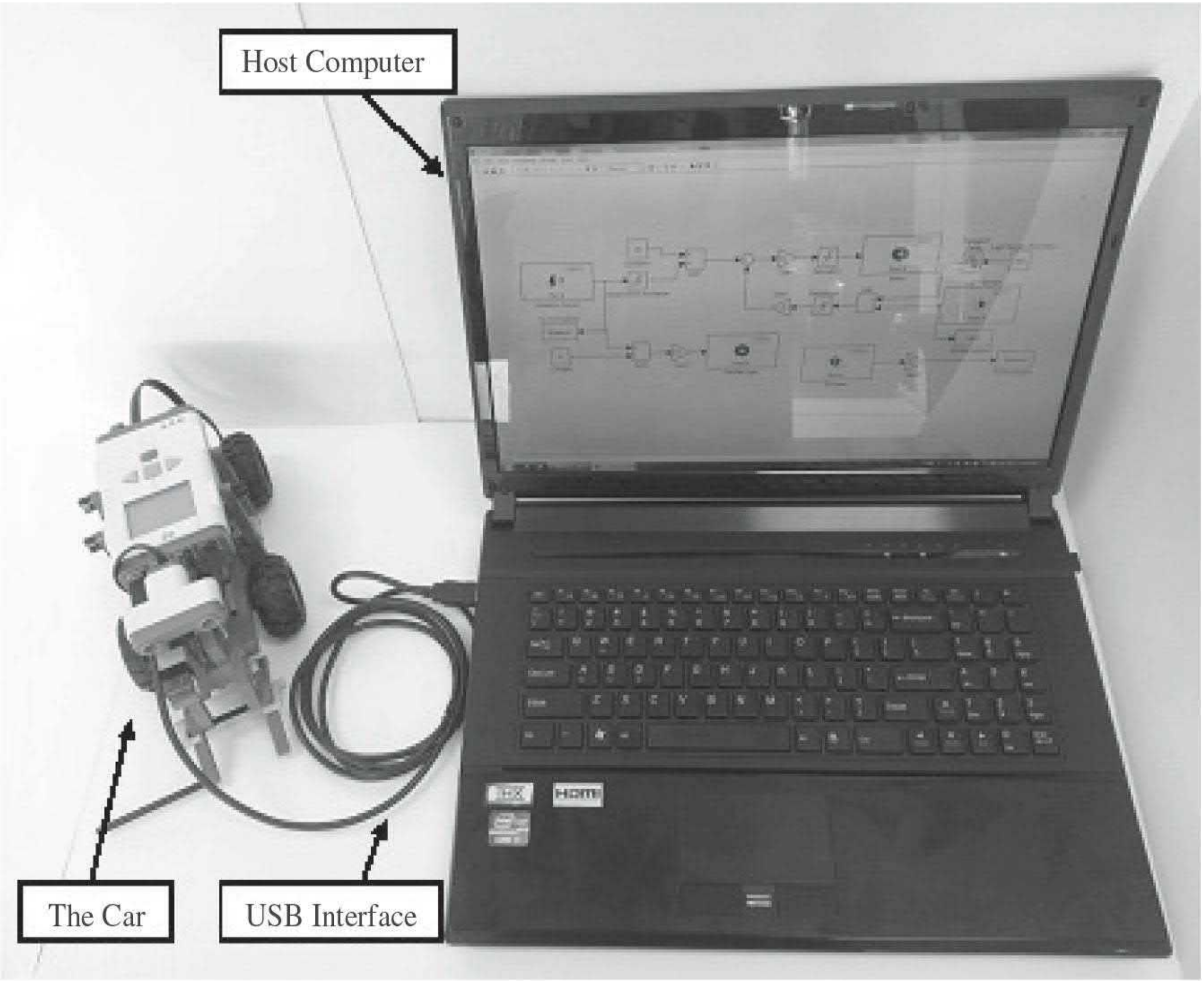


Figure 1-16 Final car product with host computer.

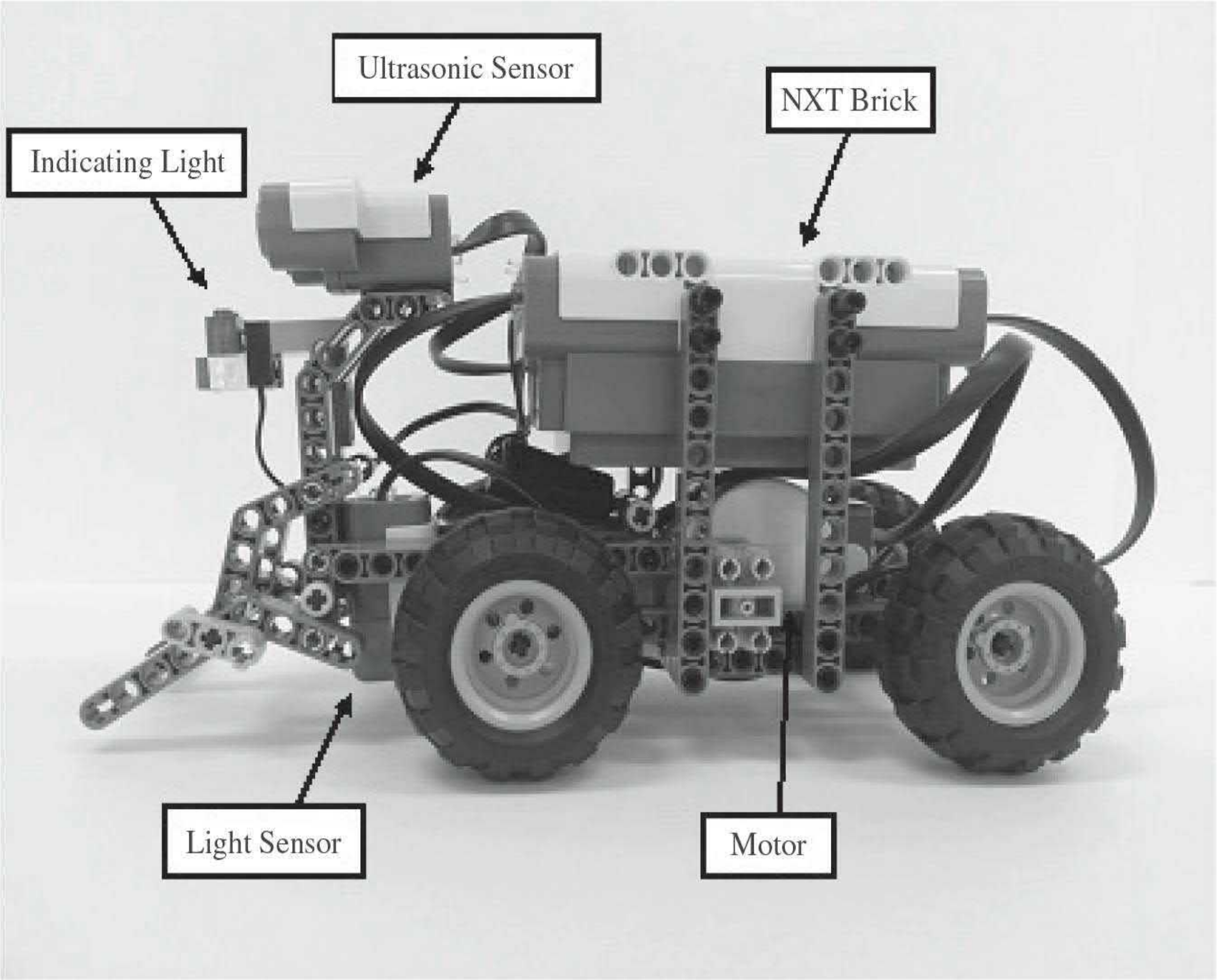


Figure 1-17 Car design—side view.

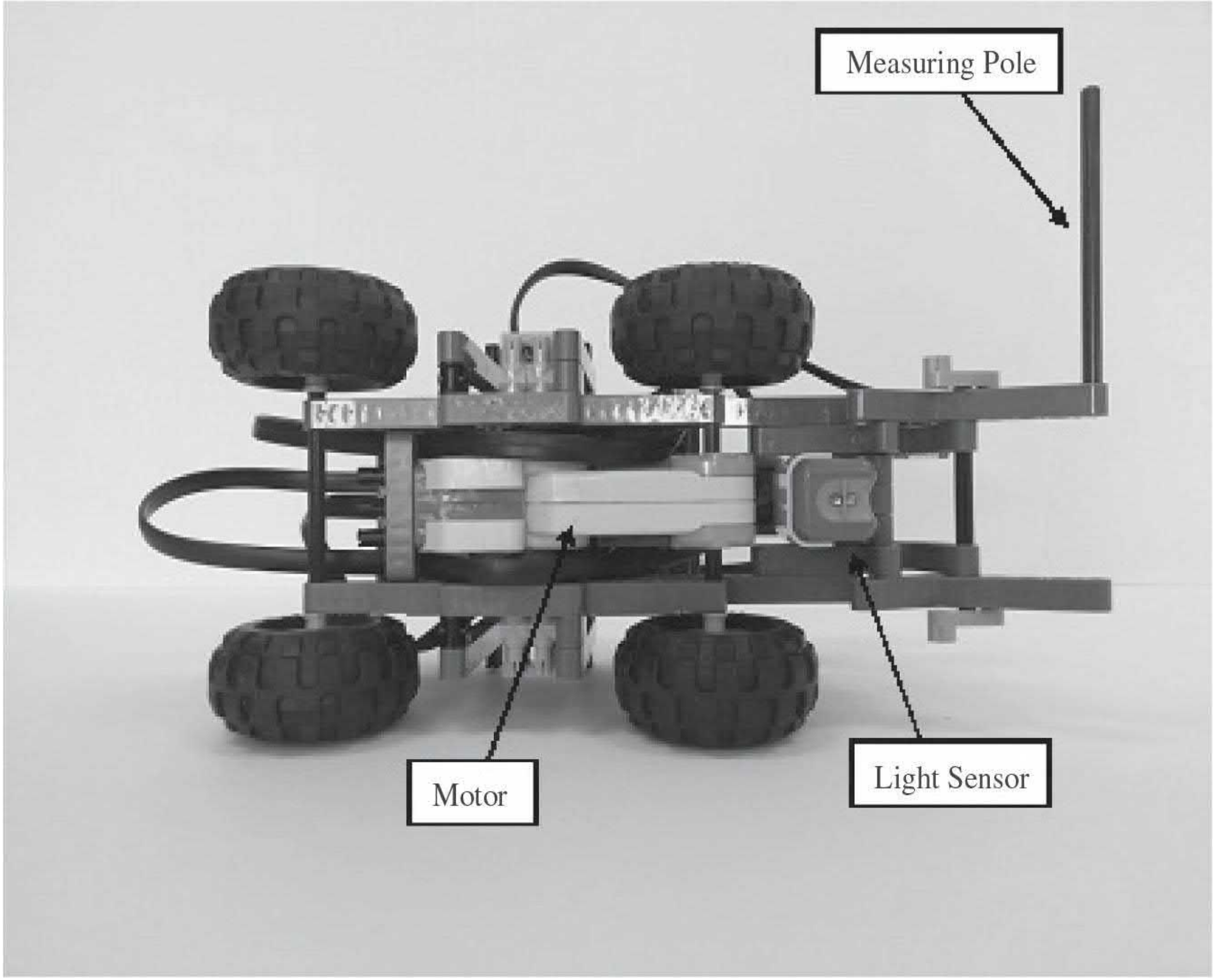


Figure 1-18 Car design—bottom view.

The Controller Design Procedure

The design of a control system for a practical problem requires a systematic treatment as follows:

- Outline the objectives of the control system.
- Specify the requirements, design criteria, and constraints (Chaps. 7 and 11).
- Develop a mathematical model of the system, including mechanical, electrical, sensors, motor, and the gearbox (Chaps. 2, 3, and 6).
- Establish how the overall system subcomponents interact, utilizing block diagrams (Chap. 4).
- Use block diagrams, signal flow graphs, or state diagrams to find the

model of the overall system—transfer function or state space model (Chap. 4).

- Study the transfer function of the system in the Laplace domain, or the state space representation of the system (Chap. 3).
- Understand the time and frequency response characteristics of the system and whether it is stable or not (Chaps. 5, 7, and 9 to 11).
- Design a controller using time response (Chaps. 7 and 11).
- Design a controller using the root locus technique (in the Laplace domain) and time response (Chaps. 7, 9, and 11).
- Design a controller using frequency response techniques (Chaps. 10 and 11).
- Design a controller using the state space approach (Chap. 8).
- Optimize the controller if necessary (Chap. 11).
- Implement the design on the experimental/practical system (Chaps. 7 and 11 and App. D).

Objective

The objective of this project is to have the LEGO car running on a white surface and stop just before hitting an obstacle—in this case a wall.

Design Criteria and Constraints

The car can only be running at full speed on a white surface. The car must stop, if the surface color is not white. The car must also stop just before hitting an obstacle.

Develop a Mathematical Model of the System

The motor drives the rear wheels. The vehicle mass, motor, gearbox, and wheel friction must be considered in the modeling process. You may use Chaps. 2 and 6 to arrive at the mathematical model of the system. Also check Sec. 7-5.

Following the process in Chaps. 6 and 7 and App. D, **the block diagram** of the system using position control (using an amplifier with gain K) and the encoder sensor position feedback is shown in Fig. 1-19, where system parameters and variables, in time domain, include

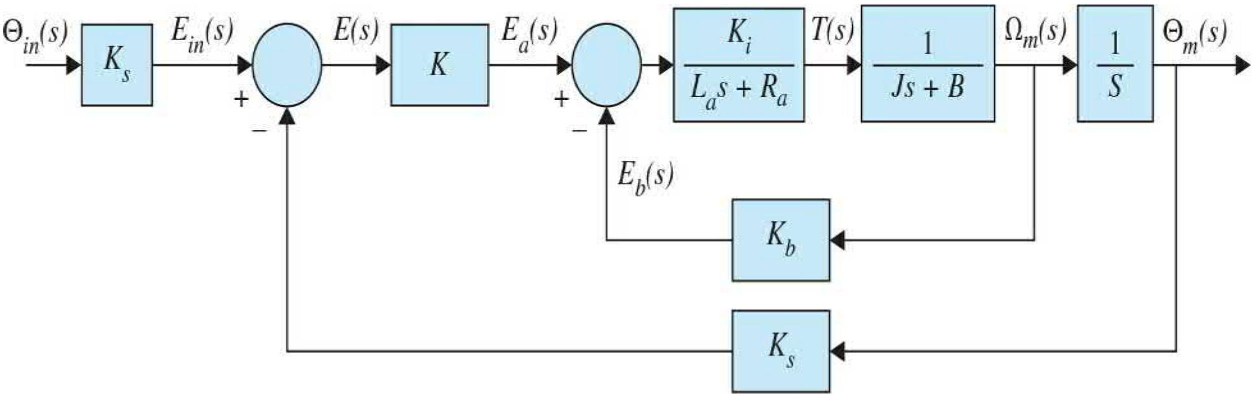


Figure 1-19 Block diagram of a position-control, armature-controlled dc motor representing the LEGO car.

R_a = armature resistance, Ω

L_a = armature inductance, H

θ_m = angular displacement of the motor gearbox shaft, radian

θ_{in} = desired angular displacement of the motor gearbox shaft, radian

ω_m = angular speed of the motor shaft, rad/s

T = torque developed by the motor, N · m

J = equivalent moment of inertia of the motor and load connected to the motor shaft, $J = J_L/n^2 + J_m$, $kg - m^2$ (refer to Chap. 2 for more details)

n = gear ratio

B = equivalent viscous-friction coefficient of the motor and load referred to the motor shaft, N · m/rad/s (in the presence of gear ratio, B must be scaled by n; refer to Chap. 2 for more details)

K_i = torque constant, N · m/A

K_b = back-emf constant, V/rad/s

K_s = equivalent encoder sensor gain, V/rad

K = position control gain (amplifier)

The closed-loop **transfer function**, in Laplace domain, in this case becomes

$$\frac{\Theta_m(s)}{\Theta_{in}(s)}=\frac{\dfrac{KK_iK_s}{R_a}}{(\tau_e s+1)\left\{Js^2+\left(B_m+\dfrac{K_bK_i}{R_a}\right)s+\dfrac{KK_iK_s}{R_a}\right\}} \tag{1-7}$$

where K_s is the sensor gain. The motor electrical time constant $\tau_e = L_a/R_a$ may be neglected for small L_a. As a result the position transfer function is simplified to

$$\frac{\Theta_m(s)}{\Theta_{in}(s)}=\frac{\dfrac{KK_iK_s}{R_aJ}}{s^2+\left(\dfrac{R_aB_m+K_iK_b}{R_aJ}\right)s+\dfrac{KK_iK_s}{R_aJ}}=\frac{\omega_n^2}{\left(s^2+2\zeta\omega_n s+\omega_n^2\right)} \tag{1-8}$$

where Eq. (1-8) is a second-order system, and

$$2\zeta\omega_n=\frac{R_aB_m+K_iK_b}{R_aJ} \tag{1-9}$$

$$\omega_n^2=\frac{KK_iK_s}{R_aJ} \tag{1-10}$$

The transfer function in Eq. (1-8) represents a **stable** system for all $K > 0$ and will not exhibit **any steady state error—**that is, it will reach the desired destination dictated by the input.

In order to study the time response behavior of the position-control system, we use Simulink. The Simulink numerical model of the system is shown in Fig. 1-20, where all system parameters may be obtained experimentally using the procedure discussed in Chap. 8, as shown in Table 1-1.

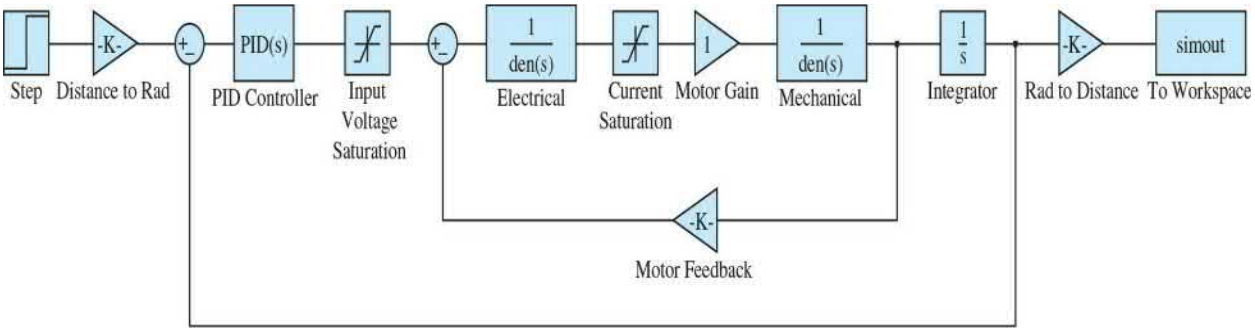


Figure 1-20 Car Simulink model.

TABLE 1-1 LEGO Car System Parameters

Car mass	$M = 574$ g
Armature resistance	$R_a = 2.27\ \Omega$
Armature inductance	$L_a = 0.0047$ H
Motor torque constant	$K_i = 0.25$ N · m/A
Back-EMF constant	$K_b = 0.25$ V/rad/s
Equivalent viscous-friction coefficient	$B = 0.003026$ kg · m^2/s or N · m/s
Total moment of inertia	$J = 0.00246$ kg · m^2

After running the simulation for the controller gain $K = 12.5$, we can plot the car travel as shown in Fig. 1-21. Note encoder output was scaled to arrive at the results shown. As shown, the car travels at a constant speed from 1 to approximately 2.7 s before stopping. From the slope of the graph in Fig. 1-21, we can obtain the maximum speed of the car as 0.4906 m/s. This is also confirmed using the speed plot in Fig. 1-22. Furthermore, from this graph, we can find the average acceleration of the car as 2.27 m/s^2. The stoppage time is dictated by the system mechanical time constant where the speed, as shown in Fig. 1-22, decays exponentially from maximum to zero. This time must be taken into consideration when the actual system meets an obstacle.

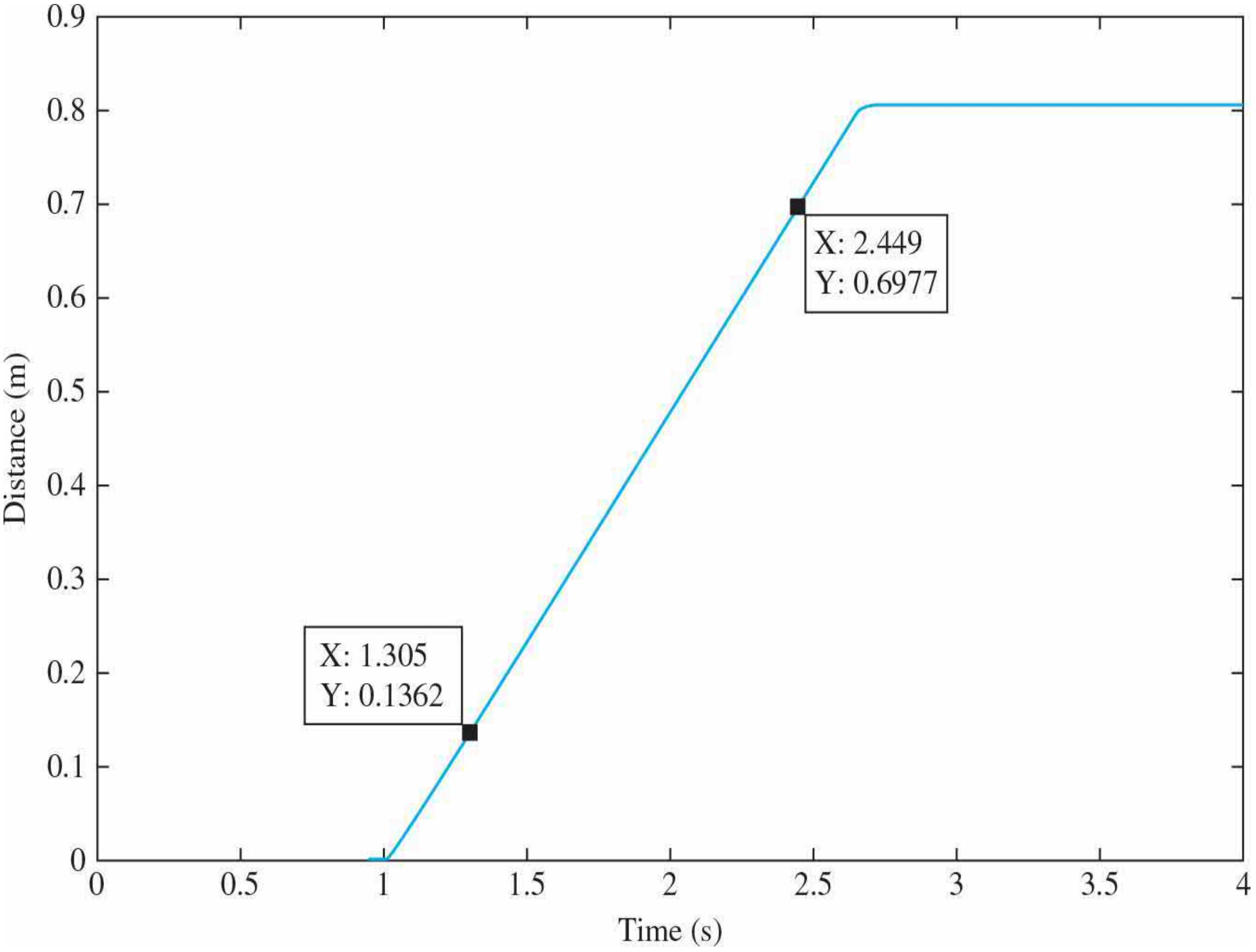


Figure 1-21 Car travel.

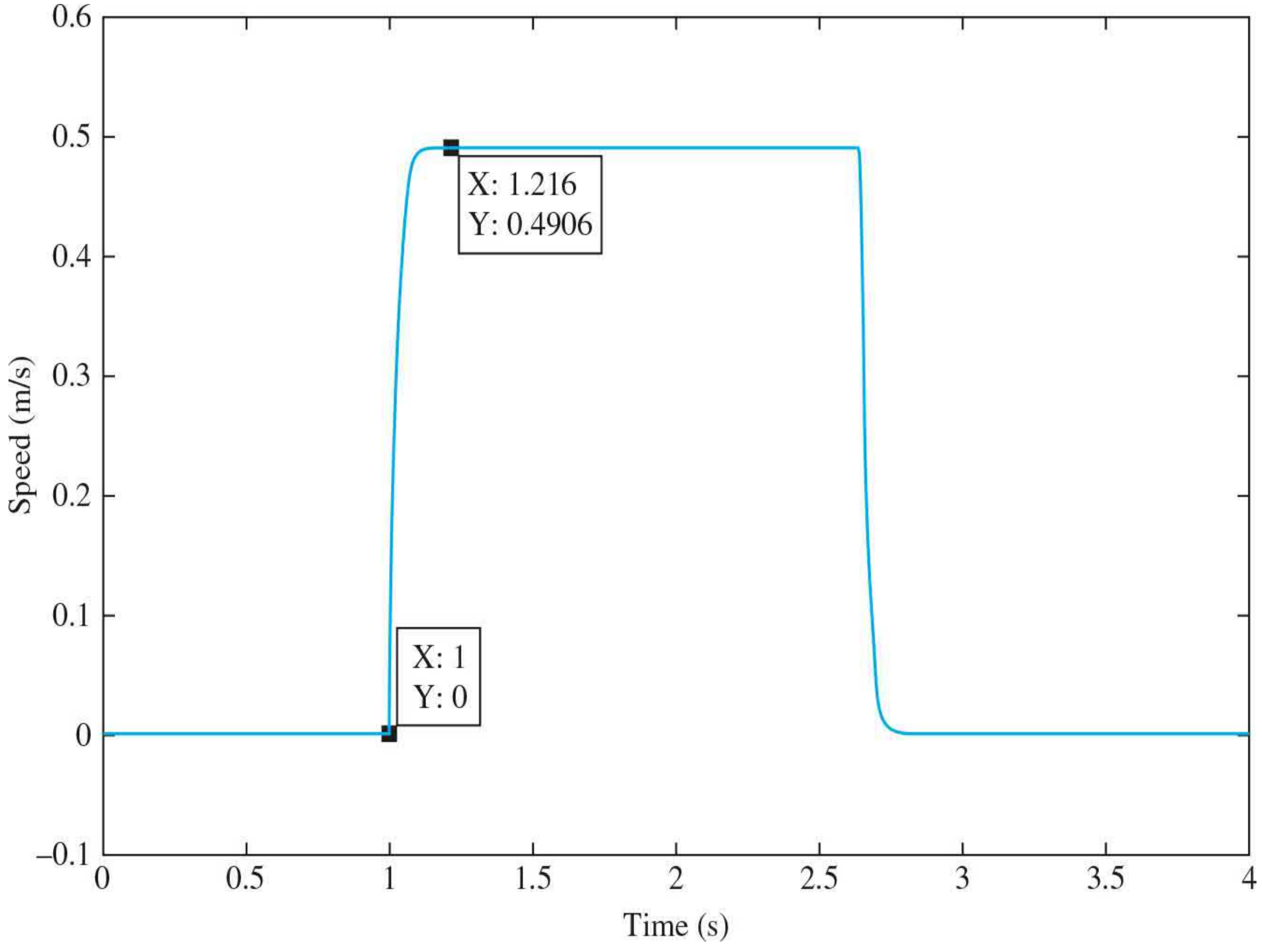


Figure 1-22 Car speed graph.

Once a satisfactory response is obtained, the control system can be tested on the **actual system**. The LEGO car utilizes Simulink to operate. The Simulink model in this case is built based on the process indicated in Chap. 8 and is shown in Fig. 1-23. Remember to enter the gain parameter $K = 12.5$.

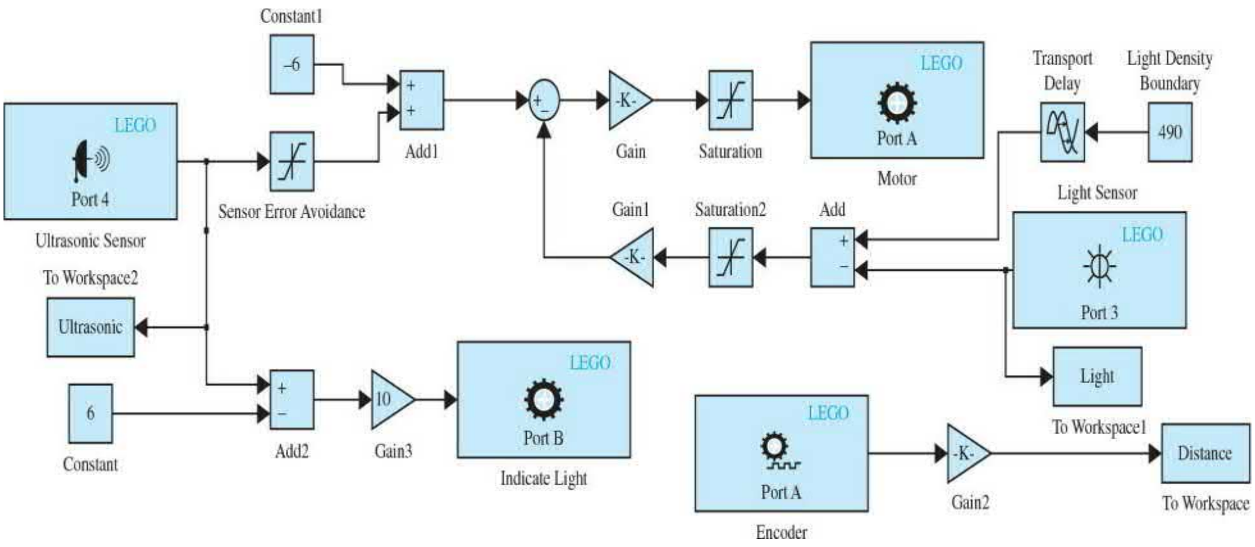


Figure 1-23 Simulink model to operate the LEGO car.

After the Simulink model is built, you can pair the host computer with the NXT brick using Bluetooth—see the instructional video. We must first start by connecting the car to the host computer using the USB cable and selecting Run on Target Hardware in the Simulink Tools menu. When the indicating light illuminates, the car is operational. For the car to run wirelessly, simply unplug the USB cable at this point.

To start up the car, you can place a strip, which is not white, underneath the light sensor, as shown in Fig. 1-24. By pulling the strip out, the car will start its run. When the car reaches an obstacle, the indicating light will turn off as it comes to a halt, as shown in Fig. 1-25—see the instructional video to learn more about the ultrasonic sensor shown in the Simulink model in Fig. 1-23. After the car has finished its run, click Stop in the Simulink program to stop the operation. All data will be stored in the computer. Using MATLAB you can plot the vehicle time response—as illustrated in Chap. 8. As shown in Fig. 1-26, the speed of the car is 0.4367 m/s, which is close to the numerical simulation results. The distance that car traveled to reach the wall is 0.8063 m. From the speed plot, shown in Fig. 1-27, the average acceleration is 1.888 m/s^2.

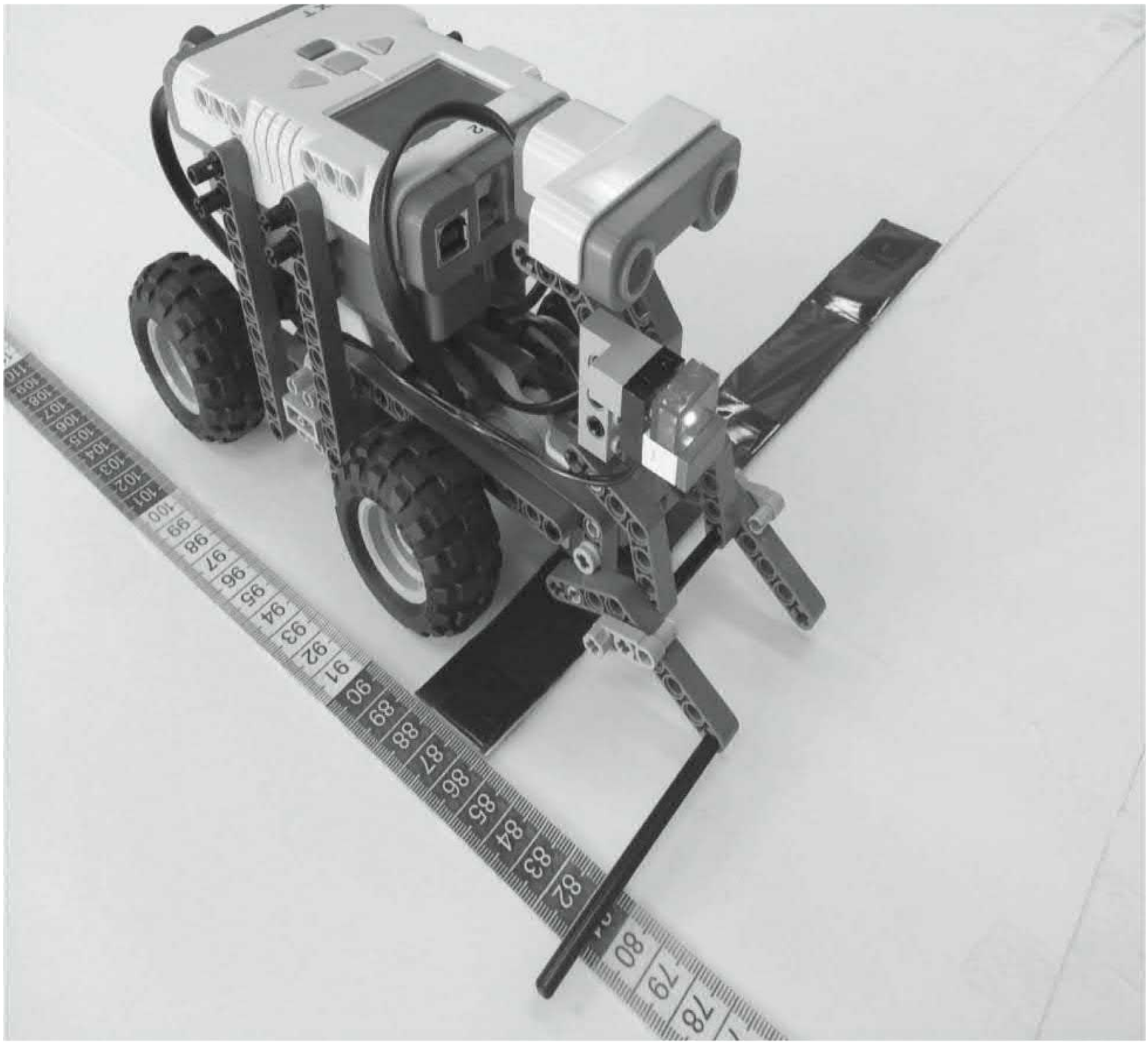


Figure 1-24 Starting position.

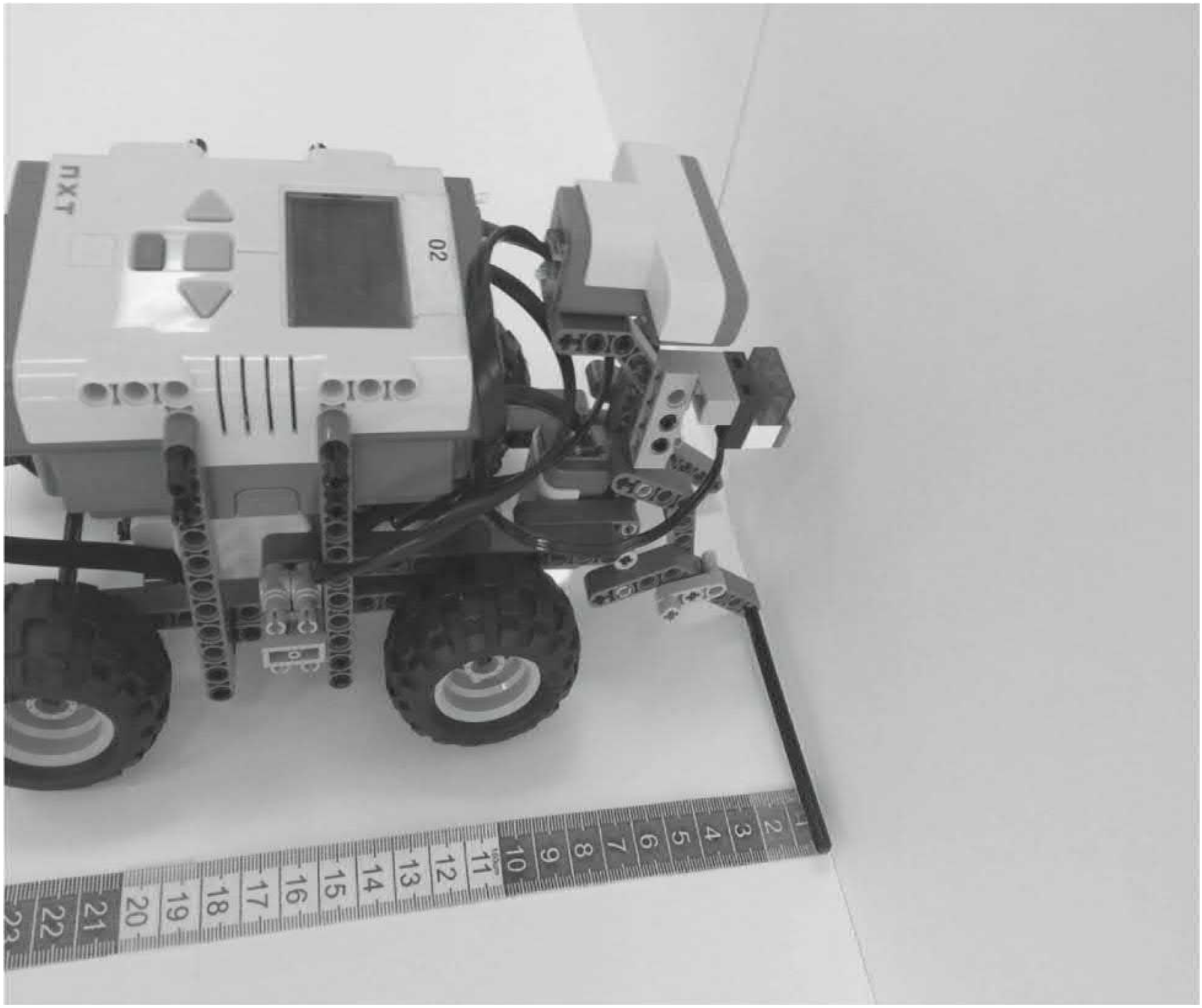


Figure 1-25 Final position.

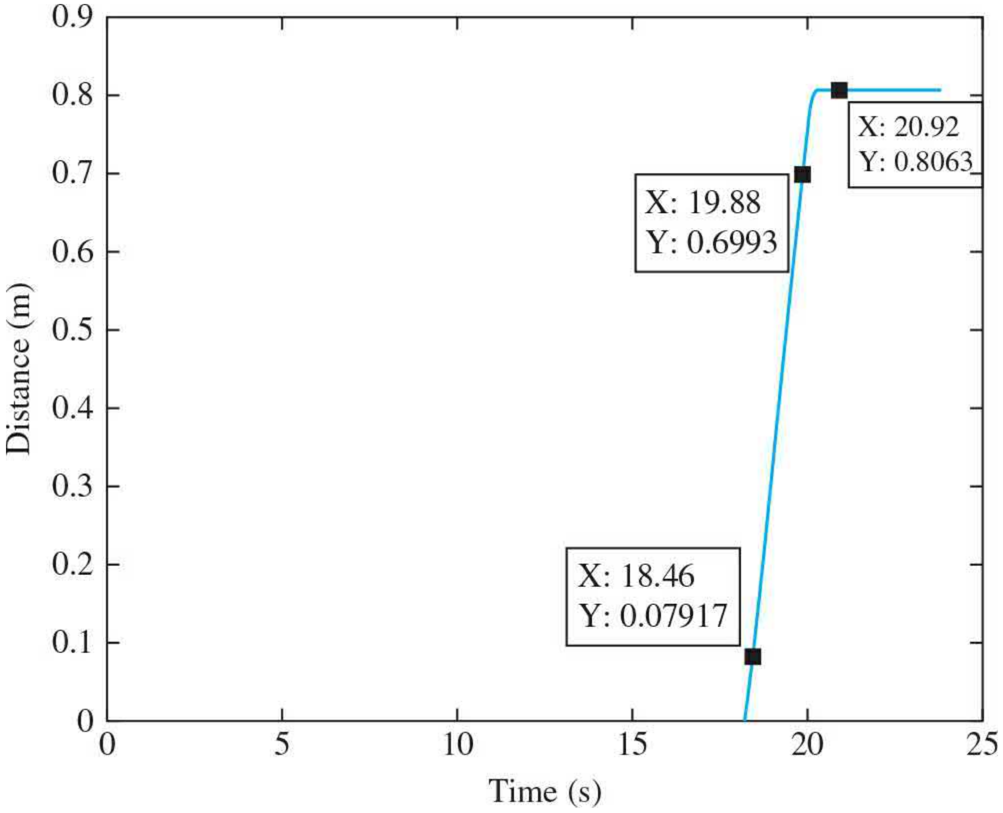


Figure 1-26 Car travel from motor encoder.

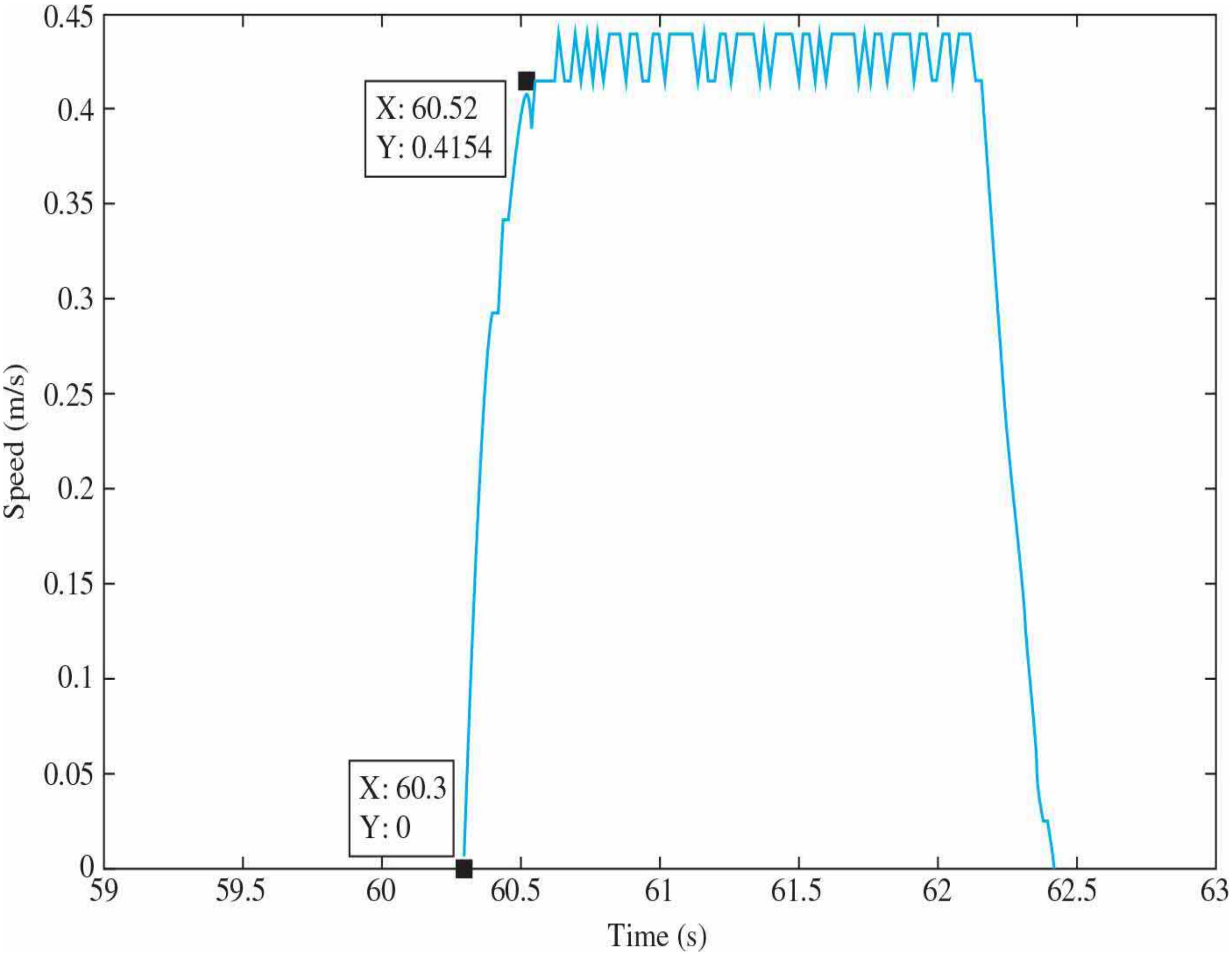


Figure 1-27 Car speed plot.

1-12 SUMMARY

In this chapter, we introduced some of the basic concepts of what a control system is and what it is supposed to accomplish. The basic components of a control system were described. By demonstrating the effects of feedback in a rudimentary way, the question of why most control systems are closed-loop systems was also clarified. Most important, it was pointed out that feedback is a double-edged sword—it can benefit as well as harm the system to be controlled. This is part of the challenging task of designing a control system, which involves consideration of such performance criteria as stability, sensitivity, bandwidth, and accuracy. Finally, various types of control systems were categorized according to the system signals, linearity, and control objectives. Several typical control-system examples were given to illustrate the analysis and design of control systems. Most systems encountered in real life are nonlinear and time varying to some extent. The concentration on the studies of linear systems is due primarily to the availability of unified and simple-to-understand analytical methods in the analysis and design of linear systems.

[1] Source: http://stateimpact.npr.org/texas/files/2011/08/Solar-Energy-Power-by-Daniel-Reese-01.jpg.

[2] Instructional YouTube video: http://youtu.be/gZo7qkWlZhs.

CHAPTER 2

Modeling of Dynamic Systems

As mentioned in Chap. 1, one of the most important tasks in the analysis and design of control systems is mathematical modeling of system subcomponents and ultimately the overall system. The models of these systems are represented by **differential equations**, which may be linear or nonlinear. In this textbook, we consider systems that are modeled by ordinary differential equations—as opposed to partial differential equations.

The analysis and design of control systems for most applications use linear (or linearized) models and are well established; while the treatment of nonlinear systems is quite complex. As a result, the control-systems engineer often has the task of determining not only how to accurately describe a system mathematically but also, more importantly, how to make proper assumptions and approximations, whenever necessary, so that the system may be realistically characterized by a linear mathematical model.

Learning Outcomes

After successful completion of this chapter, you will be able to

1. Model the differential equations of basic mechanical systems.
2. Model the differential equations of basic electrical systems.
3. Model the differential equations of basic thermal systems.
4. Model the differential equations of basic fluid systems.
5. Linearize nonlinear ordinary differential equations.
6. Discuss analogies and relate mechanical, thermal, and fluid systems to their electrical equivalents.

In this chapter, we provide a more detailed look at the modeling of components of various control systems. A control system may be composed of several components including mechanical, thermal, fluid, pneumatic, and electrical systems. In this chapter, we review basic properties of some of these systems, otherwise known as **dynamic systems**. Using the basic modeling principles such as Newton’s second law of motion, Kirchoff’s law, or conservation of mass (incompressible fluids) the model of these dynamic systems are represented by differential equations.

As mentioned earlier, because in most cases, the controller design process requires a linear model; in this chapter, we provide a review of linearization of nonlinear equations. In this chapter, we also demonstrate the similarities amongst these systems and establish analogies among mechanical, thermal, and fluid systems with electrical networks.

A control system also includes other components such as amplifiers, sensors, actuators, and computers. The modeling of these systems is discussed later in Chap. 6 because of additional theoretical requirements.

Finally, it is important to mention that the modeling materials presented in this chapter are intended to serve as a review of various second or third year university level engineering courses including dynamics, fluid mechanics, heat transfer electrical circuits, electronics, and sensors and actuators. For a more comprehensive understanding of any of these subjects, the reader is referred to courses in mentioned areas.

2-1 MODELING OF SIMPLE MECHANICAL SYSTEMS

Mechanical systems are composed of **translational**, **rotational**, or a combination of both components. The motion of mechanical elements is often directly or indirectly formulated from **Newton’s law of motion**.[1] Introductory models of these mechanical systems are based on particle dynamics, where the mass of the system is assumed to be a dimensionless particle. In order to capture the motion of realistic mechanical systems, including translation and rotational motions, rigid body dynamics models are used. Springs are used to describe flexible components and dampers are used to model friction. In the end, the resulting governing equations of motion are linear or nonlinear differential equations that can be described by up to six variables—in 3D, an object is capable of three translational motions and three rotational motions. In this textbook, we mainly look at linear and planar particle and rigid body motions.

2-1-1 Translational Motion

Translational motion can take place along a straight or curved path. The variables that are used to describe translational motion are **acceleration**, **velocity**, and **displacement**.

Newton’s law of motion states that the algebraic sum of external forces acting on a rigid body or a particle in a given direction is equal to the product of the mass of the body and its acceleration in the same direction. The law can be expressed as

$$\sum_{\text{External}} \text{forces} = Ma \tag{2-1}$$

where M denotes the mass, and a is the acceleration in the direction considered. Figure 2-1 illustrates the situation where a force is acting on a body with mass M. The force equation is written as

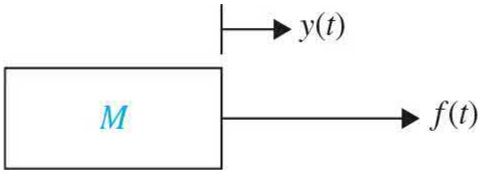


Figure 2-1 Force-mass system.

$$f(t) = Ma(t) = M\frac{d^2y(t)}{dt^2} \tag{2-2}$$

Or,

$$f(t) = M\frac{dv(t)}{dt} \tag{2-3}$$

where $a(t)$ is the acceleration, $v(t)$ denotes linear velocity, and $y(t)$ is the displacement of mass M. Note that the first step in modeling is always to draw the free-body diagram (FBD) of the system by isolating the mass and representing the effect of all attached components by their corresponding reaction forces. These forces are external forces that act on the body resulting it to accelerate. In this case, the only external force is $f(t)$. As a general rule, find the equations assuming the mass is moving along $y(t)$.

Considering Fig. 2-2, where a force *f* (*t*) is applied to a flexible structure, in this case a cantilever beam, a simple mathematical model may be obtained after approximating the system by a spring-mass-damper system.

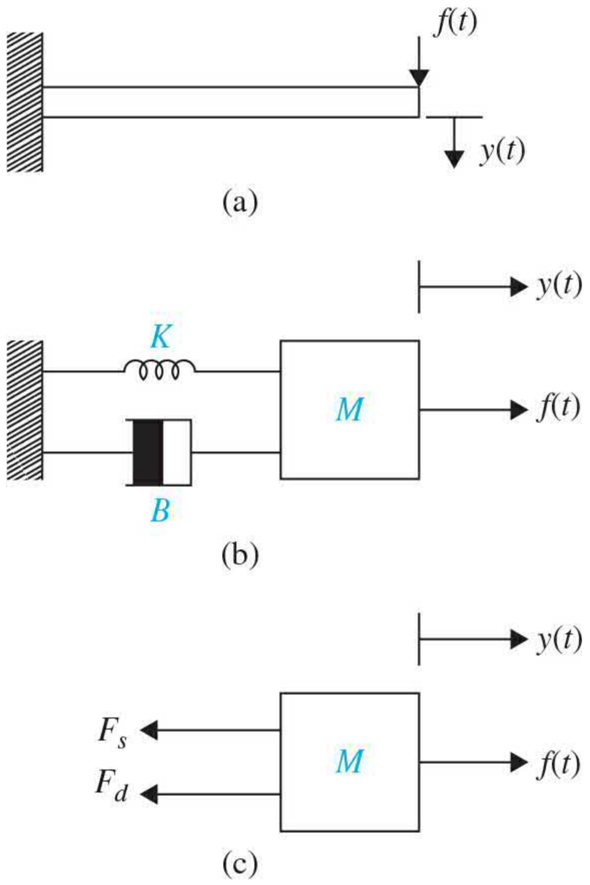


Figure 2-2 Force applied to a cantilever beam, modeled as a spring-mass-damper system. (*a*) A cantilever beam. (*b*) Spring-mass-damper equivalent model. (*c*) Free-body diagram.

In this case, in addition to the mass, the following system elements are also involved.

- **Linear spring.** In practice, a linear spring may be a model of an actual spring or a compliance of a mechanical component such as a cable or a belt—in this case a beam. In general, *an ideal spring is a massless element that stores potential energy*. The spring element in Fig. 2-2 applies a force F_s to mass *M*. Using Newton's concept of action and reaction, the mass also exerts a same force to the spring *K*, as shown in Fig. 2-3 and has the following linear model:

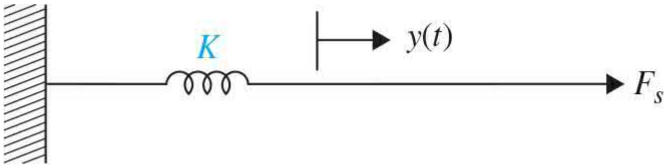


Figure 2-3 Force-spring system.

$$F_s = Ky(t) \tag{2-4}$$

where K is the spring constant, or simply stiffness. Equation (2-4) implies that the force acting on the spring is linearly proportional to the displacement (deflection) of the spring. If the spring is preloaded with a preload tension of T, then Eq. (2-4) is modified to

$$F_s - T = Ky(t) \tag{2-5}$$

- **Friction.** Whenever there is motion or tendency of motion between two physical elements, frictional forces exist. Mechanical structures also exhibit internal friction. In the case of the beam in Fig. 2-2, upon bending and releasing the structure, the resulting motion will eventually come to a halt due to this internal friction. The frictional forces encountered in physical systems are usually of a nonlinear nature. The characteristics of the frictional forces between two contacting surfaces often depend on such factors as the composition of the surfaces, the pressure between the surfaces, and their relative velocity. So an exact mathematical description of the frictional force is difficult to obtain. Three different types of friction are commonly used in practical systems: **viscous friction**, **static friction**, and **Coulomb friction**. In most cases, and in this book, in order to utilize a linear model, most frictional components are approximated as viscous friction, also known as **viscous damping**. *In viscous damping the applied force and velocity are linearly proportional*. The schematic diagram element for viscous damping is often represented by a dashpot (or damper), such as that shown in Fig. 2-3. Figure 2-4 shows the isolated dashpot, which has the following mathematical expression:

In most cases, and in this book, in order to utilize a linear model, most friction components are approximated as viscous friction, also known as

viscous damping.

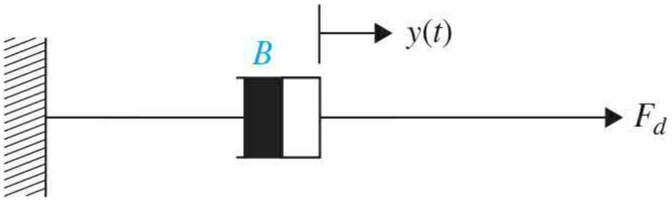


Figure 2-4 Dashpot for viscous friction.

$$F_d = B\frac{dy(t)}{dt} \tag{2-6}$$

where B is the viscous damping coefficient.

The equation of motion of the system shown in Fig. 2-2 is obtained using the free-body diagram shown in Fig. 2-2*c*—assuming the mass is pulled along $y(t)$ direction. Hence, we get

$$f(t) - F_s - F_d = Ma(t) = M\frac{d^2y(t)}{dt^2} \tag{2-7}$$

Upon substituting Eqs. (2-4) and (2-5) into (2-6) and rearranging the equation, we have

$$M\frac{d^2y(t)}{dt^2} + B\frac{dy(t)}{dt} + Ky(t) = f(t) \tag{2-8}$$

where $\dot{y}(t) = \left(\frac{dy(t)}{dt}\right)$ and $\ddot{y}(t) = \left(\frac{d^2y(t)}{dt^2}\right)$ represent velocity and acceleration, respectively. Dividing the former equation by M, we get

$$\ddot{y}(t) + \frac{B}{M}\dot{y}(t) + \frac{K}{M}y(t) = \frac{K}{M}r(t) \tag{2-9}$$

where $r(t)$ has the **same units** as $y(t)$. In control systems, it is customary to rewrite Eq. (2-9) as

$$\ddot{y}(t)+2\zeta\omega_n\dot{y}(t)+\omega_n^2 y(t)=\omega_n^2 r(t) \tag{2-10}$$

where ω_n and ζ are the ***natural frequency*** and the ***damping ratio*** of the system, respectively. Equation (2-10) is also known as the **prototype second-order system**. We define $y(t)$ as the **output** and $r(t)$ as the **input** of the system.

EXAMPLE 2-1-1 Consider the two degrees of freedom mechanical system shown in Fig. 2-5, where a mass M_1 slides along a smooth lubricated surface of mass M_2 that is connected to a wall by a spring K.

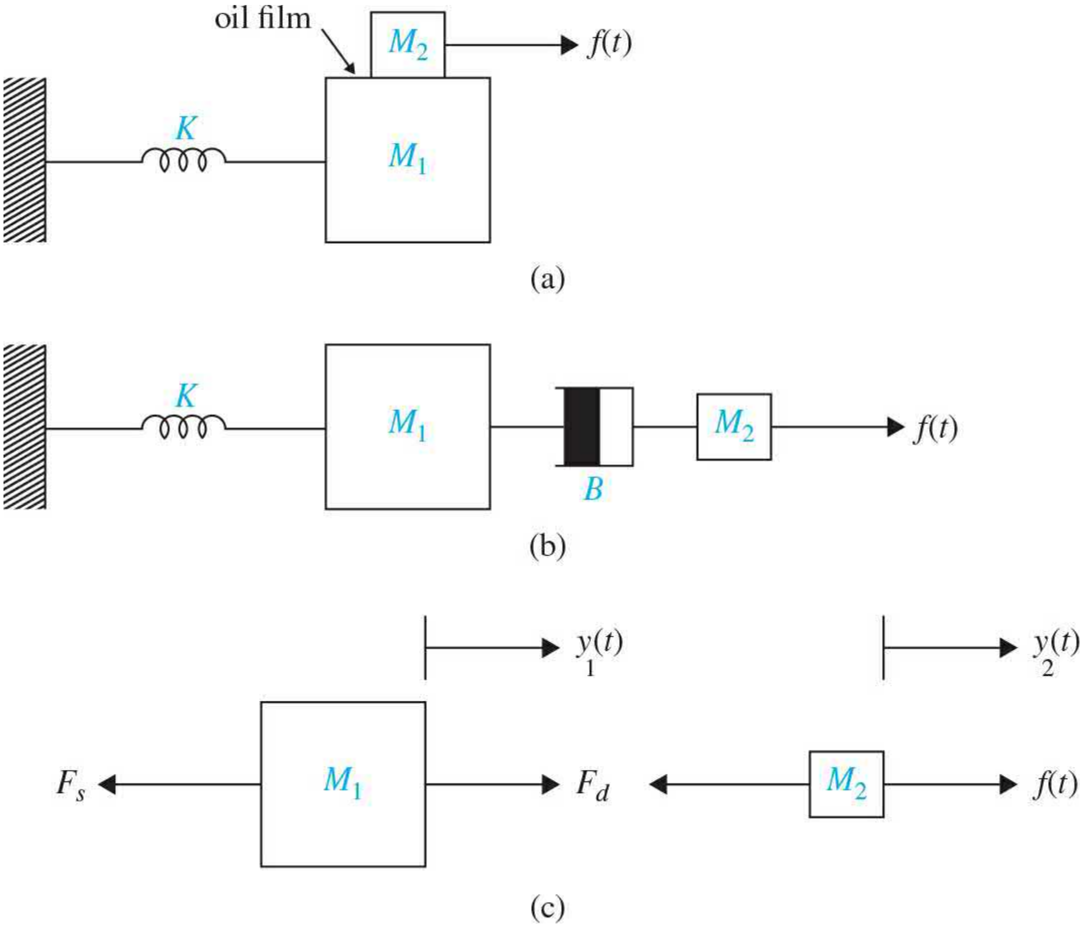


Figure 2-5 A two-degree of freedom mechanical system with spring and damper elements. (*a*) A two mass spring system. (*b*) Mass, spring, damper equivalent system. (*c*) Free-body diagram.

The displacements of masses M_1 and M_2 are measured by $y_1(t)$ and $y_2(t)$, respectively. The oil film between the two surfaces is modeled as a viscous damping element B, as shown in Fig. 2-5*b*. After drawing the free-body diagrams of the two masses, as shown in Fig. 2-5*c*, we apply Newton’s second law of motion to each mass, we get

$$\sum_{\text{External}} \text{forces} = M_1\ddot{y}_1(t) \tag{2-11}$$

Using Eqs. (2-5) and (2-6), we get

$$-Ky_1(t) + B\left(\dot{y}_2(t) - \dot{y}_1(t)\right) = M_1\ddot{y}_1(t) \tag{2-12}$$

$$\sum_{\text{External}} \text{forces} = M_2\ddot{y}_2(t) \tag{2-13}$$

Similarly, using Eqs. (2-5) and (2-6), we get

$$-B(\dot{y}_2(t) - \dot{y}_1(t)) + f(t) = M_2\ddot{y}_2(t) \tag{2-14}$$

Thus, the two second-order differential equations of motion become

$$M_1\ddot{y}_1(t) + B\left(\dot{y}_1(t) - \dot{y}_2(t)\right) + Ky_1(t) = 0 \tag{2-15}$$

$$M_2\ddot{y}_2(t) + B\left(\dot{y}_1(t) - \dot{y}_2(t)\right) = f(t) \tag{2-16}$$

▲

EXAMPLE 2-1-2 Consider the two degrees of freedom mechanical system shown in Fig. 2-6 with two masses M_1 and M_2 constrained by three springs, while a force $f(t)$ is applied to mass M_2.

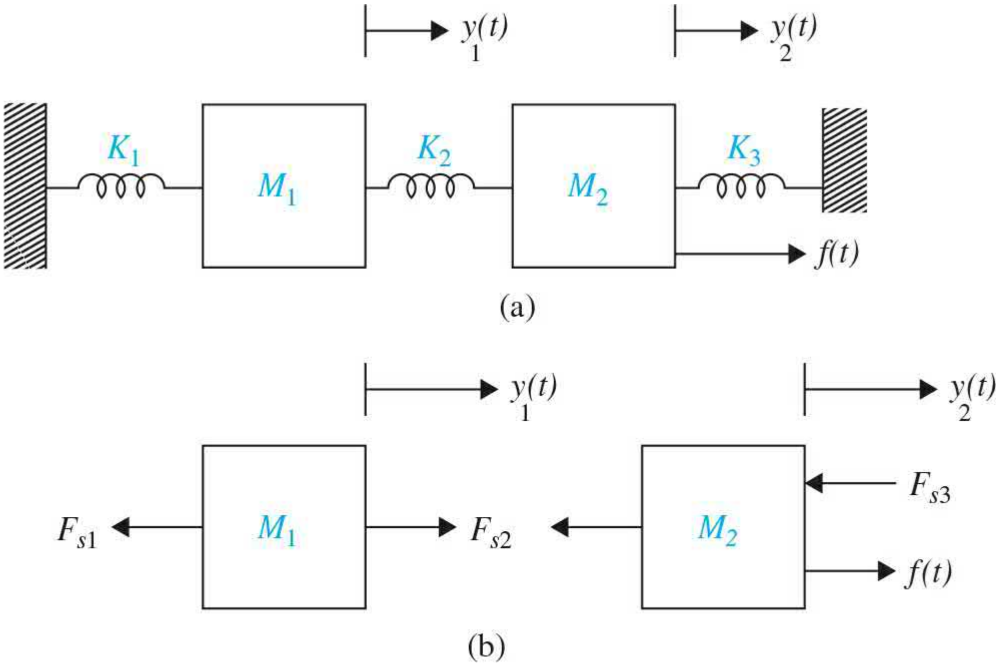


Figure 2-6 (*a*) A two-degree of freedom mechanical system with three springs. (*b*) Free-body diagram.

The displacements of masses M_1 and M_2 are measured by $y_1(t)$ and $y_2(t)$, respectively. Assuming masses are displaced in positive directions and $y_2(t) > y_1(t)$, we draw the free-body diagrams of the two masses, as shown in Fig. 2-6*b*. This is a good trick to use to get the applied spring force directions correct. So in this case, springs K_1 and K_2 are in tension while K_3 is in compression. Applying Newton's second law of motion to each mass, we get

$$\sum_{\text{External}} \text{forces} = M_1 \ddot{y}_1(t) \tag{2-17}$$

Using Eq. (2-5), and noting the deflection of springs K_1 and K_2 are $y_1(t)$ and $(y_2(t) - y_1(t))$, respectively, we get

$$-K_1 y_1(t) + K_2(y_2(t) - y_1(t)) = M_1 \ddot{y}_1(t) \tag{2-18}$$

$$\sum_{\text{External}} \text{forces} = M_2 \ddot{y}_2(t) \tag{2-19}$$

Similarly, using Eq. (2-5), we get

$$-K_2(y_2(t)-y_1(t))-K_3y_2(t)+f(t)=M_2\ddot{y}_2(t) \tag{2-20}$$

Thus, the two second-order differential equations of motion become

$$M_1\ddot{y}_1(t)+(K_1+K_2)y_1(t)-K_2y_2(t)=0 \tag{2-21}$$

$$M_2\ddot{y}_2(t)-K_2y_1(t)+(K_2+K_3)y_2(t)=f(t) \tag{2-22}$$

▲

EXAMPLE 2-1-3 Consider the three-story building shown in Fig. 2-7. Let us derive the equations of the system describing the motion of the building after a shock at the base due to an earthquake. Assuming the masses of the floors are dominant compared to those of the columns, and the columns have no internal loss of energy, the system can be modeled by three masses and three springs, as shown in Fig. 2-7*b*. The modeling approach is then identical to that in Example 2-1-2. We draw the free-body diagram, assuming $y_3(t) > y_2(t) > y_1(t)$ and obtain the final equations of the system as

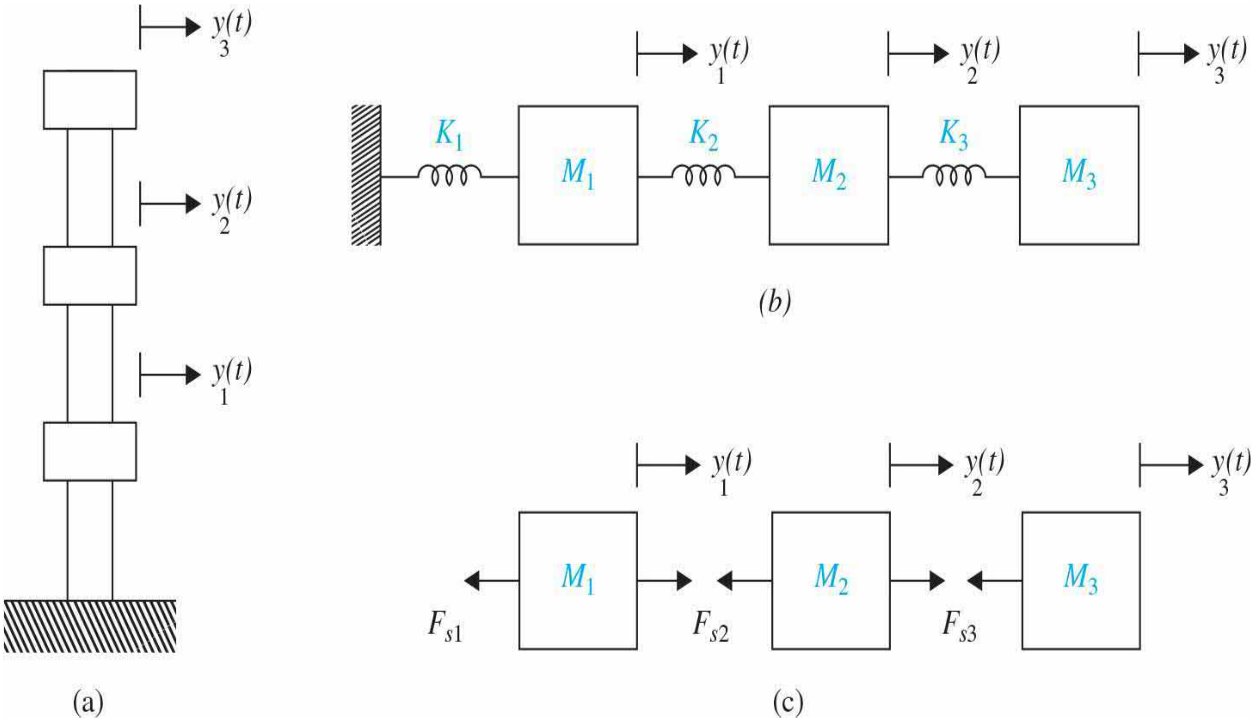


Figure 2-7 (*a*) A three-story building. (*b*) Equivalent model as a three-degree of freedom spring-mass system. (*c*) Free-body diagram.

$$M_1\ddot{y}_1(t)+(K_1+K_2)y_1(t)-K_2y_2(t)=0 \tag{2-23}$$

$$M_2\ddot{y}_2(t)-K_2y_1(t)+(K_2+K_3)y_2(t)-K_3y_3(t)=0 \tag{2-24}$$

$$M_3\ddot{y}_3(t)-K_3y_2(t)+K_3y_3(t)=0 \tag{2-25}$$

2-1-2 Rotational Motion

For most applications encountered in control systems, the rotational motion of a body can be defined as motion about a **fixed axis**.[2] Newton's second law for rotational motion states that the *algebraic sum of external moments applied to a rigid body of inertia* J *about a fixed axis, produces an angular acceleration about that axis.* Or

$$\sum_{\text{External}} \text{Moments} = J\alpha \tag{2-26}$$

where J denotes the inertia and α is the angular acceleration. The other variables generally used to describe the motion of rotation are **torque** T (normally applied from a motor), **angular velocity** ω, and **angular displacement** θ. The rotational equations of motion include the following terms:

- ***Inertia.*** A three-dimensional rigid body of mass M has three moments of inertia and three products of inertia. In this textbook, we primarily look at planar motions, governed by Eq. (2-26). *A rigid body of mass* M *has inertia*, J, *about a fixed rotational axis, which is a property related to kinetic energy of rotational motion*. The inertia of a given element depends on the geometric composition about the axis of rotation and its density. For instance, the inertia of a circular disk or shaft, of radius r and mass M, about its geometric axis is given by

$$J = \frac{1}{2}Mr^2 \tag{2-27}$$

When a torque is applied to a body with inertia J, as shown in Fig. 2-8, the torque equation is written as

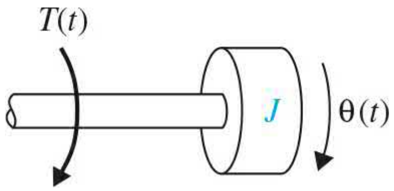


Figure 2-8 Torque-inertia system.

$$T(t) = J\alpha(t) = J\frac{d\omega(t)}{dt} = J\frac{d^2\theta(t)}{dt^2} \tag{2-28}$$

where $\theta(t)$ is the angular displacement; $\omega(t)$, the angular velocity; and $\alpha(t)$, the angular acceleration.

- ***Torsional spring.*** As with the linear spring for translational motion, a **torsional spring constant K**, in torque-per-unit angular displacement, can be devised to represent the compliance of a rod or a shaft when it is subject to an applied torque. Figure 2-9 illustrates a simple torque-spring system that can be represented by the equation

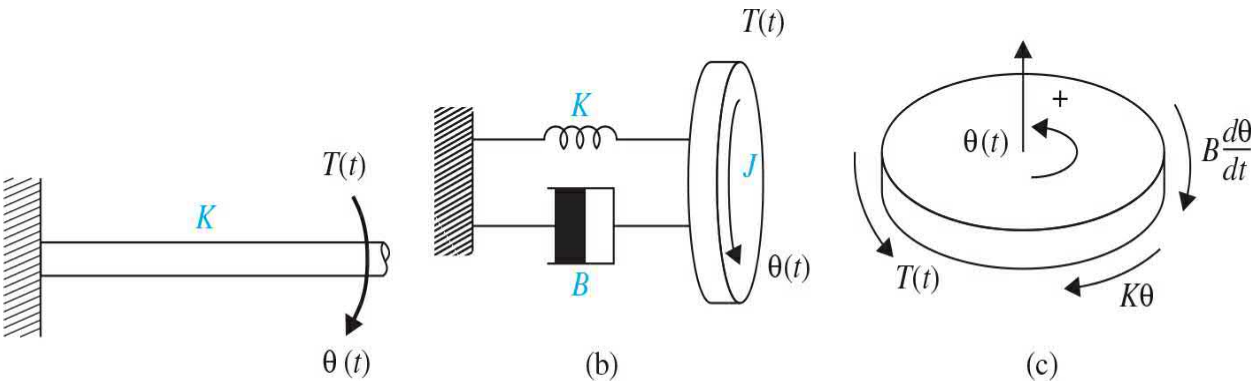


Figure 2-9 (*a*) A rod under a torsional load. (*b*) Equivalent torque torsional spring system. (*c*) Free-body diagram.

$$T_s = K\theta(t) \tag{2-29}$$

If the torsional spring is preloaded by a preload torque of *TP*, Eq. (2-36) is modified to

$$T_s - TP = K\theta(t) \tag{2-30}$$

- ***Viscous damping for rotational motion.*** The friction described for translational motion can be carried over to the motion of rotation. Therefore, Eq. (2-6) can be replaced by

$$T_d = B\frac{d\theta(t)}{dt} \tag{2-31}$$

In Fig. 2-9*b*, the internal loss of energy in a rod is represented by viscous damping *B*.

Considering the free-body diagram in Fig. 2-9*c*, we examine the reactions after application of a torque in positive direction. Note we normally use the right-hand rule to define the positive direction of rotation—in this case counterclockwise. Upon substituting Eqs. (2-29) and (2-31) into Eq. (2-26) and rearranging the equation, we have

$$J\frac{d^2\theta(t)}{dt^2} + B\frac{d\theta(t)}{dt} + K\theta(t) = T(t) \tag{2-32}$$

where $\dot{\theta}(t) = \left(\frac{d\theta(t)}{dt}\right)$ and $\ddot{\theta}(t) = \left(\frac{d^2\theta(t)}{dt^2}\right)$ represent angular velocity and acceleration, respectively. Dividing the former equation by *J*, we get

$$\ddot{\theta}(t) + \frac{B}{J}\dot{\theta}(t) + \frac{K}{J}\theta(t) = \frac{K}{J}r(t) \tag{2-33}$$

where *r*(*t*) has the **same units** as *θ*(*t*). In control systems, it is customary to rewrite Eq. (2-33) as

$$\ddot{\theta}(t) + 2\zeta\omega_n\dot{\theta}(t) + \omega_n^2\theta(t) = \omega_n^2 r(t) \tag{2-34}$$

where ω_n and *ζ* are the *natural frequency* and the *damping ratio* of the system, respectively. Equation (2-34) is also known as the **prototype second-order system**. We define *θ*(*t*) as the **output** and *r*(*t*) as the **input** of the system. Notice that this system is **analogous** to the translational system in Fig. 2-2.

EXAMPLE 2-1-4 A nonrigid coupling between two mechanical components in a control system often causes torsional resonances that can be transmitted to all parts of the system. In this case, the rotational system shown in Fig. 2-10*a* consists of a motor with a long shaft of inertia J_m. A disk representing a load with inertia J_L is mounted at the end of the motor shaft. The shaft flexibility is modeled as a torsional spring *K* and any loss of energy within the motor is represented by viscous damping of coefficient *B*. For simplicity we assume the shaft, in this case, has no internal loss of energy. Because of the flexibility in the shaft, the angular displacement at the motor end and the load are not equal—designated as θ_m and θ_L, respectively. The system, therefore, has two degrees of freedom.

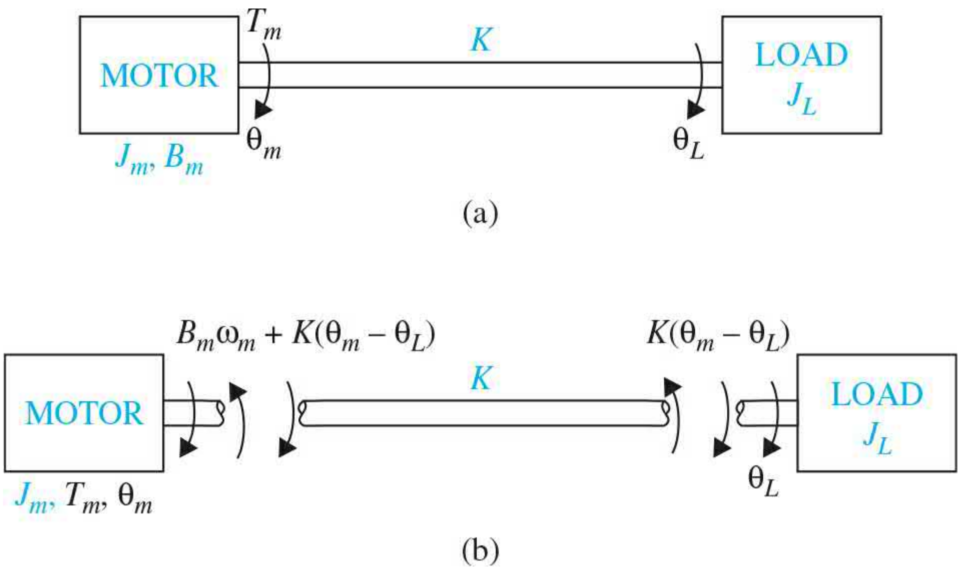


Figure 2-10 (*a*) Motor-load system. (*b*) Free-body diagram.

The system variables and parameters are defined as follows:

$T_m(t)$ = motor torque

B_m = motor viscous-friction coefficient

K = spring constant of the shaft

$\theta_m(t)$ = motor displacement

$\omega_m(t)$ = motor velocity

J_m = motor inertia

$\theta_L(t)$ = load displacement

$\omega_L(t)$ = load velocity

J_L = load inertia

The free-body diagrams of the system are shown in Fig. 2-17*b*. The two equations of the system are

$$\frac{d^2\theta_m(t)}{dt^2} = -\frac{B_m}{J_m}\frac{d\theta_m(t)}{dt} - \frac{K}{J_m}[\theta_m(t) - \theta_L(t)] + \frac{1}{J_m}T_m(t) \tag{2-35}$$

$$K[\theta_m(t) - \theta_L(t)] = J_L\frac{d^2\theta_L(t)}{dt^2} \tag{2-36}$$

Equations (2-35) and (2-36) are rearranged as

$$\frac{d^2\theta_m(t)}{dt^2}+\frac{B_m}{J_m}\frac{d\theta_m(t)}{dt}+\frac{K}{J_m}[\theta_m(t)-\theta_L(t)]=\frac{1}{J_m}T_m(t) \tag{2-37}$$

$$\frac{d^2\theta_L(t)}{dt^2}+\frac{K}{J_L}[\theta_L(t)-\theta_m(t)]=0 \tag{2-38}$$

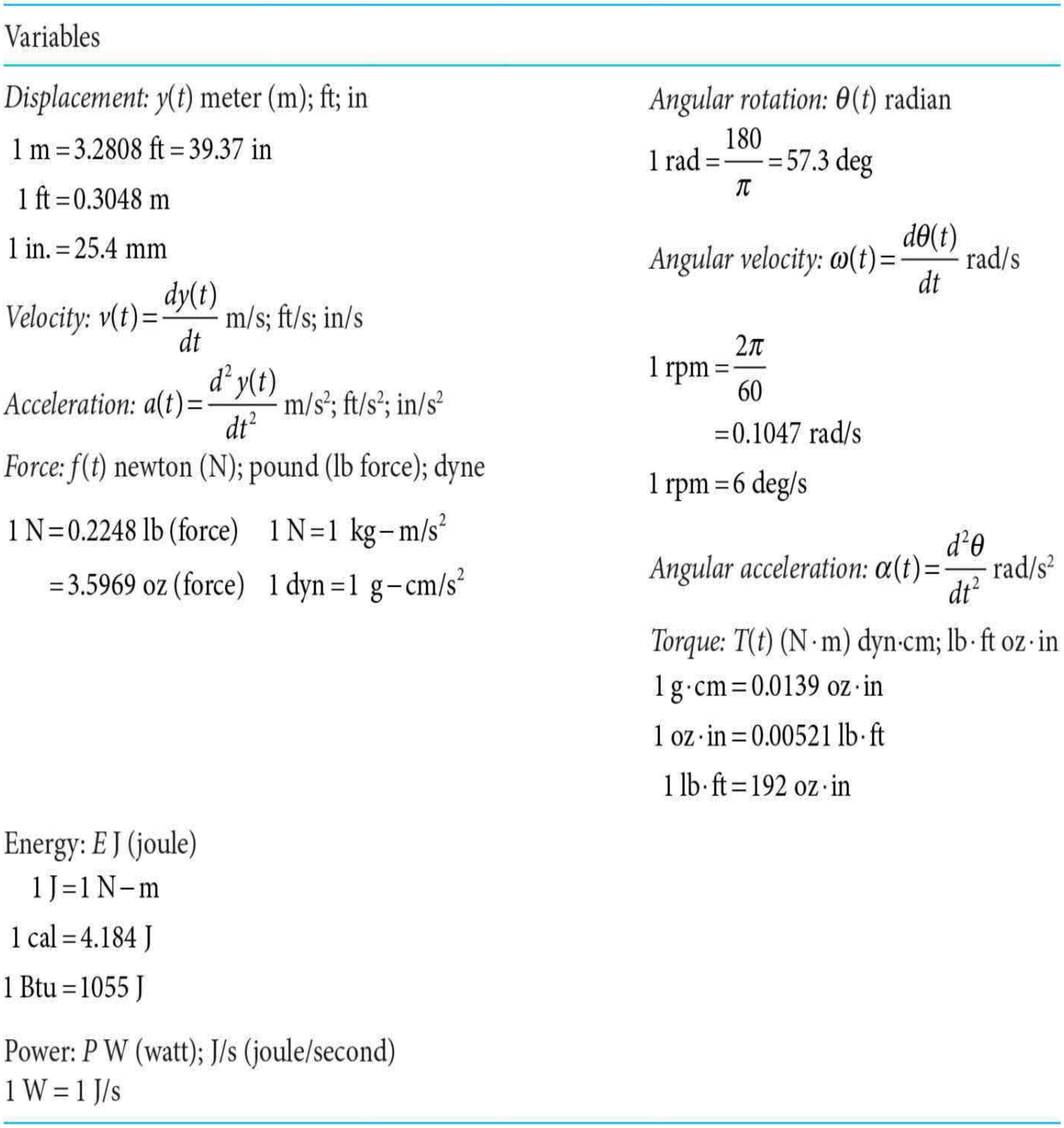
Note that if the motor shaft is **rigid**, $\theta_m = \theta_L$ and all the motor applied torque is transmitted to the load. So, in this case the overall equation of the system becomes

$$\frac{d^2\theta_m(t)}{dt^2}+\frac{B_m}{J_m+J_L}\frac{d\theta_m(t)}{dt}=\frac{1}{J_m+J_L}T_m(t) \tag{2-39}$$

Table 2-1 shows the SI and other measurement units for translational and rotational mechanical system parameters.

TABLE 2-1 Basic Translational and Rotational Mechanical System Properties and Their Units

Parameter	Symbol Used	SI Units	Other Units	Conversion Factors
Translational Motion				
Mass	M	Kilogram (kg)	Slug ft/s^2	$1\ \text{kg} = 1000\ \text{g}$ $= 2.2046\ \text{lb (mass)}$ $= 35.274\ \text{oz (mass)}$ $= 0.06852\ \text{slug}$
Spring constant	K	N/m	lb/ft	
Viscous friction coefficient	B	N/m/s	lb/ft/s	
Rotational Motion				
Inertia	J	$\text{kg} \cdot \text{m}^2$	$\text{slug} \cdot \text{ft}^2$ $\text{lb} \cdot \text{ft} \cdot \text{s}^2$ $\text{oz} \cdot \text{in} \cdot \text{s}^2$	$1\ \text{g} \cdot \text{cm} = 1.417 \times 10^{-5}\ \text{oz} \cdot \text{in} \cdot \text{s}^2$ $1\ \text{lb} \cdot \text{ft} \cdot \text{s}^2 = 192\ \text{oz} \cdot \text{in} \cdot \text{s}^2$ $= 32.2\ \text{lb} \cdot \text{ft}^2$ $1\ \text{oz} \cdot \text{in} \cdot \text{s}^2 = 386\ \text{oz} \cdot \text{in}^2$ $1\ \text{g} \cdot \text{cm} \cdot \text{s}^2 = 980\ \text{g} \cdot \text{cm}^2$
Spring constant	K	$\text{N} \cdot \text{m/rad}$	$\text{ft} \cdot \text{lb/rad}$	
Viscous friction Coefficient	B	$\text{N} \cdot \text{m/rad/s}$	$\text{ft} \cdot \text{lb/rad/s}$	



Variables

Displacement: $y(t)$ meter (m); ft; in

$1\ \text{m} = 3.2808\ \text{ft} = 39.37\ \text{in}$

$1\ \text{ft} = 0.3048\ \text{m}$

$1\ \text{in.} = 25.4\ \text{mm}$

Velocity: $v(t) = \dfrac{dy(t)}{dt}$ m/s; ft/s; in/s

Acceleration: $a(t) = \dfrac{d^2y(t)}{dt^2}$ m/s^2; ft/s^2; in/s^2

Force: $f(t)$ newton (N); pound (lb force); dyne

$1\ \text{N} = 0.2248\ \text{lb (force)} \quad 1\ \text{N} = 1\ \text{kg} - \text{m/s}^2$

$= 3.5969\ \text{oz (force)} \quad 1\ \text{dyn} = 1\ \text{g} - \text{cm/s}^2$

Angular rotation: $\theta(t)$ radian

$1\ \text{rad} = \dfrac{180}{\pi} = 57.3\ \text{deg}$

Angular velocity: $\omega(t) = \dfrac{d\theta(t)}{dt}$ rad/s

$1\ \text{rpm} = \dfrac{2\pi}{60}$

$= 0.1047\ \text{rad/s}$

$1\ \text{rpm} = 6\ \text{deg/s}$

Angular acceleration: $\alpha(t) = \dfrac{d^2\theta}{dt^2}$ rad/s^2

Torque: $T(t)$ $(\text{N} \cdot \text{m})$ dyn·cm; lb·ft oz·in

$1\ \text{g} \cdot \text{cm} = 0.0139\ \text{oz} \cdot \text{in}$

$1\ \text{oz} \cdot \text{in} = 0.00521\ \text{lb} \cdot \text{ft}$

$1\ \text{lb} \cdot \text{ft} = 192\ \text{oz} \cdot \text{in}$

Energy: E J (joule)

$1\ \text{J} = 1\ \text{N} - \text{m}$

$1\ \text{cal} = 4.184\ \text{J}$

$1\ \text{Btu} = 1055\ \text{J}$

Power: P W (watt); J/s (joule/second)

$1\ \text{W} = 1\ \text{J/s}$

2-1-3 Conversion between Translational and Rotational Motions

In motion-control systems, it is often necessary to convert rotational motion into translational motion. For instance, a load may be controlled to move along a straight line through a rotary-motor-and-lead-screw assembly, such as that shown in Fig. 2-11. Figure 2-12 shows a similar situation in which a rack-and-pinion assembly is used as a mechanical linkage. Another familiar system in motion control is the control of a mass through a pulley by a rotary motor, as shown in Fig. 2-13. The systems shown in Figs. 2-11 to 2-13 can all be represented by a simple system with an equivalent inertia connected directly to the drive motor. For instance, the mass in Fig. 2-13 can be regarded as a point mass that moves about the pulley, which has a radius *r*. By disregarding the inertia of the pulley, the equivalent inertia that the motor sees is

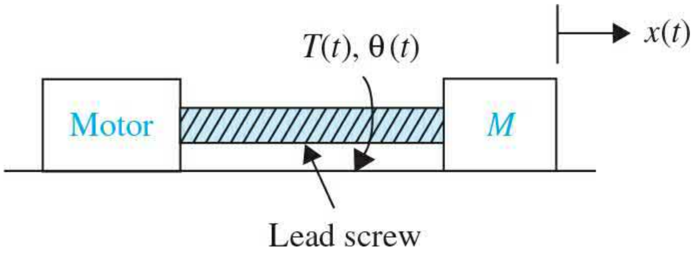


Figure 2-11 Rotary-to-linear motion control system (lead screw).

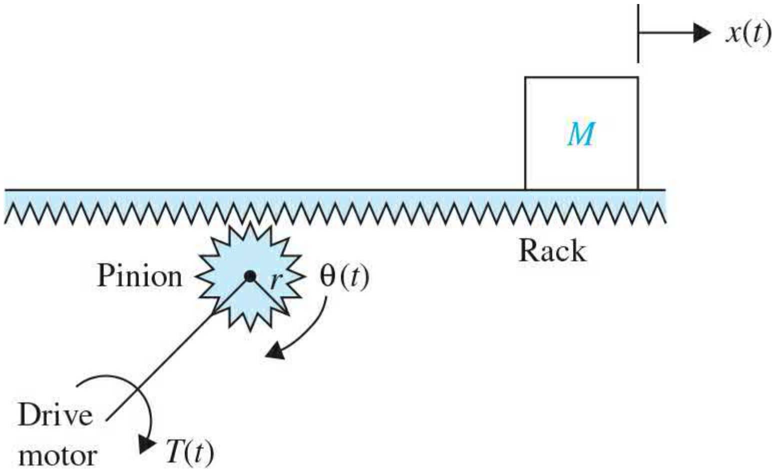


Figure 2-12 Rotary-to-linear motion control system (rack and pinion).

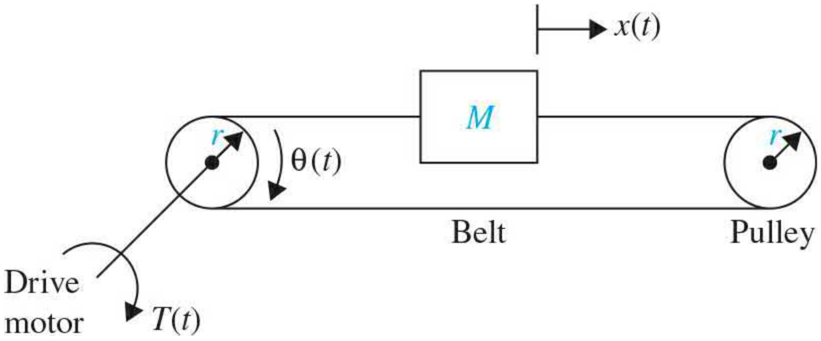


Figure 2-13 Rotary-to-linear motion control system (belt and pulley).

$$J = Mr^2 \tag{2-40}$$

If the radius of the pinion in Fig. 2-12 is *r*, the equivalent inertia that the motor sees is also given by Eq. (2-40).

Now consider the system of Fig. 2-11. The lead of the screw, *L*, is defined as the linear distance that the mass travels per revolution of the screw. In principle, the two systems in Figs. 2-12 and 2-13 are equivalent. In Fig. 2-12, the distance traveled by the mass per revolution of the pinion is $2\pi r$. By using Eq. (2-40) as the equivalent inertia for the system of Fig. 2-11, we have

$$J = M\left(\frac{L}{2\pi}\right)^2 \tag{2-41}$$

EXAMPLE 2-1-5 Classically, the quarter-car model is used in the study of vehicle suspension systems and the resulting dynamic response due to various road inputs. Typically, the inertia, stiffness, and damping characteristics of the system as illustrated in Fig. 2-14*a* are modeled in a two-degree of freedom (2-DOF) system, as shown in Fig. 2-14*b*. Although a 2-DOF system is a more accurate model, it is sufficient for the following analysis to assume a 1-DOF model, as shown in Fig. 2-14*c*.

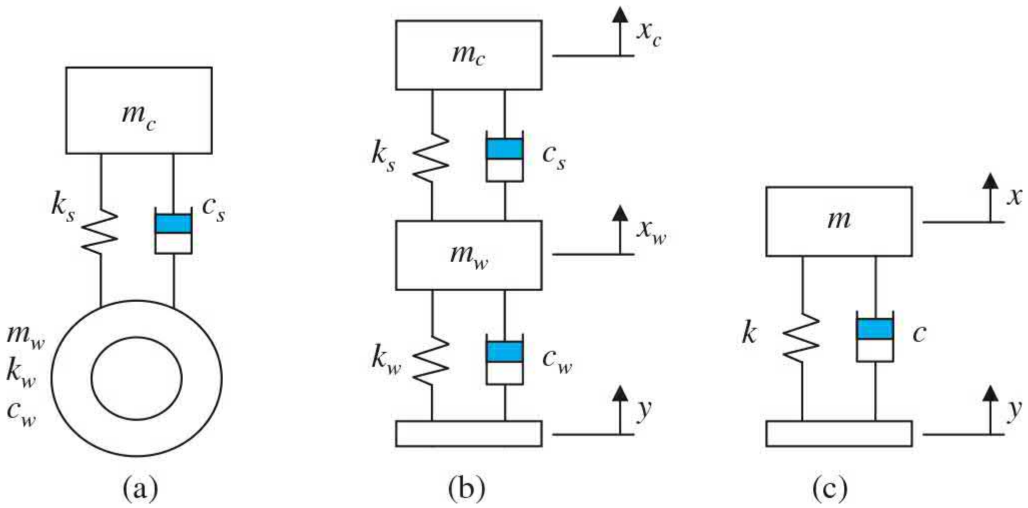


Figure 2-14 Quarter-car model realization. (*a*) Quarter car. (*b*) Two degrees of freedom model. (*c*) One degree of freedom model.

Given the system illustrated in Fig. 2-14*c*, where

m = effective ¼ car mass

k = effective stiffness

c = effective damping

x(*t*) = absolute displacement of the mass, *m*

y(*t*) = absolute displacement of the base

z(*t*) = relative displacement of the mass with respect to the base

The equation of motion of the system is defined as follows:

$$m\ddot{x}(t) = c(\dot{y}(t) - \dot{x}(t)) + k(y(t) - x(t)) \tag{2-42}$$

or

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = c\dot{y}(t) + ky(t) \tag{2-43}$$

which can be redefined in terms of the relative displacement, or bounce, by substituting the relation

$$z(t) = x(t) - y(t) \tag{2-44}$$

Dividing the result by m, Eq. (2-43) is rewritten as

$$\ddot{z}(t) + 2\zeta\omega_n\dot{z}(t) + \omega_n^2 z(t) = -\ddot{y}(t) = -a(t) \tag{2-45}$$

Note that as before ω_n and ζ are the *natural frequency* and the *damping ratio* of the system, respectively. Equation (2-45) reflects how the vehicle chassis bounces relative to the ground given an input acceleration from the ground—for example, after the wheels go through a bump.

In practice, **active control** of the suspension system may be achieved using various types of actuators including hydraulic, pneumatic, or electromechanical systems such as motors. Let’s use an active suspension that uses a dc motor in conjunction with a rack as shown in Fig. 2-15.

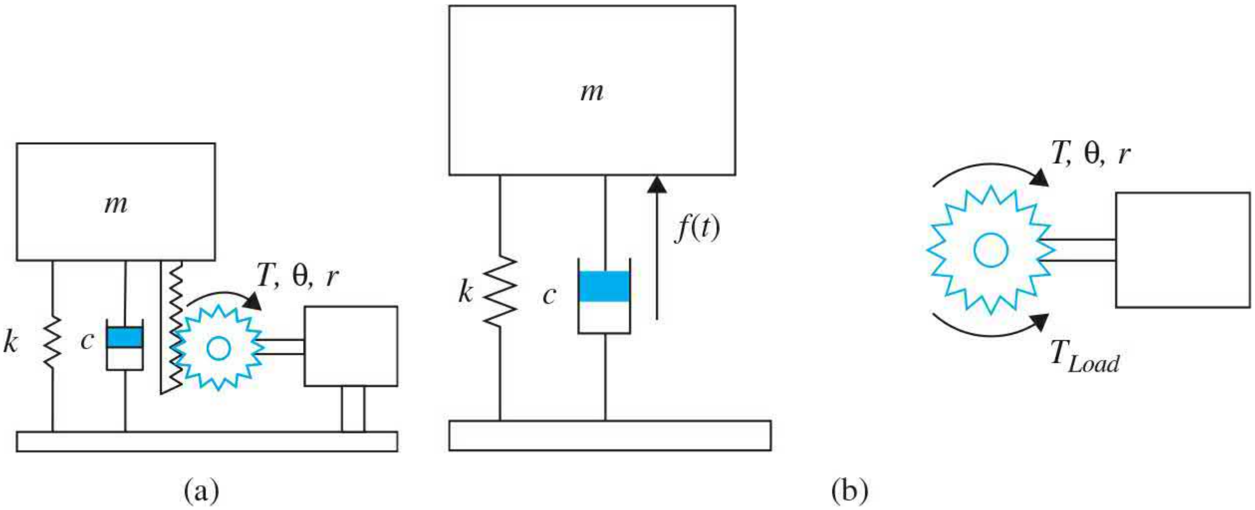


Figure 2-15 Active control of the 1-DOF quarter-car model via a dc motor and rack. (*a*) Schematics. (*b*) Free-body diagram.

In Fig. 2-15, *T*(*t*) is the torque produced by the motor with shaft rotation *θ*, and *r* is the radius of the motor drive gear. Hence, the motor torque equation is

$$T(t) = J_m\ddot{\theta} + B_m\dot{\theta} + T_{\text{Load}} \tag{2-46}$$

Defining the transmitted force from the motor assembly to the mass as *f*(*t*), the mass equation of motion is

$$m\ddot{x} + c\dot{x} + kx = c\dot{y} + ky + f \tag{2-47}$$

In order to control the vehicle bounce, we use *z*(*t*) = *x*(*t*) – *y*(*t*) to rewrite the equation as

$$m\ddot{z} + c\dot{z} + kz = f - m\ddot{y} = f(t) - ma(t) \tag{2-48}$$

Using

$$f(t) = \frac{T_{\text{Load}}}{r} \tag{2-49}$$

and noting that $z = \theta r$, Eq. (2-48) is rewritten as

$$(mr^2 + J_m)\ddot{\theta} + (cr^2 + B_m)\dot{\theta} + kr^2\theta = T(t) - mra(t) \tag{2-50}$$

or

$$J\ddot{z} + B\dot{z} + Kz = r[T(t) - mra(t)] \tag{2-51}$$

where $J = mr^2 + J_m$, $B = cr^2 + B_m$, and $K = kr^2$.

So the motor torque and be used to control the vehicle bounce caused by ground disturbances due to acceleration $a(t)$.

2-1-4 Gear Trains

A gear train, lever, or timing belt over a pulley is a mechanical device that transmits energy from one part of the system to another in such a way that force, torque, speed, and displacement may be altered. These devices can also be regarded as matching devices used to attain maximum power transfer. Two gears are shown coupled together in Fig. 2-16. The inertia and friction of the gears are neglected in the ideal case considered.

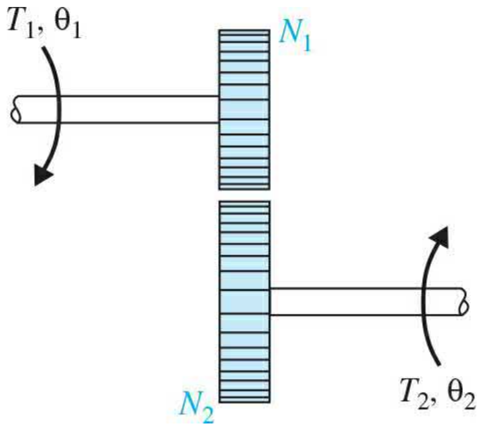


Figure 2-16 Gear train.

The relationships between the torques T_1 and T_2, angular displacement q_1 and q_2, and the teeth numbers N_1 and N_2 of the gear train are derived from the following facts:

1. The number of teeth on the surface of the gears is proportional to the radii r_1 and r_2 of the gears; that is,

$$r_1 N_2 = r_2 N_1 \tag{2-52}$$

2. The distance traveled along the surface of each gear is the same. Thus,

$$\theta_1 r_1 = \theta_2 r_2 \tag{2-53}$$

3. The work done by one gear is equal to that of the other since there are assumed to be no losses. Thus,

$$T_1 \theta_1 = T_2 \theta_2 \tag{2-54}$$

If the angular velocities of the two gears, ω_1 and ω_2, are brought into the picture, Eqs. (2-52) through (2-54) lead to

$$\frac{T_1}{T_2} = \frac{\theta_2}{\theta_1} = \frac{N_1}{N_2} = \frac{\omega_2}{\omega_1} = \frac{r_1}{r_2} \tag{2-55}$$

EXAMPLE 2-1-6 Consider motor-load assembly, shown in Fig. 2-10, with a rigid shaft of inertia J_m. If we use a gear train with gear ratio $\frac{N_1}{N_2} = n$ between the motor shaft and the load of inertia J_L, J is the equivalent moment of inertia of the motor and load, $J = J_L/n^2 + J_m$, and as a result Eq. (2-39) is revised to

$$\frac{d^2\theta_m(t)}{dt^2} + \frac{B_m}{J}\frac{d\theta_m(t)}{dt} = \frac{1}{J}T_m(t) \tag{2-56}$$

▲

In practice, gears do have inertia and friction between the coupled gear teeth that often cannot be neglected. An equivalent representation of a gear train with viscous friction and inertia considered as lumped parameters is shown in Fig. 2-17, where T denotes the applied torque, T_1 and T_2 are the transmitted torque, and B_1 and B_2 are the viscous friction coefficients. The torque equation for gear 2 is

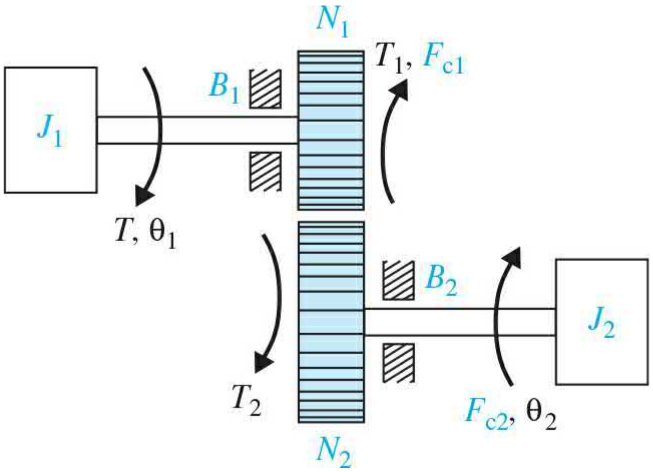


Figure 2-17 Gear train with friction and inertia.

$$T_2(t) = J_2 \frac{d^2\theta_2(t)}{dt^2} + B_2 \frac{d\theta_2(t)}{dt} \tag{2-57}$$

The torque equation on the side of gear 1 is

$$T(t) = J_1 \frac{d^2\theta_1(t)}{dt^2} + B_1 \frac{d\theta_1(t)}{dt} + T_1(t) \tag{2-58}$$

Using Eq. (2-55), Eq. (2-57), after premultiplication by $\frac{N_1}{N_2}$, is converted to

$$T_1(t) = \frac{N_1}{N_2} T_2(t) = \left(\frac{N_1}{N_2}\right)^2 J_2 \frac{d^2\theta_1(t)}{dt^2} + \left(\frac{N_1}{N_2}\right)^2 B_2 \frac{d\theta_1(t)}{dt} \tag{2-59}$$

Equation (2-59) indicates that it is possible to reflect inertia, friction, compliance, torque, speed, and displacement from one side of a gear train to the other. The following quantities are obtained when reflecting from gear 2 to gear 1:

$$\begin{aligned}
&\text{Inertia}: \left(\frac{N_1}{N_2}\right)^2 J_2 \\
&\text{Viscous-friction coefficient}: \left(\frac{N_1}{N_2}\right)^2 B_2 \\
&\text{Torque}: \frac{N_1}{N_2} T_2 \qquad (2\text{-}60) \\
&\text{Angular displacement}: \frac{N_1}{N_2} \theta_2 \\
&\text{Angular velocity}: \frac{N_1}{N_2} \omega_2
\end{aligned}$$

Similarly, gear parameters and variables can be reflected from gear 1 to gear 2 by simply interchanging the subscripts in the preceding expressions. If a torsional spring effect is present, the spring constant is also multiplied by $(N_1/N_2)^2$ in reflecting from gear 2 to gear 1. Now substituting Eq. (2-59) into Eq. (2-58), we get

$$T(t) = J_{1e} \frac{d^2\theta_1(t)}{dt^2} + B_{1e} \frac{d\theta_1(t)}{dt} \qquad (2\text{-}61)$$

where

$$J_{1e} = J_1 + \left(\frac{N_1}{N_2}\right)^2 J_2 \qquad (2\text{-}62)$$

$$B_{1e} = B_1 + \left(\frac{N_1}{N_2}\right)^2 B_2 \qquad (2\text{-}63)$$

EXAMPLE 2-1-7 Given a load that has inertia of 0.05 oz·in·s**2**, find the inertia and frictional torque reflected through a 1:5 gear train (N_1/N_2 = 1/5, with N_2 on the load side). The reflected inertia on the side of N_1 is $(1/5)^2$ × 0.05 = 0.002 oz·in·s**2**. ▲

2-1-5 Backlash and Dead Zone (Nonlinear Characteristics)

Backlash and dead zone are commonly found in gear trains and similar mechanical linkages where the coupling is not perfect. In a majority of situations, backlash may give rise to undesirable inaccuracy, oscillations, and instability in control systems. In addition, it has a tendency to wear out the mechanical elements. Regardless of the actual mechanical elements, a physical model of backlash or dead zone between an input and an output member is shown in Fig. 2-18. The model can be used for a rotational system as well as for a translational system. The amount of backlash is $b/2$ on either side of the reference position.

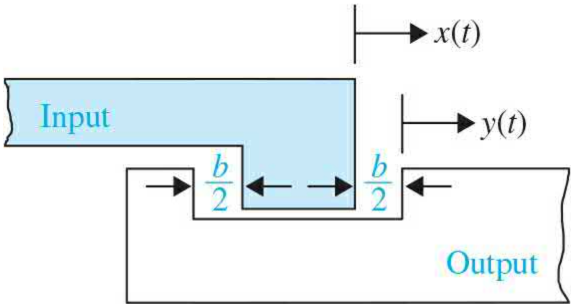


Figure 2-18 Physical model of backlash between two mechanical elements.

In general, the dynamics of the mechanical linkage with backlash depend on the relative inertia-to-friction ratio of the output member. If the inertia of the output member is very small compared with that of the input member, the motion is controlled predominantly by friction. This means that the output member will not coast whenever there is no contact between the two members. When the output is driven by the input, the two members will travel together until the input member reverses its direction; then the output member will be at a standstill until the backlash is taken up on the other side, at which time it is assumed that the output member instantaneously takes on the velocity of the input member. The transfer characteristic between the input and output displacements of a system with backlash with negligible output inertia is shown in Fig. 2-19.

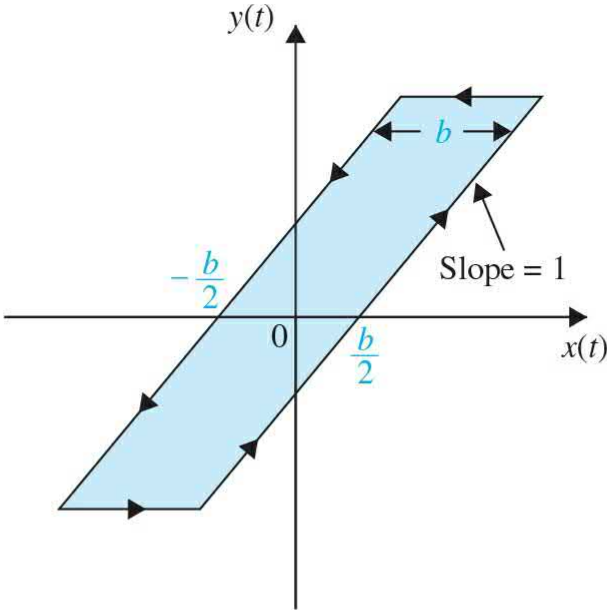


Figure 2-19 Input–output characteristic of backlash.

2-2 INTRODUCTION TO MODELING OF SIMPLE ELECTRICAL SYSTEMS

In this chapter, we address modeling of electrical networks with simple passive elements such as resistors, inductors, and capacitors. The mathematical models of these systems are governed by ordinary differential equations. Later in Chap. 6, we address operational amplifiers, which are active electrical elements and their models are more relevant to controller systems discussions.

2-2-1 Modeling of Passive Electrical Elements

Consider Fig. 2-20, which shows the basic passive electrical elements: resistors, inductors, and capacitors.

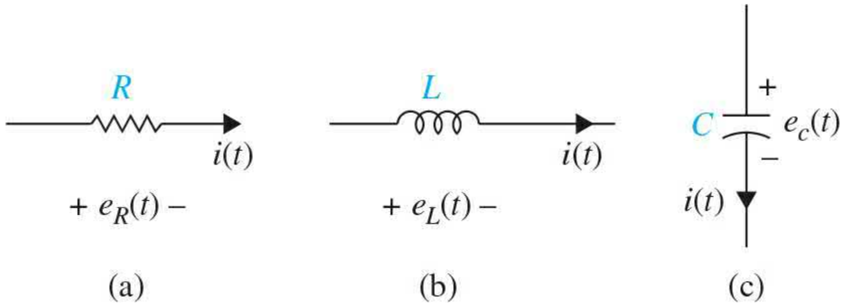


Figure 2-20 Basic passive electrical elements. (*a*) A resistor. (*b*) An inductor. (*c*) A capacitor.

***Resistors*. Ohm's law** states that the voltage drop, $e_R(t)$, across a resistor *R* is proportional to the current *i*(*t*) going through the resistor. Or

$$e_R(t) = i(t)R \tag{2-64}$$

***Inductors*.** The voltage drop, $e_L(t)$, across an inductor *L* is proportional to the time rate of change of current *i*(*t*) going through the inductor. Thus,

$$e_L(t) = L\frac{di(t)}{dt} \tag{2-65}$$

***Capacitor*.** The voltage drop, $e_C(t)$, across a capacitor *C* is proportional to the integral current *i*(*t*) going through the capacitor with respect to time. Therefore,

$$e_c(t) = \int \frac{i(t)}{C}dt \tag{2-66}$$

2-2-2 Modeling of Electrical Networks

The classical way of writing equations of electric networks is based on the loop method or the node method, both of which are formulated from the two laws of Kirchhoff, which state:

> ***Current law or loop method*.** The algebraic summation of all currents entering a node is zero.
>
> ***Voltage law or node method*.** The algebraic sum of all voltage drops around a complete closed loop is zero.

EXAMPLE 2-2-1 Let us consider the **RLC** network shown in Fig. 2-21. Using the voltage law

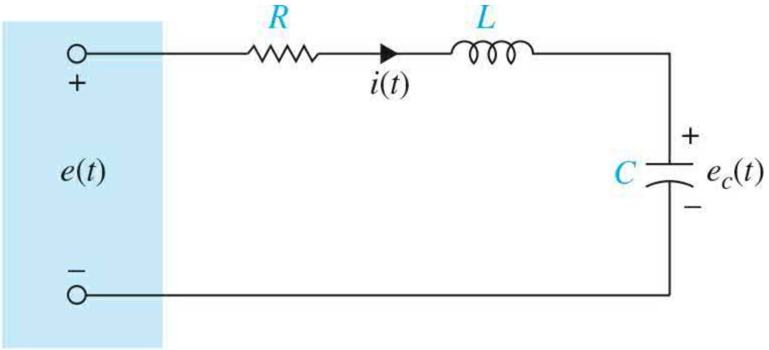


Figure 2-21 RLC network. Electrical schematics.

$$e(t) = e_R + e_L + e_c \tag{2-67}$$

where e_R = Voltage across the resistor *R*

e_L = Voltage across the inductor *L*

e_c = Voltage across the capacitor *C*

Or

$$e(t) = +e_c(t) + Ri(t) + L\frac{di(t)}{dt} \tag{2-68}$$

Using current in *C*,

$$C\frac{de_c(t)}{dt} = i(t) \tag{2-69}$$

and substituting for *i*(*t*) in Eq. (2-68), we get the equation of the RLC network as

$$LC\frac{d^2e_c(t)}{dt^2} + RC\frac{de_c(t)}{dt} + e_c(t) = e(t) \tag{2-70}$$

Dividing the former equation by *LC* and using $\dot{e}_c(t) = \left(\frac{de_c(t)}{dt}\right)$ and $\ddot{e}_c(t) = \left(\frac{d^2e_c(t)}{dt^2}\right)$, we get

$$\ddot{e}_c(t)+\frac{R}{L}\dot{e}_c(t)+\frac{1}{LC}e_c(t)=\frac{1}{LC}e(t) \tag{2-71}$$

In control systems it is customary to rewrite Eq. (2-70) as

$$\ddot{e}_c(t)+2\zeta\omega_n\dot{e}_c(t)+\omega_n^2 e_c(t)=\omega_n^2 e(t) \tag{2-72}$$

where ω_n and ζ are the *natural frequency* and the *damping ratio* of the system, respectively. As in Eq. (2-10), Eq. (2-72) is also known as the **prototype second-order system**. We define $e_c(t)$ as the **output** and $e(t)$ as the **input** of the system, where both terms have same units. Notice that this system is also **analogous** to the translational Mechanical system in Fig. 2-2. ▲

EXAMPLE 2-2-2 Another example of an electric network is shown in Fig. 2-22. The voltage across the capacitor is $e_c(t)$ and the currents of the inductors are $i_1(t)$ and $i_2(t)$, respectively. The equations of the network are

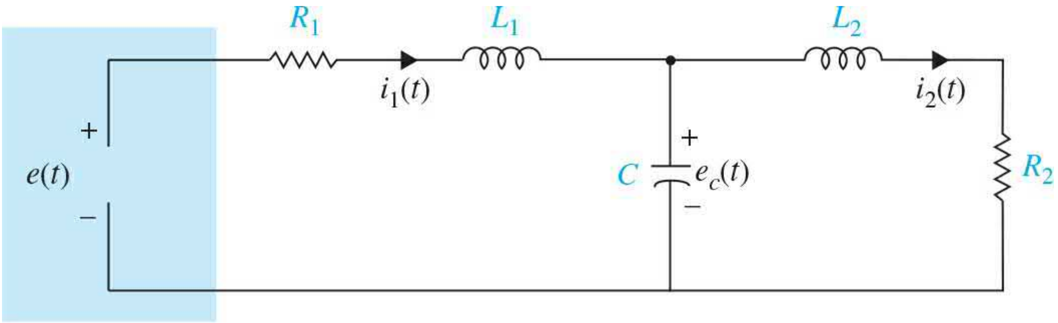


Figure 2-22 Electrical schematic for network of Example 2-2-2.

$$L_1\frac{di_1(t)}{dt}+R_1 i_1(t)+e_c(t)=e(t) \tag{2-73}$$

$$L_2\frac{di_2(t)}{dt}+R_2 i_2(t)=e_c(t) \tag{2-74}$$

$$C\frac{de_c(t)}{dt}=i_1(t)-i_2(t) \tag{2-75}$$

Differentiating Eqs. (2-73) and (2-74) and substituting Eq. (2-75), we get

$$L_1\frac{d^2i_1(t)}{dt^2}+R_1\frac{di_1(t)}{dt}+i_1(t)-i_2(t)=e(t) \tag{2-76}$$

$$L_2\frac{d^2i_2(t)}{dt^2}+R_2\frac{di_2(t)}{dt}-i_1(t)+i_2(t)=0 \tag{2-77}$$

Exploring the similarity of this system with that represented in Example 2-1-4, we find the two systems are analogous when $R_2 = 0$—compare Eqs. (2-76) and (2-77) with Eqs. (2-37) and (2-38). ▲

EXAMPLE 2-2-3 Consider the *RC* circuit shown in Fig. 2-23. Find the differential equation of the system. Using the voltage law

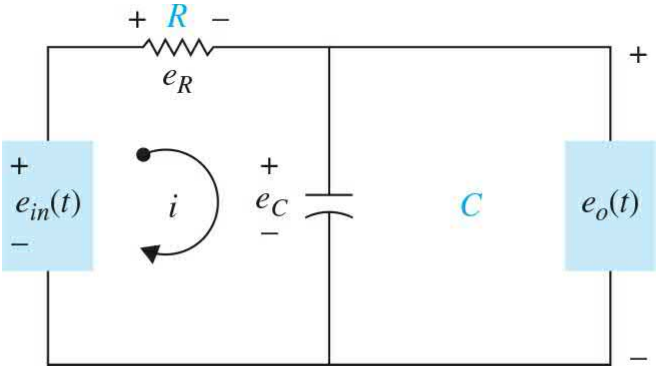


Figure 2-23 A simple electrical *RC* circuit.

$$e_{in}(t)=e_R(t)+e_C(t) \tag{2-78}$$

where

$$e_R=iR \tag{2-79}$$

and the voltage across the capacitor $e_c(t)$ is

$$e_C(t)=\frac{1}{C}\int idt \tag{2-80}$$

But from Fig. 2-21

$$e_o(t) = \frac{1}{C}\int i\,dt = e_C(t) \tag{2-81}$$

If we differentiate Eq. (2-81) with respect to time, we get

$$\frac{i}{C} = \frac{de_o(t)}{dt} \tag{2-82}$$

or

$$C\dot{e}_o(t) = i \tag{2-83}$$

This implies that Eq. (2-78) can be written in an input–output form

$$RC\dot{e}_o(t) + e_o(t) = e_{in}(t) \tag{2-84}$$

where the $\tau = RC$ is also known as the **time constant** of the system. The significance of this term is discussed later in Chaps. 3, 6, and 7. Using this term the equation of the system is rewritten in the form of a **standard first-order prototype** system.

$$\dot{e}_o(t) + \frac{1}{\tau}e_o(t) = \frac{1}{\tau}e_{\text{in}}(t) \tag{2-85}$$

Notice that Eq. (2-85) is analogous to Eq. (2-8), when $M = 0$. ▲

EXAMPLE 2-2-4 Consider the *RC* circuit shown in Fig. 2-24. Find the differential equation of the system.

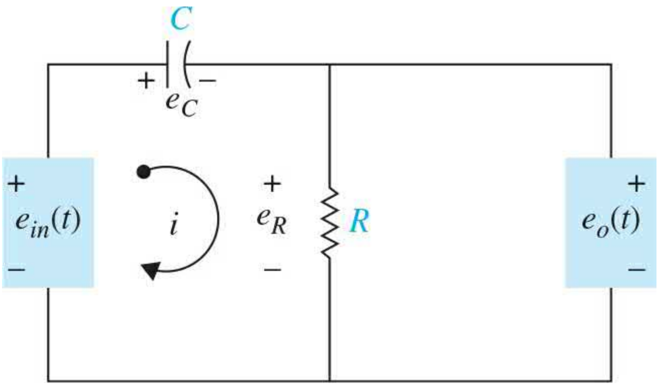


Figure 2-24 Simple electrical *RC* circuit.

As before, we have

$$e_{\text{in}}(t) = e_c(t) + e_R(t) \tag{2-86}$$

or

$$e_{\text{in}}(t) = \frac{1}{C}\int i\,dt + iR \tag{2-87}$$

But $e_o(t) = iR$. So

$$e_{\text{in}}(t) = \frac{\int e_o(t)\,dt}{RC} + e_o(t) \tag{2-88}$$

is the differential equation of the system. To solve Eq. (2-88), we differentiate once with respect to time

$$\dot{e}_{\text{in}}(t) = \frac{e_o(t)}{RC} + \dot{e}_o(t) \tag{2-89}$$

where, again, $\tau = RC$ is the **time constant** of the system. ▲

EXAMPLE 2-2-5 Consider the voltage divider of Fig. 2-25. Given an input voltage $e_0(t)$, find the output voltage $e_1(t)$ in the circuit composed of two resistors R_1 and R_2.

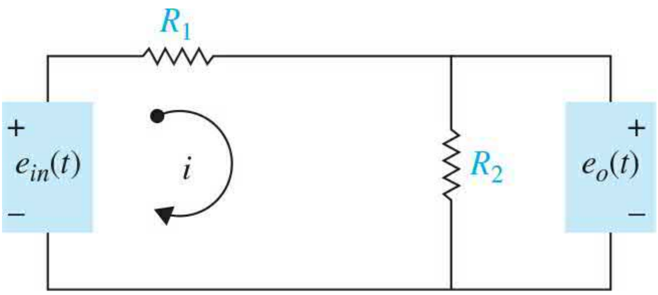


Figure 2-25 A voltage divider.

The currents in the resistors are

$$i = \frac{e_{\text{in}}(t) - e_o(t)}{R_1} \tag{2-90}$$

$$i = \frac{e_o(t)}{R_2} \tag{2-91}$$

Equating Eqs. (2-90) and (2-91), we have

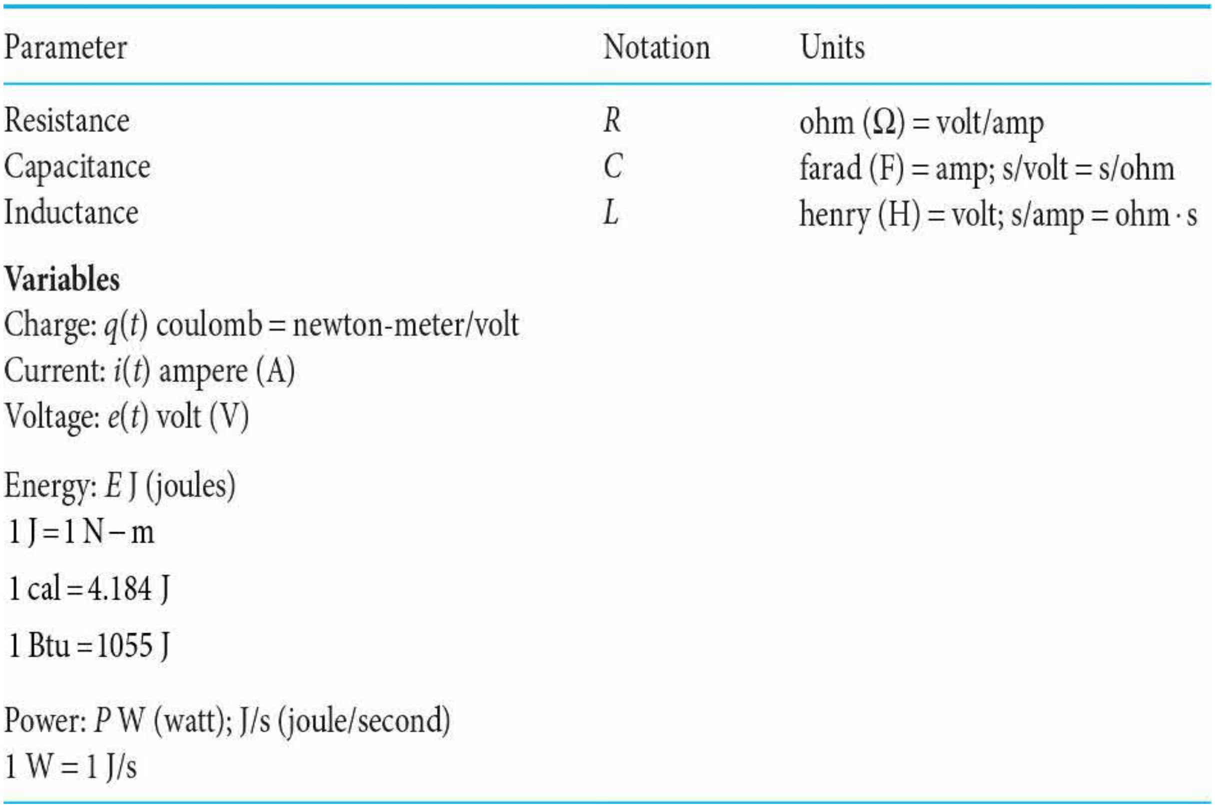
$$\frac{e_{\text{in}}(t) - e_o(t)}{R_1} = \frac{e_o(t)}{R_2} \tag{2-92}$$

Rearrangement of this equation yields the following equation for the voltage divider:

$$e_o(t) = \frac{R_2}{R_1 + R_2} e_{\text{in}}(t) \tag{2-93}$$

The SI and most other measurement units for variables in electrical systems are the same, as shown in Table 2-2.

TABLE 2-2 Basic Electrical System Properties and Their Units



Parameter	Notation	Units
Resistance	R	ohm (Ω) = volt/amp
Capacitance	C	farad (F) = amp; s/volt = s/ohm
Inductance	L	henry (H) = volt; s/amp = ohm · s

Variables

Charge: $q(t)$ coulomb = newton-meter/volt

Current: $i(t)$ ampere (A)

Voltage: $e(t)$ volt (V)

Energy: E J (joules)

$1\,\mathrm{J} = 1\,\mathrm{N} - \mathrm{m}$

$1\,\mathrm{cal} = 4.184\,\mathrm{J}$

$1\,\mathrm{Btu} = 1055\,\mathrm{J}$

Power: P W (watt); J/s (joule/second)

$1\,\mathrm{W} = 1\,\mathrm{J/s}$

2-3 INTRODUCTION TO MODELING OF THERMAL AND FLUID SYSTEMS

In this section, we review thermal and fluid systems. Knowledge of these systems is important in many mechanical and chemical engineering control system applications such as in power plants, fluid power control systems or temperature control system. Because of the complex mathematics associated with these nonlinear systems, we only focus on basic and simplified models.

2-3-1 Elementary Heat Transfer Properties[3]

In a thermal system, we look at transfer of heat among different components. The two key variables in a thermal process are temperature T and thermal storage or heat stored Q, which has the same units as energy (e.g., J or joules in SI units). Also heat transfer systems include thermal capacitance and resistance properties, which are analogous to same properties mentioned in electrical systems. Heat transfer is related to the heat flow rate q, which has the units of power. That is,

$$q = \dot{Q} \tag{2-94}$$

As in the electrical systems, the concept of **capacitance** in a heat transfer problem is related to storage (or discharge) of heat in a body. The capacitance *C* is related to the change of the body temperature *T* with respect to time and the rate of heat flow *q*:

> **Capacitance** in a heat transfer problem is related to storage (or discharge) of heat in a body.

$$q = C\dot{T} \tag{2-95}$$

where the thermal capacitance *C* can be stated as a product of *r* material density, *c* material specific heat, and volume *V*:

$$C = \rho c_p V \tag{2-96}$$

In a thermal system, there are three different ways that heat is transferred. That is by **conduction**, **convection**, or **radiation**.

Conduction

Thermal conduction describes how an object conducts heat. In general this type of heat transfer happens in solid materials due to a temperature difference between two surfaces. In this case, heat tends to travel from the hot to the cold region. The transfer of energy in this case takes place by molecule diffusion and in a direction perpendicular to the object surface. Considering one-directional steady-state heat conduction along *x*, as shown in Fig. 2-26, the rate of heat transfer is given by

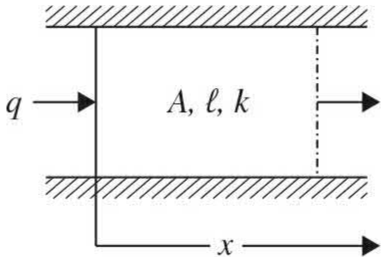


Figure 2-26 One-directional heat conduction flow.

$$q = \frac{kA}{\ell} \Delta T = D_{1-2} \Delta T \tag{2-97}$$

where q is the rate of heat transfer (flow), k is the thermal conductivity related to the material used, A is the area normal to the direction of heat flow x, and $\Delta T = T_1 - T_2$ is the difference between the temperatures at $x = 0$ and $x = l$, or T_1 and T_2. Note in this case, assuming a perfect insulation, the heat conduction in other directions is zero. Also note that

$$D_{1-2} = \frac{1}{\frac{\ell}{kA}} = \frac{1}{R} \tag{2-98}$$

where R is also known as **thermal resistance**. So the rate of heat transfer q may be represented in terms of R as

> **Thermal resistance** is a property of materials to resist the flow of heat.

$$q = \frac{\Delta T}{R} \tag{2-99}$$

Convection

This type of heat transfer occurs between a solid surface and a fluid exposed to it, as shown in Fig. 2-27. At the boundary where the fluid and the solid surface meet, the heat transfer process is by conduction. But once the fluid is exposed to the heat, it can be replaced by new fluid. In thermal convection, the heat flow is given by

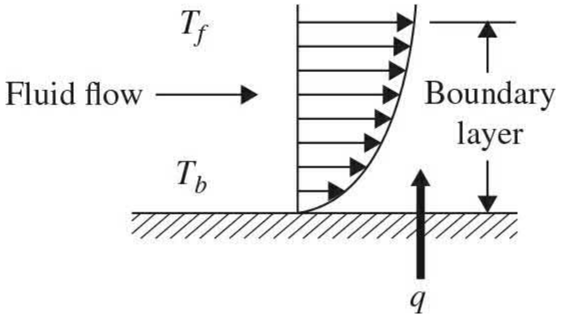


Figure 2-27 Fluid-boundary heat convection.

$$q = hA\Delta T = D_0 \Delta T \tag{2-100}$$

where q is the rate of heat transfer or heat flow, h is the coefficient of convective heat transfer, A is the area of heat transfer, and $\Delta T = T_b - T_f$ is the difference between the boundary and fluid temperatures. The term hA may be denoted by D_0, where

$$D_0 = hA = \frac{1}{R} \tag{2-101}$$

Again, the rate of heat transfer q may be represented in terms of thermal resistance R. Thus,

$$q = \frac{\Delta T}{R} \tag{2-102}$$

Radiation

The rate of heat transfer through radiation between two separate objects is determined by the Stephan-Boltzmann law,

$$q = \sigma A\left(T_1^4 - T_2^4\right) \tag{2-103}$$

where q is the rate of heat transfer, σ is the Stephan-Boltzmann constant and is equal to 5.667×10^{-8} W/m^2.K^4, A is the area normal to the heat flow, and T_1 and T_1 are the absolute temperatures of the two bodies. Note that Eq. (2-103) applies to directly opposed ideal radiators of equal surface area A that perfectly absorb all the heat without reflection (Fig. 2-28).

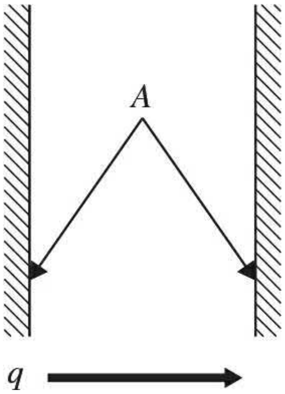
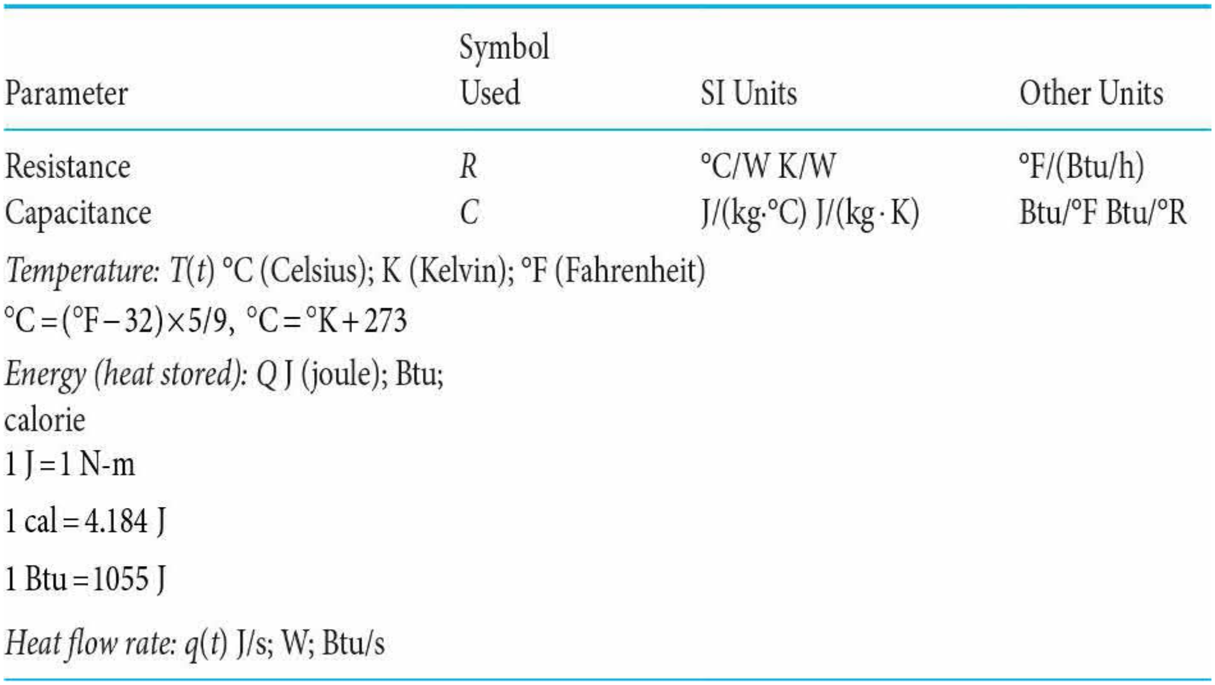


Figure 2-28 A simple heat radiation system with directly opposite ideal radiators.

The SI and other measurement units for variables in thermal systems are shown in Table 2-3.

TABLE 2-3 Basic Thermal System Properties and Their Units



Parameter	Symbol Used	SI Units	Other Units
Resistance	R	°C/W K/W	°F/(Btu/h)
Capacitance	C	J/(kg·°C) J/(kg · K)	Btu/°F Btu/°R
Temperature: $T(t)$ °C (Celsius); K (Kelvin); °F (Fahrenheit)			
°C = (°F − 32) × 5/9, °C = °K + 273			
Energy (heat stored): Q J (joule); Btu; calorie			
1 J = 1 N-m			
1 cal = 4.184 J			
1 Btu = 1055 J			
Heat flow rate: $q(t)$ J/s; W; Btu/s			

EXAMPLE 2-3-1 A rectangular object is composed of a material that is in contact with fluid on its top side while being perfectly insulated on three other sides, as shown in Fig. 2-29. Find the equations of the heat transfer process for the following:

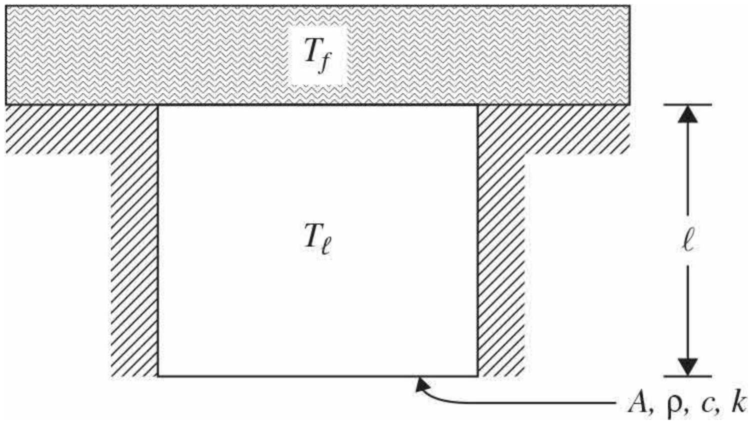


Figure 2-29 Heat transfer problem between a fluid and an insulated solid object.

T_l = solid object temperature; assume that the temperature distribution is uniform

T_f = top fluid temperature

ℓ = length of the object

A = cross-sectional area of the object

p = material density

c = material specific heat

k = material thermal conductivity

h = coefficient of convective heat transfer

SOLUTION The rate of heat storage in the solid from Eq. (2-95) is

$$q = \rho c A \ell \left(\frac{dT_\ell}{dt} \right) \tag{2-104}$$

Also, the convection rate of heat transferred from the fluid is

$$q = hA(T_f - T_\ell) \tag{2-105}$$

The energy balance equation for the system dictates q to be the same in Eqs. (2-104) and (2-105). Hence, upon introducing thermal capacitance C from Eq. (2-95) and the convective thermal resistance R from Eq. (2-99) and substituting the right-hand sides of Eq. (2-104) into Eq. (2-105), we get

$$RC\dot{T}_\ell + T_\ell = T_f \tag{2-106}$$

where the $RC = \tau$ is also known as the **time constant** of the system. Notice that Eq. (2-106) is analogous to the electrical system modeled by Eq. (2-84).

2-3-2 Elementary Fluid System Properties[4]

In this section, we derive the equations of fluid systems. The key application in control systems associated with fluid systems is in the area of fluid power control. Understanding the behavior of fluid systems will help appreciating the models of hydraulic actuators. In fluid systems, there are five parameters of importance—pressure, flow mass (and flow rate), temperature, density, and flow volume (and volume rate). Our focus will primarily be on **incompressible fluid systems** because of their application to elements of popular industrial control systems such as hydraulic actuators and dampers. In case of incompressible fluids, the fluid volume remains constant, and just like electrical systems, they can be modeled by passive components including resistance, capacitance, and inductance.

> For an **incompressible fluid**, density ρ **is constant**, and the fluid capacitance C is the ratio of the volumetric fluid flow rate q to the rate of pressure P.

To understand these concepts better, we must look at the **fluid continuity equation** or **the law of conservation of mass**. For the control volume shown in Fig. 2-30 and the net mass flow rate $q_m = \rho q$, we have

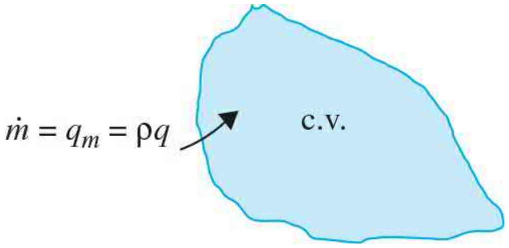


Figure 2-30 Control volume and the net mass flow rate.

$$m = \int \rho q\, dt \tag{2-107}$$

where *m* is the net mass flow, *ρ* is fluid density, $\frac{dV}{dt} = q = q_i - q_o$ is the net volumetric fluid flow rate (volume flow rate of the ingoing fluid q_i minus volume flow rate of the outgoing fluid q_o). The **conservation of mass** states

$$\frac{dm}{dt} = \rho q = \frac{d}{dt}(M_{cv}) = \frac{d}{dt}(\rho V) \tag{2-108}$$

$$\frac{dm}{dt} = \rho \dot{V} + V\dot{\rho} \tag{2-109}$$

where *m* is the net mass flow rate, M_{cv} is the mass of the control volume (or for simplicity “the container” fluid), and *V* is the container volume. Note

$$\frac{dV}{dt} = q_i - q_o = q \tag{2-110}$$

which is also known as the **conservation of volume** for the fluid. For an **incompressible fluid,** ***ρ*** **is constant**. Hence setting *ρ* = 0 in Eq. (2-109), the conservation of mass for an incompressible fluid is

$$\dot{m} = \rho \dot{V} = \rho q \tag{2-111}$$

Capacitance—Incompressible Fluids

Similar to the electrical capacitance, fluid capacitance relates to how energy can be stored in a fluid system. **The fluid capacitance C is the change in the fluid volume that is stored over the pressure change**. Alternatively capacitance is defined as the ratio of the volumetric fluid flow rate *q* to the rate of pressure *P* as follows:

$$C = \frac{\dot{V}}{\dot{P}} = \frac{q}{\dot{P}} \tag{2-112}$$

or

$$q = C\dot{P} \tag{2-113}$$

EXAMPLE 2-3-2 In a one-tank liquid-level system, the fluid pressure in the tank that is filled to height *h* (also known as head), shown in Fig. 2-31, is the weight of the fluid over the cross-sectional area, or

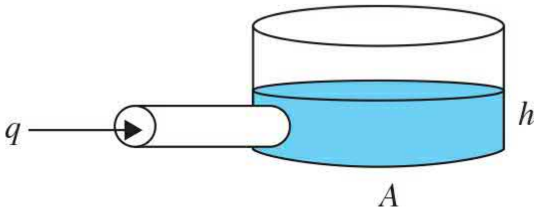


Figure 2-31 Incompressible fluid flow into an open-top cylindrical container.

$$P = \frac{\rho V g}{A} = \frac{\rho h g A}{A} = \rho h g \tag{2-114}$$

As a result, from Eqs. (2-112) and noting *V* = *Ah*, we get

$$C = \frac{\dot{V}}{\dot{P}} = \frac{A\dot{h}}{\rho g \dot{h}} = \frac{A}{\rho g} \tag{2-115}$$

▲

In general, the fluid density *ρ* is nonlinear and may depend on temperature and pressure. This nonlinear dependency *ρ*(*P*,*T*), known as **the equation of state**, may be linearized using the first-order Taylor series relating *r* to *P* and *T*:

$$\rho = \rho_{\text{ref}} + \left(\frac{\partial \rho}{\partial P}\right)_{P_{\text{ref}},T_{\text{ref}}} (P - P_{\text{ref}}) + \left(\frac{\partial \rho}{\partial T}\right)_{P_{\text{ref}},T_{\text{ref}}} (T - T_{\text{ref}}) \tag{2-116}$$

where ρ_{ref}, P_{ref}, and T_{ref} are constant reference values of density, pressure, and temperature, respectively. In this case,

$$\beta = \frac{1}{\rho_{\text{ref}}}\left(\frac{\partial \rho}{\partial P}\right)_{P_{\text{ref}},T_{\text{ref}}} \tag{2-117}$$

$$\alpha = -\frac{1}{\rho_{\text{ref}}}\left(\frac{\partial \rho}{\partial T}\right)_{P_{\text{ref}},T_{\text{ref}}} \tag{2-118}$$

are the **bulk modulus** and the **thermal expansion coefficient**, respectively. In most cases of interest, however, the temperatures of the fluid entering and flowing out of the container are almost the same. Hence recalling the control volume in Fig. 2-30, the conservation of mass Eq. (2-108) reflects both changes in volume and density as

$$\frac{dm}{dt} = \frac{d\rho}{dt}V + \rho\frac{dV}{dt} \tag{2-119}$$

If the container of volume *V* is a **rigid object**, *V* = 0. Hence,

$$\frac{dm}{dt} = \frac{d\rho}{dt}V \tag{2-120}$$

Substituting the time derivative of Eq. (2-116), assuming no temperature dependency, into Eq. (2-120) and using Eq. (2-117), the capacitance relation may be obtained as

$$q = \frac{V}{\beta}\dot{P} = C\dot{P} \tag{2-121}$$

In general, density may depend on temperature and pressure. In the latter case, the fluid is considered to be **compressible**.

Note that $\frac{dm}{dt} = q_m = \rho_{\text{ref}}q$ was used to arrive at Eq. (2-121). As a result in the case of a **compresible** fluid inside a rigid object, the capacitance is

$$C = \frac{V}{\beta} \tag{2-122}$$

EXAMPLE 2-3-3 In practice, accumulators are fluid capacitors, which may be modeled as a spring-loaded piston systems as shown in Fig. 2-32. In this case, assuming a spring-loaded piston of area *A* traveling inside a rigid cylindrical container, using the conservation of mass Eq. (2-119) for compressible fluids, we get

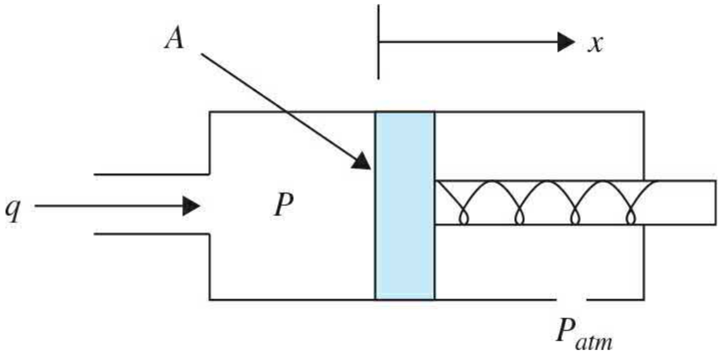


Figure 2-32 A spring-loaded piston system.

$$\rho_{\text{ref}}\, q = \frac{d\rho}{dt} V + \rho_{\text{ref}} \frac{dV}{dt} \tag{2-123}$$

Assuming a compressible fluid with no temperature dependency, taking a time derivative of Eq. (2-116) and using Eq. (2-117), we have

$$\frac{d\rho}{dt} = \frac{\rho_{\text{ref}}}{\beta} \frac{dP}{dt} \tag{2-124}$$

Combining Eqs. (2-123) and (2-124) and using Eq. (2-122), the pressure rise rate within the varying control volume of Fig. 2-42 is shown as

$$\dot{P} = \frac{\beta}{V}(q - \dot{V}) = \frac{1}{C}(q - \dot{V}) \tag{2-125}$$

where $\dot{V} = A\dot{x}$. This equation reflects that the rate of change of pressure inside a varying control volume is related to the entering fluid volumetric flow rate and the rate of change of chamber volume itself.

Inductance—Incompressible Fluids

Fluid inductance is also referred to as fluid **inertance** in relation to the inertia of a moving fluid inside a passage (line or a pipe). Inertance occurs mainly in long lines, but it can also occur where an external force (e.g., caused by a pump) causes a significant change in the flow rate. In the case shown in Fig. 2-33, assuming a frictionless pipe with a uniform fluid flow moving at the speed *v*, in order to accelerate the fluid, an external force *F* is applied. From Newton's second law,

Inductance (or **inertance**) occurs mainly in long pipes or where an external force causes a significant change in the flow rate.

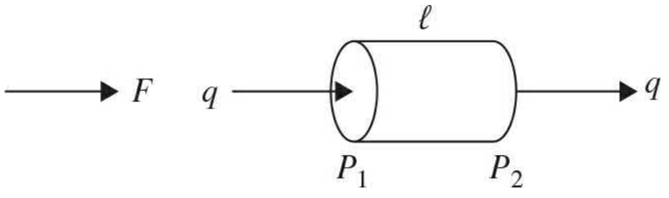


Figure 2-33 A uniform incompressible fluid flow forced through a frictionless pipe.

$$\begin{aligned} F &= A\Delta P = M\dot{v} = \rho A\ell\dot{v} \\ \Delta P &= (P_1 - P_2) \end{aligned} \tag{2-126}$$

But

$$\dot{V} = Av = q \tag{2-127}$$

So

$$(P_1 - P_2) = L\dot{q} \tag{2-128}$$

where

$$L = \frac{\rho \ell}{A} \tag{2-129}$$

is known as the **fluid inductance.** Note that the concept of inductance is rarely discussed in the case of compressible fluids and gasses.

Resistance—Incompressible Fluids

As in the electrical systems, fluid resistors dissipate energy. However, there is no unique definition for this term. In this textbook, we adopt the most common term, which relates fluid resistance to pressure change. For the system shown in Fig. 2-34, the force resisting the fluid passing through a passage like a pipe is

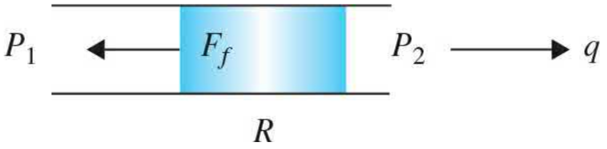


Figure 2-34 Flow of an incompressible fluid through a pipe and a fluid resistor *R*.

In this textbook, fluid **resistance** relates the pressure drop to the volumetric flow rate *q*.

$$F_f = A\Delta P = A(P_1 - P_2) \tag{2-130}$$

where $\Delta P = P_1 - P_2$ is the pressure drop and *A* is the cross-sectional area of the pipe. Depending on the type of flow (i.e., laminar or turbulent) the fluid resistance relationship can be linear or nonlinear and relates the pressure drop to the volumetric flow rate *q*. For a laminar flow, we define

$$\Delta P = Rq \tag{2-131}$$

$$R = \frac{\Delta P}{q} \tag{2-132}$$

where *q* is the volume flow rate. Table 2-4 shows resistance *R* for various passage cross sections, assuming a laminar flow.

TABLE 2-4 Equations of Resistance R for Laminar Flows

Fluid Resistance	
Symbols used	**Fluid volume flow rate:** q **Pressure drop:** $\Delta P = P_{12} = P_1 - P_2$ **Laminar resistance:** R μ**: Fluid viscosity** $w =$ width; $h =$ height; $\ell =$ length; $d =$ diameter
General case	$R = \dfrac{32\,\mu\ell}{Ad_h^2}$ $d_h =$ **hydraulic diameter** $= \dfrac{4A}{perimeter}$
Circular cross section	$R = \dfrac{128\mu\ell}{\pi d^4}$
Square cross section	$R = \dfrac{32\mu\ell}{w^4}$
Rectangular cross section	$R = \dfrac{8\mu\ell}{\dfrac{wh^3}{(1+h/w)^2}}$
Rectangular cross section: approximation	$R = \dfrac{12\mu\ell}{wh^3}$ $w/h =$ small
Annular cross section	$R = \dfrac{8\mu\ell}{\pi d_o d_i^3 \left(1 - \dfrac{d_i}{d_o}\right)}$ $d_o =$ outer diameter; $d_i =$ inner diameter
Annular cross section: approximation	$R = \dfrac{12\mu\ell}{\pi d_o d_i^3}$ $d_o/d_i =$ small

When the flow becomes **turbulent**, the pressure drop relation Eq. (2-131) is rewritten as

$$\Delta P = R_T q^n \tag{2-133}$$

where R_T is the turbulent resistance and n is a power varying depending on the boundary used—for example, n = 7/4 for a long pipe and, most useful, n = 2 for a flow through an orifice or a valve.

In order to get a sense of the laminar and turbulent flows and their corresponding resistance terms, you may wish to conduct a simple experiment by applying a force on the plunger syringe filled with water. If you push the plunger with a gentle force, the water is expelled easily from the other end through the syringe orifice. However, application of a strong force would cause a strong resistance. In the former case, you encounter a mild resistance due to the laminar flow, while in the latter case the resistance is high because of the turbulent flow.

EXAMPLE 2-3-4 For the liquid-level system shown in Fig. 2-35, water or any **incompressible fluid** (i.e., fluid density *r* is constant) enters the tank from the top and exits through the valve with resistance *R* in the bottom. The fluid height (also known as head) in the tank is *h* and is **variable**. The valve resistance is *R*. Find the system equation for the input, q_i, and output, *h*.

A One-Tank Liquid-Level System

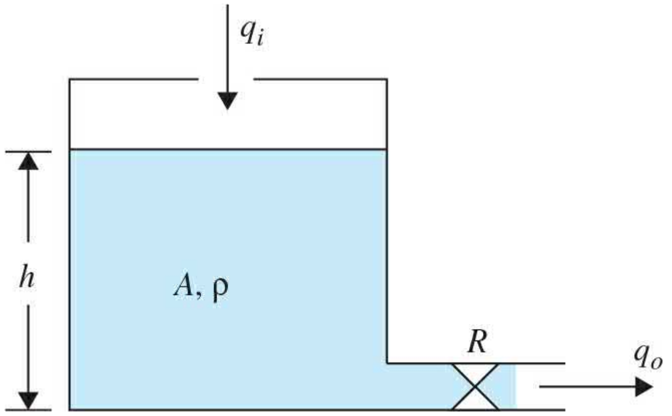


Figure 2-35 A single-tank liquid-level system.

SOLUTION The conservation of mass suggests

$$\frac{dm}{dt}=\frac{d(\rho V)}{dt}=\rho q_i-\rho q_o \tag{2-134}$$

where ρq_i and ρq_o are the mass flow rate in and out of the valve, respectively. Because the fluid density ρ is a constant, the conservation of volume also applies, which suggests the time rate of change of the fluid volume inside the tank is equal to the difference of incoming and outgoing flow rates.

$$\frac{d(V)}{dt}=\frac{d(Ah)}{dt}=q_i-q_o \tag{2-135}$$

Recall from Eq. (2-112) the tank fluid **capacitance** is

$$C=\frac{\dot{V}}{\dot{P}}=\frac{A\dot{h}}{\rho g\dot{h}}=\frac{A}{\rho g} \tag{2-136}$$

where P is the rate of change of fluid pressure at the outlet valve. From Eq. (2-132), resistance R at the valve, assuming a laminar flow, is defined as

$$R=\frac{\Delta P}{q_o} \tag{2-137}$$

where $\Delta P = P_o - P_{atm}$ is the pressure drop across the valve. Relating the pressure to fluid height h, which is variable, we get

$$P_o=P_{\text{atm}}+\rho gh \tag{2-138}$$

where P_o is in the pressure at the valve and P_{atm} is the atmospheric pressure. Hence, from Eq. (2-137), we obtain

$$q_o = \frac{\rho g h}{R} \tag{2-139}$$

After combining Eqs. (2-134) and (2-139), and using the relationship for capacitance from Eq. (2-136), we get the system equation

$$RC\frac{dh}{dt} + h = \frac{R}{\rho g} q_i \tag{2-140}$$

Or using Eq. (2-139) we can also find the system equation in terms of the volumetric flow rate

$$RC\dot{q}_o + q_o = q_i \tag{2-141}$$

where system time constant is $\tau = RC$. This system is **analogous** to the electrical system represented by Eq. (2-85).

EXAMPLE 2-3-5 The liquid-level system shown in Fig. 2-36 is the same as that in Fig. 2-35, except the drainage pipe is long with the length of ℓ.

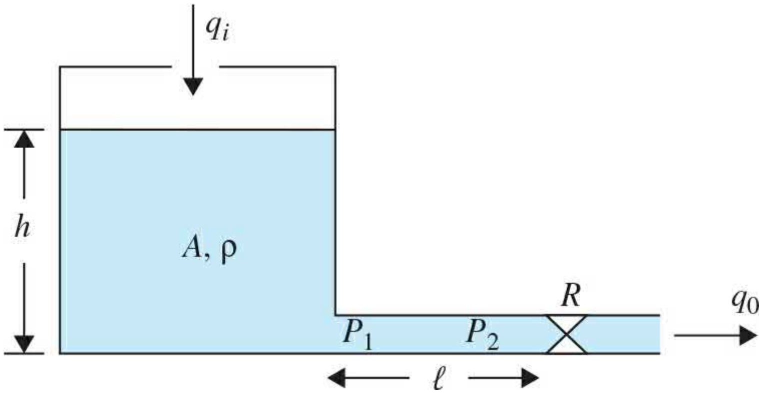


Figure 2-36 A single-tank liquid-level system.

In this case the pipe will have the following inductance

$$(P_1 - P_2) = \ell \dot{q}_o \tag{2-142}$$

As in the previous example, at the valve the resistance is

$$R = \frac{P_2 - P_{\text{atm}}}{q_o} \tag{2-143}$$

And the tank fluid capacitance is the same as Example 2-3-4:

$$C = \frac{A}{\rho g} \tag{2-144}$$

Substituting Eq. (2-143) into Eq. (2-142) and using $P_1 = P_{atm} + \rho gh$, we get

$$\rho gh = L\dot{q}_o + Rq_o \tag{2-145}$$

But from the conservation of volume we also have

$$\frac{d(V)}{dt} = A\dot{h} = q_i - q_o \tag{2-146}$$

Differentiating Eq. (2-145) we can modify Eq. (2-146) in terms of input, q_i, and output, q_o. That is,

$$\frac{L}{\rho g}\ddot{q}_o + \frac{R}{\rho g}\dot{q}_o = \frac{1}{A}q_i - \frac{1}{A}\dot{q}_o \tag{2-147}$$

Using the capacitance formula in Eq. (2-144), Eq. (2-147) is modified to

$$LC\ddot{q}_o + RC\dot{q}_o + q_o = q_i \tag{2-148}$$

▲

EXAMPLE 2-3-6 Consider a double-tank system, as shown in Fig. 2-37, with h_1 and h_1 representing the two tank heights and R_1 and R_2 representing the two valve resistances, respectively. We label the pressure in the bottom of tanks 1 and 2 as P_1 and P_2, respectively. Further, the pressure at the outlet of tank 2 is $P_3 = P_{atm}$. Find the differential equations.

A Two-Tank Liquid-Level System

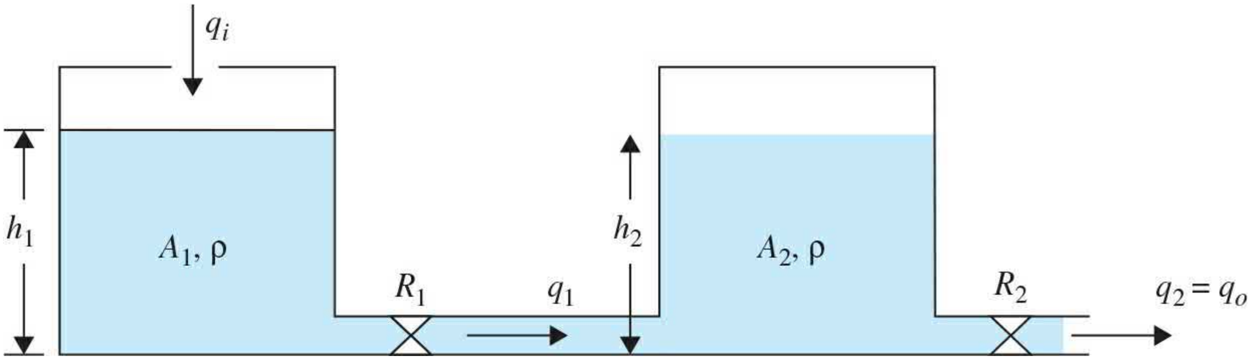


Figure 2-37 Two-tank liquid-level system.

SOLUTION Using the same approach as Example 2-3-4, it is not difficult to see for tank 1:

$$\frac{d(V_1)}{dt} = A_1\dot{h}_1 = q_i - q_1$$

$$= q_i - \frac{P_1 - P_2}{R_1} = q_i - \frac{(P_{\text{atm}} + \rho g h_1) - (P_{\text{atm}} + \rho g h_2)}{R_1} \tag{2-149}$$

and for tank 2:

$$\frac{d(V_2)}{dt} = A_2\dot{h}_2 = q_1 - q_2 = \frac{P_1 - P_2}{R_1} - \frac{P_2 - P_3}{R_2}$$

$$= \frac{(P_{\text{atm}} + \rho g h_1) - (P_{\text{atm}} + \rho g h_2)}{R_1} - \frac{(P_{\text{atm}} + \rho g h_2) - P_{\text{atm}}}{R_2} \tag{2-150}$$

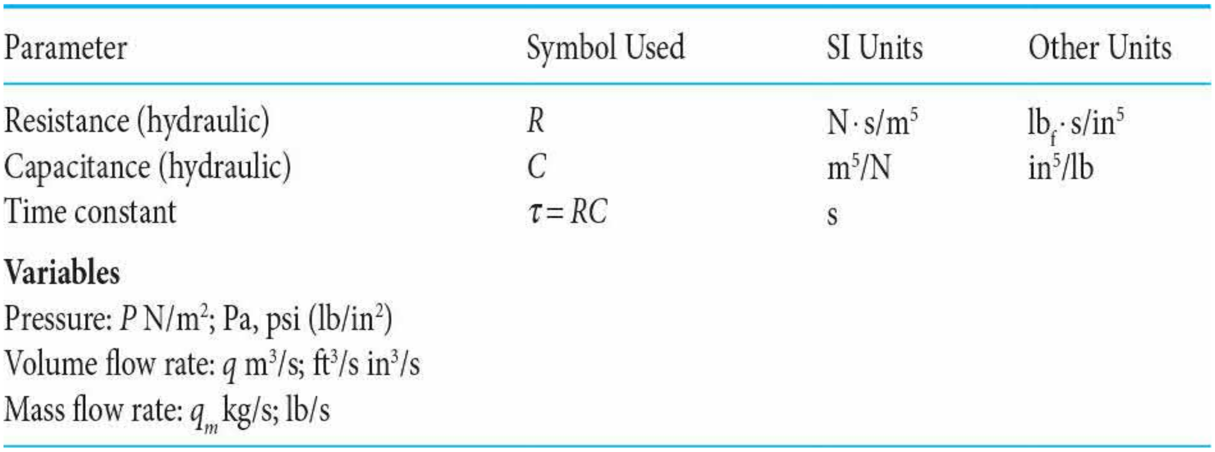
Thus, the equations of the system are

$$A_1\dot{h}_1 + \frac{\rho g h_1}{R_1} - \frac{\rho g h_2}{R_1} = q_i \tag{2-151}$$

$$A_2\dot{h}_2 - \frac{\rho g h_1}{R_1} + \left(\frac{1}{R_1} + \frac{1}{R_2}\right)\rho g h_2 = 0 \tag{2-152}$$

The SI and other measurement units for variables in fluid systems are tabulated in Table 2-5.

TABLE 2-5 Basic Fluid System Properties and Their Units



Parameter	Symbol Used	SI Units	Other Units
Resistance (hydraulic)	R	$\mathrm{N \cdot s/m^5}$	$\mathrm{lb_f \cdot s/in^5}$
Capacitance (hydraulic)	C	$\mathrm{m^5/N}$	$\mathrm{in^5/lb}$
Time constant	$\tau = RC$	s	
Variables			
Pressure: P $\mathrm{N/m^2}$; Pa, psi ($\mathrm{lb/in^2}$)			
Volume flow rate: q $\mathrm{m^3/s}$; $\mathrm{ft^3/s}$ $\mathrm{in^3/s}$			
Mass flow rate: q_m kg/s; lb/s			

2-4 LINEARIZATION OF NONLINEAR SYSTEMS

From the discussions given in the preceding sections on basic system modeling, we should realize that most components found in physical systems have nonlinear characteristics. In practice, we may find that some devices have moderate nonlinear characteristics, or nonlinear properties that would occur if they were driven into certain operating regions. For these devices, the modeling by linear-system models may give quite accurate analytical results over a relatively wide range of operating conditions. However, there are numerous physical devices that possess strong nonlinear characteristics. For these devices, a linearized model is valid only for limited range of operation and often only at the operating point at which the linearization is carried out. More importantly, when a nonlinear system is linearized at an operating point, the linear model may contain time-varying elements.

2-4-1 Linearization Using Taylor Series: Classical Representation

In general, Taylor series may be used to expand a nonlinear function f

(*x*(*t*)) about a reference or operating value $x_0(t)$. An operating value could be the equilibrium position in a spring-mass-damper, a fixed voltage in an electrical system, steady-state pressure in a fluid system, and so on. A function *f* (*x*(*t*)) can therefore be represented in a form

$$f(x(t)) = \sum_{i=1}^{n} c_i (x(t) - x_0(t))^i \tag{2-153}$$

where the constant c_i represents the derivatives of *f* (*x*(*t*)) with respect to *x*(*t*) and evaluated at the operating point $x_0(t)$. That is,

$$c_i = \frac{1}{i!}\frac{d^i f(x_0)}{dx^i} \tag{2-154}$$

Or

$$\begin{aligned} f(x(t)) = f(x_0(t)) + \frac{df(x_0(t))}{dt}(x(t)-x_0(t)) + \frac{1}{2}\frac{d^2 f(x_0(t))}{dt^2}(x(t)-x_0(t))^2 \\ + \frac{1}{6}\frac{d^3 f(x_0(t))}{dt^3}(x(t)-x_0(t))^3 + \cdots + \frac{1}{n!}\frac{d^n f(x_0(t))}{dt^n}(x(t)-x_0(t))^n \end{aligned} \tag{2-155}$$

If $\Delta(x) = x(t) - x_0(t)$ is small, the series Eq. (2-155) converges, and a linearization scheme may be used by replacing *f* (*x*(*t*)) with the first two terms in Eq. (2-155). That is,

$$\begin{aligned} f(x(t)) &\approx f(x_0(t)) + \frac{df(x_0(t))}{dt}(x(t)-x_0(t)) \\ &= c_0 + c_1 \Delta x \end{aligned} \tag{2-156}$$

The following examples serve to illustrate the linearization procedure just described.

EXAMPLE 2-4-1 Find the equation of motion of a simple (ideal) pendulum with a mass *m* and a massless rod of length ℓ, *hinged at point O*, as shown in Fig. 2-38.

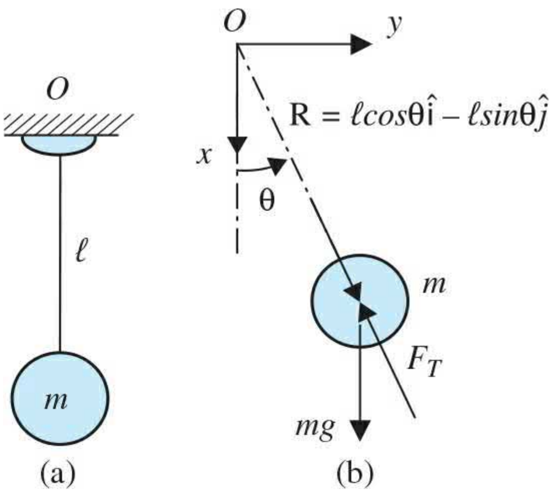


Figure 2-38 (*a*) A simple pendulum. (*b*) Free-body diagram of mass *m*.

SOLUTION Assume the mass is moving in the positive direction as defined by angle *θ*. Note that *θ* is measured from the *x* axis in the counterclockwise direction. The first step is to draw the free-body diagram of the components of the system, that is, mass and the rod, as shown in Fig. 2-38*b*. For the mass *m*, the equations of motion are

$$\sum F_x = ma_x \tag{2-157}$$

$$\sum F_y = ma_y \tag{2-158}$$

where F_x and F_y are the external forces applied to mass *m*, and a_x and a_y are the components of acceleration of mass *m* along *x* and *y*, respectively. If the position vector from point *O* to mass *m* is designated by vector **R**, acceleration of mass *m* is the second time derivative of **R**, and is a vector **a** with tangential and centripetal components. Using the rectangular coordinate frame (*x*, *y*) representation, acceleration vector is

$$\mathbf{a} = \frac{d^2\mathbf{R}}{dt^2} = \frac{d^2(\ell\cos\theta\hat{i} + \ell\sin\theta\hat{j})}{dt^2}$$

$$= (-\ell\ddot{\theta}\sin\theta - \ell\dot{\theta}^2\cos\theta)\hat{i} + (\ell\ddot{\theta}\cos\theta - \ell\dot{\theta}^2\sin\theta)\hat{j} \qquad (2\text{-}159)$$

where $\hat{i}$ and $\hat{j}$ are the unit vectors along *x* and *y* directions, respectively. As a result,

$$a_x = (-\ell\ddot{\theta}\sin\theta - \ell\dot{\theta}^2\cos\theta) \qquad (2\text{-}160)$$

$$a_y = (\ell\ddot{\theta}\cos\theta - \ell\dot{\theta}^2\sin\theta) \qquad (2\text{-}161)$$

Considering the external forces applied to mass, we have

$$\sum F_x = -F_T\cos\theta + mg \qquad (2\text{-}162)$$

$$\sum F_y = -F_T\sin\theta \qquad (2\text{-}163)$$

Equations (2-241) and (2-242) may therefore be rewritten as

$$-F_T\cos\theta + mg = m(-\ell\ddot{\theta}\sin\theta - \ell\dot{\theta}^2\cos\theta) \qquad (2\text{-}164)$$

$$-F_T\sin\theta = m(\ell\ddot{\theta}\cos\theta - \ell\dot{\theta}^2\sin\theta) \qquad (2\text{-}165)$$

Premultiplying Eq. (2-164) by (–sin*θ*) and Eq. (2-165) by (cos*θ*) and adding the two, we get

$$-mg\sin\theta = m\ell\ddot{\theta} \qquad (2\text{-}166)$$

where

$$\sin^2\theta + \cos^2\theta = 1 \qquad (2\text{-}167)$$

After rearranging, Eq. (2-167) is rewritten as

$$m\ell\ddot{\theta} + mg\sin\theta = 0 \qquad (2\text{-}168)$$

or

$$\ddot{\theta}+\frac{g}{\ell}\sin\theta=0 \tag{2-169}$$

In brief, using static equilibrium position $\theta = 0$ as the operating point, for small motions the linearization of the system implies $\theta \approx \theta$ as shown in Fig. 2-39.

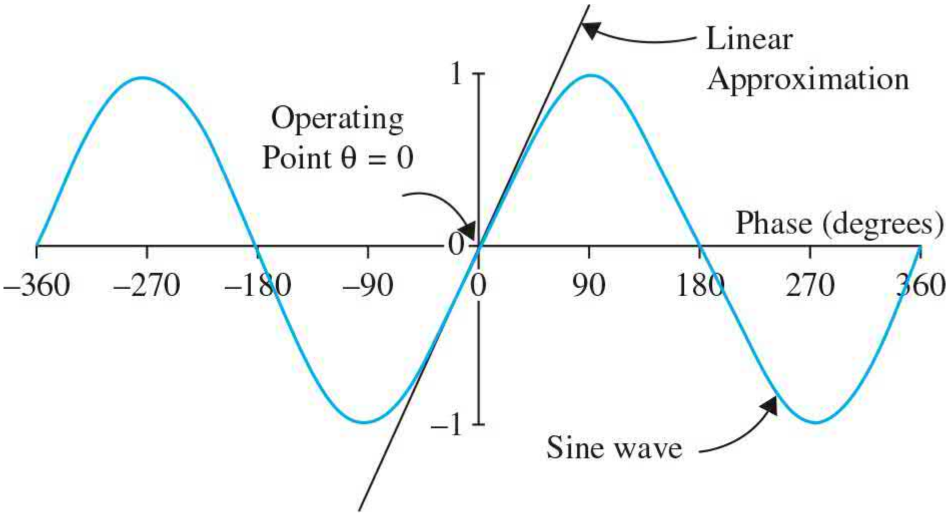


Figure 2-39 Linearization of $\theta \approx \theta$ about $\theta = 0$ operating point.

Hence, the linear representation of the system is $\ddot{\theta}+\frac{g}{\ell}\theta=0$. Or

$$\ddot{\theta}+\omega_n^2\theta=0 \tag{2-170}$$

where $\omega_n=\sqrt{\frac{g}{\ell}}$ rad/s is the natural frequency of the linearized model. ▲

EXAMPLE 2-4-2 For the pendulum shown in Fig. 2-38, rederive the differential equation using the moment equation.

SOLUTION The free-body diagram for the moment equation is shown in Fig. 2-38*b*. Applying the moment equation about the fixed point *O*,

$$\begin{aligned} \sum M_o &= m\ell^2\alpha \\ -\ell\sin\theta\cdot mg &= m\ell^2\ddot{\theta} \end{aligned} \tag{2-171}$$

Rearranging the equation in the standard input–output differential equation form,

$$m\ell^2\ddot{\theta} + mg\ell\sin\theta = 0 \tag{2-172}$$

or

$$\ddot{\theta} + \frac{g}{\ell}\sin\theta = 0 \tag{2-173}$$

which is the same result obtained previously. For small motions, as in the Example 2-4-1,

$$\sin\theta \approx \theta \tag{2-174}$$

The linearized differential equations is

$$\ddot{\theta} + \omega_n^2\theta = 0 \tag{2-175}$$

where, as before

$$\omega_n = \sqrt{\frac{g}{\ell}} \tag{2-176}$$

2-5 ANALOGIES

In this section, we demonstrate the similarities among mechanical, thermal, and fluid systems with electrical networks. As an example, let us compare Eqs. (2-10) and (2-71). It is not difficult to see that the mechanical system in Fig. 2-2 is **analogous** to a **series *RLC* electric network** shown in Fig. 2-21. These systems are shown in Fig. 2-40. In order to exactly see how parameters, *M*, *B*, and *K* are related to *R*, *L*, and *C*; or how the variables *y*(*t*) and *f* (*t*) are related to *i*(*t*) and *e*(*t*), we need to compare Eqs. (2-8) and (2-59).

Thus,

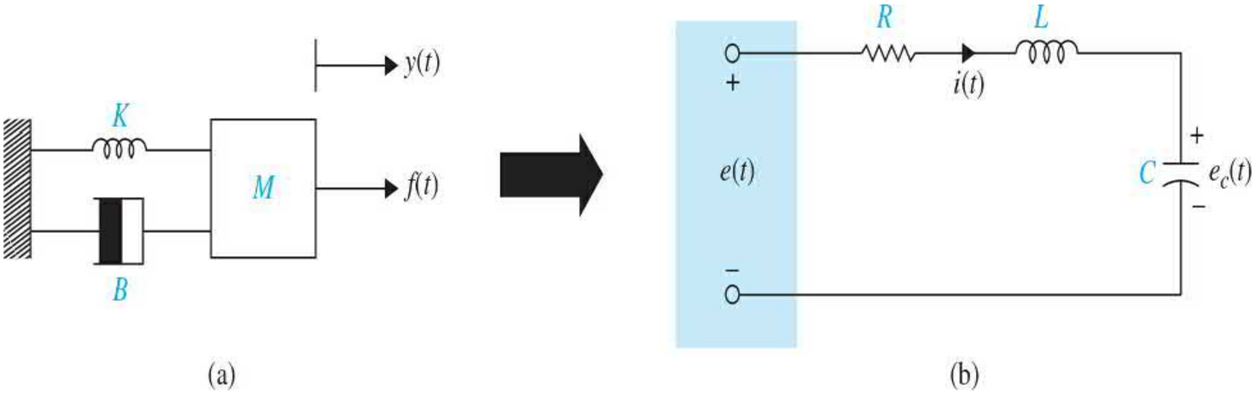


Figure 2-40 Analogy of a spring-mass-damper system to a series *RLC* network. (*a*) A spring-mass-damper system. (*b*) A series *RLC* equivalent.

Using a **force-voltage** analogy, the spring-mass-damper system in Fig. 2-2 is **analogous** to a series RLC electric network shown in Fig. 2-19.

$$M\frac{d^2y(t)}{dt^2}+B\frac{dy(t)}{dt}+Ky(t)=f(t) \tag{2-177}$$

$$L\frac{di(t)}{dt}+Ri(t)+\frac{1}{C}\int i(t)\,dt=e(t) \tag{2-178}$$

This comparison is more properly made upon integrating Eq. (2-177) with respect to time. That is,

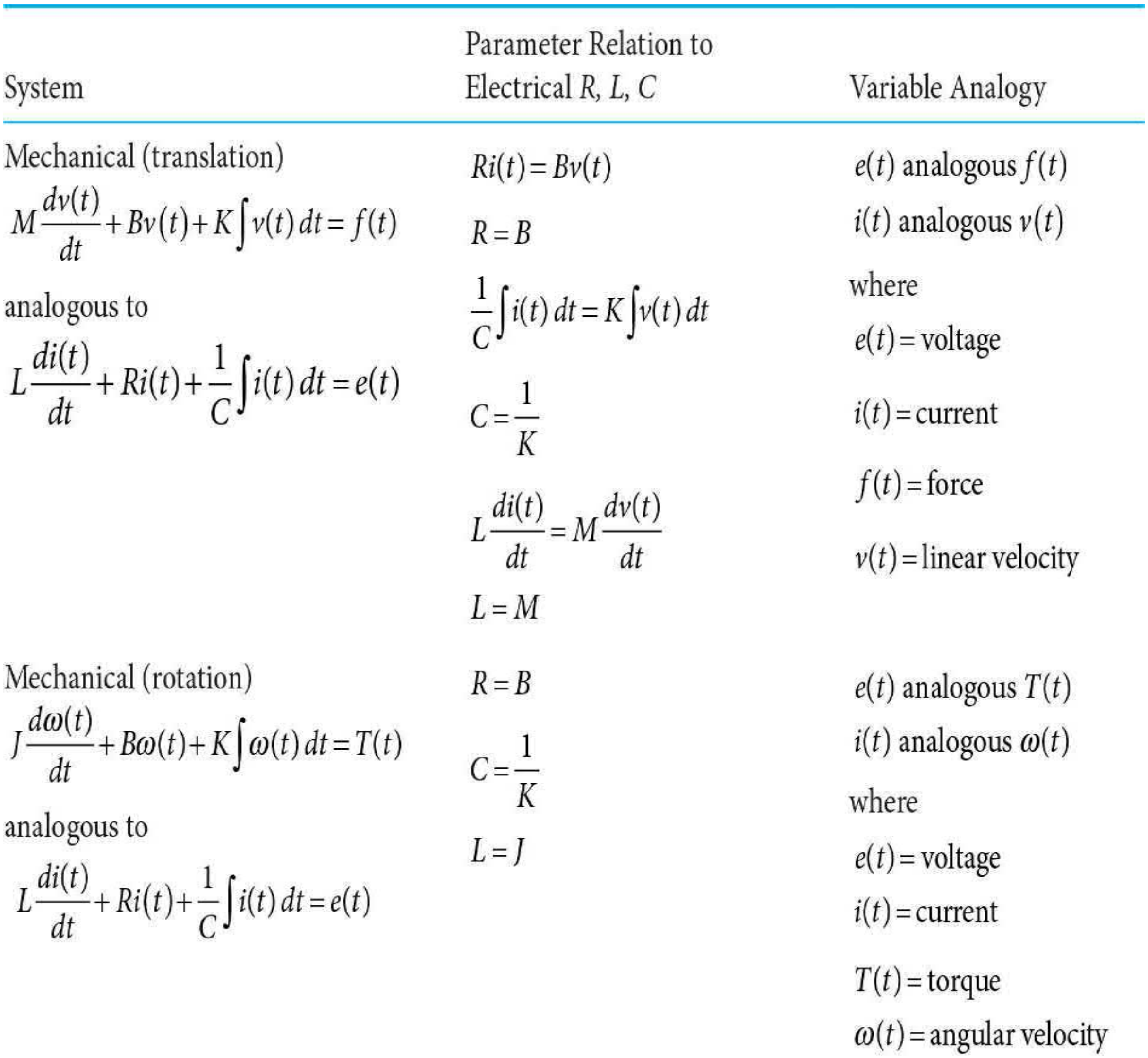
$$M\frac{dv(t)}{dt}+Bv(t)+K\int v(t)\,dt=f(t) \tag{2-179}$$

where *v*(*t*) represents the velocity of mass *m*. As a result, with this comparison, mass *M* is analogous to inductance *L*, the spring constant *K* is analogous to the inverse of capacitance 1/*C*, and the viscous-friction coefficient *B* is analogous to resistance *R*. Similarly, *v*(*t*) and *f* (*t*) are analogous to *i*(*t*) and *e*(*t*), respectively. This type of analogy is also known as

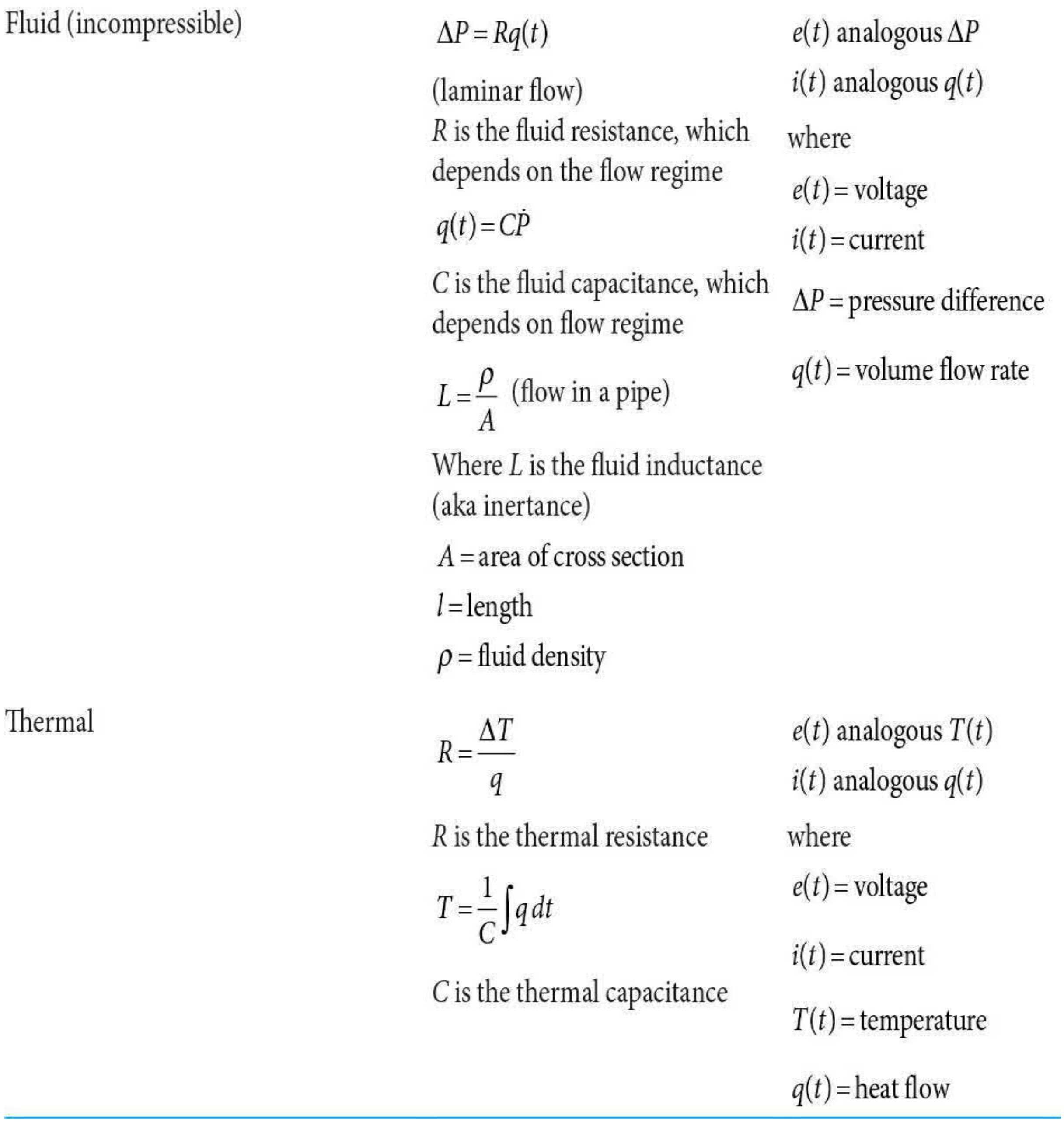
force-voltage analogy. Similar assessment can be made by comparing the rotational system in Eq. (2-32) with the *RLC* network of Example 2-4-1.

Using a parallel *RLC* network with current as a source, some literature use a force-current analogy that is not discussed here.[5] Comparing the thermal, fluid, and electrical systems, similar analogies may be obtained, as shown in Table 2-6.

TABLE 2-6 Mechanical, Thermal, and Fluid Systems and Their Electrical Equivalents



System	Parameter Relation to Electrical *R*, *L*, *C*	Variable Analogy
Mechanical (translation) $M\frac{dv(t)}{dt}+Bv(t)+K\int v(t)\,dt=f(t)$ analogous to $L\frac{di(t)}{dt}+Ri(t)+\frac{1}{C}\int i(t)\,dt=e(t)$	$Ri(t)=Bv(t)$ $R=B$ $\frac{1}{C}\int i(t)\,dt=K\int v(t)\,dt$ $C=\frac{1}{K}$ $L\frac{di(t)}{dt}=M\frac{dv(t)}{dt}$ $L=M$	$e(t)$ analogous $f(t)$ $i(t)$ analogous $v(t)$ where $e(t)=$ voltage $i(t)=$ current $f(t)=$ force $v(t)=$ linear velocity
Mechanical (rotation) $J\frac{d\omega(t)}{dt}+B\omega(t)+K\int \omega(t)\,dt=T(t)$ analogous to $L\frac{di(t)}{dt}+Ri(t)+\frac{1}{C}\int i(t)\,dt=e(t)$	$R=B$ $C=\frac{1}{K}$ $L=J$	$e(t)$ analogous $T(t)$ $i(t)$ analogous $\omega(t)$ where $e(t)=$ voltage $i(t)=$ current $T(t)=$ torque $\omega(t)=$ angular velocity



Fluid (incompressible)	$\Delta P = Rq(t)$ (laminar flow) R is the fluid resistance, which depends on the flow regime $q(t) = C\dot{P}$ C is the fluid capacitance, which depends on flow regime $L = \frac{\rho}{A}$ (flow in a pipe) Where L is the fluid inductance (aka inertance) A = area of cross section l = length ρ = fluid density	$e(t)$ analogous ΔP $i(t)$ analogous $q(t)$ where $e(t)$ = voltage $i(t)$ = current ΔP = pressure difference $q(t)$ = volume flow rate
Thermal	$R = \frac{\Delta T}{q}$ R is the thermal resistance $T = \frac{1}{C}\int q\,dt$ C is the thermal capacitance	$e(t)$ analogous $T(t)$ $i(t)$ analogous $q(t)$ where $e(t)$ = voltage $i(t)$ = current $T(t)$ = temperature $q(t)$ = heat flow

EXAMPLE 2-5-1 For the liquid-level system shown in Fig. 2-35, $C = \frac{A}{\rho g}$ is the capacitance and $R = \frac{\rho g h}{q_o}$ is the resistance. As a result, system time constant is $\tau = RC$.

A One-Tank Liquid-Level System

SOLUTION In order to design a speed, position, or any type of control system, the first task at hand is to arrive at a mathematical model of the system. This will help us to "properly" develop the best controller for the required task (e.g., proper positioning of the arm in a pick and place operation).

A general advice is to use the simplest model you can that is "good enough!" In this case, we can assume the effective mass of the arm and the mass of payload are concentrated at the end of a massless rod, as shown in Fig. 2-41. You can experimentally arrive at what mass *m* in your model should be. See App. D for details.

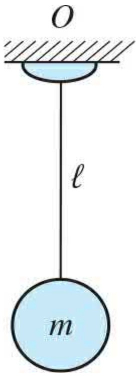


Figure 2-41 One-degree-of-freedom arm with required components.

As in Example 2-2-1, is moving in the positive direction as defined by angle θ. Note that θ is measured from the *x* axis in the counterclockwise direction. For the mass *m*, the equations of motion may be obtained by taking a moment about point *O*, we get

$$\sum M_O = T = m\ell^2\ddot{\theta} \tag{2-180}$$

where *T* is the external torque applied by the motor to accelerate the mass.

In Chap. 6, we significantly augment this model by adding the model of the motor. ▲

2-6 PROJECT: INTRODUCTION TO LEGO MINDSTORMS NXT MOTOR—MECHANICAL MODELING

This section provides a simple, yet practical, project for you to better appreciate the theoretical concepts that have been discussed so far.

The goal of this project is further to build a one-degree-of-freedom robot using the LEGO MINDSTORMS NXT motor, shown in Fig. 2-42, and to arrive at the mathematical model of the mechanical one-degree of freedom arm. This example is followed through in Chaps. 6, 7, 8, and 11. The detailed discussion on this topic is provided in App. D, with the **objective** to provide you with a series of experiments for **measuring** a dc motor’s **electrical** and **mechanical** properties, and ultimately, to create a **mathematical model** for the motor and the robot arm shown in Fig. 2-42 for controller design purposes.

> The first objective of this project is to help you better understand how to measure a dc motor’s electrical and mechanical characteristics and ultimately create a model for the motor.

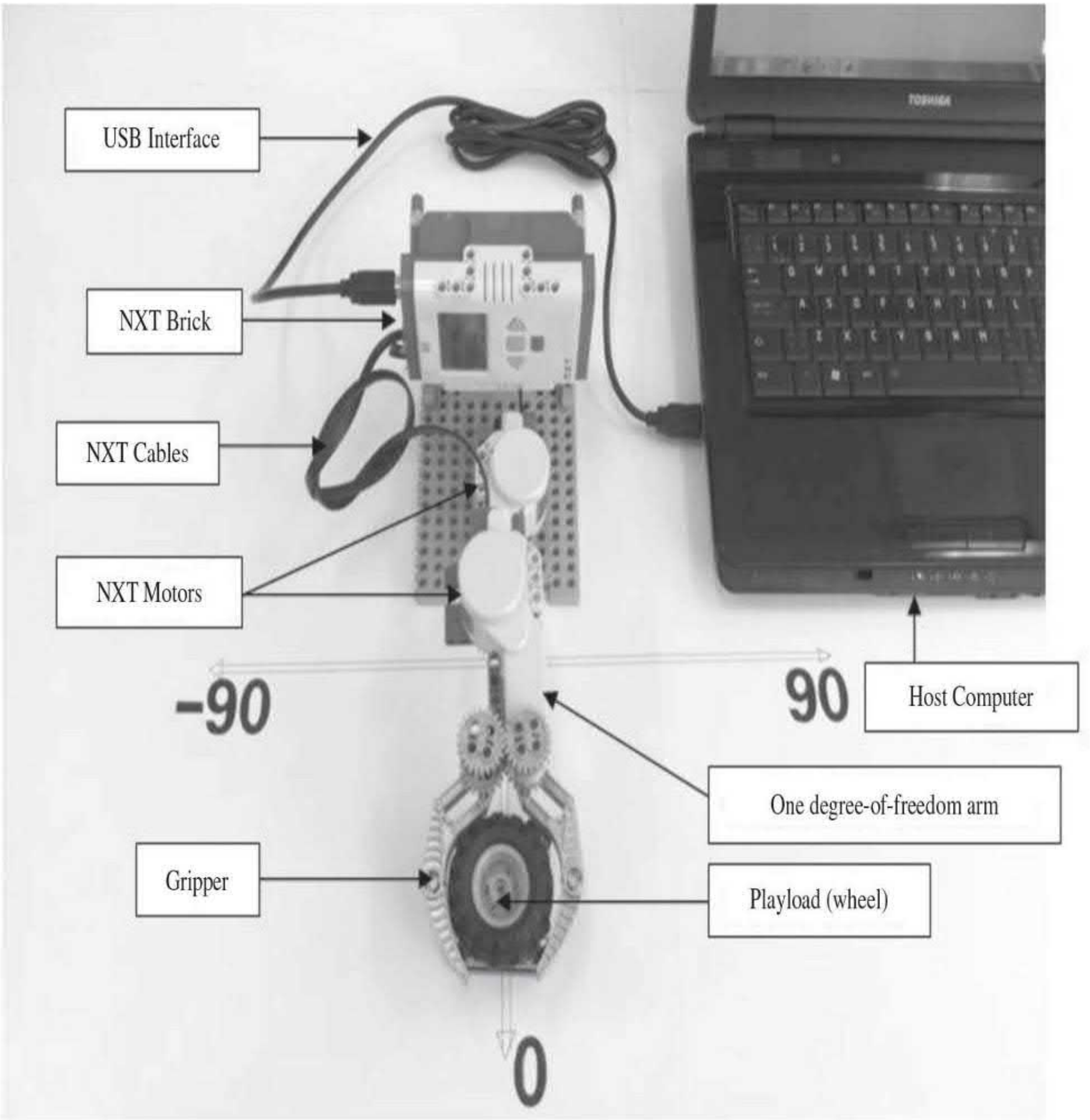


Figure 2-42 A simplified model of a one-degree-of-freedom robotic arm.

As shown in Fig. 2-42, the components of our robotic system include an NXT brick, an NXT motor, and several LEGO pieces found in the basic LEGO MINDSTORMS kit, which are used here to construct a one-degree-of-freedom arm. The arm is to pick up a payload and drop it into the cup, which is located at a specified angle while data are sampled in Simulink. Programming is done on the host computer using Simulink and is uploaded on to the NXT brick via USB interface. The brick then provides both power and control to the arm via the NXT cables. Additionally, there is an **optical encoder** located behind the motor which measures the rotational position of the output shaft with one-degree resolution. The host computer samples encoder data from the NXT brick via a Bluetooth connection. In order for the host computer to recognize the NXT brick, the host computer must be paired with the NXT brick when setting up the Bluetooth connection.[6]

2-7 SUMMARY

This chapter is devoted to the mathematical modeling of basic dynamic systems, including various examples of mechanical, electrical, thermal, and fluid systems. Using the basic modeling principles such as Newton’s second law of motion, Kirchhoff’s law, or conservation of mass the model of these dynamic systems are represented by differential equations, which may be linear or nonlinear. However, due to space limitations and the intended scope of this text, only some of the physical devices used in practice are described.

Because nonlinear systems cannot be ignored in the real world, and this book is not devoted to the subject, we introduced the linearization of nonlinear systems at a nominal operating point. Once the linearized model is determined, the performance of the nonlinear system can be investigated under the small-signal conditions at the designated operating point.

Finally, in this chapter we establish analogies between mechanical, thermal, and fluid systems with equivalent electrical networks.

REFERENCES

1. W. J. Palm III, *Modeling, Analysis, and Control of Dynamic Systems*, 2nd Ed., John Wiley & Sons, New York, 1999.
2. K. Ogata, *Modern Control Engineering*, 4th Ed., Prentice Hall, New Jersey, 2002.
3. I. Cochin and W. Cadwallender, *Analysis and Design of Dynamic Systems*, 3rd Ed., Addison-Wesley, New York, 1997.
4. A. Esposito, *Fluid Power with Applications*, 5th Ed., Prentice Hall, New Jersey, 2000.
5. H. V. Vu and R. S. Esfandiari, *Dynamic Systems*, Irwin/McGraw-

Hill, Boston, 1997.

6. J. L. Shearer, B. T. Kulakowski, and J. F. Gardner, *Dynamic Modeling and Control of Engineering Systems*, 2nd Ed., Prentice Hall, New Jersey, 1997.

7. R. L. Woods and K. L. Lawrence, *Modeling and Simulation of Dynamic Systems*, Prentice Hall, New Jersey, 1997.

8. E. J. Kennedy, *Operational Amplifier Circuits*, Holt, Rinehart and Winston, Fort Worth, TX, 1988.

9. J. V. Wait, L. P. Huelsman, and G. A. Korn, *Introduction to Operational Amplifier Theory and Applications*, 2nd Ed., McGraw-Hill, New York, 1992.

10. B. C. Kuo and F. Golnaraghi, *Automatic Control Systems*, 8th Ed., John Wiley & Sons, New York, 2003.

11. F. Golnaraghi and B. C. Kuo, *Automatic Control Systems*, 9th Ed., John Wiley & Sons, New York, 2010.

PROBLEMS

PROBLEMS FOR SEC. 2-1

2-1. Find the equation of the motion of the mass-spring system shown in Fig. 2P-1. Also calculate the natural frequency of the system.

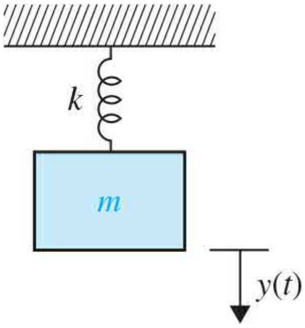


Figure 2P-1

2-2. Find its single spring-mass equivalent in the five-spring one-mass system shown in Fig. 2P-2. Also calculate the natural frequency of the system.

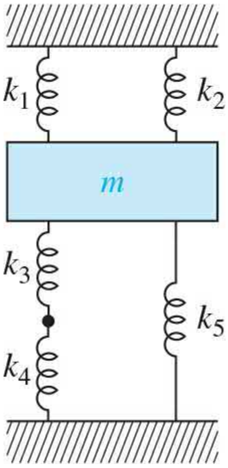


Figure 2P-2

2-3. Find the equation of the motion for a simple model of a vehicle suspension system hitting a bump. As shown in Fig. 2P-3, the mass of wheel and its mass moment of inertia are *m* and *J*, respectively. Also calculate the natural frequency of the system.

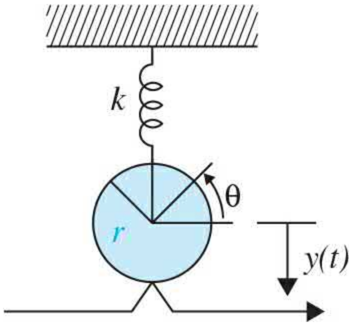


Figure 2P-3

2-4. Write the force equations of the linear translational systems shown in Fig. 2P-4.

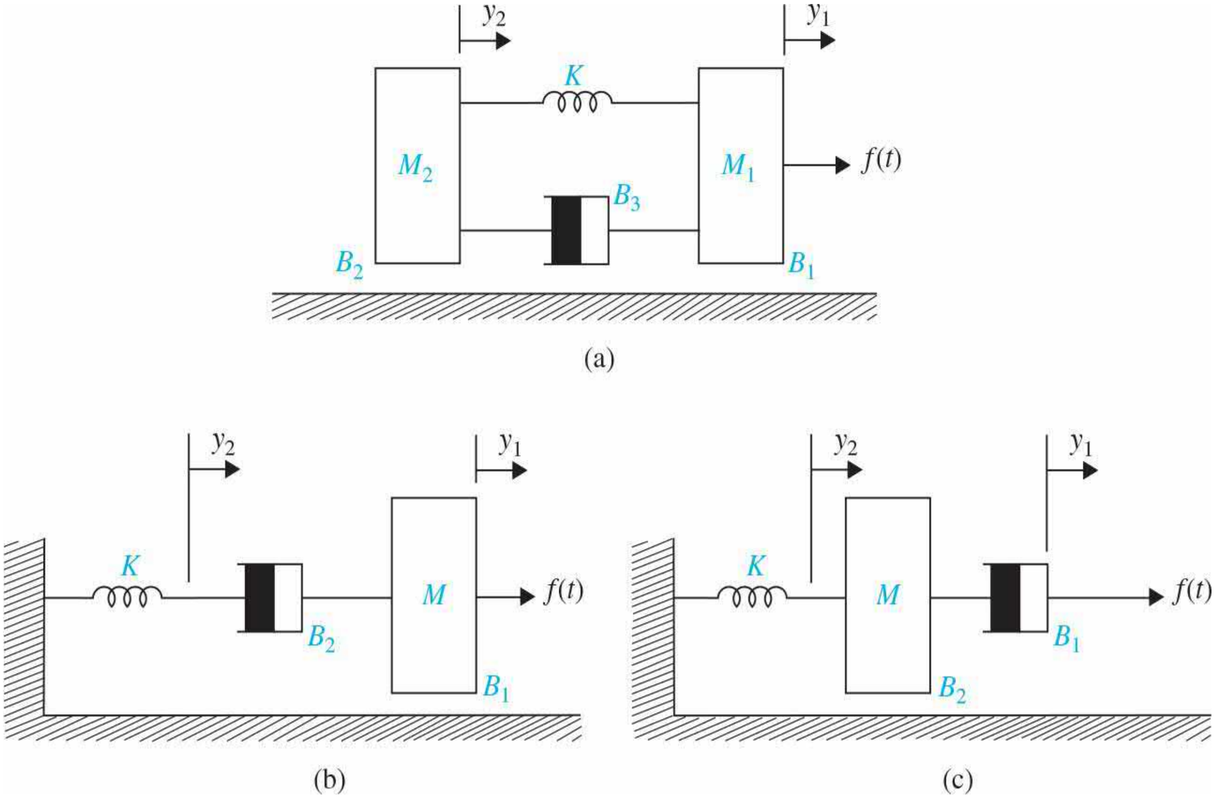


Figure 2P-4

2-5. Write the force equations of the linear translational system shown in Fig. 2P-5.

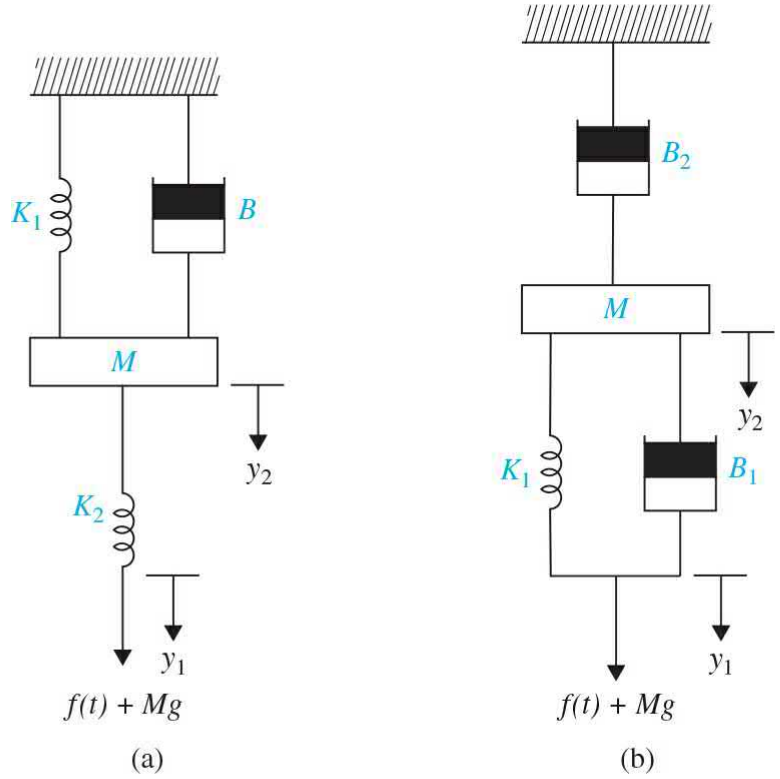


Figure 2P-5

2-6. Consider a train consisting of an engine and a car, as shown in Fig. 2P-6.

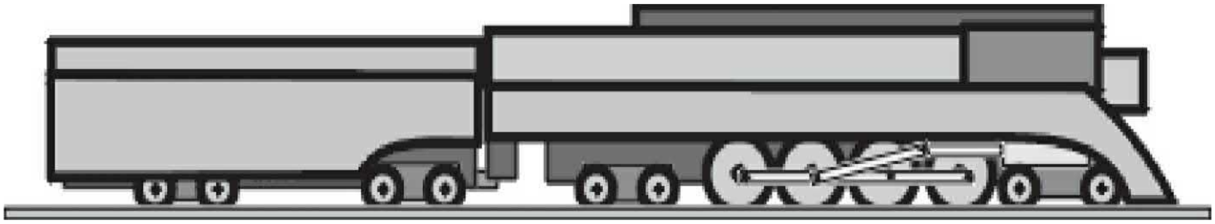


Figure 2P-6

A controller is applied to the train so that it has a smooth start and stop, along with a constant-speed ride. The mass of the engine and the car are *M* and *m*, respectively. The two are held together by a spring with the stiffness coefficient of *K*. *F* represents the force applied by the engine, and *m* represents the coefficient of rolling friction. If the train only travels in one direction:

(a) Draw the free-body diagram.

(b) Find the equations of motion.

2-7. A vehicle towing a trailer through a spring-damper coupling hitch is shown in Fig. 2P-7. The following parameters and variables are defined: M is the mass of the trailer; K_h, the spring constant of the hitch; B_h, the viscous-damping coefficient of the hitch; B_t, the viscous-friction coefficient of the trailer; $y_1(t)$, the displacement of the towing vehicle; $y_2(t)$, the displacement of the trailer; and $f(t)$, the force of the towing vehicle.

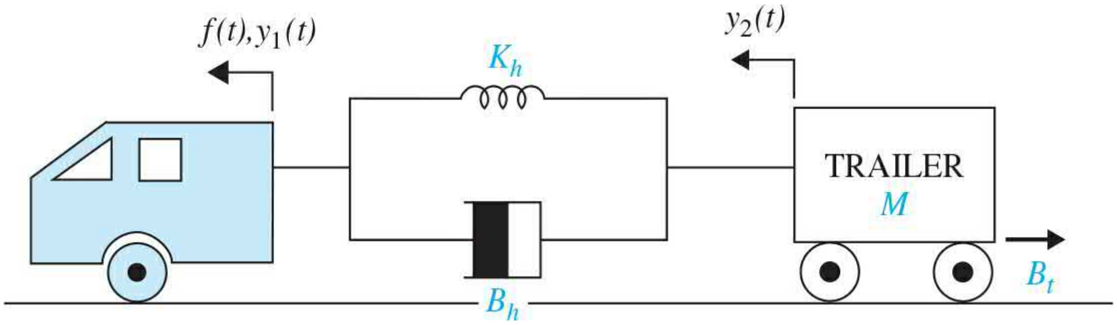


Figure 2P-7

Write the differential equation of the system.

2-8. Assume that the displacement angles of the pendulums shown in Fig. 2P-8 are small enough that the spring always remains horizontal. If the rods with the length of L are massless and the spring is attached to the rods ⅞ from the top, find the state equation of the system.

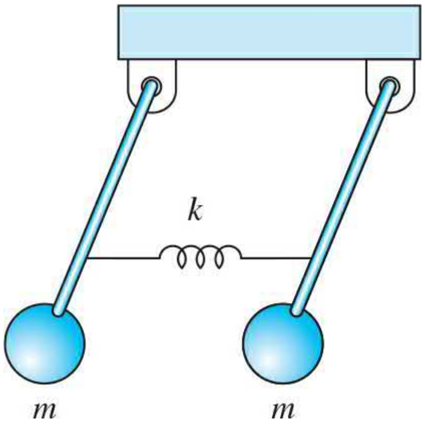


Figure 2P-8

2-9. (**Challenge Problem**) Figure 2P-9 shows an inverted pendulum on a cart.

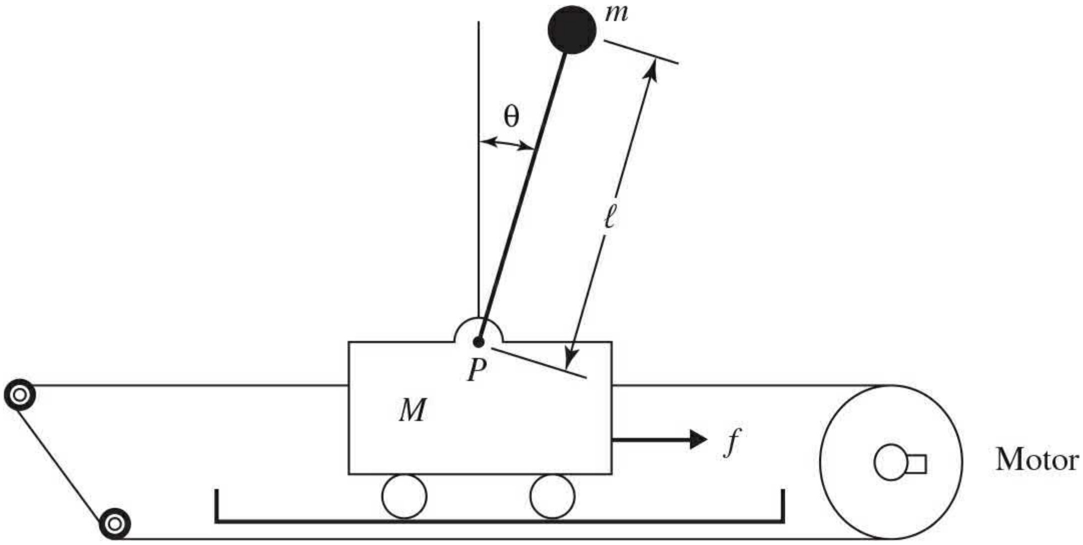


Figure 2P-9

If the mass of the cart is represented by M and the force f is applied to hold the bar at the desired position, then

(a) Draw the free-body diagram.

(b) Determine the dynamic equation of the motion.

2-10. (**Challenge Problem**) A two-stage inverted pendulum on a cart is shown in Fig. 2P-10.

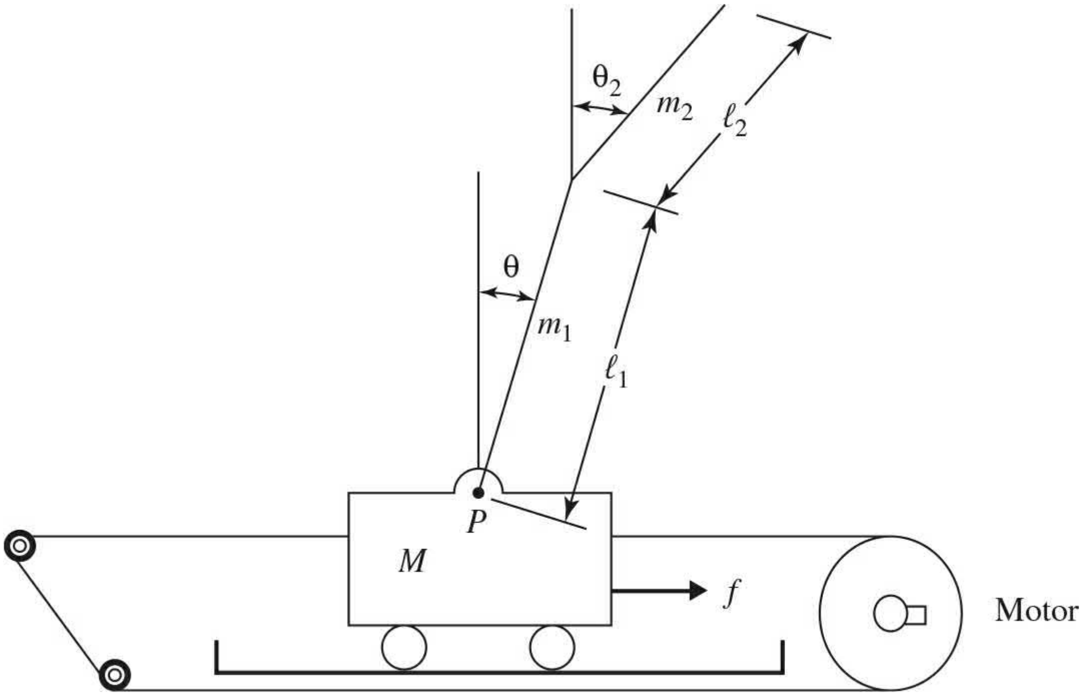


Figure 2P-10

If the mass of the cart is represented by *M* and the force *f* is applied to hold the bar at the desired position, then

(a) Draw the free-body diagram of mass *M*.

(b) Determine the dynamic equation of the motion.

2-11. (**Challenge Problem**) Figure 2P-11 shows a well-known “ball and beam” system in control systems. A ball is located on a beam to roll along the length of the beam. A lever arm is attached to the one end of the beam and a servo gear is attached to the other end of the lever arm. As the servo gear turns by an angle θ, the lever arm goes up and down, and then the angle of the beam is changed by α. The change in angle causes the ball to roll along the beam. A controller is desired to manipulate the ball’s position.

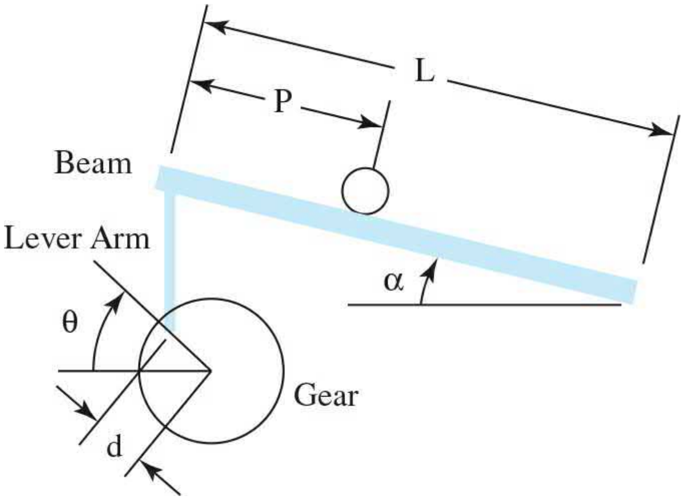


Figure 2P-11

Assuming:
m = mass of the ball
r = radius of the ball
d = lever arm offset
g = gravitational acceleration
L = length of the beam
J = ball’s moment of inertia
p = ball position coordinate
α = beam angle coordinate
θ = servo gear angle

Determine the dynamic equation of the motion.

2-12. The motion equations of an aircraft are a set of six nonlinear coupled differential equations. Under certain assumptions, they can be decoupled and linearized into the longitudinal and lateral equations. Figure 4P-12 shows a simple model of airplane during its flight. Pitch control is a longitudinal problem, and an autopilot is designed to control the pitch of the airplane.

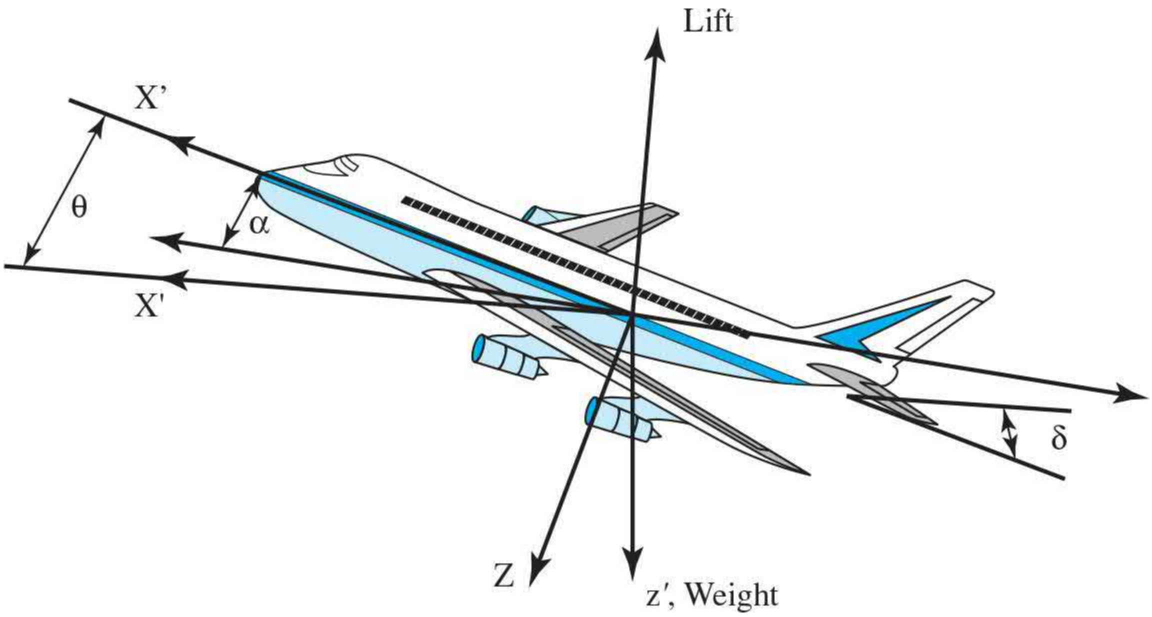


Figure 2P-12

Consider that the airplane is in steady-cruise at constant altitude and velocity, which means the thrust and drag cancel out and the lift and weight balance out each other. To simplify the problem, assume that change in pitch angle does not affect the speed of an aircraft under any circumstance.

Determine the longitudinal equations of motion of the aircraft.

2-13. Write the torque equations of the rotational systems shown in Fig. 2P-13.

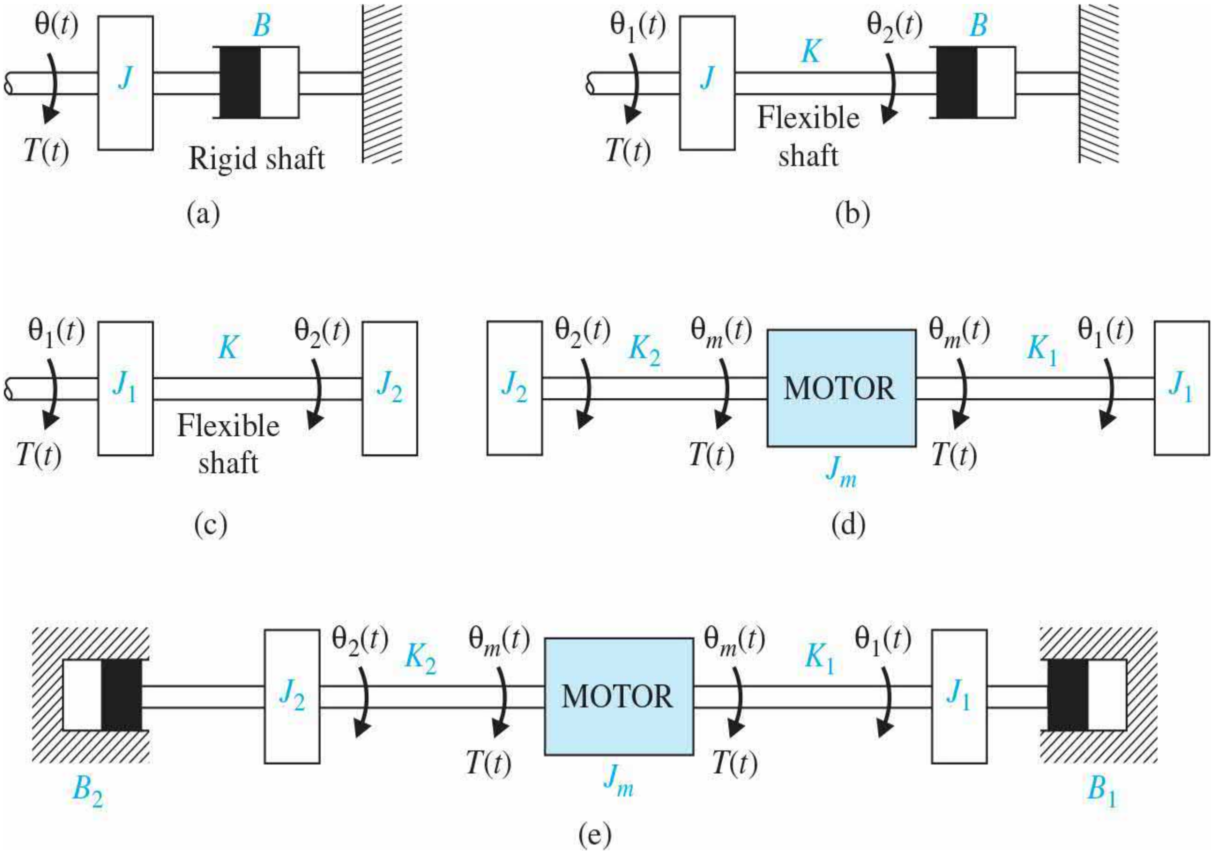


Figure 2P-13

2-14. Write the torque equations of the gear-train system shown in Fig. 2P-14. The moments of inertia of gears are lumped as J_1, J_2, and J_3. $T_m(t)$ is the applied torque; N_1, N_2, N_3, and N_4 are the number of gear teeth. Assume rigid shafts.

(a) Assume that J_1, J_2, and J_3 are negligible. Write the torque equations of the system. Find the total inertia the motor sees.

(b) Repeat part (a) with the moments of inertia J_1, J_2, and J_3.

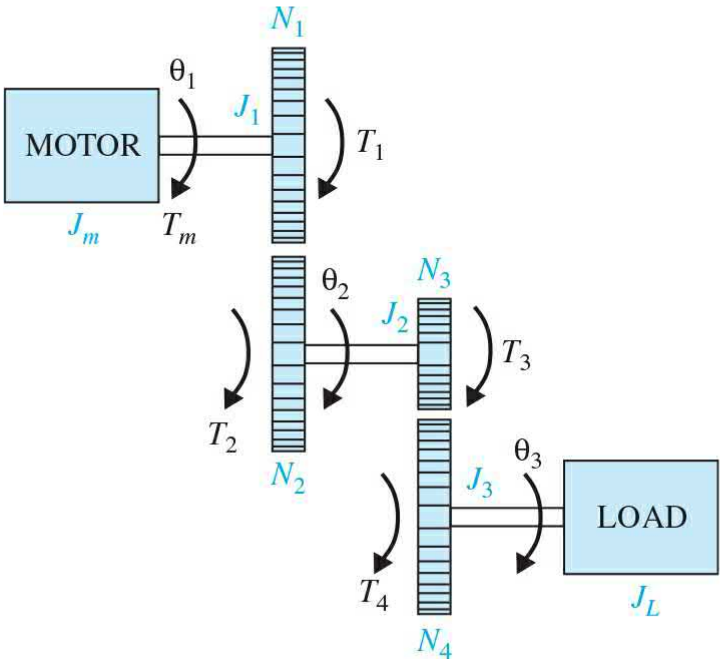


Figure 2P-14

2-15. Figure 2P-15 shows a motor-load system coupled through a gear train with gear ratio $n = N_1/N_2$. The motor torque is $T_m(t)$, and $T_L(t)$ represents a load torque.

(a) Find the optimum gear ratio n^* such that the load acceleration $\alpha_L = d^2\theta_L/dt^2$ is maximized.

(b) Repeat part (a) when the load torque is zero.

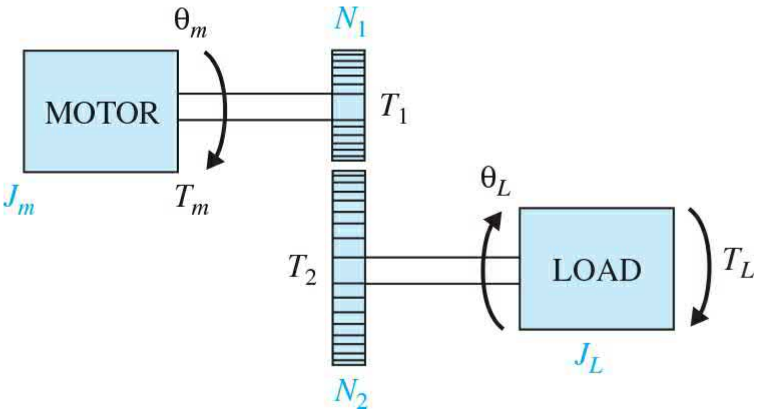


Figure 2P-15

2-16. Figure 2P-16 shows the simplified diagram of the printwheel control system of a word processor. The printwheel is controlled by a dc motor through belts and pulleys. Assume that the belts are rigid. The following parameters and variables are defined: $T_m(t)$ is the motor torque; $\theta_m(t)$, the motor displacement; $y(t)$, the linear displacement of the printwheel; J_m, the motor inertia; B_m, the motor viscous-friction coefficient; r, the pulley radius; M, the mass of the printwheel.

Write the differential equation of the system.

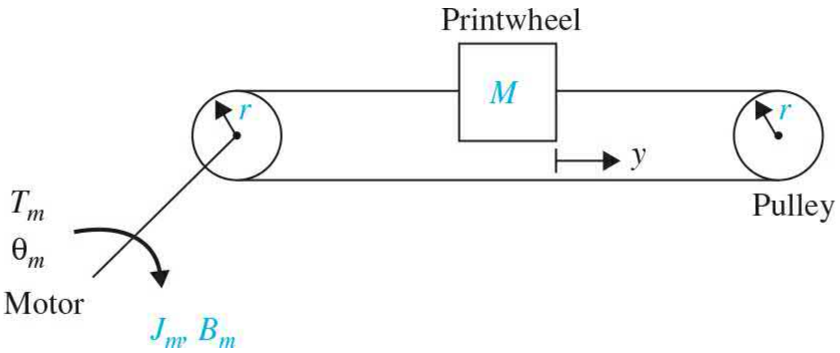


Figure 2P-16

2-17. Figure 2P-17 shows the diagram of a printwheel system with belts and pulleys. The belts are modeled as linear springs with spring constants K_1 and K_2.

Write the differential equations of the system using θ_m and y as the dependent variables.

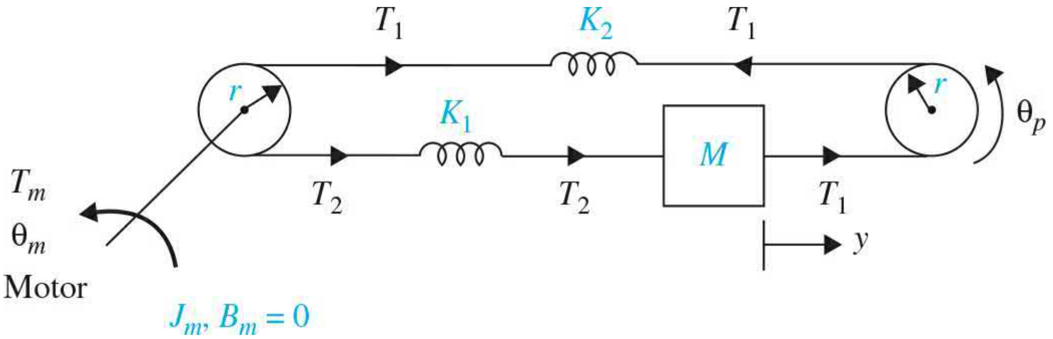


Figure 2P-17

2-18. Classically, the quarter-car model is used in the study of vehicle suspension systems and the resulting dynamic response due to various road inputs. Typically, the inertia, stiffness, and damping characteristics of the system as illustrated in Fig. 2P-18*a* are modeled in a two-degree of freedom (2-DOF) system, as shown in Fig. 2P-18***b*.** Although a 2-DOF system is a more accurate model, it is sufficient for the following analysis to assume a 1-DOF model, as shown in 2P-18*c*.

Find the equations of motion for absolute motion x and the relative motion (bounce) $z = x - y$.

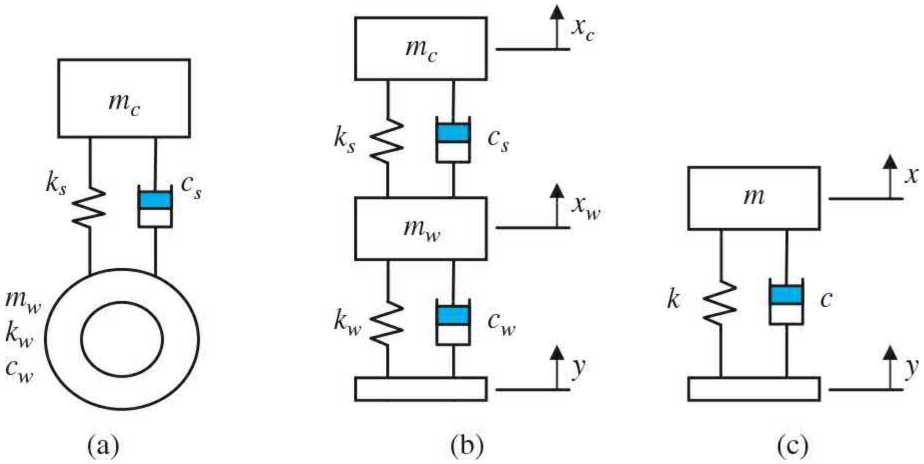


Figure 2P-18 Quarter-car model realization. (*a*) Quarter car. (*b*) Two degrees of freedom. (*c*) One degree of freedom.

2-19. The schematic diagram of a motor-load system is shown in Fig. 2P-19. The following parameters and variables are defined: $T_m(t)$, the motor torque; $\omega_m(t)$, the motor velocity; $\theta_m(t)$, the motor displacement; $\omega_L(t)$, the load velocity; $\theta_L(t)$, the load displacement; K, the torsional spring constant; J_m, the motor inertia; B_m, the motor viscous-friction coefficient; and B_L, the load viscous-friction coefficient.

Write the torque equations of the system.

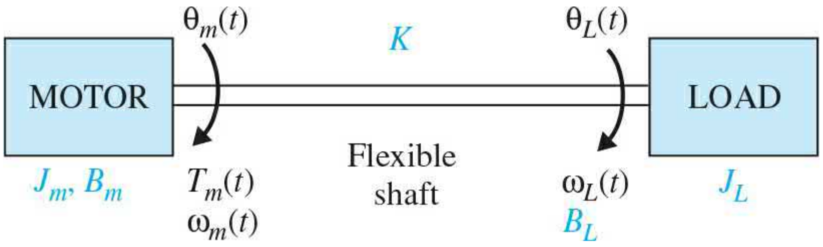


Figure 2P-19

2-20. This problem deals with the attitude control of a guided missile. When traveling through the atmosphere, a missile encounters aerodynamic forces that tend to cause instability in the attitude of the missile. The basic concern from the flight-control standpoint is the lateral force of the air, which tends to rotate the missile about its center of gravity. If the missile centerline is not aligned with the direction in which the center of gravity C is traveling, as shown in Fig. 2P-20, with angle θ, which is also called the angle of attack, a side force is produced by the drag of the air through which the missile travels. The total force F_α may be considered to be applied at the center of pressure P. As shown in Fig. 2P-20, this side force has a tendency to cause the missile to tumble end over end, especially if the point P is in front of the center of gravity C. Let the angular acceleration of the missile about the point C, due to the side force, be denoted by α_F. Normally, α_F is directly proportional to the angle of attack θ and is given by

$$\alpha_F = \frac{K_F d_1}{J}\theta$$

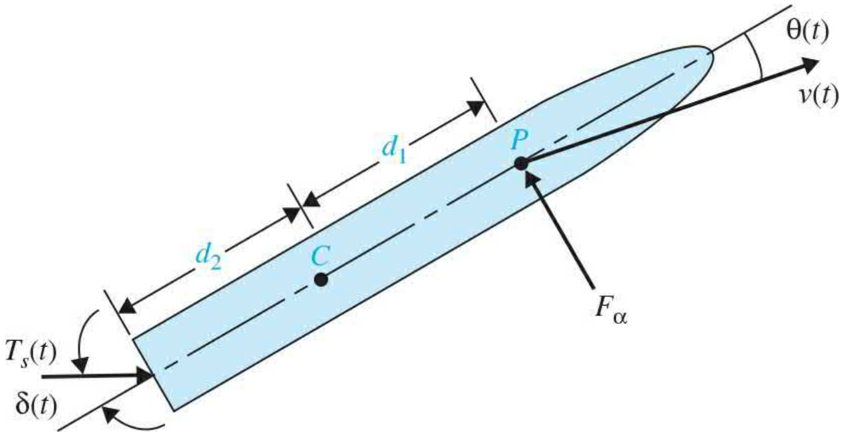


Figure 2P-20

where K_F = a constant that depends on such parameters as dynamic pressure, velocity of the missile, air density, and so on, and

J = missile moment of inertia about C

d_1 = distance between C and P

The main objective of the flight-control system is to provide the stabilizing action to counter the effect of the side force. One of the standard control means is to use gas injection at the tail of the missile to deflect the direction of the rocket engine thrust T_s, as shown in Fig. 2P-20.

(a) Write a torque differential equation to relate among T_s, δ, θ, and the system parameters given. Assume that δ is very small, so that sin $\delta(t)$ is approximated by $\delta(t)$.

(b) Repeat parts (a) with points *C* and *P* interchanged. The d_1 in the expression of α_F should be changed to d_2.

2-21. Figure 2P-21*a* shows a well-known “broom-balancing” system in control systems. The objective of the control system is to maintain the broom in the upright position by means of the force $u(t)$ applied to the car as shown. In practical applications, the system is analogous to a one-dimensional control problem of the balancing of a unicycle or a missile immediately after launching. The free-body diagram of the system is shown in Fig. 2P-21*b*, where

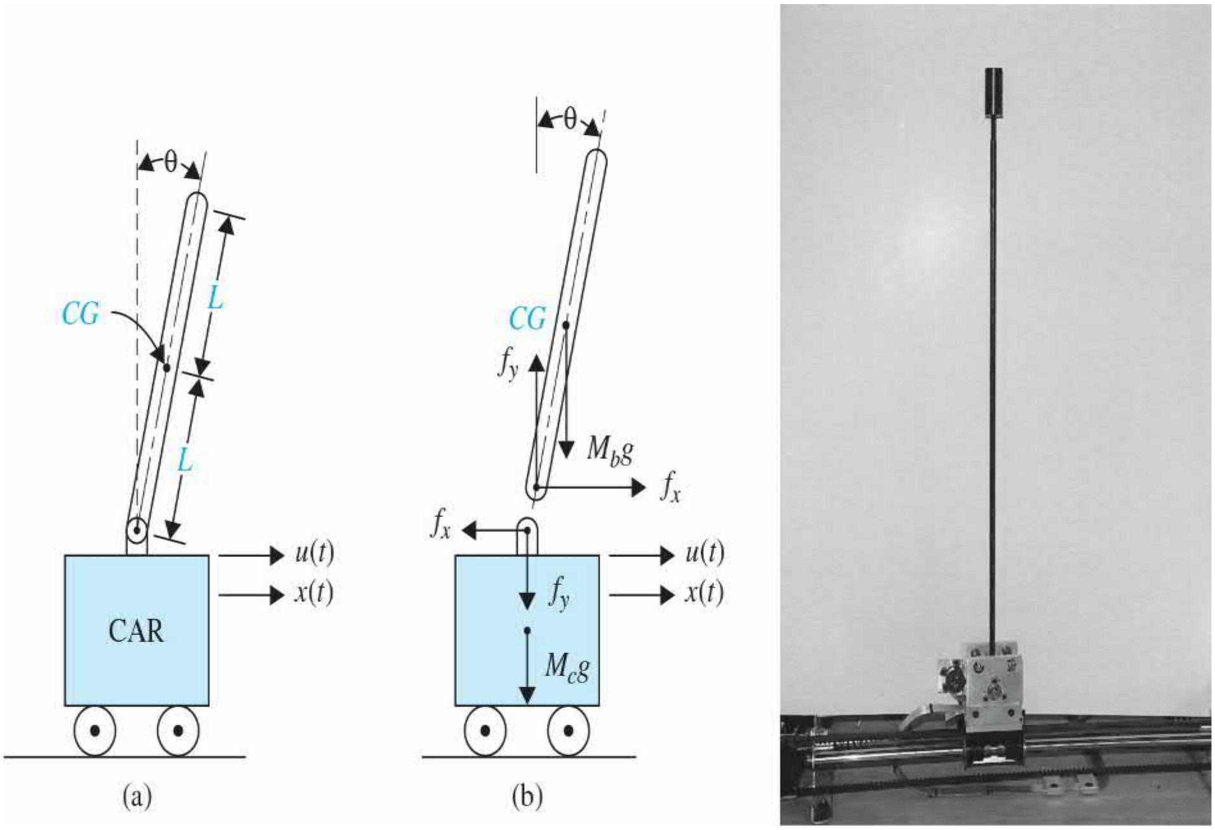


Figure 2P-21

- f_x = force at broom base in horizontal direction
- f_y = force at broom base in vertical direction
- M_b = mass of broom
- g = gravitational acceleration
- M_c = mass of car
- J_b = moment of inertia of broom about center of gravity $CG = M_bL_2/3$

(a) Write the force equations in the *x* and the *y* directions at the pivot point of the broom. Write the torque equation about the center of gravity *CG* of the broom. Write the force equation of the car in the horizontal direction.

(b) Compare your results with those in Prob. 2-9.

2-22. Most machines and devices have rotating parts. Even a small irregularity in the mass distribution of rotating components can cause vibration, which is called rotating unbalanced. Figure 2P-22 represents the schematic of a rotating unbalanced mass of *m*. Assume that the frequency of rotation of the machine is *ω*.

Derive the equations of motion of the system.

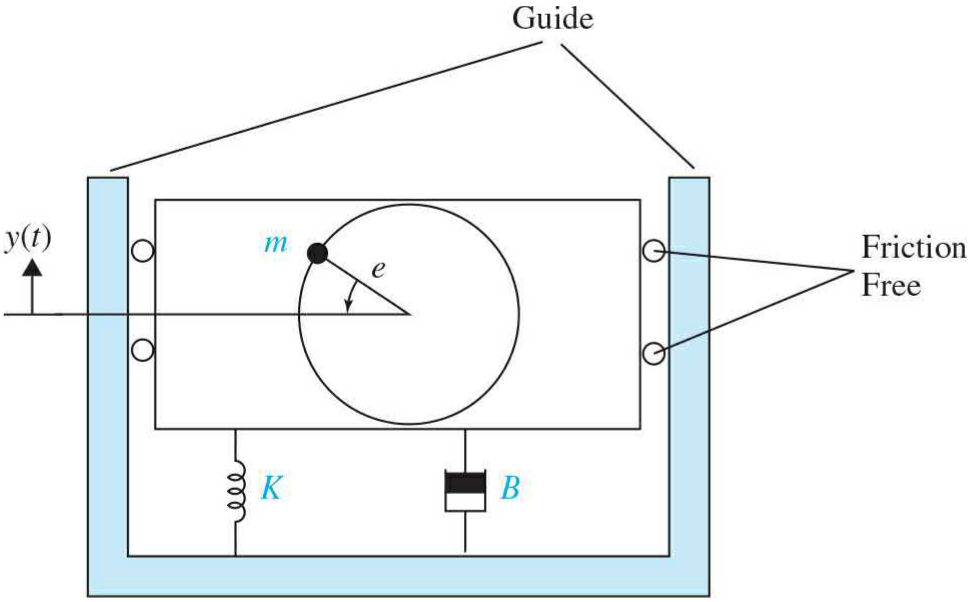


Figure 2P-22

2-23. Vibration absorbers are used to protect machines that work at the constant speed from steady-state harmonic disturbance. Figure 2P-23 shows a simple vibration absorber.

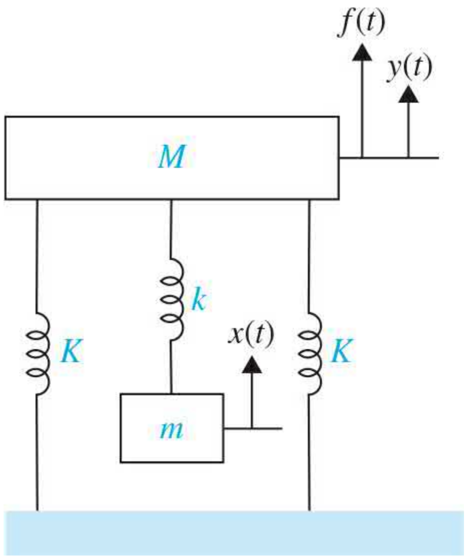


Figure 2P-23

Assuming the harmonic force $F(t) = A\sin(\omega t)$ is the disturbance applied to the mass M, derive the equations of motion of the system.

2-24. Figure 2P-24 represents a vibration absorption system.

Assuming the harmonic force $F(t) = A\sin(\omega t)$ is the disturbance applied to the mass M, derive the equations of motion of the system.

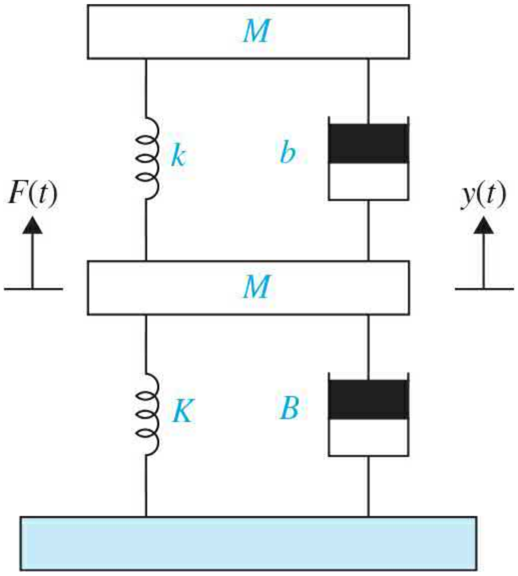


Figure 2P-24

2-25. An accelerometer is a transducer as shown in Fig. 2P-25. Find the dynamic equation of motion.

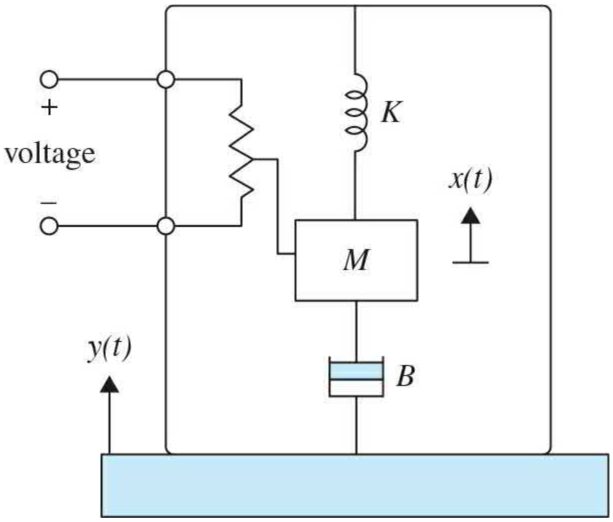


Figure 2P-25

PROBLEMS FOR SEC. 2-2

2-26. Consider the electrical circuits shown in Fig. 2P-26*a* and ***b.***

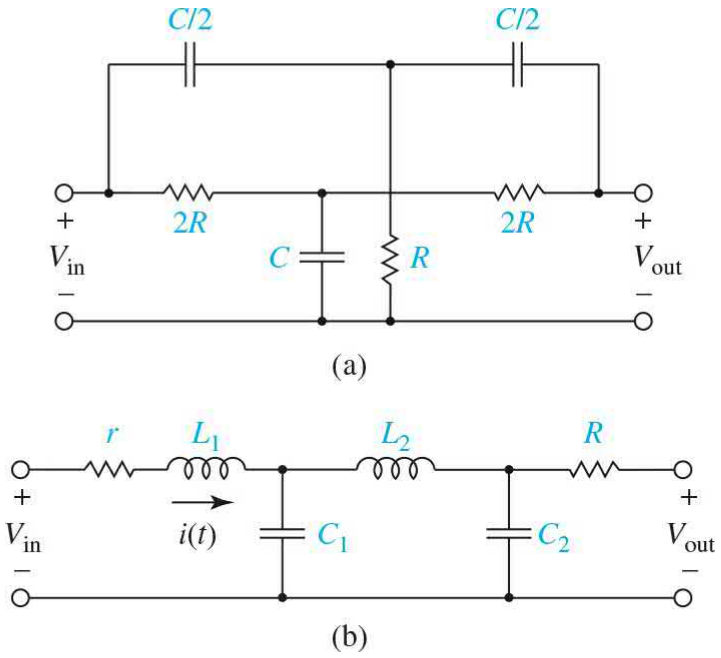


Figure 2P-26

For each circuit find the dynamic equations.

2-27. In a strain gauge circuit, the electrical resistance in one or more of the branches of the bridge circuit, shown in Fig. 2P-27, varies with the strain of the surface to which it is rigidly attached to. The change in resistance results in a differential voltage that is related to the strain. The bridge is composed of two voltage dividers, so the differential voltage Δe can be expressed as the difference in e_1 and e_2.

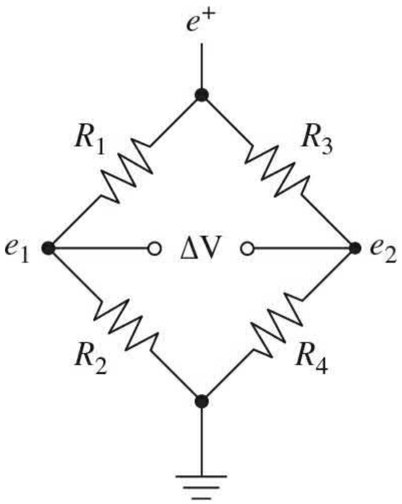


Figure 2P-27

(a) Find Δe.

(b) If the resistance R_2 is has a fixed value of R_2^*, plus a small increment in resistance, δR, then $R_2 = R_2^* + \delta R$. For equal resistance values ($R_1 = R_3 = R_4 = R_2^* = R$), rewrite the bridge equation (i.e., for Δe).

2-28. Figure 2P-28 shows a circuit made up of two *RC* circuits. Find the dynamic equations of the system.

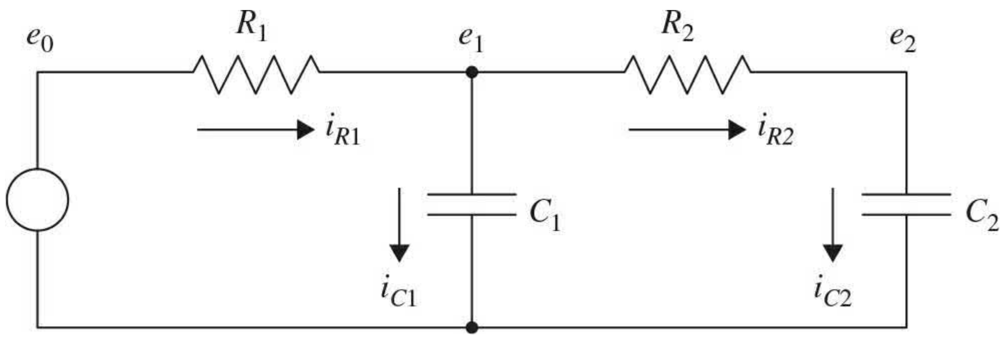


Figure 2P-28

2-29. For the Parallel *RLC* Circuit, shown in Fig. 2P-29, find the dynamic equations of the system.

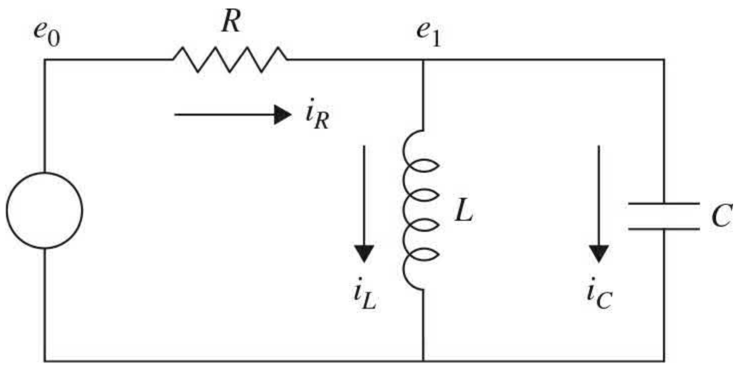


Figure 2P-29

PROBLEMS FOR SEC. 2-3

2-30. Hot oil forging in quenching vat with its cross-sectional view is shown in Fig. 2P-30.

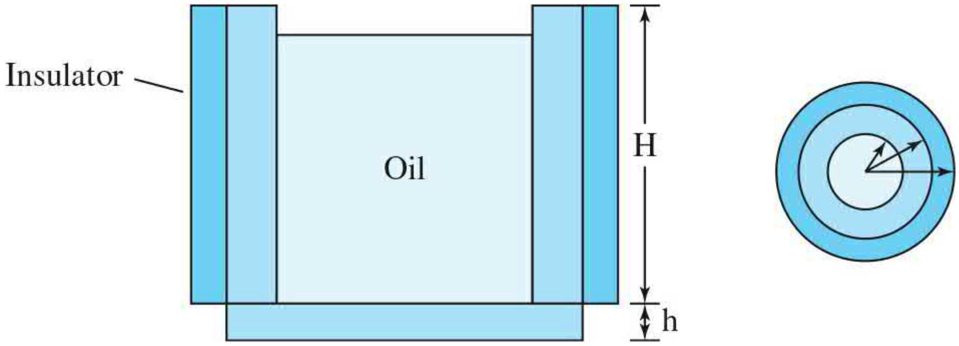


Figure 2P-30

The radii shown in Fig. 20-30 are r_1, r_2, and r_3 from inside to outside. The heat is transferred to the atmosphere from the sides and bottom of the vat and also the surface of the oil with a convective heat coefficient of k_o. Assuming:

k_v = thermal conductivity of the vat
k_i = thermal conductivity of the insulator
c_o = specific heat of the oil
d_o = density of the oil
c = specific heat of the forging
m = mass of the forging
A = surface area of the forging
h = thickness of the bottom of the vat
T_a = ambient temperature

Determine the system model when the temperature of the oil is desired.

2-31. A power supply within an enclosure is shown in Fig. 2P-31. Because the power supply generates lots of heat, a heat sink is usually attached to dissipate the generated heat. Assuming the rate of heat generation within the power supply is known and constant, Q, the heat transfers from the power supply to the enclosure by radiation and conduction, the frame is an ideal insulator, and the heat sink temperature is constant and equal to the atmospheric temperature, determine the model of the system that can give the temperature of the power supply during its operation. Assign any needed parameters.

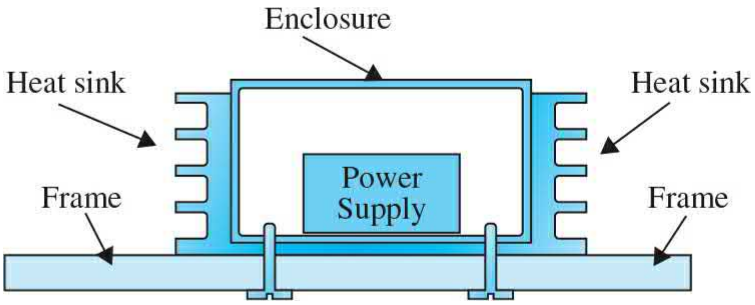


Figure 2P-31

2-32. Figure 2P-32 shows a heat exchanger system.

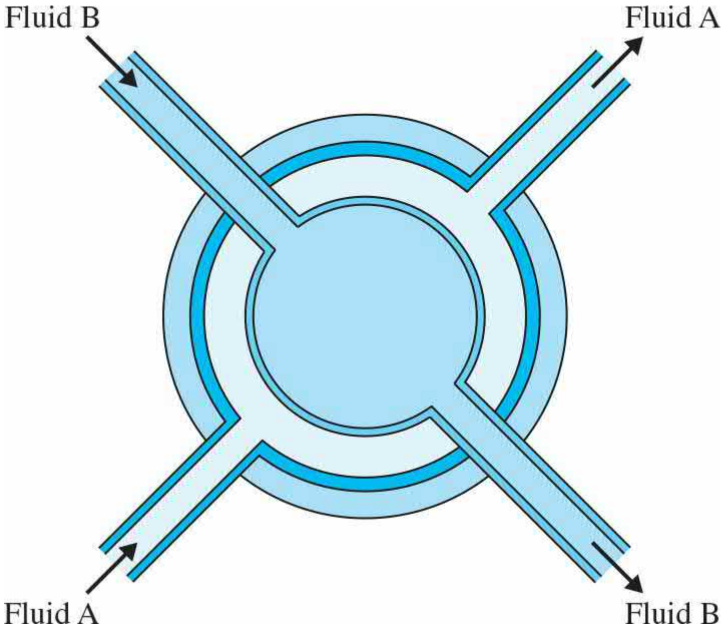


Figure 2P-32

Assuming the simple material transport model represents the rate of heat energy gain for this system, then

$$(\dot{m}c)(T_2 - T_1) = q_{\text{gained}}$$

where m represents the mass flow, T_1 and T_2 are the entering and leaving fluid temperature, and c shows the specific heat of fluid.

If the length of the heat exchanger cylinder is L, derive a model to give the temperature of fluid B leaving the heat exchanger. Assign any required parameters, such as radii, thermal conductivity coefficients, and the thickness.

2-33. Vibration can also be exhibited in fluid systems. Figure 2P-33 shows a U-tube manometer.

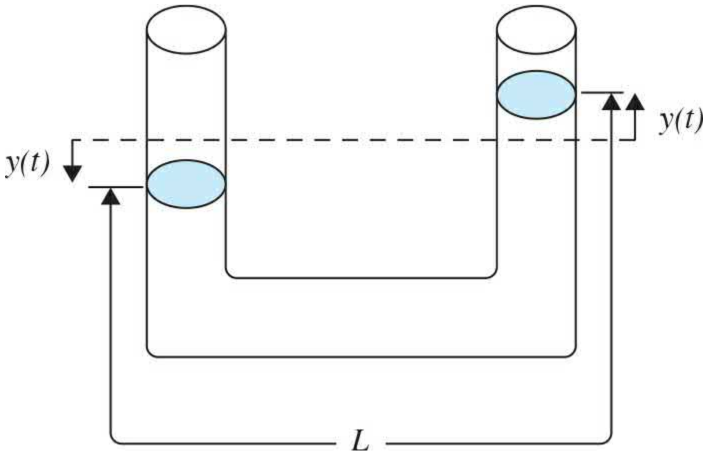


Figure 2P-33

Assume the length of fluid is L, the weight density is m, and the cross-section area of the tube is A.

(a) Write the state equation of the system.

(b) Calculate the natural frequency of oscillation of the fluid.

2-34. A long pipeline connects a water reservoir to a hydraulic generator system as shown in Fig. 2P-34.

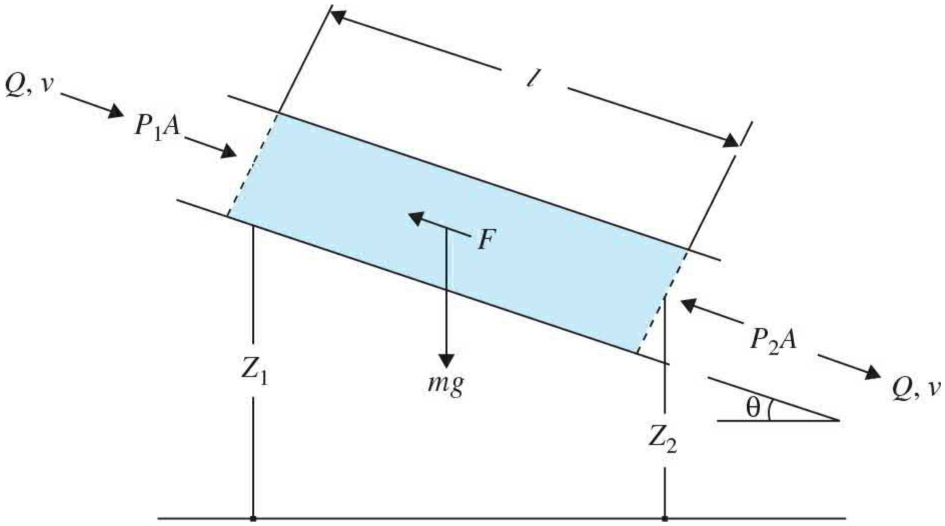


Figure 2P-34

At the end of the pipeline, there is a valve controlled by a speed controller. It may be closed quickly to stop the water flow if the generator loses its load. Determine the dynamic model for the level of the surge tank. Consider the turbine-generator is an energy converter. Assign any required parameters.

2-35. A simplified oil well system is shown in Fig. 2P-35. In this figure, the drive machinery is replaced by the input torque, $T_{in}(t)$. Assuming the pressure in the surrounding rock is fixed at P and the walking beam moves through small angles, determine a model for this system during the upstroke of the pumping rod.

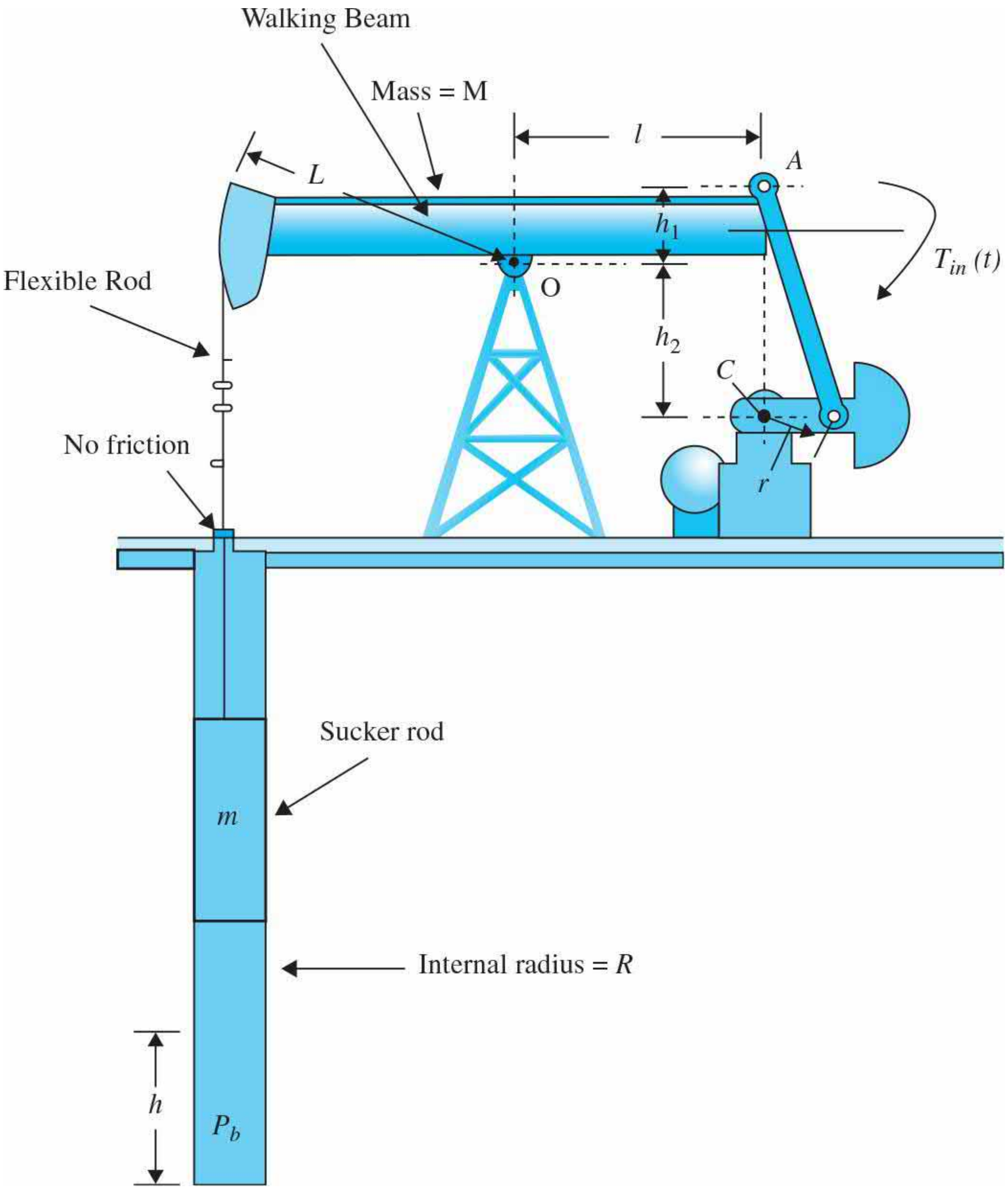


Figure 2P-35

2-36. Figure 2P-36 shows a two-tank liquid-level system. Assume that Q_1 and Q_2 are the steady-state inflow rates, and H_1 and H_2 are steady-state heads. If the other quantities shown in Fig. 2P-36 are supposed to be small, derive the state-space model of the system when h_1 and h_2 are outputs of the system and q_{i1} and q_{i2} are the inputs.

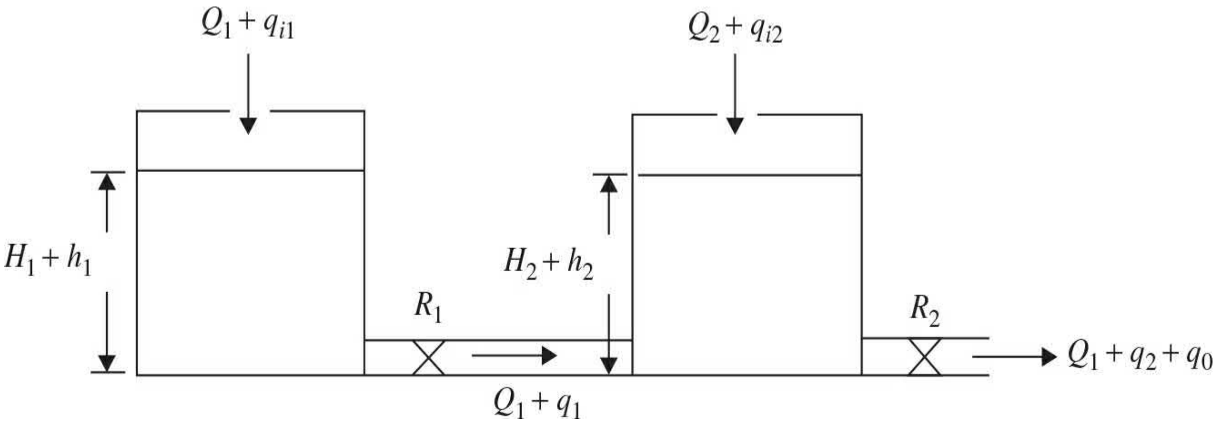


Figure 2P-36

PROBLEMS FOR SEC. 2-4

2-37. Figure 2P-37 shows a typical grain scale.

Assign any required parameters.

(a) Find the free-body diagram.

(b) Derive a model for the grain scale that determines the waiting time for the reading of the weight of grain after placing on the scale platform.

(c) Develop an analogous electrical circuit for this system.

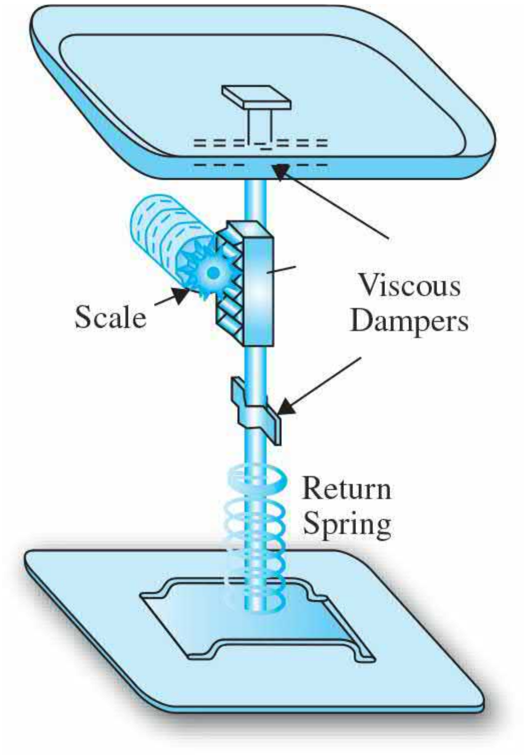


Figure 2P-37

2-38. Develop an analogous electrical circuit for the mechanical system shown in Fig. 2P-38.

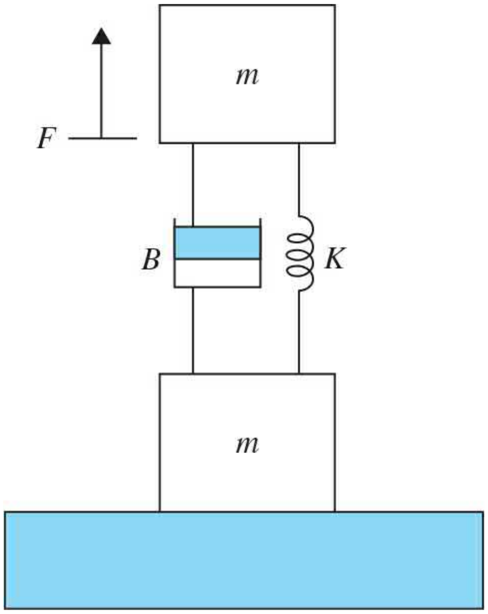


Figure 2P-38

2-39. Develop an analogous electrical circuit for the fluid hydraulic system shown in Fig. 2P-39.

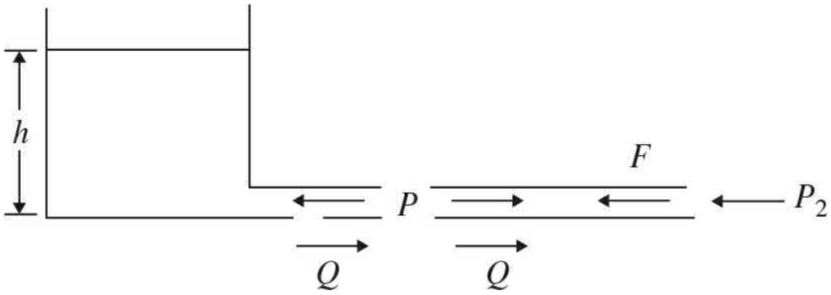


Figure 2P-39

PROBLEMS FOR SEC. 2-5

See Chap. 3 for more linearization problems.

[1] In more complex applications, advanced modeling topics such as Lagrange's approach may be used as alternatives to Newton's modeling approach.

[2] Rotations about an arbitrary axis or an axis passing through the rigid body center of mass are represented by different equations. The reader should refer to a textbook on dynamics of rigid bodies for a more detailed exposure to this topic.

[3] For more in-depth study of this subject, refer to Refs. 1 to 7.

[4] For a more in-depth study of this subject, refer to Refs. 1 to 7.

[5] In a **force-current** analogy, $f(t)$ and $v(t)$ are analogous to $i(t)$ and $e(t)$, respectively, while M, K, and B are analogous to C, $1/L$, and $1/R$, respectively.

[6] For instructions on setting up the Bluetooth connection, visit http://www.mathworks.com/matlabcentral/fileexchange/35206-simulink-support-package-for-lego-mindstorms-nxt-hardware/content/lego/legodemos/html/publish_lego_communication.html#4.

CHAPTER 3

Solution of Differential Equations of Dynamic Systems

Before starting this chapter, the reader is encouraged to refer to App. B to review the theoretical background related to complex variables.

As mentioned in Chap. 2, the design process of a control system starts with development of a mathematical model of the system, represented by differential equations. In this textbook, as in many conventional control engineering applications, we consider systems that are modeled by ordinary differential equations—as opposed to partial differential equations.

Learning Outcomes

After successful completion of this chapter, you will be able to

1. Convert linear time-invariant ordinary differential equations into the Laplace domain.
2. Find the transfer function, poles and zeros of differential equations, represented in the Laplace domain.
3. Find the response of linear time-invariant differential equations using inverse Laplace transforms.
4. Understand the behavior of first and prototype second-order differential equations.
5. Find the state space representation of linear time-invariant differential equations.
6. Find the response of state space equations using inverse Laplace transforms.
7. Find transfer functions using the state space approach.
8. Find the state space representation, from the transfer function of the system.

Once we obtain the equations of the system, we need to develop a set of analytical and numerical tools that can assist us with a clear understanding of the performance of the system. This is an important step in advance of extending the design of a control system to a prototype or the actual system. Two most common tools for studying the behavior of (i.e., the solution of) a control system are the transfer function and the state-variable methods. Transfer functions are based on the Laplace transform technique and are valid only for linear time-invariant systems, whereas the state equations can be applied to linear as well as nonlinear systems.

In this chapter, we review ordinary time-invariant differential equations and how they are treated utilizing the Laplace transforms or the state space approach. The main objectives of this chapter are

- To review ordinary time-invariant differential equations.
- To review the fundamentals of Laplace transforms.
- To demonstrate the applications of Laplace transforms to solve linear ordinary differential equations.
- To introduce the concept of transfer functions and how to apply them to the modeling of linear time-invariant systems.
- To introduce state space systems.
- To provide examples on how Laplace transforms and state space systems are used to solve differential equations.

Because some of these topics are considered as review material for the reader, the treatment of these subjects will not be exhaustive. Additional supplemental material can be found in appendices.

3-1 INTRODUCTION TO DIFFERENTIAL EQUATIONS

As discussed in Chap. 2, a wide range of systems in engineering are modeled mathematically by differential equations. These equations generally involve derivatives (or integrals) of the dependent variables with respect to the independent variable—usually time. For instance, as shown in Sec. 2-2-2, a series electric *RLC* (resistance-inductance-capacitance) network, shown in Fig. 3-1*a*, can be represented by the differential equation:

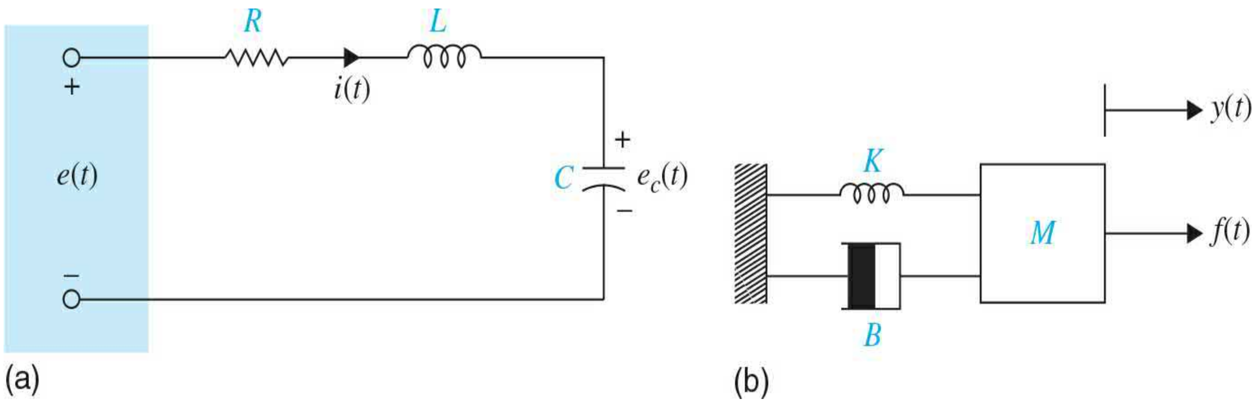


Figure 3-1 (*a*) A series *RLC* network. (*b*) A spring-mass-damper system.

$$LC\frac{d^2e_c(t)}{dt^2}+RC\frac{de_c(t)}{dt}+e_c(t)=e(t) \tag{3-1}$$

or alternatively,

$$L\frac{di(t)}{dt}+Ri(t)+\int\frac{i(t)}{C}dt=e(t) \tag{3-2}$$

Similarly from Sec. 2-1-1, a simple mass-spring-dashpot mechanism, shown in Fig. 3-1*b*, may be modeled using the Newton's second law as

$$M\frac{d^2y(t)}{dt^2}(t)+B\frac{dy(t)}{dt}+Ky(t)=f(t) \tag{3-3}$$

Recall from Chap. 2 that these two systems are analogous, and they can both be represented by a **standard prototype second-order** system of the form:

$$\ddot{y}(t)+2\zeta\omega_n\dot{y}(t)+\omega_n^2y(t)=\omega_n^2u(t) \tag{3-4}$$

3-1-1 Linear Ordinary Differential Equations

In general, the differential equation of an *n*th-order system is written as

$$\frac{d^n y(t)}{dt^n} + a_{n-1}\frac{d^{n-1}y(t)}{dt^{n-1}} + \cdots + a_1\frac{dy(t)}{dt} + a_0 y(t)$$
$$= b_m\frac{d^m u(t)}{dt^m} + b_{m-1}\frac{d^{m-1}u(t)}{dt^{m-1}} + \cdots + b_1\frac{du(t)}{dt} + b_0 u(t) \tag{3-5}$$

which is also known as a **linear ordinary differential equation** if the coefficients $a_0, a_1, \ldots, a_{n-1}$ and $b_0, b_1, \ldots, b_m$ are real constants and are not functions of $y(t)$. In control systems, the terms $u(t)$ and $y(t)$ are known as the input and the output of the system, respectively, for $t \ge t_0$. Note that, in general, there may be more than one input function applied to Eq. (3-5). But since the system is linear, we can utilize the principle of superposition and study the effect of each input on the system separately.

A first-order linear ordinary differential equation is therefore in the general form

$$\frac{dy(t)}{dt} + a_0 y(t) = f(t) \tag{3-6}$$

and the second-order general form of a linear ordinary differential equation is

$$\frac{d^2 y(t)}{dt^2} + a_1\frac{dy(t)}{dt} + a_0 y(t) = f(t) \tag{3-7}$$

In this text, we primarily study systems represented by ordinary differential equations. As we saw, in Chap. 2, there are systems—for example, fluid and heat transfer systems—that are actually modeled by partial differential equations. In that case, these system equations are modified—under special circumstances such as restricting the fluid flow in one direction—and are converted into ordinary differential equations.

3-1-2 Nonlinear Differential Equations

Many physical systems are nonlinear and must be described by nonlinear differential equations. For instance, the differential equation describing the motion of a pendulum of mass m and length ℓ, Eq. (2-169), is

$$\ddot{\theta} + \frac{g}{\ell}\sin\theta = 0 \tag{3-8}$$

Because in Eq. (3-8), sin$\theta(t)$ is nonlinear, the system is a **nonlinear system**. In order to be able to treat a nonlinear system, in most engineering practices, the corresponding equations are linearized, about a certain operating point, and are converted into a linear ordinary differential equation form. In this case using static equilibrium position $\theta = 0$ as the operating point, for small motions, the linearization of the system implies $\theta \approx \theta$, or

$$\ddot{\theta} + \frac{g}{\ell}\theta = 0 \tag{3-9}$$

See Sec. 2-4 for more details on Taylor series linearization technique. Also you may wish to refer to Sec. 3-9 where this topic is revisited in matrix format.

3-2 LAPLACE TRANSFORM

Laplace transform technique is one of the mathematical tools used to solve linear ordinary differential equations. This approach is very popular in the study of control systems because of the following two features:

> ***Laplace transform** is one of the techniques used to solve linear ordinary differential equations.*

1. The homogeneous and the particular components in the solution of the differential equation are obtained in one operation.
2. The Laplace transform converts the differential equation into simple to manipulate algebraic form—in what is known as the *s*-domain.

3-2-1 Definition of the Laplace Transform

Given the real function $f(t)$ that satisfies the condition:

$$\int_0^\infty \left| f(t)e^{-\sigma t} \right| dt < \infty \tag{3-10}$$

for some finite, real **σ**, the Laplace transform of *f* (*t*) is defined as

$$F(s) = \int_{0^-}^{\infty} f(t)e^{-st}dt \tag{3-11}$$

or

$$F(s) = \text{Laplace transform of } f(t) = \mathcal{L}[f(t)] \tag{3-12}$$

The variable *s* is referred to as the **Laplace operator**, which is a complex variable; that is, $s = \sigma + j\omega$, where σ is the real component, $j = \sqrt{-1}$ and ω is the imaginary component.

The defining equation in Eq. (3-12) is also known as the **one-sided Laplace transform**, as the integration is evaluated from $t = 0$ to ∞. This simply means that all information contained in $f(t)$ prior to $t = 0$ is ignored or considered to be zero. This assumption does not impose any limitation on the applications of the Laplace transform to linear systems, since in the usual time-domain studies, time reference is often chosen at $t = 0$. Furthermore, for a physical system when an input is applied at $t = 0$, the response of the system does not start sooner than $t = 0$.[1] Such a system is also known as being **causal** or simply **physically realizable**.

The following examples illustrate how Eq. (3-12) is used for the evaluation of the Laplace transform of $f(t)$.

EXAMPLE 3-2-1 Let $f(t)$ be a unit-step function that is defined as

$$f(t) = u_s(t) = \begin{cases} 0, & t < 0 \\ 1, & t \geq 0 \end{cases} \tag{3-13}$$

The Laplace transform of $f(t)$ is obtained as

$$F(s) = L[u_s(t)] = \int_0^{\infty} u_s(t)e^{-st}\,dt = -\frac{1}{s}e^{-st}\bigg|_0^{\infty} = \frac{1}{s} \tag{3-14}$$

Eq. (3-14) is valid if

$$\int_0^{\infty} \left|u_s(t)e^{-\sigma t}\right|dt = \int_0^{\infty} \left|e^{-\sigma t}\right|dt < \infty \tag{3-15}$$

which means that the real part of *s*, σ , must be greater than zero. Having said that, *in practice*, we simply refer to the Laplace transform of the unit-step function as 1/*s*. ▲

EXAMPLE 3-2-2 Consider the exponential function

$$f(t)=e^{-\alpha t} \quad t\geq 0 \tag{3-16}$$

where α is a real constant. The Laplace transform of *f*(*t*) is written

$$F(s)=\int_0^\infty e^{-\alpha t}e^{-st}\,dt=\left.\frac{e^{-(s+\alpha)t}}{s+\alpha}\right|_0^\infty=\frac{1}{s+\alpha} \tag{3-17}$$

▲

Toolbox 3-2-1

Use the MATLAB symbolic toolbox to find the Laplace transforms.

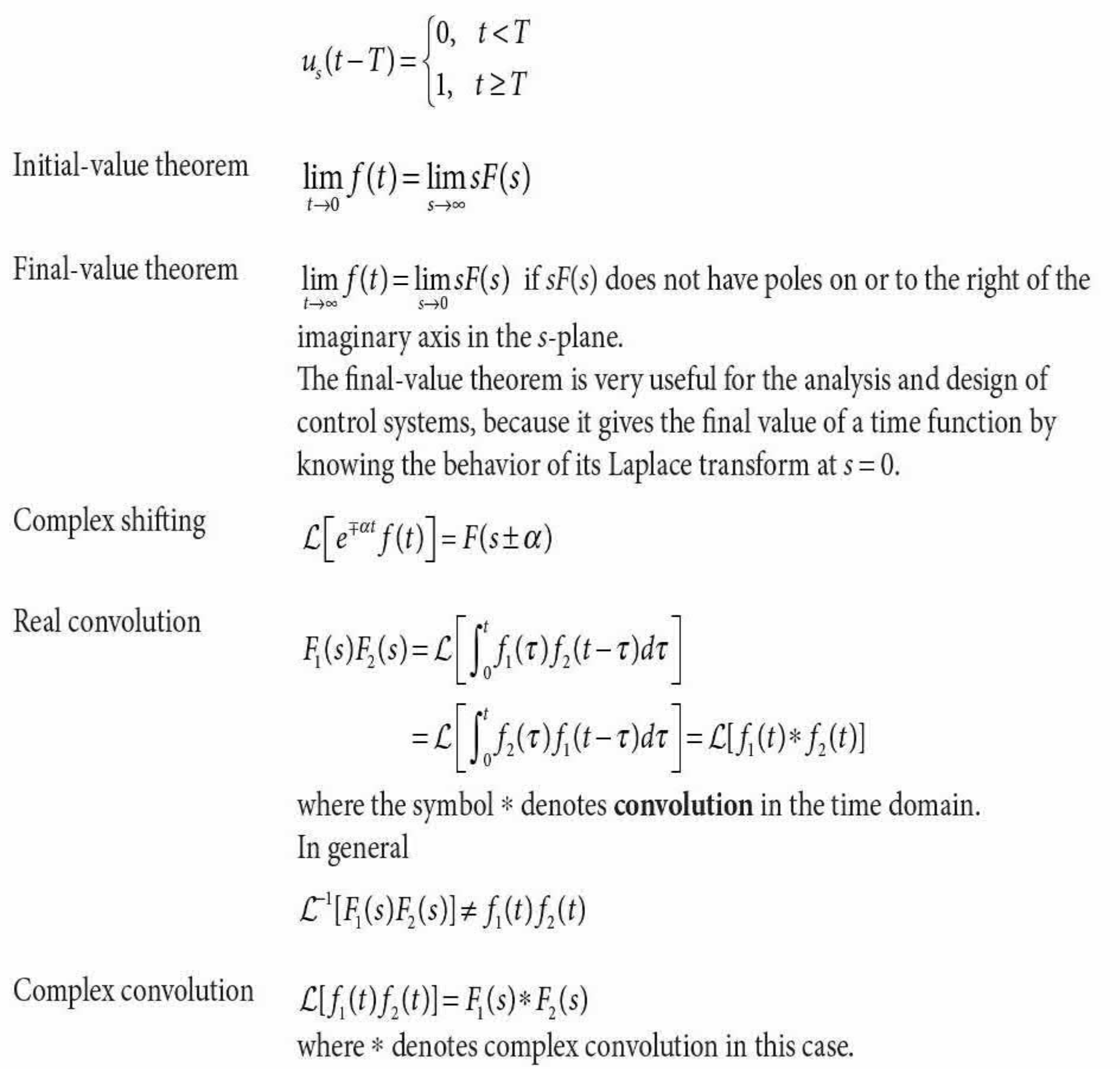
```
>> syms t
>> f = t^4
f =
t^4
>> laplace(f)
ans =
24/s^5
```

3-2-2 Important Theorems of the Laplace Transform

The applications of the Laplace transform in many instances are simplified by utilization of the properties of the transform. These properties are presented in Table 3-1 for which no proofs are given here.

TABLE 3-1 Theorems of Laplace Transforms

Multiplication by a constant	$\mathcal{L}[kf(t)] = kF(s)$
Sum and difference	$\mathcal{L}[f_1(t) \pm f_2(t)] = F_1(s) \pm F_2(s)$
Differentiation	$\mathcal{L}\left[\frac{df(t)}{dt}\right] = sF(s) - f(0)$ $\mathcal{L}\left[\frac{d^n f(t)}{dt^n}\right] = s^n F(s) - s^{n-1} f(0) - s^{n-2} f^{(1)}(0) - \cdots - f^{(n-1)}(0)$ where $f^{(k)}(0) = \left.\frac{d^k f(t)}{dt^k}\right\|_{t=0}$ denotes the kth-order derivative of $f(t)$ with respect to t, evaluated at $t = 0$.
Integration	$\mathcal{L}\left[\int_0^t f(\tau) d\tau\right] = \frac{F(s)}{s}$ $\mathcal{L}\left[\int_0^{t_n}\int_0^{t_{n-1}} \cdots \int_0^{t_1} f(t) d\tau dt_1 dt_2 \cdots dt_{n-1}\right] = \frac{F(s)}{s^n}$
Shift in time	$\mathcal{L}[f(t-T)u_s(t-T)] = e^{-Ts}F(s)$ where $u_s(t–T)$ denotes the unit-step function that is shifted in time to the right by T.



	$u_s(t-T)=\begin{cases}0, & t<T\\1, & t\geq T\end{cases}$
Initial-value theorem	$\lim_{t\to 0} f(t)=\lim_{s\to\infty} sF(s)$
Final-value theorem	$\lim_{t\to\infty} f(t)=\lim_{s\to 0} sF(s)$ if $sF(s)$ does not have poles on or to the right of the imaginary axis in the s-plane. The final-value theorem is very useful for the analysis and design of control systems, because it gives the final value of a time function by knowing the behavior of its Laplace transform at $s=0$.
Complex shifting	$\mathcal{L}\left[e^{\mp\alpha t}f(t)\right]=F(s\pm\alpha)$
Real convolution	$F_1(s)F_2(s)=\mathcal{L}\left[\int_0^t f_1(\tau)f_2(t-\tau)d\tau\right]$ $=\mathcal{L}\left[\int_0^t f_2(\tau)f_1(t-\tau)d\tau\right]=\mathcal{L}[f_1(t)*f_2(t)]$ where the symbol $*$ denotes **convolution** in the time domain. In general $\mathcal{L}^{-1}[F_1(s)F_2(s)]\neq f_1(t)f_2(t)$
Complex convolution	$\mathcal{L}[f_1(t)f_2(t)]=F_1(s)*F_2(s)$ where $*$ denotes complex convolution in this case.

3-2-3 Transfer Function

In classical control, **transfer functions** are used to represent input-output relations between variables. Let us consider the following *n*th-order differential equation with constant real coefficients:

$$\frac{d^n y(t)}{dt^n}+a_{n-1}\frac{d^{n-1}y(t)}{dt^{n-1}}+\cdots+a_1\frac{dy(t)}{dt}+a_0y(t)$$
$$=b_m\frac{d^m u(t)}{dt^m}+b_{m-1}\frac{d^{m-1}u(t)}{dt^{m-1}}+\cdots+b_1\frac{du(t)}{dt}+b_0u(t) \qquad (3\text{-}18)$$

> The **transfer function** between a pair of input and output variables is the ratio of the Laplace transform of the output to the Laplace transform of the input.

The coefficients $a_0, a_1, \ldots, a_{n-1}$ and $b_0, b_1, \ldots b_m$ are real constants. Once the **input** $u(t)$ for $t \geq t_0$ and the initial conditions of $y(t)$ and the derivatives of $y(t)$ are specified at the initial time $t = t_0$, the **output** response $y(t)$ for $t \geq t_0$ is determined by solving Eq. (3-18).

The **transfer function** of Eq. (3-18) is a function $G(s)$, defined as

$$G(s) = \mathcal{L}[g(t)] \tag{3-19}$$

Taking the Laplace transform on both sides of the equation and assume **zero initial conditions**. The result is

$$\left(s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0\right)Y(s) = \left(b_m s^m + b_{m-1}s^{m-1} + \cdots + b_1 s + b_0\right)U(s) \tag{3-20}$$

The transfer function between $u(t)$ and $y(t)$ is given by

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^m + b_{m-1}s^{m-1} + \cdots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0} \tag{3-21}$$

The properties of the transfer function[2] are summarized as follows:

- The transfer function of a linear system of differential equations is the ratio of the Laplace transform of the output to the Laplace transform of the input.
- All initial conditions of the system are set to zero.
- The transfer function is independent of the input of the system.

3-2-4 Characteristic Equation

The characteristic equation *of a linear system is defined as the equation obtained by setting the denominator polynomial of the transfer function to zero*. Thus, for the system described by differential equation shown in Eq. (3-18), the characteristic equation of the system is obtained from the denominator of the transfer function in Eq. (3-21) so that

$$s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 = 0 \tag{3-22}$$

3-2-5 Analytic Function

A function G(s) *of the complex variable* s *is called an* ***analytic function*** *in a region of the* s*-plane if the function and all its derivatives exist in the region*. For instance, the function

$$G(s) = \frac{1}{s(s+1)} \tag{3-23}$$

is analytic at every point in the *s*-plane except at the points *s* = 0 and *s* = –1. At these two points, the value of the function is infinite. As another example, the function *G*(*s*) = *s* + 2 is analytic at every point in the finite *s*-plane.

3-2-6 Poles of a Function

A pole, also known as a singularity, plays a very important role in the studies of classical control theory. Loosely speaking, the **poles** of a transfer function in Eq. (3-21) are the points in the **Cartesian** coordinate frame (aka the *s*-plane) at which the function becomes infinite. In other words, the poles are also the roots of the characteristic Eq. (3-22), which make the denominator of *G*(*s*) to become equal to zero.[3]

If the denominator of *G*(*s*) includes the factor $(s - p_i)^r$, for *r* =1, the pole at $s = p_i$ is called a **simple pole**; for *r* = 2, the pole at $s = p_i$ is of order two; etc.

As an example, the function

$$G(s) = \frac{10(s+2)}{s(s+1)(s+3)^2} \tag{3-24}$$

has a pole of order 2 at *s* = –3 and simple poles at *s* = 0 and *s* = –1. It can also be said that the function *G*(*s*) is analytic in the *s*-plane except at these poles. See Fig. 3-2 for the graphical representation of the finite poles of the system in the *s*-plane.

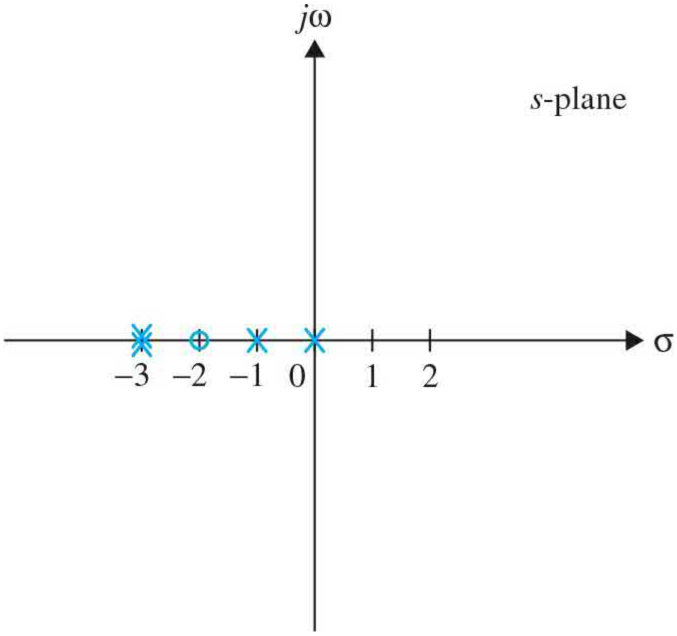


Figure 3-2 Graphical representation of $G(s)=\dfrac{10(s+2)}{s(s+1)(s+3)^2}$ in the s-plane: × poles and O zeros.

3-2-7 Zeros of a Function

The **zeros** of a transfer function $G(s)$ in Eq. (3-24) are the points in the s-plane at which the function becomes zero.

If the numerator of $G(s)$ includes the factor $(s - z_i)^r$, for r =1, the zero at $s = z_i$ is called a **simple zero**; for $r = 2$, the zero at $s = z_i$ is of order two; and so on.

In other words, the zeros of $G(s)$ are also the roots of the numerator equation in Eq. (3-24).[4] For example, the function in Eq. (3-24) has a simple zero at $s = –2$.

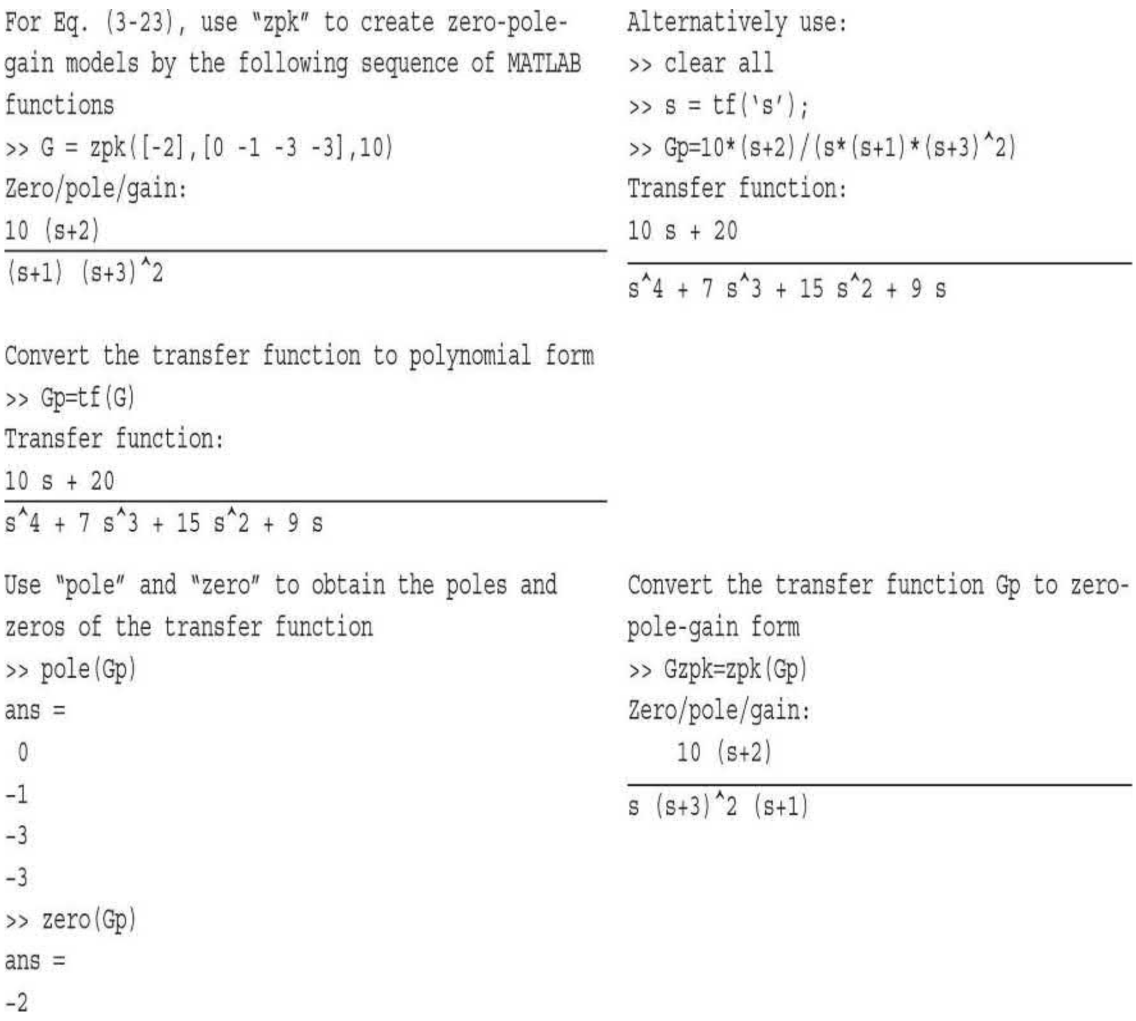
Mathematically speaking, the total number of poles equals the total number of zeros, counting the multiple-order poles and zeros and taking into account the poles and zeros at infinity. The function in Eq. (3-24) has four finite poles at s = 0, –1, –3, and –3; there is one finite zero at s = –2, but there are three zeros at infinity, because

$$\lim_{s\to\infty} G(s) = \lim_{s\to\infty}\frac{10}{s^3} = 0 \tag{3-25}$$

Therefore, the function has a total of four poles and four zeros in the entire *s*-plane, including infinity. See Fig. 3-2 for the graphical representation of the finite zeros of the system.

Practically speaking, we only consider the finite poles and zeros of a function.

Toolbox 3-2-2



For Eq. (3-23), use "zpk" to create zero-pole-gain models by the following sequence of MATLAB functions

```
>> G = zpk([-2],[0 -1 -3 -3],10)
Zero/pole/gain:
10 (s+2)
------------------
(s+1) (s+3)^2
```

Alternatively use:

```
>> clear all
>> s = tf('s');
>> Gp=10*(s+2)/(s*(s+1)*(s+3)^2)
Transfer function:
10 s + 20
-----------------------------
s^4 + 7 s^3 + 15 s^2 + 9 s
```

Convert the transfer function to polynomial form

```
>> Gp=tf(G)
Transfer function:
10 s + 20
-----------------------------
s^4 + 7 s^3 + 15 s^2 + 9 s
```

Use "pole" and "zero" to obtain the poles and zeros of the transfer function

```
>> pole(Gp)
ans =
 0
-1
-3
-3
>> zero(Gp)
ans =
-2
```

Convert the transfer function Gp to zero-pole-gain form

```
>> Gzpk=zpk(Gp)
Zero/pole/gain:
    10 (s+2)
--------------------
s (s+3)^2 (s+1)
```

3-2-8 Complex Conjugate Poles and Zeros

When dealing with the time response of control systems, see Chap. 7, complex-conjugate poles (or zeros) play an important role, and as a result, they deserve special treatment here. Consider the transfer function

$$G(s)=\frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2} \tag{3-26}$$

Let us assume that the value of ζ is less than 1, so that $G(s)$ has a pair of **simple complex-conjugate poles** at

$$s=s_1=-\sigma+j\omega \text{ and } s=s_2=-\sigma-j\omega \tag{3-27}$$

where

$$\sigma=\zeta\omega_n \tag{3-28}$$

and

$$\omega=\omega_n\sqrt{1-\zeta^2} \tag{3-29}$$

The poles in Eq. (3-27) are represented in **rectangular form,** where $j=\sqrt{-1}$, $(-\sigma,\omega)$, and $(-\sigma,-\omega)$ are real and imaginary coefficients of s_1 and s_2, respectively. Focusing on $(-\sigma,\omega)$, it represents a point in the s-plane as shown in Fig. 3-3. A point in a rectangular coordinate frame may also be defined by a vector R and an angle ϕ. It is then easy to see that

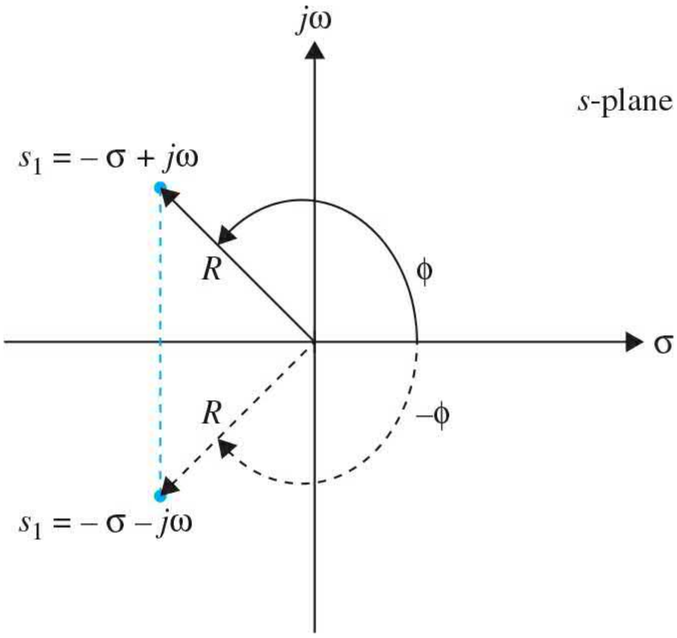


Figure 3-3 A pair of complex conjugate poles in the *s*-plane.

$$\begin{aligned} -\sigma &= R\sin\phi \\ \omega &= R\cos\phi \end{aligned} \tag{3-30}$$

where

R = magnitude of *s*

ϕ = phase of *s* and is measured from the *σ* (real) axis. Right-hand rule convention: positive phase is in counterclockwise direction.

Hence,

$$\begin{aligned} R &= \sqrt{\sigma^2 + \omega^2} \\ \phi &= \tan^{-1}\frac{\omega}{-\sigma} \end{aligned} \tag{3-31}$$

Introducing Eq. (3-30) into s_1 in Eq. (3-27), we get

$$s = s_1 = R(\cos\phi + j\sin\phi) \tag{3-32}$$

Upon comparison of Taylor series of the terms involved, it is easy to confirm

$$e^{j\phi} = \cos\phi + j\sin\phi \tag{3-33}$$

Equation (3-33) is also known as the **Euler formula**. As a result, s_1 in Eq. (3-1) may also be represented in **polar form** as

$$s = s_1 = Re^{j\phi} = R\angle\phi \tag{3-34}$$

Note that the **conjugate** of the complex pole in Eq. (3-34) is

$$s = s_2 = R(\cos\phi - j\sin\phi) = Re^{-j\phi} = R\angle-\phi \tag{3-35}$$

3-2-9 Final-Value Theorem

The final-value theorem is very useful for the analysis and design of control systems because it gives the final value of a time function by knowing the behavior of its Laplace transform at *s* = 0. The theorem states: *If the*

Laplace transform of f (*t*) *is F*(*s*), *and if sF*(*s*) *is analytic* (see Sec. 3-2-5 on the definition of an analytic function) *on the imaginary axis and in the right half of the s-plane, then*

$$\lim_{t\to\infty} f(t) = \lim_{s\to 0} sF(s) \tag{3-36}$$

The final-value theorem is *not* valid if *sF*(*s*) contains any pole whose real part is zero or positive, which is equivalent to the analytic requirement of *sF*(*s*) in the right-half *s*-plane, as stated in the theorem. The following examples illustrate the care that must be taken in applying the theorem.

EXAMPLE 3-2-3 Consider the function

$$F(s) = \frac{5}{s(s^2+s+2)} \tag{3-37}$$

Because *sF*(*s*) is analytic on the imaginary axis and in the right-half *s*-plane, the final-value theorem may be applied. Using Eq. (3-36), we have

$$\lim_{t\to\infty} f(t) = \lim_{s\to 0} sF(s) = \lim_{s\to 0} \frac{5}{s^2+s+2} = \frac{5}{2} \tag{3-38}$$

EXAMPLE 3-2-4 Consider the function

$$F(s) = \frac{\omega}{s^2+\omega^2} \tag{3-39}$$

which is the Laplace transform of *f*(*t*) = sin *ωt*. Because the function *sF*(*s*) has two poles on the imaginary axis of the *s*-plane, the final-value theorem *cannot* be applied in this case. In other words, although the final-value theorem would yield a value of zero as the final value of *f*(*t*), the result is erroneous.

3-3 INVERSE LAPLACE TRANSFORM BY

PARTIAL-FRACTION EXPANSION

Given the Laplace transform $F(s)$, the operation of obtaining $f(t)$ is termed the inverse Laplace transformation and is denoted by

$$f(t) = \text{Inverse Laplace transform of } F(s) = \mathcal{L}^{-1}[F(S)] \tag{3-40}$$

The inverse Laplace transform integral is given as

$$f(t) = \frac{1}{2\pi j}\int_{c-j\infty}^{c+j\infty} F(s)e^{st}ds \tag{3-41}$$

where c is a real constant that is greater than the real parts of all the singularities of $F(s)$. Equation (3-41) represents a line integral that is to be evaluated in the s-plane. In a majority of the problems in control systems, the evaluation of the inverse Laplace transform does not rely on the use of the inversion integral of Eq. (3-41). For simple functions, the inverse Laplace transform operation can be carried out simply by referring to the **Laplace transform table**, such as the one given in App. C. For complex functions, the inverse Laplace transform can be carried out by first performing a **partial-fraction expansion** on $F(s)$ and then using the Transform table. You can also use the MATLAB symbolic tool to find the inverse Laplace transform of a function.

3-3-1 Partial Fraction Expansion

When the Laplace transform solution of a differential equation is a rational function in s, it can be written as

$$G(s) = \frac{Q(s)}{P(s)} \tag{3-42}$$

where $P(s)$ and $Q(s)$ are polynomials of s. It is assumed that *the order of* $P(s)$ *in s is greater than that of* $Q(s)$. The polynomial $P(s)$ may be written as

$$P(s) = s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 \tag{3-43}$$

where $a_0, a_1, \ldots, a_{n-1}$ are real coefficients. The methods of partial-fraction expansion will now be given for the cases of simple poles, multiple-order poles, and complex-conjugate poles of *G*(*s*). The idea here is to simplify *G*(*s*) as much as possible to allow us find its inverse Laplace transform at ease without referring to tables.

G(*s*) Has Simple Poles

If all the poles of *G*(*s*) are simple and real, Eq. (3-42) can be written as

$$G(s)=\frac{Q(s)}{P(s)}=\frac{Q(s)}{(s+s_1)(s+s_2)\cdots(s+s_n)} \tag{3-44}$$

where $s_1 \neq s_2 \neq \ldots \neq s_n$. Applying the partial-fraction expansion, Eq. (3-43) is written as

$$G(s)=\frac{K_{s1}}{s+s_1}+\frac{K_{s2}}{s+s_2}+\cdots+\frac{K_{sn}}{s+s_n} \tag{3-45}$$

or

$$G(s)=\frac{K_{s1}}{s+s_1}+\frac{K_{s2}}{s+s_2}+\cdots+\frac{K_{\mathrm{sn}}}{s+s_n}=\frac{Q(s)}{(s+s_1)(s+s_2)\cdots(s+s_n)} \tag{3-46}$$

The coefficient $K_{si}(i = 1,2,...,n)$ is determined by multiplying both sides of Eq. (3-46) by the factor $(s + s_i)$ and then setting s equal to $-s_i$. To find the coefficient K_{s1}, for instance, we multiply both sides of Eq. (3-46) by $(s + s_1)$ and let $s = -s_1$. Thus,

$$K_{s_1}=[(s+s_1)G(s)]\Big|_{s=-s_1} \tag{3-47}$$

or

$$\begin{aligned} K_{s1} &= K_{s1}+\cancel{\frac{(-s_1+s_1)}{-s_1+s_2}}K_{s2}+\cdots+\cancel{\frac{(-s_1+s_1)}{-s_1+s_n}}K_{sn} \\ &= \frac{Q(-s_1)}{(s_2-s_1)(s_3-s_1)\cdots(s_n-s_1)} \end{aligned} \tag{3-48}$$

EXAMPLE 3-3-1 Consider the function

$$G(s)=\frac{5s+3}{(s+1)(s+2)(s+3)}=\frac{5s+3}{s^3+6s^2+11s+6} \tag{3-49}$$

which is written in the partial-fraction expanded form

$$G(s)=\frac{K_{-1}}{s+1}+\frac{K_{-2}}{s+2}+\frac{K_{-3}}{s+3} \tag{3-50}$$

The coefficients K_{-1},K_{-2}, and K_{-3} are determined as follows:

$$K_{-1}=[(s+1)G(s)]\big|_{s=-1}=\frac{5(-1)+3}{(2-1)(3-1)}=-1 \tag{3-51}$$

$$K_{-2}=[(s+2)G(s)]\big|_{s=-2}=\frac{5(-2)+3}{(1-2)(3-2)}=7 \tag{3-52}$$

$$K_{-3}=[(s+3)G(s)]\big|_{s=-3}=\frac{5(-3)+3}{(1-3)(2-3)}=-6 \tag{3-53}$$

Thus, Eq. (3-49) becomes

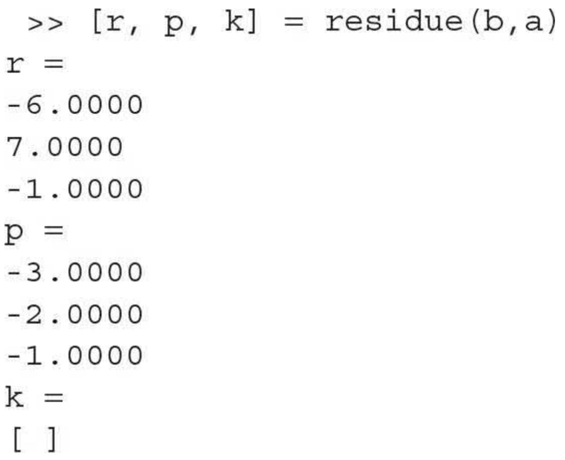
$$G(s)=\frac{-1}{s+1}+\frac{7}{s+2}-\frac{6}{s+3} \tag{3-54}$$

Toolbox 3-3-1

For Example 3-3-1, Eq. (3-49) is a ratio of two polynomials.

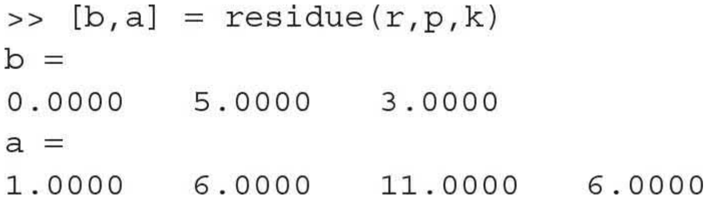
```
>> b = [5 3] % numerator polynomial coefficients
>> a = [1,6,11,6] % denominator polynomial coefficients
```

You can calculate the partial fraction expansion as



```
 >> [r, p, k] = residue(b,a)
r =
-6.0000
7.0000
-1.0000
p =
-3.0000
-2.0000
-1.0000
k =
[ ]
```

Note r represents the numerators of Eq. (3-54), and p represents the corresponding pole values. Now, convert the partial fraction expansion back to polynomial coefficients.



```
>> [b,a] = residue(r,p,k)
b =
0.0000    5.0000    3.0000
a =
1.0000    6.0000    11.0000    6.0000
```

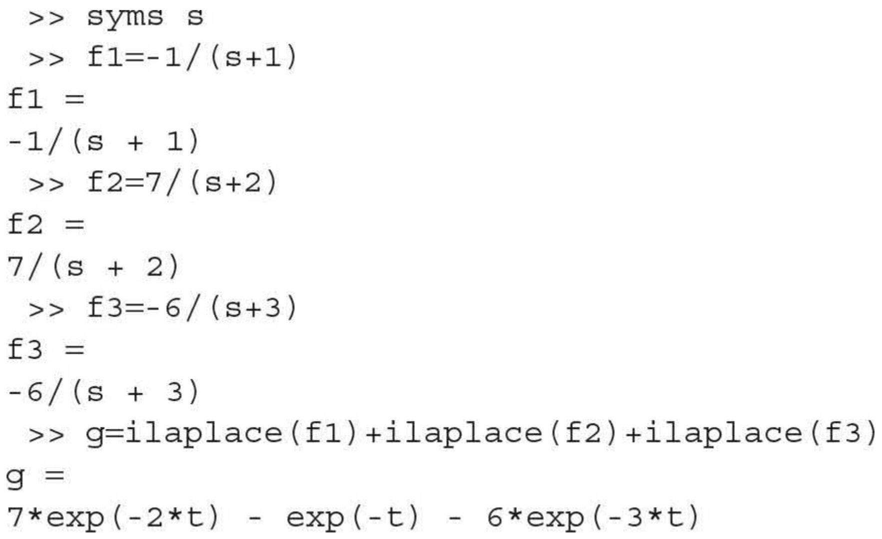
Note b and a represent the coefficients of the numerator and denominator polynomials in Eq. (3-49), respectively. Note also that the result is normalized for the leading coefficient in the denominator.

Taking the **inverse Laplace transform** on both sides using the Laplace table in App. C or using the following MATLAB toolbox, we get

$$g(t) = -e^{-t} + 7e^{-2t} - 6e^{-3t} \quad t \geq 0 \tag{3-55}$$

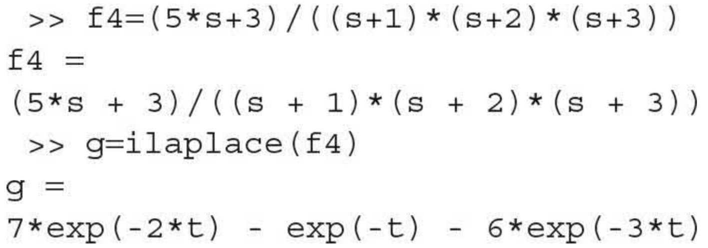
Toolbox 3-3-2

For Example 3-3-1, Eq. (3-54) is composed of three functions, which we call f1, f2, and f3. Using the Symbolic functions in MATLAB we have



```
 >> syms s
 >> f1=-1/(s+1)
f1 =
-1/(s + 1)
 >> f2=7/(s+2)
f2 =
7/(s + 2)
 >> f3=-6/(s+3)
f3 =
-6/(s + 3)
 >> g=ilaplace(f1)+ilaplace(f2)+ilaplace(f3)
g =
7*exp(-2*t) - exp(-t) - 6*exp(-3*t)
```

Note *g* is the inverse Laplace transform of *G*(*s*) in Eq. (3-54), as shown in Eq. (3-55). Alternatively you may also directly find the inverse Laplace transform of Eq. (3-49).



```
 >> f4=(5*s+3)/((s+1)*(s+2)*(s+3))
f4 =
(5*s + 3)/((s + 1)*(s + 2)*(s + 3))
 >> g=ilaplace(f4)
g =
7*exp(-2*t) - exp(-t) - 6*exp(-3*t)
```

G(*s*) Has Multiple-Order Poles

If *r* of the *n* poles of *G*(*s*) are identical—that is the pole at $s = -s_i$ is of multiplicity *r* – *G*(*s*) is written as

$$G(s) = \frac{Q(s)}{P(s)} = \frac{Q(s)}{(s+s_1)(s+s_2)\cdots(s+s_{n-r})(s+s_i)^r} \qquad (3\text{-}56)$$

($i \neq 1,2,...,n-r$), then *G*(*s*) can be expanded as

$$G(s) = \frac{K_{s1}}{s+s_1} + \frac{K_{s2}}{s+s_2} + \cdots + \frac{K_{s(n-r)}}{s+s_{n-r}}$$

$$|\leftarrow n-r \text{ terms of simple poles} \rightarrow|$$

$$+ \frac{A_1}{s+s_i} + \frac{A_2}{(s+s_i)^2} + \cdots + \frac{A_r}{(s+s_i)^r} \tag{3-57}$$

$$|\leftarrow r \text{ terms of repeated poles} \rightarrow|$$

Then $(n - r)$ coefficients, $K_{s1}, K_{s2}, \ldots, K_{s(n-r)}$, which correspond to simple poles, may be evaluated by the method described in Eq. (3-47). The determination of the coefficients that correspond to the multiple-order poles is described as follows:

$$A_r = \left[(s+s_i)^r G(s)\right]\Big|_{s=-s_i} \tag{3-58}$$

$$A_{r-1} = \frac{d}{ds}\left[(s+s_i)^r G(s)\right]\Big|_{s=-s_i} \tag{3-59}$$

$$A_{r-2} = \frac{1}{2!}\frac{d^2}{ds^2}\left[(s+s_i)^r G(s)\right]\Big|_{s=-s_i} \tag{3-60}$$

$$\vdots$$

$$A_1 = \frac{1}{(r-1)!}\frac{d^{r-1}}{ds^{r-1}}\left[(s+s_i)^r G(s)\right]\Big|_{s=-s_i} \tag{3-61}$$

EXAMPLE 3-3-2 Consider the function

$$G(s) = \frac{1}{s(s+1)^3(s+2)} = \frac{1}{s^5+5s^4+9s^3+7s^2+2s} \tag{3-62}$$

By using the format of Eq. (3-57), $G(s)$ is written as

$$G(s) = \frac{K_0}{s} + \frac{K_{-2}}{s+2} + \frac{A_1}{s+1} + \frac{A_2}{(s+1)^2} + \frac{A_3}{(s+1)^3} \tag{3-63}$$

The coefficients corresponding to the simple poles are

$$K_0 = [sG(s)]\Big|_{s=0} = \frac{1}{2} \tag{3-64}$$

$$K_{-2} = [(s+2)G(s)]\Big|_{s=-2} = \frac{1}{2} \tag{3-65}$$

and those of the third-order pole are

$$A_3 = [(s+1)^3 G(s)]\Big|_{s=-1} = -1 \tag{3-66}$$

$$A_2 = \frac{d}{ds}[(s+1)^3 G(s)]\Bigg|_{s=-1} = \frac{d}{ds}\left[\frac{1}{s(s+2)}\right]\Bigg|_{s=-1} = 0 \tag{3-67}$$

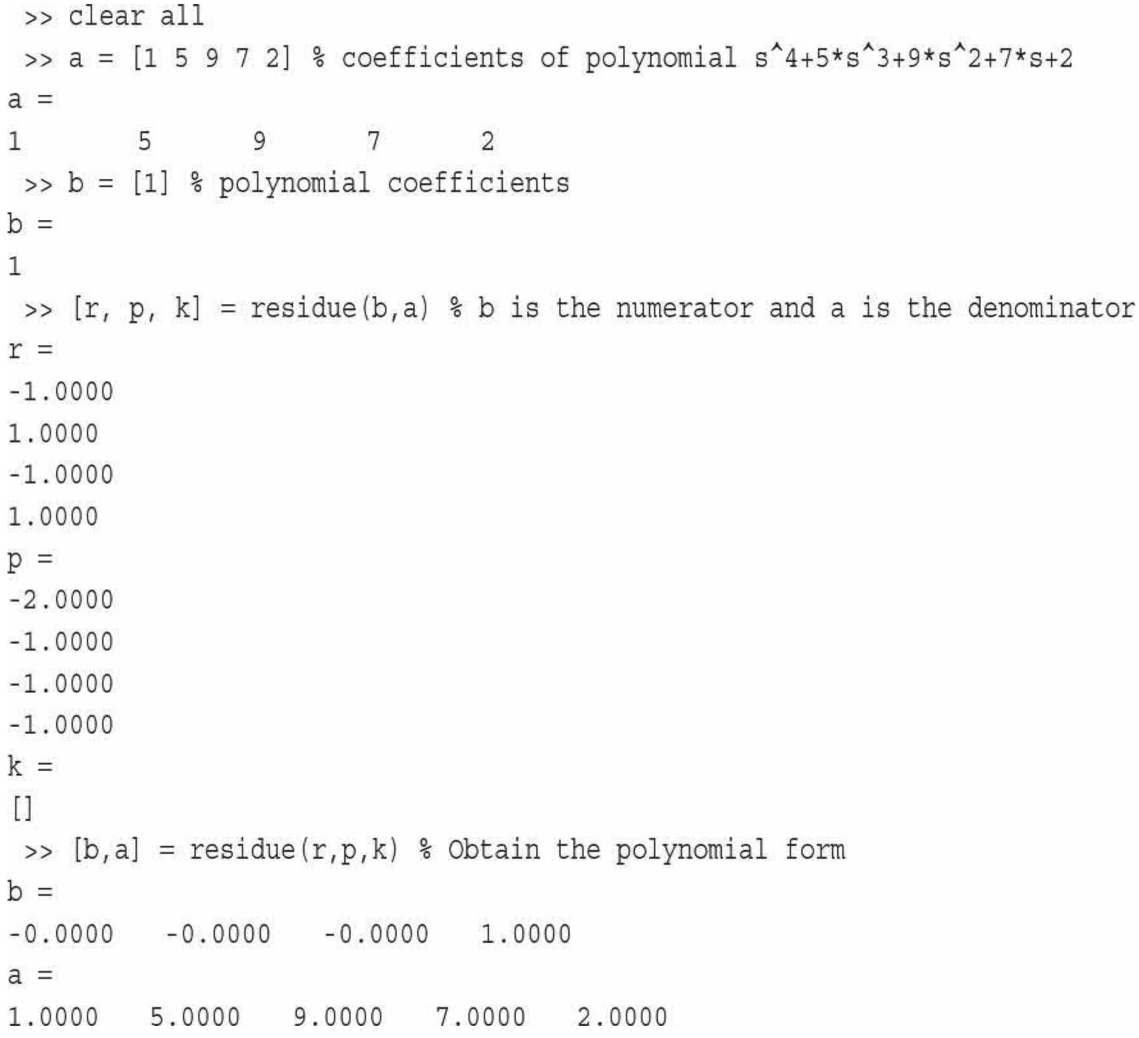
$$A_1 = \frac{1d^2}{2!ds^2}[(s+1)^3 G(s)]\Bigg|_{s=-1} = \frac{1d^2}{2ds^2}\left[\frac{1}{s(s+2)}\right]\Bigg|_{s=-1} = -1 \tag{3-68}$$

The completed partial-fraction expansion is

$$G(s) = \frac{1}{2s} + \frac{1}{2(s+2)} - \frac{1}{s+1} - \frac{1}{(s+1)^3} \tag{3-69}$$

Toolbox 3-3-3

For Example 3-3-2, Eq. (3-62) is a ratio of two polynomials.



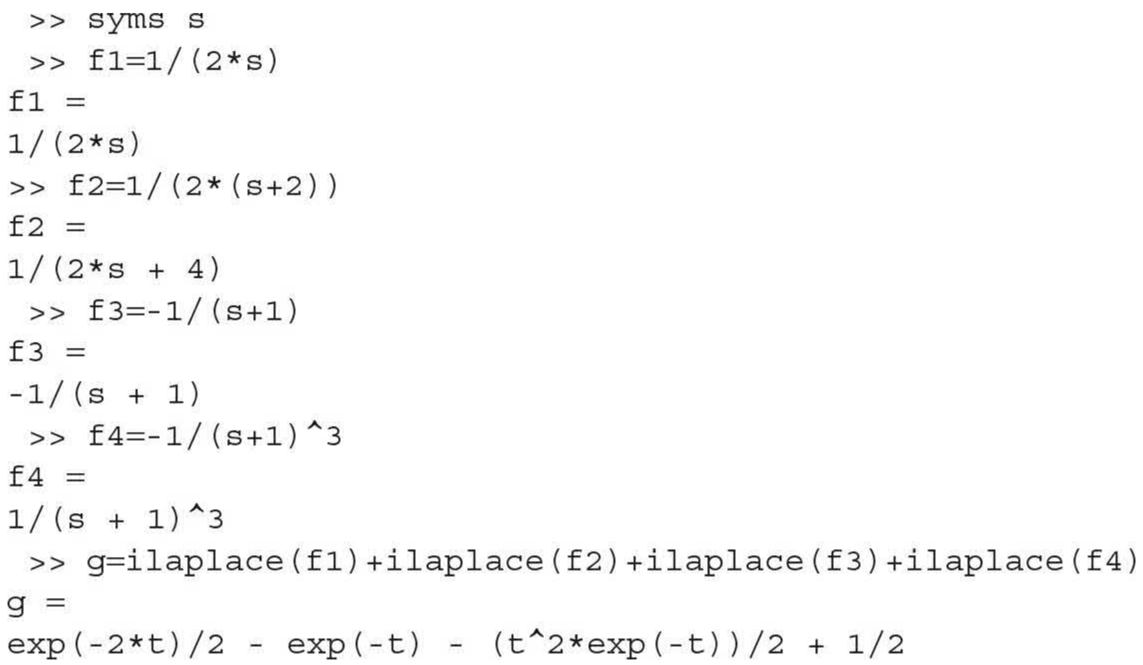
```
 >> clear all
 >> a = [1 5 9 7 2] % coefficients of polynomial s^4+5*s^3+9*s^2+7*s+2
a =
1       5       9       7       2
 >> b = [1] % polynomial coefficients
b =
1
 >> [r, p, k] = residue(b,a) % b is the numerator and a is the denominator
r =
-1.0000
1.0000
-1.0000
1.0000
p =
-2.0000
-1.0000
-1.0000
-1.0000
k =
[]
 >> [b,a] = residue(r,p,k) % Obtain the polynomial form
b =
-0.0000   -0.0000   -0.0000   1.0000
a =
1.0000   5.0000   9.0000   7.0000   2.0000
```

Taking the **inverse Laplace transform** on both sides using the Laplace table in App. C or using the following MATLAB toolbox, we get

$$g(t)=\frac{1}{2}-e^{-t}+\frac{e^{-2t}}{2}+\frac{t^2e^{-t}}{2}t\geq 0 \tag{3-70}$$

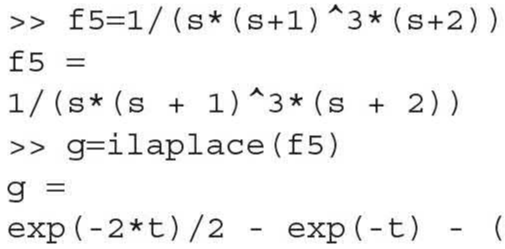
Toolbox 3-3-4

For Example 3-3-2, Eq. (3-69) is composed of four functions, which we call f1, f2, f3, and f4. Using the Symbolic functions in MATLAB, we have



```
 >> syms s
 >> f1=1/(2*s)
f1 =
1/(2*s)
>> f2=1/(2*(s+2))
f2 =
1/(2*s + 4)
 >> f3=-1/(s+1)
f3 =
-1/(s + 1)
 >> f4=-1/(s+1)^3
f4 =
1/(s + 1)^3
 >> g=ilaplace(f1)+ilaplace(f2)+ilaplace(f3)+ilaplace(f4)
g =
exp(-2*t)/2 - exp(-t) - (t^2*exp(-t))/2 + 1/2
```

Note *g* is the inverse Laplace transform of *G*(*s*) in Eq. (3-69), as shown in Eq. (3-70). Alternatively you may also directly find the inverse Laplace transform of Eq. (3-63).



```
>> f5=1/(s*(s+1)^3*(s+2))
f5 =
1/(s*(s + 1)^3*(s + 2))
>> g=ilaplace(f5)
g =
exp(-2*t)/2 - exp(-t) - (t^2*exp(-t))/2 + 1/2
```

G(*s*) Has Simple Complex-Conjugate Poles

The partial-fraction expansion of Eq. (3-42) is valid also for simple complex-conjugate poles. As discussed in Sec. 3-2-8, complex-conjugate poles are of special interest in control system studies and as a result require a more special attention.

Suppose that *G*(*s*) of Eq. (3-42) contains a pair of complex poles $s = -\sigma + j\omega$ and $s = -\sigma - j\omega$. The corresponding coefficients of these poles are found by using Eq. (3-45),

$$K_{-\sigma+j\omega} = (s+\sigma-j\omega)G(s)\Big|_{s=-\sigma+j\omega} \tag{3-71}$$

$$K_{-\sigma+j\omega} = (s+\sigma-j\omega)G(s)\Big|_{s=-\sigma+j\omega} \tag{3-71}$$

The procedure for finding the coefficients in Eqs. (3-71) and (3-72) is illustrated through the following example.

EXAMPLE 3-3-3 Consider the **second-order prototype** function

$$G(s) = \frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2} \tag{3-73}$$

Let us assume that the value of ζ is less than 1, so that the poles of $G(s)$ are complex. Then, $G(s)$ is expanded as follows:

$$G(s) = \frac{K_{-\sigma+j\omega}}{s+\sigma-j\omega} + \frac{K_{-\sigma-j\omega}}{s+\sigma+j\omega} \tag{3-74}$$

where

$$\begin{aligned} \sigma &= \zeta\omega_n \\ \omega &= \omega_n\sqrt{1-\zeta^2} \end{aligned} \tag{3-75}$$

The coefficients in Eq. (3-73) are determined as

$$K_{-\sigma+j\omega} = (s+\sigma-j\omega)G(s)\Big|_{s=-\sigma+j\omega} = \frac{\omega_n^2}{2j\omega} \tag{3-76}$$

$$K_{-\sigma-j\omega} = (s+\sigma+j\omega)G(s)\Big|_{s=-\sigma-j\omega} = -\frac{\omega_n^2}{2j\omega} \tag{3-77}$$

The complete partial-fraction expansion of Eq. (3-73) is

$$G(s) = \frac{\omega_n^2}{2j\omega}\left[\frac{1}{s+\sigma-j\omega} - \frac{1}{s+\sigma+j\omega}\right] \tag{3-78}$$

Taking the **inverse Laplace transform** on both sides of the last equation gives

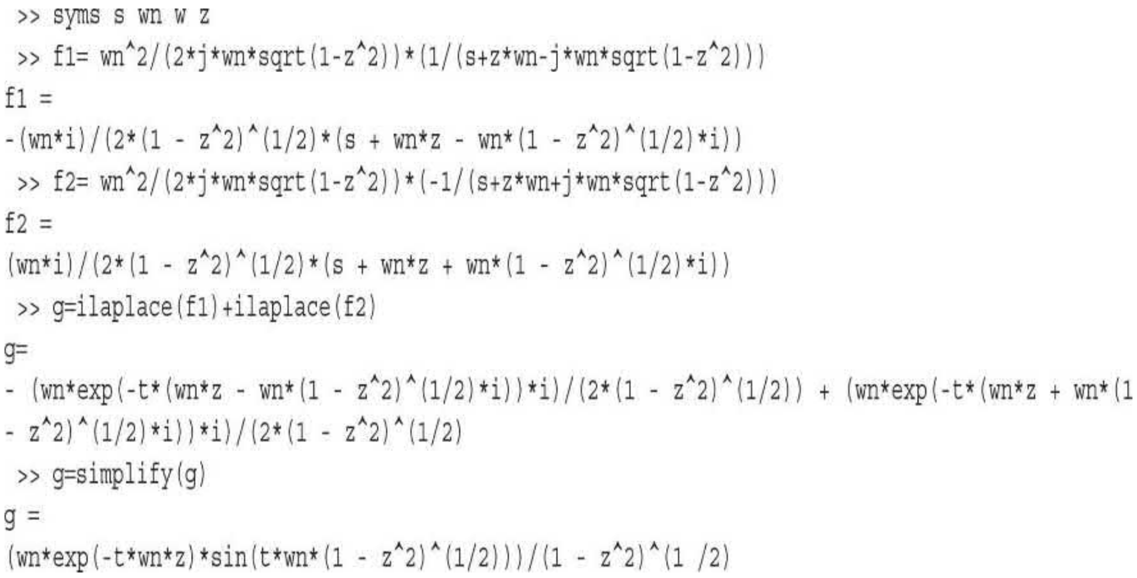
$$g(t)=\frac{\omega_n^2}{2j\omega}e^{-\sigma t}(e^{j\omega t}-e^{-j\omega t})\quad t\geq 0 \tag{3-79}$$

or

$$g(t)=\frac{\omega_n}{\sqrt{1-\zeta^2}}e^{-\zeta\omega_n t}\sin\left(\omega_n\sqrt{1-\zeta^2}t\right)\quad t\geq 0 \tag{3-80}$$

Toolbox 3-3-5

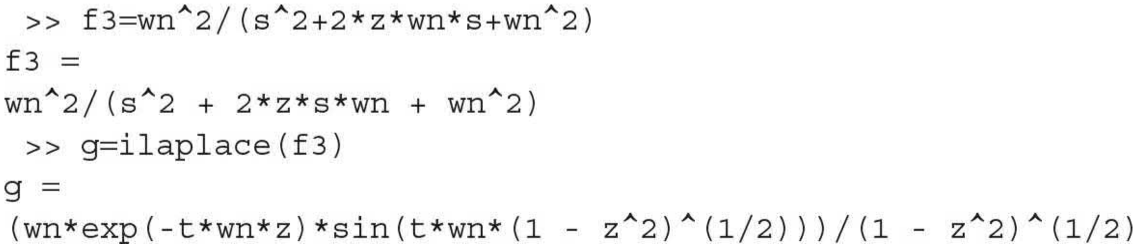
For Example 3-3-3, Eq. (3-73) is composed of two functions, which we call f1 and f2. Using the Symbolic functions in MATLAB, we have



```
 >> syms s wn w z
 >> f1= wn^2/(2*j*wn*sqrt(1-z^2))*(1/(s+z*wn-j*wn*sqrt(1-z^2)))
f1 =
-(wn*i)/(2*(1 - z^2)^(1/2)*(s + wn*z - wn*(1 - z^2)^(1/2)*i))
 >> f2= wn^2/(2*j*wn*sqrt(1-z^2))*(-1/(s+z*wn+j*wn*sqrt(1-z^2)))
f2 =
(wn*i)/(2*(1 - z^2)^(1/2)*(s + wn*z + wn*(1 - z^2)^(1/2)*i))
 >> g=ilaplace(f1)+ilaplace(f2)
g=
- (wn*exp(-t*(wn*z - wn*(1 - z^2)^(1/2)*i))*i)/(2*(1 - z^2)^(1/2)) + (wn*exp(-t*(wn*z + wn*(1
- z^2)^(1/2)*i))*i)/(2*(1 - z^2)^(1/2)
 >> g=simplify(g)
g =
(wn*exp(-t*wn*z)*sin(t*wn*(1 - z^2)^(1/2)))/(1 - z^2)^(1 /2)
```

Note g is the inverse Laplace transform of $G(s)$ in Eq. (3-78), as shown in Eq. (3-80). We have used the symbolic simplify command to convert g to trigonometric format. Note also that in MATLAB (i) and (j) both represent SQRT(-1).

Alternatively you may also directly find the inverse Laplace transform of Eq. (3-73).



```
 >> f3=wn^2/(s^2+2*z*wn*s+wn^2)
f3 =
wn^2/(s^2 + 2*z*s*wn + wn^2)
 >> g=ilaplace(f3)
g =
(wn*exp(-t*wn*z)*sin(t*wn*(1 - z^2)^(1/2)))/(1 - z^2)^(1/2)
```

3-4 APPLICATION OF THE LAPLACE TRANSFORM TO THE SOLUTION OF LINEAR ORDINARY DIFFERENTIAL EQUATIONS

As we saw in Chap. 2, mathematical models of most components of control systems are represented by first- or second-order differential equations. In this textbook, we primarily study **linear ordinary differential equations** with constant coefficients such as the first-order linear system:

$$\frac{dy(t)}{dt} + a_0 y(t) = f(t) \tag{3-81}$$

or the second-order linear system:

$$\frac{d^2 y(t)}{dt^2} + a_1 \frac{dy(t)}{dt} + a_0 y(t) = f(t) \tag{3-82}$$

Linear ordinary differential equations can be solved by the Laplace transform method with the aid of the theorems on Laplace transform given in Sec. 3-2, the partial-fraction expansion, and the table of Laplace transforms. The procedure is outlined as follows:

1. Transform the differential equation to the *s*-domain by Laplace transform using the Laplace transform table.
2. Manipulate the transformed algebraic equation and obtain the output variable.
3. Perform partial-fraction expansion to the transformed algebraic equation.
4. Obtain the inverse Laplace transform from the Laplace transform

table.

Let us examine two specific cases, first- and second-order prototype systems. The prototype forms of differential equations provide a common format of representing various components of a control system. The significance of this representation became evident in Chap. 2 and becomes more evident when we study the **time response** of control systems in Chap. 7.

3-4-1 First-Order Prototype System

In Chap. 2, we demonstrated that fluid, electrical, thermal, and mechanical systems are modeled by differential equations. Figure 3-4 shows four cases of mechanical, electrical, fluid, and thermal systems that are modeled by first-order differential equations, which may ultimately be represented by the **first-order prototype** of the form

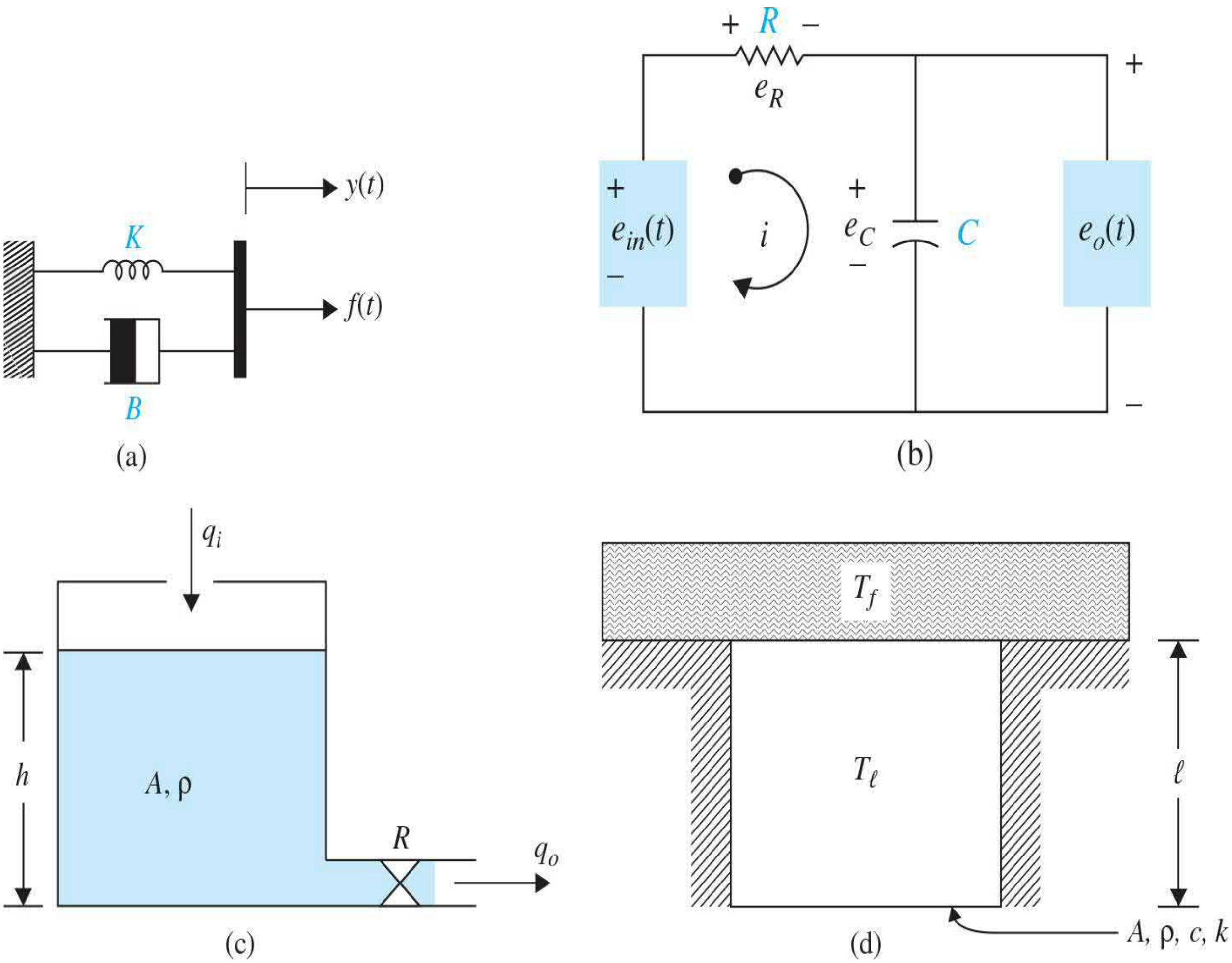


Figure 3-4 (*a*) A spring-dashpot mechanism. (*b*) A series *RC* network. (*c*) A one-tank liquid level system. (*d*) A heat transfer problem.

The **prototype** forms of differential equations provide a common format of representing various components of a control system.

$$\frac{dy(t)}{dt}+\frac{1}{\tau}y(t)=\frac{1}{\tau}u(t) \tag{3-83}$$

where $\boldsymbol{\tau}$ is known as the **time constant** of the system, which is a measure of how fast the system responds to initial conditions of external excitations. Note that the input in Eq. (3-83) is scaled by $1/\boldsymbol{\tau}$ for cosmetic reasons.

In the spring-damper (no mass) system in Fig. 3-4*a*

$$\dot{y}(t)+\frac{K}{B}y(t)=\frac{Ku(t)}{B} \tag{3-84}$$

where $Ku(t) = f(t)$ is the applied force to the system, and $\tau=\frac{B}{K}$ is the time constant, which is the ratio of damping constant B and spring stiffness K. In this case, displacements $y(t)$ and $u(t)$ are the output and input variables, respectively.

In the *RC* circuit in Fig. 3-4*b*, the output voltage $e_o(t)$ satisfies the following differential equation:

$$\dot{e}_o(t)+\frac{1}{RC}e_o(t)=\frac{1}{RC}e_{in}(t) \tag{3-85}$$

where $e_{in}(t)$ is the input voltage and $\boldsymbol{\tau} = RC$ is the time constant.

In the one-tank liquid level system in Fig. 3-4*c*, the system equation is defined in terms of the output volumetric flow rate q_o as

$$\dot{q}_o+\frac{q_o}{RC}=\frac{q_i}{RC} \tag{3-86}$$

where q_i is the input flow rate, $RC = \boldsymbol{\tau}$ is the system time constant, R is fluid resistance, and C is the tank capacitance.

Finally, in the thermal system represented in Fig. 3-4*d*, C is thermal capacitance C, R is convective thermal resistance, $RC = \boldsymbol{\tau}$ is system time constant, T_ℓ is solid object temperature, and T_f is top fluid temperature. The equation of the system representing the heat transfer process is

$$\dot{T}_\ell + \frac{T_\ell}{RC} = \frac{T_f}{RC} \tag{3-87}$$

As evident from Eqs. (3-84) through (3-87), they are all represented by the first-order prototype system Eq. (3-83), and to understand their respective behaviors, we can solve Eq. (3-83) for a *test input*—in this case a unit step input:

$$u(t) = u_s(t) = \begin{cases} 0, & t < 0, \\ 1, & t \geq 0 \end{cases} \tag{3-88}$$

The unit step input is basically a constant input applied to the system, and by solving for the differential equation, we examine how the output responds to this input. Rewriting Eq. (3-83) as

$$u_s(t) = \tau \frac{dy(t)}{dt} + y(t) \tag{3-89}$$

If $y(0) = \frac{dy(0)}{dt} = 0, \mathcal{L}(u_s(t)) = \frac{1}{s}$ and $L(y(t)) = Y(s)$, we have

$$\frac{1}{s} = s\tau Y(s) + Y(s) \tag{3-90}$$

Or, as a result, the output in s-domain is

$$Y(s) = \frac{1}{s}\frac{1}{\tau s + 1} \tag{3-91}$$

Notice that the system has a pole at zero due to the input and one at $s = -1/\boldsymbol{\tau}$ as shown in Fig. 3-5. For a positive $\boldsymbol{\tau}$, the pole in the left-half s-plane.

Using partial fractions, Eq. (3-91) becomes

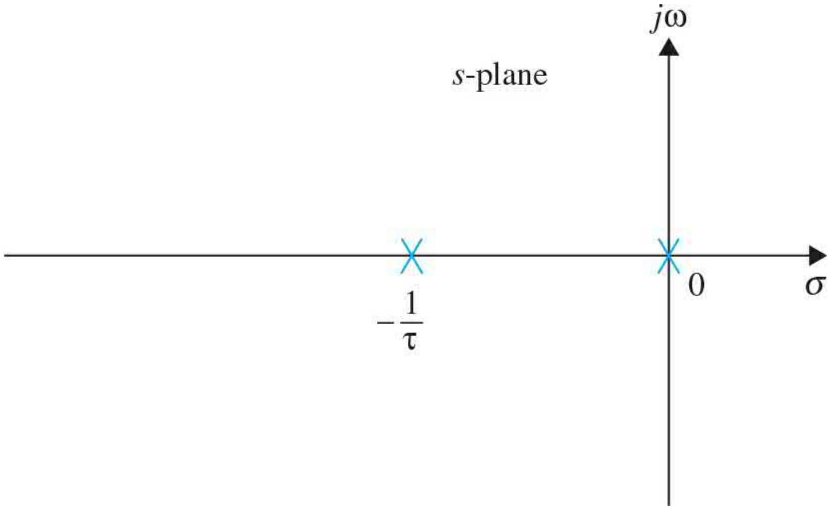


Figure 3-5 Pole configuration of the transfer function of a prototype first-order system.

$$Y(t) = \frac{K_0}{s} + \frac{K_{-1/\tau}}{\tau s + 1} \tag{3-92}$$

where $K_0 = 1$ and $K_{-1/\tau} = -1$. Applying the inverse Laplace transform to Eq. (3-92), we get the time response of Eq. (3-83).

$$y(t) = 1 - e^{-t/\tau} \tag{3-93}$$

where *t* is the time for *y*(*t*) to reach 63 percent of its **final value** of $\lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = 1$.

Typical unit-step response of *y*(*t*) is shown in Fig. 3-5 for a general value of *t*. As the value of time constant τ decreases, the system response approaches faster to the final value.

Toolbox 3-4-1

The inverse Laplace transform for Eq. (3-91) is obtained using the MATLAB Symbolic Toolbox by the following sequence of MATLAB functions.

```
>> syms s tau;
>> ilaplace(1/(tau*s^2+s))
ans =
 1 - exp(-t/tau)
```

The result is Eq. (3-93).

Note, the sym command lets you construct symbolic variables and expressions, and the command

```
>> syms s tau;
```

is equivalent to

```
>> s=sym(‘s’);
>> tau=sym(‘tau’);
```

Time response of Eq. (3-83), shown in Fig. 3-6, for a given value τ = 0.1 s is obtained using

```
>> clear all;
>> t = 0:0.01:1;
>> tau = 0.1;
>> plot(1-exp(-t/tau));
```

You may wish to confirm that at time t = 0.1 s, y(t) = 0.63.

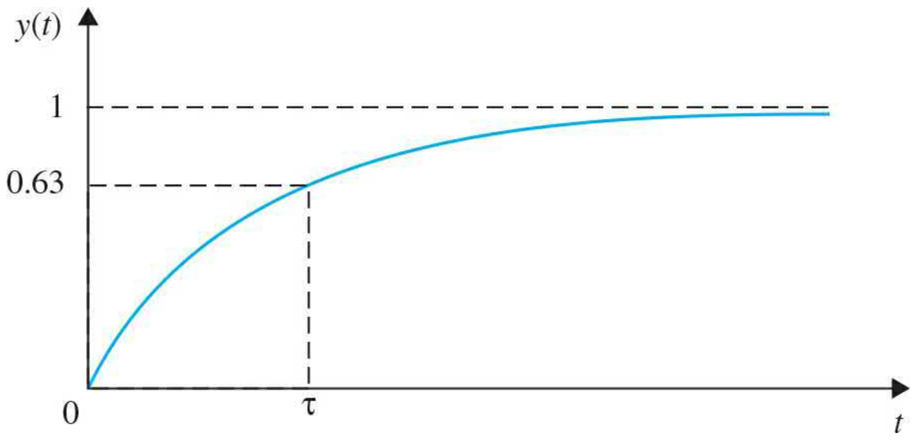


Figure 3-6 Unit-step response of a first-order prototype *differential equation.*

3-4-2 Second-Order Prototype System

Similar to the previous section, various mechanical, electrical, and fluid systems discussed in Chap. 2 may be modeled as a **second-order prototype** system—see, for example, the systems shown in Fig. 3-1, represented by Eqs. (3-1) and (3-3). The standard second-order prototype system has the form:

> Damping ratio ζ plays an important role in the time response of the prototype second-order system.

$$\frac{d^2y(t)}{dt^2}+2\zeta\omega_n\frac{dy(t)}{dt}+\omega_n^2y(t)=\omega_n^2u(t) \tag{3-94}$$

where ζ is known as the damping ratio, ω_n is the natural frequency of the system, $y(t)$ is the output variable, and $u(t)$ is the input. As in Sec. 3-4-1, we can solve Eq. (3-94) for a *test input*—in this case a unit step input:

$$u(t)=u_s(t)=\begin{cases}0, & t<0,\\ 1, & t\geq 0\end{cases} \tag{3-95}$$

If $y(0)=\frac{dy(0)}{dt}=0,\ \mathcal{L}(u_s(t))=U(s)=\frac{1}{s}$, and $L(y(t)) = Y(s)$, the output relation in the s-domain is

$$Y(s)=\frac{1}{s}\frac{\omega_n^2}{s^2+2\zeta\omega_ns+\omega_n^2} \tag{3-96}$$

where the transfer function of the system is

$$G(s)=\frac{Y(s)}{U(s)}=\frac{\omega_n^2}{s^2+2\zeta\omega_ns+\omega_n^2} \tag{3-97}$$

The characteristic equation of the prototype second-order system is obtained by setting the denominator of Eq. (3-97) to zero:

$$\Delta(s)=s^2+2\zeta\omega_ns+\omega_n^2=0 \tag{3-98}$$

The two poles of the system are the roots of the characteristic equation, expressed as

$$s_1, s_2 = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1} \tag{3-99}$$

From the poles of the system shown in Eq. (3-99), it is clear that the solution to Eq. (3-96) has a direct correlation to the value of the damping ratio ζ. The damping ratio determines if the poles in Eq. (3-99) are real or complex. In order to get a clear picture of the time behavior of the system, we first find the inverse Laplace transform of Eq. (3-96) for three important cases $\zeta < 1$, $\zeta > 1$.

System Is Critically Damped $\zeta = 1$

When the two roots of the characteristic equation are real and equal, we call the system **critically damped**. From Eq. (3-99), we see that critical damping occurs when $\zeta = 1$. In this case, the output relation in the *s*-domain, represented by Eq. (3-96), is rewritten as

$$Y(s) = \frac{1}{s}\frac{\omega_n^2}{s^2 + 2\omega_n s + \omega_n^2} = \frac{1}{s}\frac{\omega_n^2}{(s+\omega_n)^2} \tag{3-100}$$

Further, the transfer function in Eq. (3-98) becomes

$$G(s) = \frac{\omega_n^2}{(s+\omega_n)^2} \tag{3-101}$$

where $G(s)$ has two repeated poles at $s = -\omega_n$, as shown in Fig. 3-7. In order to find the solution of the differential equation, in this case, we obtain the partial fraction representation of Eq. (3-100) following the process defined in Example 3-3-2. Hence, by using the format of Eq. (3-57), $Y(s)$ is written as

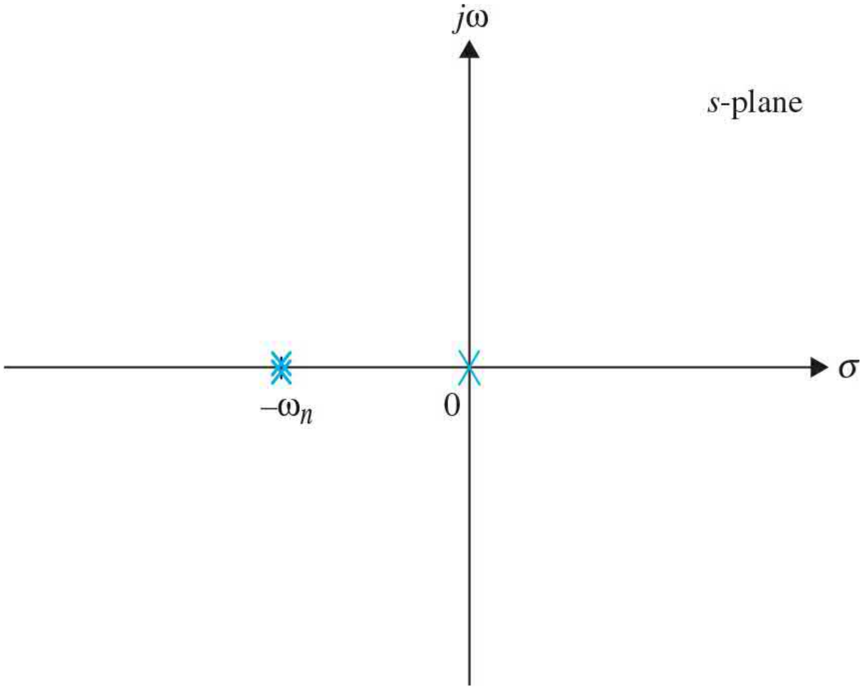


Figure 3-7 Poles of $Y(s)$ in a critically damped prototype first-order system with a unit step input.

$$Y(s) = \frac{K_0}{s} + \frac{A_1}{(s+\omega_n)} + \frac{A_2}{(s+\omega_n)^2} \tag{3-102}$$

where

$$K_0 = \left[(s)\frac{1}{s}\frac{\omega_n^2}{(s+\omega_n)^2}\right]\Bigg|_{s=0} = 1 \tag{3-103}$$

$$A_2 = \left[(s+\omega_n)^2\frac{1}{s}\frac{\omega_n^2}{(s+\omega_n)^2}\right]\Bigg|_{s=-\omega_n} = -1 \tag{3-104}$$

$$A_1 = \frac{d}{ds}\left[(s+\omega_n)^2\frac{1}{s}\frac{\omega_n^2}{(s+\omega_n)^2}\right]\Bigg|_{s=-\omega_n} = -1 \tag{3-105}$$

The completed partial-fraction expansion is

$$Y(s) = \frac{1}{s} - \frac{1}{(s+\omega_n)} - \frac{1}{(s+\omega_n)^2} \qquad (3\text{-}106)$$

Taking the **inverse Laplace transform** on both sides using the Laplace table in App. C or using the following MATLAB toolbox, we get

$$y(t) = 1 - e^{-\omega_n t} - te^{-\omega_n t} \quad t \geq 0 \qquad (3\text{-}107)$$

Toolbox 3-4-2

Equation (3-106) is composed of three functions, which we call f1, f2, and f3. Using the Symbolic functions in MATLAB, we have

```
>> syms s wn
>> f1= 1/s
f1 =
1/s
>> f2=-1/(s+wn)
f2 =
-1/(s + wn)
>> f3=-1/(s+wn)^2
f3 =
-1/ (s + wn)^2
 >> y=ilaplace(f1)+ilaplace(f2)+ilaplace(f3)
y =
1 - t*exp(-t*wn) - exp(-t*wn)
```

Alternatively, we can directly find the inverse Laplace transform of Eq. (3-100).

```
>> syms s wn
>> f1= 1/s
f1 =
1/s
 >> f2= wn^2/(s+wn)^2
f2 =
wn^2/(s + wn)^2
 >> y=ilaplace(f1*f2)
y =
1 - t*exp(-t*wn) - exp(-t*wn)
```

Note y is the inverse Laplace transform of Y(s) in Eq. (3-100), as shown in Eq. (3-107).

System Is Overdamped ζ > 1

When the two roots of the characteristic equation are distinct and real, we call the system **overdamped**. From Eq. (3-99), we see that an overdamped scenario occurs when $\zeta > 1$. In this case the output relation in the *s*-domain, represented by Eq. (3-96) is rewritten as

$$Y(s) = \frac{1}{s}\frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{3-108}$$

Further, the transfer function in Eq. (3-108) becomes

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{3-109}$$

where *G*(*s*) has two poles at

$$s_1, s_2 = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1} \tag{3-110}$$

Let’s define

$$\sigma = \zeta\omega_n \tag{3-111}$$

as the damping factor, and

$$\omega = \omega_n\sqrt{\zeta^2 - 1} \tag{3-112}$$

is, for the purpose of reference, loosely called the conditional (or damped) frequency of the system—note the system will not exhibit oscillations in the overdamped case, so usage of the term frequency is not an accurate term. We use the following numerical example for easier understanding of the approach.

EXAMPLE 3-4-1 Consider Eq. (3-108) with $\zeta = \frac{3\sqrt{2}}{4}$ and $\omega_n = \sqrt{2}$ rad/s:

$$Y(s) = \frac{1}{s}\frac{2}{s^2+3s+2} = \frac{1}{s}\frac{2}{(s+1)(s+2)} \tag{3-113}$$

The transfer function of the system $G(s)$ in Eq. (3-109) has two poles at $s_1 = 1$ and $s_2 = 2$, as shown in Fig. 3-8. In order to find the solution of the differential equation, in this case, we obtain the partial fraction representation of Eq. (3-113) following the process defined in Example 3-3-1. Hence, by using the format of Eq. (3-45), $Y(s)$ is written as

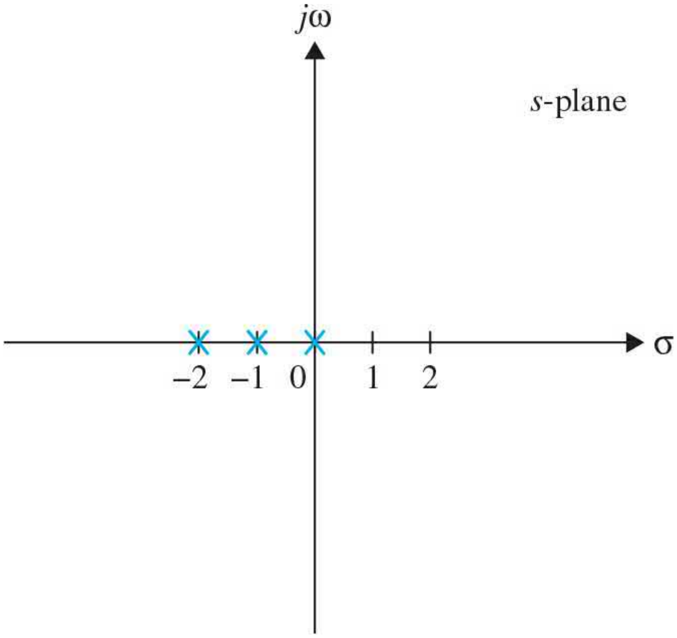


Figure 3-8 Poles of $Y(s)$ in an overdamped prototype first-order system with a unit-step input.

$$Y(s) = \frac{K_0}{s} + \frac{K_{-1}}{(s+1)} + \frac{K_{-2}}{(s+2)} \tag{3-114}$$

where

$$K_0 = \left[(s)\frac{1}{s}\frac{2}{(s+1)(s+2)} \right]\Bigg|_{s=0} = 1 \tag{3-115}$$

$$K_{-1} = \left[(s+1)\frac{1}{s}\frac{2}{(s+1)(s+2)} \right]\Bigg|_{s=-1} = -2 \tag{3-116}$$

$$K_{-2} = \left[(s+2)\frac{1}{s}\frac{2}{(s+1)(s+2)} \right]\Bigg|_{s=-2} = 1 \tag{3-117}$$

The completed partial-fraction expansion is

$$Y(s) = \frac{1}{s} - \frac{2}{(s+1)} + \frac{1}{(s+2)} \tag{3-118}$$

Taking the inverse Laplace transform on both sides using the Laplace table in App. C or using the following MATLAB toolbox, we get

$$y(t) = 1 - 2e^{-t} + e^{-2t} \quad t \geq 0 \tag{3-119}$$

Toolbox 3-4-3

Equation (3-118) is composed of three functions, which we call f1, f2, and f3. Using the Symbolic functions in MATLAB, we have

```
 >> syms s
 >> f1=1/s
f1 =
1/s
 >> f2=-2/(s+1)
f2 =
-2/(s + 1)
 >> f3=1/(s+2)
f3 =
1/(s + 2)
 >> y=ilaplace(f1)+ilaplace(f2)+ilaplace(f3)
y =
exp(-2*t) - 2*exp(-t) + 1
```

Alternatively we can directly find the inverse Laplace transform of Eq. (3-113),

```
 >> syms s
 >> y=ilaplace(2/(s*(s+1)*(s+2)))
y =
exp(-2*t) - 2*exp(-t) + 1
```

which is the same as Eq. (3-119).

EXAMPLE 3-4-2 Consider a **modified second-order prototype** of the form:

$$\frac{d^2y(t)}{dt^2}+2\zeta\omega_n\frac{dy(t)}{dt}+\omega_n^2y(t)=A\omega_n^2u(t) \tag{3-120}$$

where *A is a constant*. The transfer function of this system is

$$G(s)=\frac{A\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2} \tag{3-121}$$

Assigning the following values to the differential equation parameters, we get

$$\frac{d^2y(t)}{dt^2}+3\frac{dy(t)}{dt}+2y(t)=5\mu_s(t) \tag{3-122}$$

where $u_s(t)$ is the unit-step function. The initial conditions are $y(0) = -1$ and $\dot{y}(0)=\left.\frac{dy(t)}{dt}\right|_{t=0}=2.$ To solve the differential equation, we first take the Laplace transform on both sides of Eq. (3-122):

$$s^2Y(s)-sy(0)-\dot{y}(0)+3sY(s)-3y(0)+2Y(s)=5/s \tag{3-123}$$

Substituting the values of the initial conditions into the last equation and solving for *Y*(*s*), we get

$$Y(s)=\frac{-s^2-s+5}{s(s^2+3s+2)}=\frac{-s^2-s+5}{s(s+1)(s+2)} \qquad (3\text{-}124)$$

Equation (3-124) is expanded by partial-fraction expansion to give

$$Y(s)=\frac{5}{2s}-\frac{5}{s+1}+\frac{3}{2(s+2)} \qquad (3\text{-}125)$$

Taking the inverse Laplace transform of Eq. (3-125), we get the complete solution as

$$y(t)=\frac{5}{2}-5e^{-t}+\frac{3}{2}e^{-2t} \quad t\geq 0 \qquad (3\text{-}126)$$

The first term in Eq. (3-126) is the **steady-state** or the **particular solution**; the last two terms represent the **transient** or **homogeneous solution**. Unlike the classical method, which requires separate steps to give the **transient and the steady-state responses** or solutions, the Laplace transform method gives the entire solution in one operation.

If only the magnitude of the steady-state solution of *y*(*t*) is of interest, the final-value theorem of Eq. (3-36) may be applied. Thus,

The terms **transient and steady-state responses** are used to indicate the **homogeneous and particular solutions** of differential equations.

$$\lim_{t\to\infty} y(t)=\lim_{s\to 0} sY(s)=\lim_{s\to 0}\frac{-s^2-s+5}{s^2+3s+2}=\frac{5}{2} \qquad (3\text{-}127)$$

where, in order to ensure the validity of the final-value theorem, we have first checked and found that the poles of function $sY(s)$ are all in the left-half s-plane. ▲

From Example 3-4-2, it is important to highlight that in control systems, the terms **transient and steady-state responses** are used to indicate the **homogeneous and particular solutions** of differential equations. We study these topics in detail in Chap. 7.

System Is Underdamped ζ < 1

When the two roots of the characteristic equation are complex with equal negative real parts, we call the system **underdamped**. From Eq. (3-99), we see that underdamped scenario occurs when $0 < \zeta < 1$. In this case, the output relation in the s-domain, represented by Eq. (3-96), is written as

$$Y(s) = \frac{1}{s}\frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{3-128}$$

Further, the transfer function in Eq. (3-128) becomes

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{3-129}$$

where $G(s)$ has two complex-conjugate poles at

$$s_1, s_2 = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \tag{3-130}$$

where the j term was cosmetically introduced to reflect that the poles are complex-conjugate. Let’s define

$$\sigma = \zeta\omega_n \tag{3-131}$$

as the damping factor, and

$$\omega = \omega_n\sqrt{1-\zeta^2} \tag{3-132}$$

as the conditional (or damped) frequency of the system. Figure 3-9 illustrates the relationships among the location of the characteristic equation roots and σ, ζ, ω_n and ω. For the complex-conjugate roots shown,

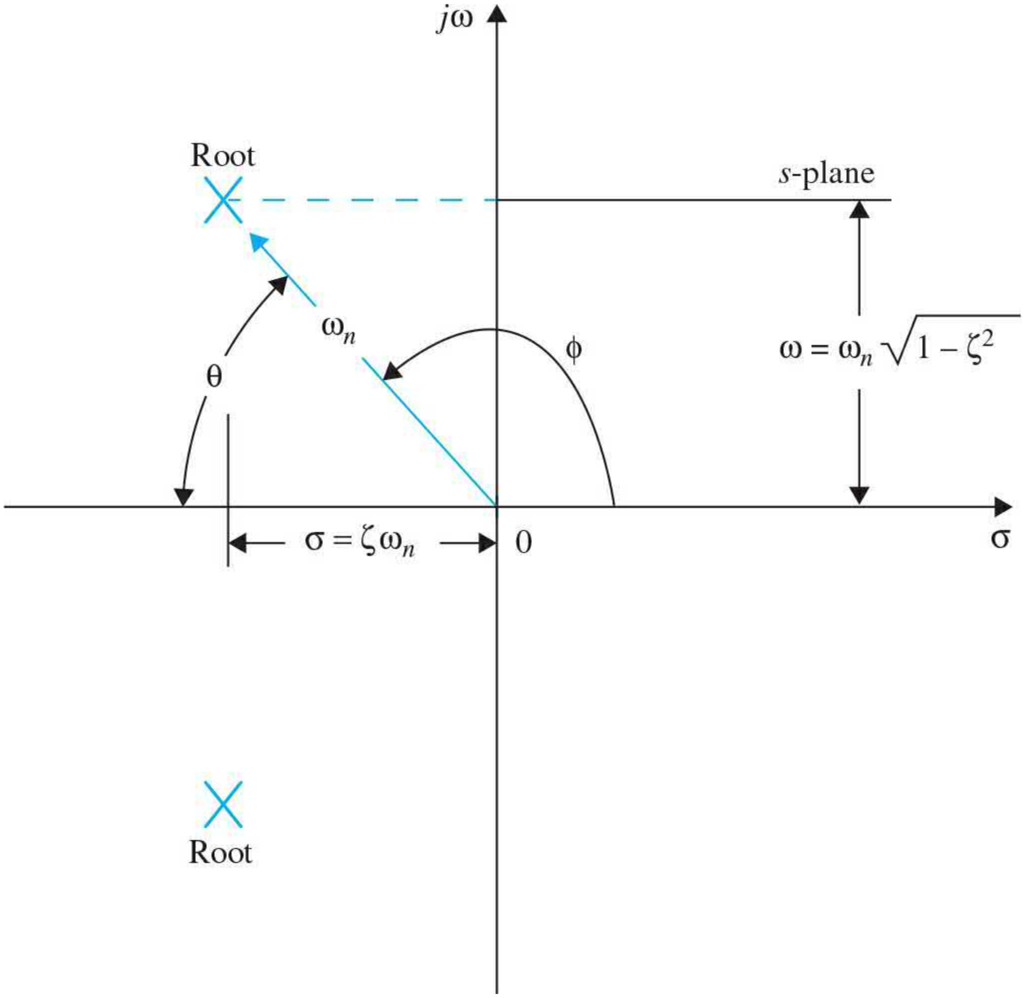


Figure 3-9 Relationships among the characteristic-equation roots of the prototype second-order system and σ, ζ, ω_n, and ω.

- ω_n is the radial distance from the roots to the origin of the *s*-plane, or $\omega_n = \sqrt{(\zeta\omega_n)^2 + \omega_n^2(1-\zeta^2)}$.
- σ is the real part of the roots.
- ω is the imaginary part of the roots.
- ζ is *the* cosine of the angle between the radial line to the roots and the negative axis when the roots are in the left-half *s*-plane, or $\zeta = \cos\theta$.

The partial-fraction expansion of Eq. (3-128) is written as

$$Y(s) = \frac{K_0}{s} + \frac{K_{-\sigma+j\omega}}{s+\sigma-j\omega} + \frac{K_{-\sigma-j\omega}}{s+\sigma+j\omega} \tag{3-133}$$

where

$$K_0 = sY(s)\big|_{s=0} = 1 \tag{3-134}$$

$$K_{-\sigma+j\omega} = (s+\sigma-j\omega)Y(s)\big|_{s=-\sigma+j\omega} = \frac{e^{-j\phi}}{2j\sqrt{1-\zeta^2}} \tag{3-135}$$

$$K_{-\sigma-j\omega} = (s+\sigma+j\omega)Y(s)\big|_{s=-\sigma-j\omega} = \frac{-e^{-j\phi}}{2j\sqrt{1-\zeta^2}} \tag{3-136}$$

The angle ϕ is given by

$$\phi = \pi - \cos^{-1}\zeta \tag{3-137}$$

and is illustrated in Fig. 3-9. The inverse Laplace transform of Eq. (3-128) is now written as

$$\begin{aligned} y(t) &= 1 + \frac{1}{2j\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t}\left[e^{j(\omega t-\phi)} - e^{-j(\omega t-\phi)}\right] \\ &= 1 + \frac{1}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t} \sin\left[\omega_n\sqrt{1-\zeta^2}\,t - \phi\right] \quad t \ge 0 \end{aligned} \tag{3-138}$$

where Euler's formula from Eq. (3-33) has been used to convert exponential terms inside the brackets in Eq. (3-138) to a sine function. Substituting Eq. (3-137) into Eq. (3-138) for ϕ, we have

$$y(t) = 1 - \frac{1}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t} \sin\left[\left(\omega_n\sqrt{1-\zeta^2}\right)t + \cos^{-1}\zeta\right] \quad t \ge 0 \tag{3-139}$$

EXAMPLE 3-4-3 Consider the linear differential equation

$$\frac{d^2y(t)}{dt^2} + 34.5\frac{dy(t)}{dt} + 1000y(t) = 1000u_s(t) \tag{3-140}$$

The initial values of $y(t)$ and $dy(t)/dt$ are zero. Taking the Laplace transform on both sides of Eq. (3-140), and solving for *Y*(*s*), we have

$$Y(s) = \frac{1000}{s(s^2 + 34.5s + 1000)} = \frac{\omega_n^2}{s\left(s^2 + 2\zeta\omega_n s + \omega_n^2\right)} \qquad (3\text{-}141)$$

where, using the second-order prototype representation, $\zeta = 0.5455$ and $\omega_n = 31.6228$. The inverse Laplace transform of Eq. (3-141) can be executed substituting these values into Eq. (3-139). Or

$$y(t) = 1 - 1.193e^{-17.25t}\sin(26.5t + 0.9938) \quad t \geq 0 \qquad (3\text{-}142)$$

where

$$\theta = \cos^{-1}\zeta = 0.9938\ \text{rad}\left(= 56.94^\circ \frac{\pi\ \text{rad}}{180^\circ}\right) \qquad (3\text{-}143)$$

$$\sigma = \zeta\omega_n = 17.25 \qquad (3\text{-}144)$$

$$\omega = \omega_n\sqrt{1 - \zeta^2} = 26.5 \qquad (3\text{-}145)$$

Notice the final value of *y*(*t*) = 1 in this case, which implies the output perfectly follows the input at steady state. See Fig. 3-10 for the time response plot obtained from the following MATLAB toolbox.

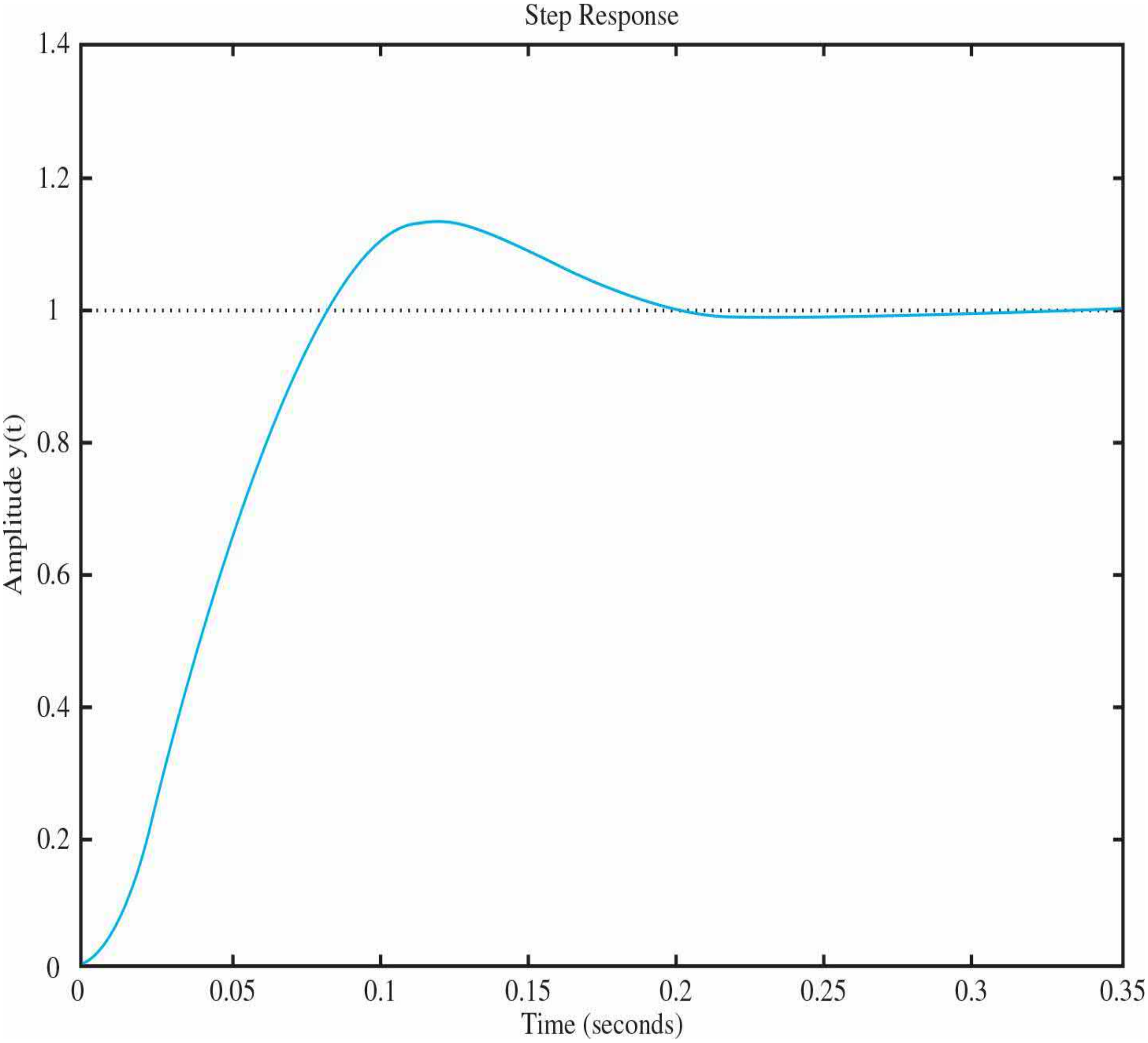
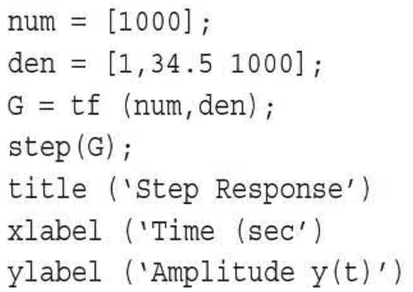


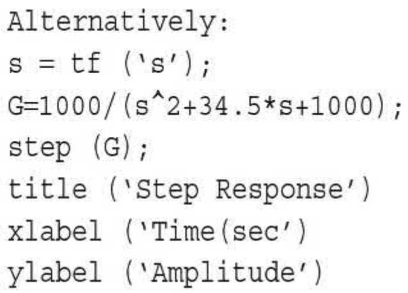
Figure 3-10 Time response $y(t)$ of the second-order system in Eq. (3-140) for a unit-step input.

Toolbox 3-4-4

Time response plot of Eq. (3-140) for a unit-step input may be obtained using



```
num = [1000];
den = [1,34.5 1000];
G = tf (num,den);
step(G);
title ('Step Response')
xlabel ('Time (sec')
ylabel ('Amplitude y(t)')
```



```
Alternatively:
s = tf ('s');
G=1000/(s^2+34.5*s+1000);
step (G);
title ('Step Response')
xlabel ('Time(sec')
ylabel ('Amplitude')
```

“step” produces the time response of a function for a unit-step input.

3-4-3 Second-Order Prototype System—Final Observations

The effects of the system parameters ζ and ω_n on the step response $y(t)$ of the prototype second-order system can be studied by referring to the roots of the characteristic equation in Eq. (3-89).

Using the next toolbox, we can plot the unit-step time responses of Eq. (3-96) for various positive values of ζ and for a fixed natural frequency $\omega_n = 10$ rad/s. As seen, the response becomes more oscillatory with larger **overshoot** as ζ decreases. When $\zeta \geq 1$, the step response does not exhibit any overshoot; that is, $y(t)$ never exceeds its final value.

Toolbox 3-4-5

The corresponding time responses for Fig. 3-11 are obtained by the following sequence of MATLAB functions:

```
clear all
wn=10;
for l=[0.2 0.4 0.6 0.8 1 1.2 1.4 1.6 1.8 2]
t=0:0.1:50;
num = [wn.^2];
den = [1 2*l*wn wn.^2];
t=0:0.01:2;
step(num,den,t)
hold on;
end
xlabel(‘Time(secs)’)
ylabel(‘Amplitude y(t)’)
```

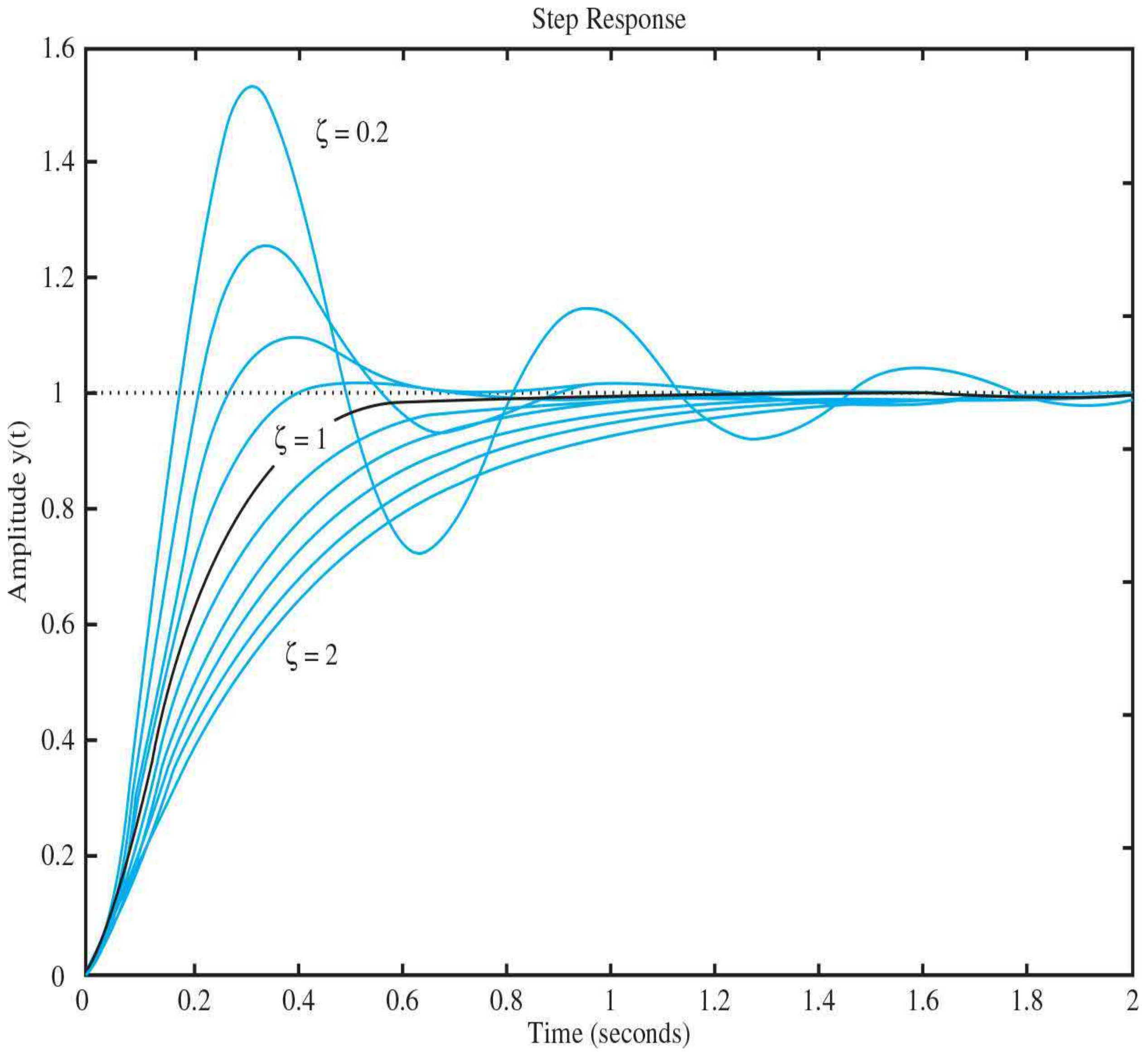


Figure 3-11 Unit-step responses of the prototype second-order system with various damping ratios.

The effect of damping of the second-order system on the characteristic equation roots—that is, poles of the transfer function in Eq. (3-97)—is further illustrated by Figs. 3-12 and 3-13. In Fig. 3-12, ω_n is held constant while the damping ratio ζ is varied from $-\infty$ to $+\infty$. Based on the values of ζ, the classification of the system dynamics appears in Table 3-2.

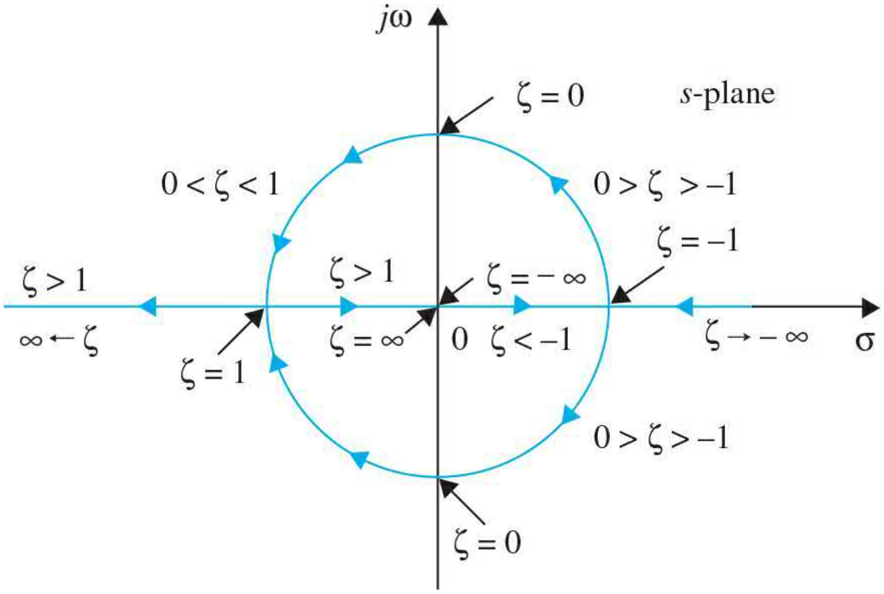


Figure 3-12 Locus of roots of the characteristic equation of the prototype second-order system.

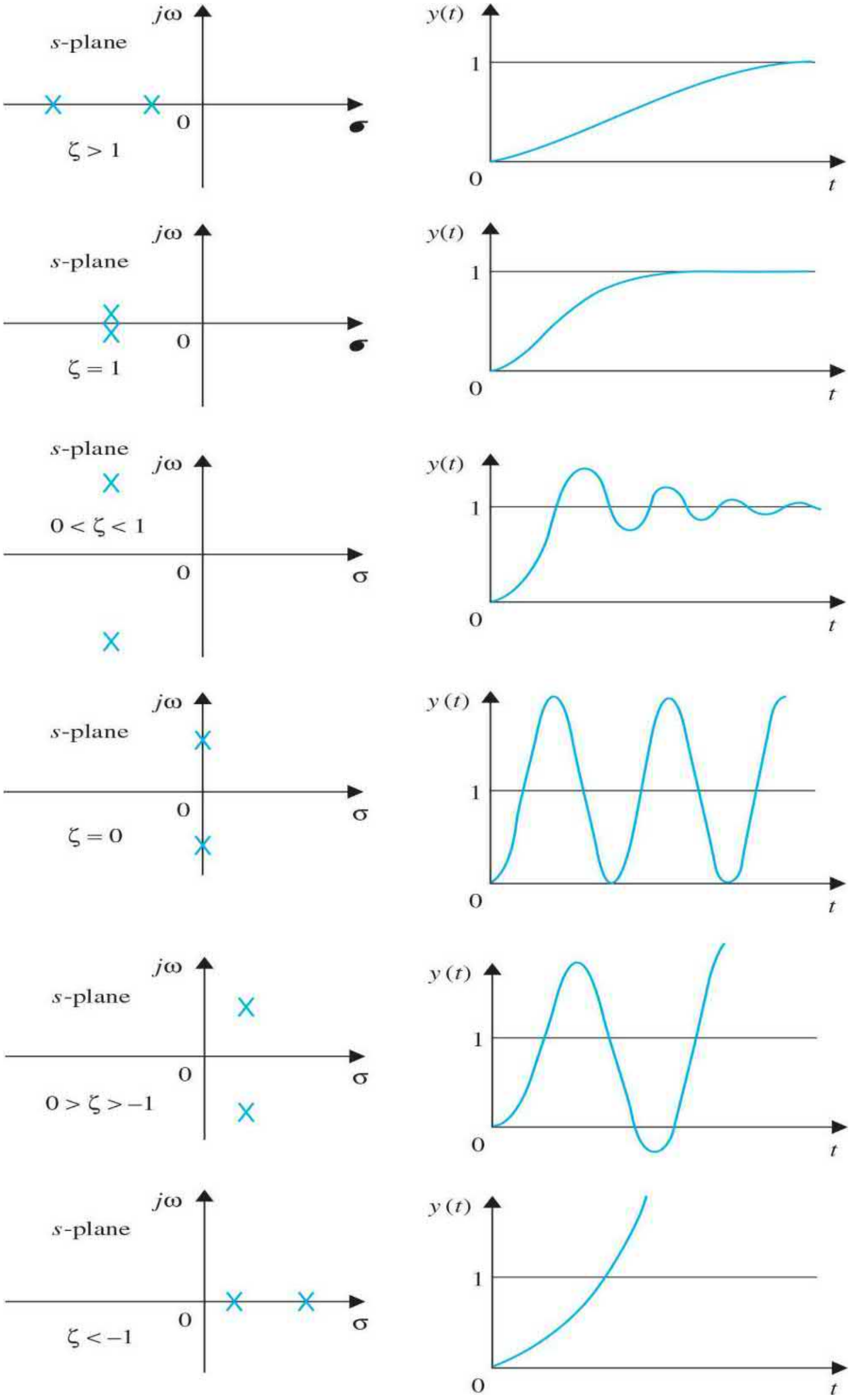
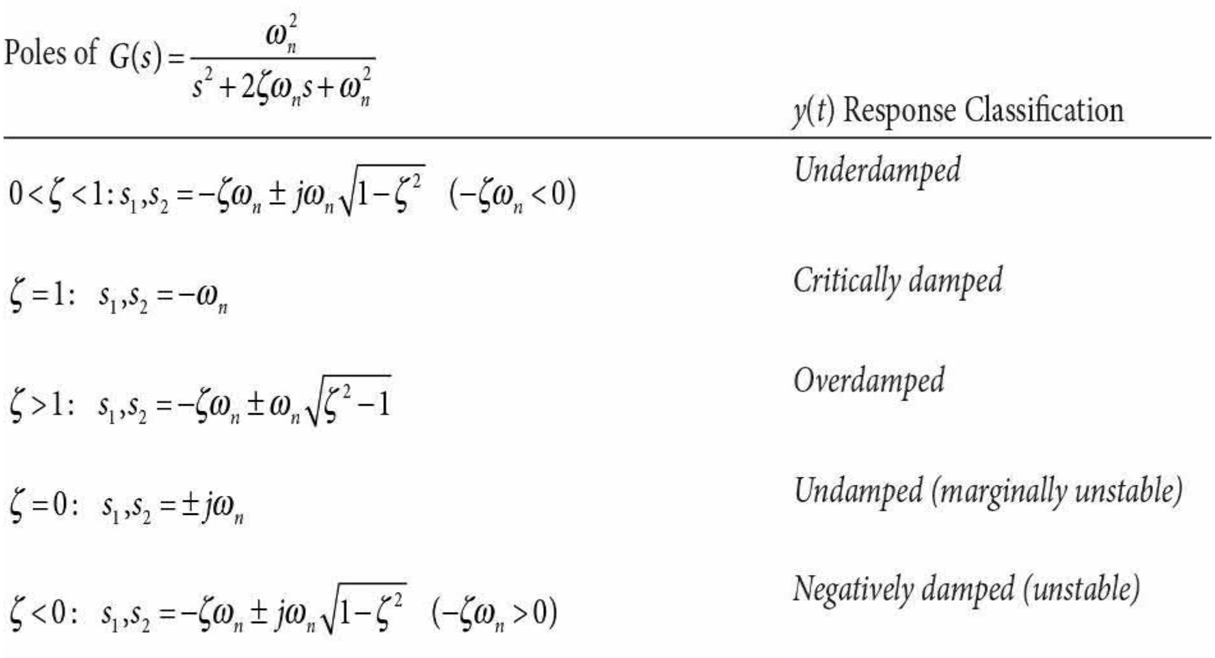

jω
s-plane
0
σ
ζ > 1
y(t)
1
0
t
jω
s-plane
0
σ
ζ = 1
y(t)
1
0
t
s-plane
jω
0 < ζ < 1
0
σ
y(t)
1
0
t
jω
s-plane
0
σ
ζ = 0
y (t)
1
0
t
jω
s-plane
0
σ
0 > ζ > −1
y (t)
1
0
t
jω
s-plane
0
σ
ζ < −1
y (t)
1
0
t

Figure 3-13 Step-response comparison for various characteristic-equation-root locations in the *s*-plane.

TABLE 3-2 Classification of the System Response Based on the Values of *ζ*



Poles of $G(s)=\dfrac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$	$y(t)$ Response Classification
$0<\zeta<1: s_1,s_2=-\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \quad (-\zeta\omega_n<0)$	*Underdamped*
$\zeta=1: \quad s_1,s_2=-\omega_n$	*Critically damped*
$\zeta>1: \quad s_1,s_2=-\zeta\omega_n \pm \omega_n\sqrt{\zeta^2-1}$	*Overdamped*
$\zeta=0: \quad s_1,s_2=\pm j\omega_n$	*Undamped (marginally unstable)*
$\zeta<0: \quad s_1,s_2=-\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \quad (-\zeta\omega_n>0)$	*Negatively damped (unstable)*

- The left-half *s*-plane corresponds to positive damping; that is, the damping factor or damping ratio is positive. Positive damping causes the unit-step response to settle to a constant final value in the steady state due to the negative exponent of exp($-\zeta\omega_n t$). The system is stable.
- The right-half *s*-plane corresponds to **negative damping**. Negative damping gives a response that **grows in magnitude without bound**, and the system is **unstable**.
- The imaginary axis corresponds to zero damping ($\sigma = 0$ or $\zeta = 0$). Zero damping results in a sustained oscillation response, and the system is marginally stable or marginally unstable.

Figure 3-13 illustrates typical unit-step responses that correspond to the various root locations already shown.

In this section, we demonstrated that the location of the characteristic equation roots plays an important role in the time response of a prototype second-order system—or any control system for that matter. In practical applications, only **stable** systems that correspond to $\zeta > 0$ are of interest.

3-5 IMPULSE RESPONSE AND TRANSFER FUNCTIONS OF LINEAR SYSTEMS

An alternative way to define the **transfer function** is to use the impulse response, which is defined in the following sections.

3-5-1 Impulse Response

Consider that a linear time-invariant system has the input $u(t)$ and output $y(t)$. As shown in Fig. 3-14, a rectangular pulse function $u(t)$ of a very large magnitude $\hat{u}/2\varepsilon$ becomes an impulse function for very small durations as $\varepsilon \to 0$. The equation representing Fig. 3-14 is

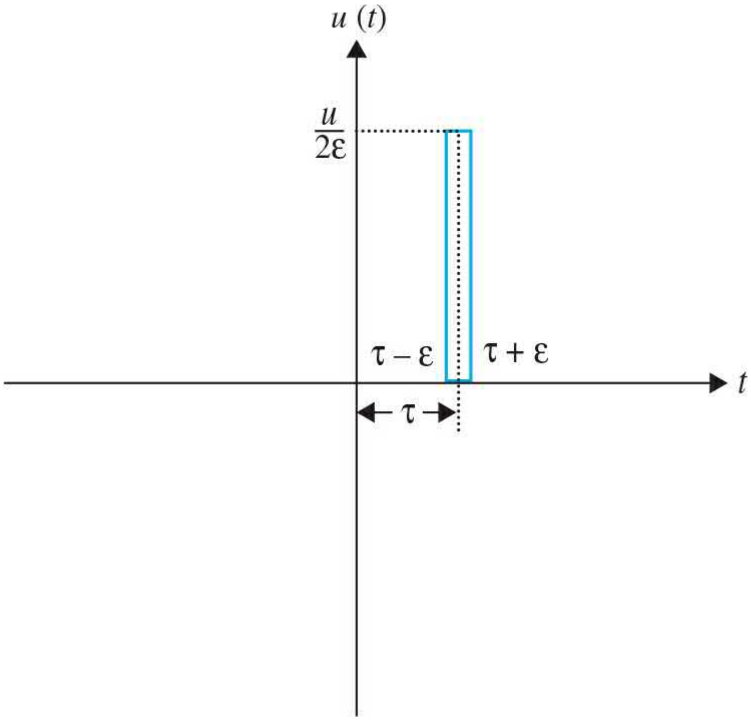


Figure 3-14 Graphical representation an impulse function.

$$u(t)=\begin{cases} 0 & t\le\tau-\varepsilon \\ \dfrac{\hat{u}}{2\varepsilon} & \tau-\varepsilon<t<\tau+\varepsilon \\ 0 & t\ge\tau+\varepsilon \end{cases} \tag{3-146}$$

For *û* = 1, *u*(*t*) = *δ*(*t*), is also known as unit impulse or **Dirac delta** function with the following properties:

$$\begin{aligned} &\delta(t-\tau)=0;\quad t\ne\tau \\ &\int_{\tau-\varepsilon}^{\tau+\varepsilon}\delta(t-\tau)dt=1;\quad \varepsilon>0 \\ &\int_{\tau-\varepsilon}^{\tau+\varepsilon}\delta(t-\tau)f(t)dt=f(t);\quad \varepsilon>0 \end{aligned} \tag{3-147}$$

where *f*(*t*) is any function of time. For *t* = 0 in Eq. (3-146) taking the Laplace transform, using Eq. (3-11), and noting the actual limits of the integral are defined from *t* = 0^- to *t* = ∞, the **Laplace transform of *ζ* (*t*)**, using the third property in Eq. (3-147) **is unity**, that is, as *ε* → 0

$$\begin{aligned} \mathcal{L}[\delta(t)] &= \int_{0^-}^{\infty}\delta(t)e^{-st}dt \\ &= \int_{\tau-\varepsilon}^{\tau+\varepsilon}\delta(t-\tau)f(t)dt=e^{-st}\Big|_{\tau-\varepsilon}^{\tau+\varepsilon}=1 \end{aligned} \tag{3-148}$$

In the following examples, we obtain the impulse response of a prototype second-order system.

EXAMPLE 3-5-1 For the following second-order prototype system find the impulse response:

$$\frac{d^2y(t)}{dt^2}+2\zeta\omega_n\frac{dy(t)}{dt}+\omega_n^2y(t)=\omega_n^2u(t) \tag{3-149}$$

For zero initial conditions,

$$G(s)=\frac{Y(s)}{U(s)}=\frac{\omega_n^2}{s^2+2\zeta\omega_ns+\omega_n^2} \tag{3-150}$$

is the transfer function of system (3-149). For *u*(*t*) = *ζ*(*t*), since *L*[*ζ*(*t*)] = *U*(*s*) = 1, using the inverse Laplace calculations in Example 3-3-3, the impulse response *y*(*t*) = *g*(*t*) for 0 < *ζ* < 1 is

$$y(t) = \frac{\omega_n}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t} \sin\left(\omega_n \sqrt{1-\zeta^2}\, t\right) \quad t \geq 0 \tag{3-151}$$

▲

EXAMPLE 3-5-2 Consider the linear differential equation

$$\frac{d^2 y(t)}{dt^2} + 34.5 \frac{dy(t)}{dt} + 1000\, y(t) = 1000\delta(t) \tag{3-152}$$

Following the solution in Eq. (3-151), the impulse response is

$$y(t) = 37.73 e^{-0.5455t} \sin(26.5t) \quad t \geq 0 \tag{3-153}$$

Using Toolbox 3-5-1, the time response plot of Eq. (3-153) is shown in Fig. 3-15. ▲

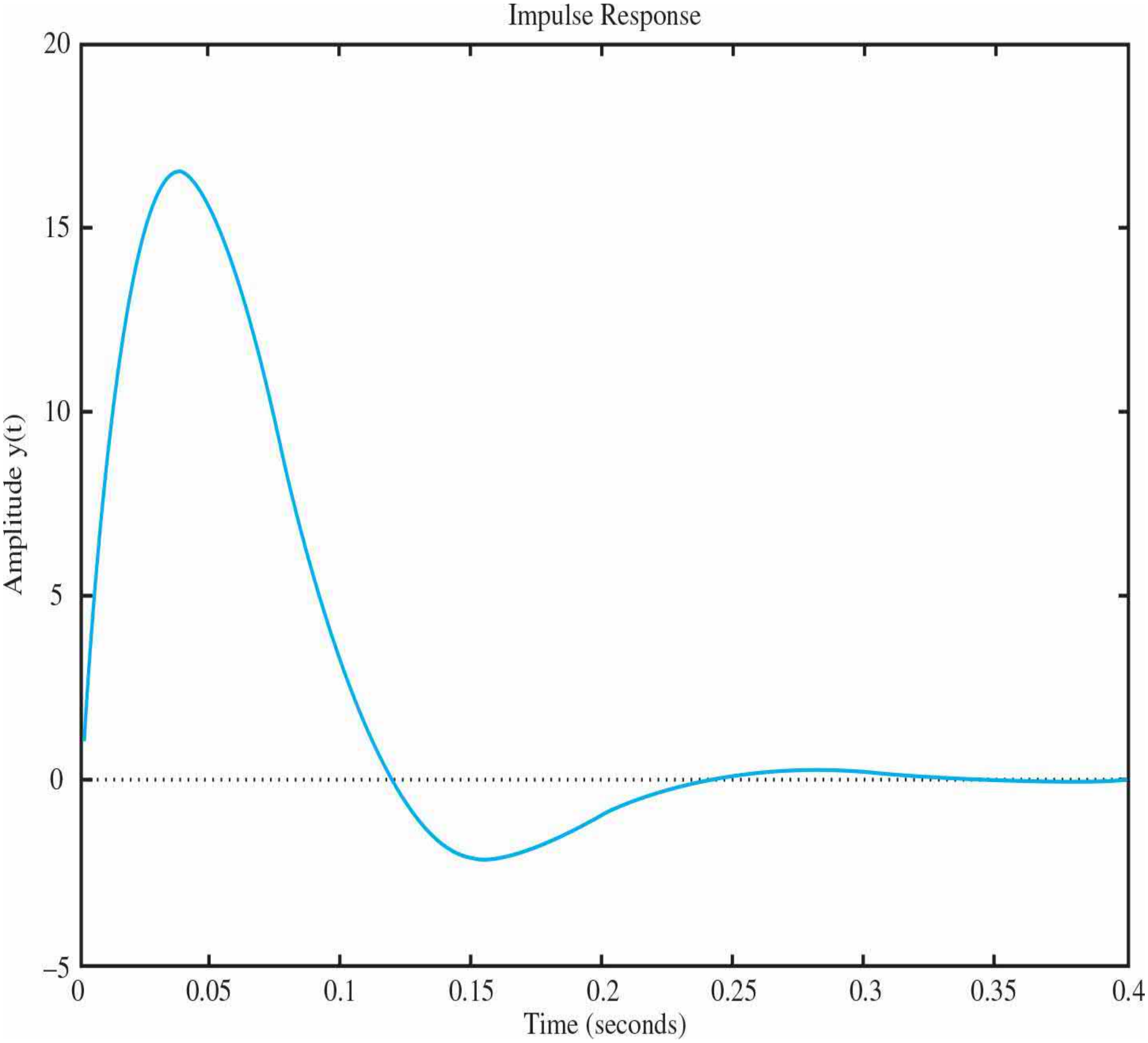
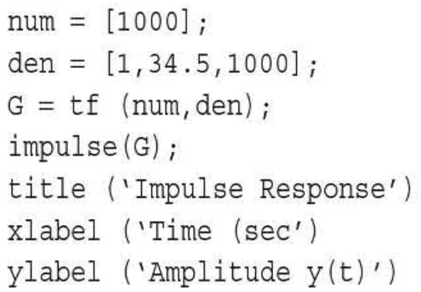


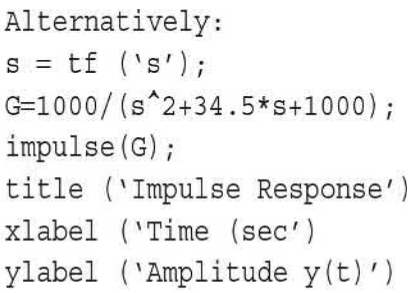
Figure 3-15 Impulse response *y*(*t*) of the second-order system in Eq. (3-153).

Toolbox 3-5-1

The unit impulse response of Eq. (3-152) may also be obtained using MATLAB.



```
num = [1000];
den = [1,34.5,1000];
G = tf (num,den);
impulse(G);
title ('Impulse Response')
xlabel ('Time (sec')
ylabel ('Amplitude y(t)')
```



```
Alternatively:
s = tf ('s');
G=1000/(s^2+34.5*s+1000);
impulse(G);
title ('Impulse Response')
xlabel ('Time (sec')
ylabel ('Amplitude y(t)')
```

"impulse" produces the time response of a function for an impulse input.

3-5-2 Time Response Using the Impulse Response

It is important to point out that *the response of any system can be characterized by its impulse response* $g(t)$, which is defined as the output for a unit-impulse input $\delta(t)$. Once the impulse response of a linear system is known, *the output of the system* $y(t)$, *for any input*, $u(t)$, *can be found by using the transfer function*. Recall,

The time response of any linear system, for any given input can be found by using the impulse response.

$$G(s) = \frac{\mathcal{L}(y(t))}{\mathcal{L}(u(t))} = \frac{Y(s)}{U(s)} \tag{3-154}$$

is the **transfer function** of the system. For more details see, for example, Ref. 14. We demonstrate this concept through the following example.

EXAMPLE 3-5-3 For the second-order prototype system in Eq. (3-149) use the impulse response $g(t)$ from Example 3-5-1 to find the time response for a unit-step input $u(t) = u_s(t)$.

The Laplace transform of Eq. (3-149) for zero initial conditions is

$$\mathcal{L}[y(t)] = Y(s) = \frac{1}{s}\frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{G(s)}{s} \tag{3-155}$$

Recall from Eq. (3-139), the time response of this system was obtained to be

$$y(t) = 1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}\sin\left[\left(\omega_n\sqrt{1-\zeta^2}\right)t + \cos^{-1}\zeta\right] \quad t \geq 0 \tag{3-156}$$

Using Eq. (3-155) and the convolution properties of Laplace transforms, from Table 3-1, we have

$$\mathcal{L}[y(t)] = \frac{G(s)}{U(s)} = \frac{G(s)}{s} = \mathcal{L}[u_s * g(t)] = \mathcal{L}\left[\int_0^t u_s g(t-\tau)d\tau\right] \tag{3-157}$$

As a result from Eq. (3-157), the output *y*(*t*) is therefore

$$\int_0^t u_s g(t-\tau)d\tau = \int_0^t \frac{\omega_n}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n(t-\tau)} \sin\left(\omega_n\sqrt{1-\zeta^2}(t-\tau)\right)d\tau \quad t \geq 0 \tag{3-158}$$

or, after some manipulations, we get

$$y(t) = 1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}\sin\left[\left(\omega_n\sqrt{1-\zeta^2}\right)t + \theta\right] \quad t \geq 0 \tag{3-159}$$

where $\theta = \cos^{-1}\zeta$. Obviously, Eqs. (3-159) and (3-156) are identical. ▲

3-5-3 Transfer Function (Single-Input, Single-Output Systems)

Let *G*(*s*) denote the transfer function of a single-input, single-output (SISO) system, with input *u*(*t*), output *y*(*t*), and impulse response *g*(*t*). We can formalize the findings in Sec. 3-5-1 and conclude the following.

The transfer function is alternatively defined as the Laplace transform of the impulse response, with all the initial conditions set to zero.

Hence, the transfer function *G*(*s*) is defined as

$$G(s)=\mathcal{L}[g(t)]=\frac{Y(s)}{U(s)} \tag{3-160}$$

with all the initial conditions set to zero, and *Y*(*s*) and *U*(*s*) are the Laplace transforms of *y*(*t*) and *u*(*t*), respectively.

3-6 SYSTEMS OF FIRST-ORDER DIFFERENTIAL EQUATIONS: STATE EQUATIONS

State equations provide an alternative to the transfer function approach, discussed earlier, to study differential equations. This technique particularly provides a powerful means to treat and analyze higher-order differential equations and is highly utilized in modern control theory and more advanced topics in control systems, such as optimal control design.

In general, an *n*th-order differential equation can be decomposed into *n* first-order differential equations. Because, in principle, first-order differential equations are simpler to solve than higher-order ones, first-order differential equations are used in the analytical studies of control systems. As an example, for the differential equation in Eq. (3-2), shown here

$$L\frac{di(t)}{dt}+Ri(t)+\int\frac{i(t)}{C}dt=e(t) \tag{3-161}$$

if we let

$$x_1(t)=\int i(t)dt \tag{3-162}$$

and

$$x_2(t) = \frac{dx_1(t)}{dt} = i(t) \tag{3-163}$$

then Eq. (3-161) is decomposed into the following two first-order differential equations:

$$\frac{dx_1(t)}{dt} = x_2(t) \tag{3-164}$$

$$\frac{dx_2(t)}{dt} = -\frac{1}{LC}x_1(t) - \frac{R}{L}x_2(t) + \frac{1}{L}e(t) \tag{3-165}$$

Alternatively, for the differential equation in Eq. (3-3),

$$M\frac{d^2y(t)}{dt^2}(t) + B\frac{dy(t)}{dt} + Ky(t) = f(t) \tag{3-166}$$

if we let

$$x_1(t) = y(t) \tag{3-167}$$

and

$$x_2(t) = \frac{dx_1(t)}{dt} = \frac{dy(t)}{dt} \tag{3-168}$$

then Eq. (3-166) is decomposed into the following two first-order differential equations:

$$\frac{dx_1(t)}{dt} = x_2(t) \tag{3-169}$$

$$\frac{dx_2(t)}{dt} = -\frac{B}{M}x_2(t) - \frac{K}{M}x_1(t) + \frac{1}{M}f(t) \tag{3-170}$$

In a similar manner, for Eq. (3-5), let us define

$$\begin{aligned} x_1(t) &= y(t) \\ x_2(t) &= \frac{dy(t)}{dt} \\ &\vdots \\ x_n(t) &= \frac{d^{n-1}y(t)}{dt^{n-1}} \end{aligned} \tag{3-171}$$

then the *n*th-order differential equation is decomposed into *n* first-order differential equations:

$$\begin{aligned} \frac{dx_1(t)}{dt} &= x_2(t) \\ \frac{dx_2(t)}{dt} &= x_3(t) \\ &\vdots \\ \frac{dx_n(t)}{dt} &= -a_0x_1(t) - a_1x_2(t) - \cdots - a_{n-2}x_{n-1}(t) - a_{n-1}x_n(t) + f(t) \end{aligned} \tag{3-172}$$

Notice that the last equation is obtained by equating the highest-ordered derivative term in Eq. (3-5) to the rest of the terms. In control systems theory, the set of first-order differential equations in Eq. (3-172) is called the **state equations**, and x_1,x_2,...,x_n, are called the **state variables**. Finally, *the minimum number of state variables needed is usually the same as the order n of the differential equation of the system*.

The minimum number of state variables needed to represent a differential equation, is usually the same as the order of the differential equation of the system.

EXAMPLE 3-6-1 Considering the two degrees of freedom mechanical system shown in Fig. 3-16 with two masses M_1 and M_2 constrained by three springs, while a fore *f*(*t*)is applied to mass M_2.

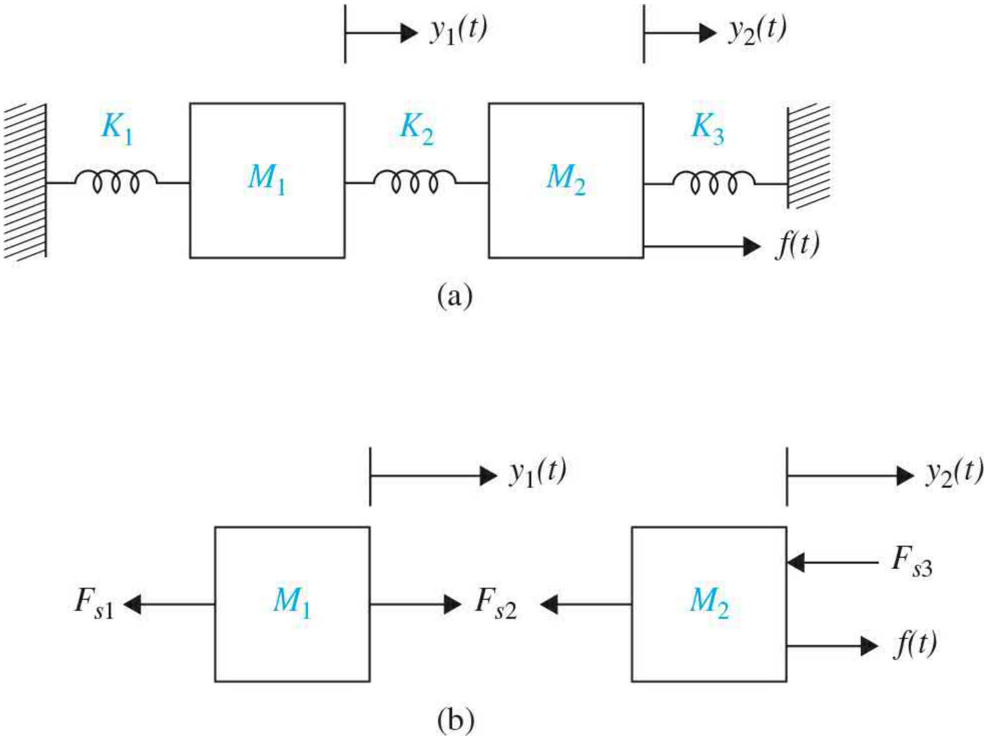


Figure 3-16 A two degree of freedom mechanical system with three springs.

The displacements of masses M_1 and M_2 are measured by $y_1(t)$ and $y_2(t)$, respectively. From Example 2-1-2, the two second-order differential equations of motion are

$$M_1\ddot{y}_1(t)+(K_1+K_2)y_1(t)-K_2y_2(t)=0 \tag{3-173}$$

$$M_2\ddot{y}_2(t)-K_2y_1(t)+(K_2+K_3)y_2(t)=f(t) \tag{3-174}$$

if we let

$$x_1(t)=y_1(t) \tag{3-175}$$

$$x_2(t)=y_2(t) \tag{3-176}$$

and

$$x_3(t) = \frac{dx_1(t)}{dt} = \frac{dy_1(t)}{dt} \tag{3-177}$$

$$x_4(t) = \frac{dx_2(t)}{dt} = \frac{dy_2(t)}{dt} \tag{3-178}$$

Then the two second-order differential equations are decomposed into four first-order differential equations, that is, the following state equations:

$$\frac{dx_1(t)}{dt} = x_3(t) \tag{3-179}$$

$$\frac{dx_2(t)}{dt} = x_4(t) \tag{3-180}$$

$$\frac{dx_3(t)}{dt} = -\frac{(K_1 + K_2)}{M_1} x_1(t) + \frac{K_2}{M_1} x_2(t) \tag{3-181}$$

$$\frac{dx_4(t)}{dt} = \frac{K_2}{M_2} x_1(t) - \frac{(K_2 + K_3)}{M_2} x_2(t) + \frac{f(t)}{M_2} \tag{3-182}$$

Note that the process of selecting the state variables is not unique, and we could have used the following representation:

$$x_1(t) = y_1(t) \tag{3-183}$$

$$x_2(t) = \frac{dx_1(t)}{dt} = \frac{dy_1(t)}{dt} \tag{3-184}$$

$$x_3(t) = y_2(t) \tag{3-185}$$

$$x_4(t) = \frac{dx_3(t)}{dt} = \frac{dy_2(t)}{dt} \tag{3-186}$$

As a result the state equations become

$$\frac{dx_1(t)}{dt} = x_2(t) \tag{3-187}$$

$$\frac{dx_2(t)}{dt} = -\frac{(K_1 + K_2)}{M_1}x_1(t) + \frac{K_2}{M_1}x_3(t) \tag{3-188}$$

$$\frac{dx_3(t)}{dt} = x_4(t) \tag{3-189}$$

$$\frac{dx_4(t)}{dt} = \frac{K_2}{M_2}x_1(t) - \frac{(K_2 + K_3)}{M_2}x_3(t) + \frac{f(t)}{M_2} \tag{3-190}$$

3-6-1 Definition of State Variables

The state of a system refers to the past, present, and future conditions of the system. From a mathematical perspective, it is convenient to define a set of state variables and state equations to model dynamic systems. As stated earlier, the variables $x_1(t), x_2(t), \ldots, x_n(t)$ defined in Eq. (3-171) are the **state variables** of the *n*th-order system described by Eq. (3-5), and the *n* first-order differential equations, in Eq. (3-172), are the **state equations**. In general, there are some basic rules regarding the definition of a state variable and what constitutes a state equation. The state variables must satisfy the following conditions:

- At any initial time $t = t_0$, the state variables $x_1(t_0), x_2(t_0), \ldots, x_n(t_0)$ define the **initial states** of the system.
- Once the inputs of the system for $t \geq t_0$ and the initial states just defined are specified, the state variables should completely define the future behavior of the system.

The state variables of a system are defined as a **minimal set** of variables, $x_1(t), x_2(t), \ldots, x_n(t)$, such that knowledge of these variables at any time t_0 and information on the applied input at time t_0 are sufficient to determine the state of the system at any time $t > t_0$. Hence, the **space state form** for *n* state variables is

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{3-191}$$

where **x(t)** is the state vector having *n* rows,

$$\mathbf{x}(t)=\begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix} \tag{3-192}$$

and **u(t)** is the input vector with *p* rows,

$$\mathbf{u}(t)=\begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_p(t) \end{bmatrix} \tag{3-193}$$

The coefficient matrices **A** and **B** are defined as

$$\mathbf{A}=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}(n\times n) \tag{3-194}$$

$$\mathbf{B}=\begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix}(n\times p) \tag{3-195}$$

EXAMPLE 3-6-2 For the system described by Eqs. (3-187) through (3-190),

$$\mathbf{x}(t)=\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} \tag{3-196}$$

$$\mathbf{u}(t) = f(t) \tag{3-197}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\dfrac{(K_1 + K_2)}{M_1} & 0 & \dfrac{K_2}{M_1} & 0 \\ 0 & 0 & 0 & 1 \\ \dfrac{K_2}{M_2} & 0 & -\dfrac{(K_2 + K_3)}{M_2} & 0 \end{bmatrix} (4 \times 4) \tag{3-198}$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dfrac{1}{M_2} \end{bmatrix} (4 \times 1) \tag{3-199}$$

3-6-2 The Output Equation

One should not confuse the state variables with the outputs of a system. An **output** of a system is a variable that *can be measured*, but a state variable does not always need to satisfy this requirement. For instance, in an electric motor, such state variables as the winding current, rotor velocity, and displacement can be measured physically, and these variables all qualify as output variables. On the other hand, magnetic flux can also be regarded as a state variable in an electric motor because it represents the past, present, and future states of the motor, but it cannot be measured directly during operation and therefore does not ordinarily qualify as an output variable. In general, an output variable can be expressed as an algebraic combination of the state variables. For the system described by Eq. (3-5), if $y(t)$ is designated as the output, then the output equation is simply $y(t) = x_1(t)$. In general,

$$\mathbf{y}(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_q(t) \end{bmatrix} = \mathbf{Cx}(t) + \mathbf{Du}(t) \tag{3-200}$$

$$\mathbf{C} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qn} \end{bmatrix} \tag{3-201}$$

$$\mathbf{D} = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1p} \\ d_{21} & d_{22} & \cdots & d_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ d_{q1} & d_{q2} & \cdots & d_{qp} \end{bmatrix} \tag{3-202}$$

We will utilize these concepts in the modeling of various dynamical systems next.

EXAMPLE 3-6-3 Consider the second-order differential equation, which was also studied in Example 3-4-1.

$$\frac{d^2y(t)}{dt^2} + 3\frac{dy(t)}{dt} + 2y(t) = 2u(t) \tag{3-203}$$

If we let

$$x_1(t) = y(t) \tag{3-204}$$

and

$$x_2(t) = \frac{dx_1(t)}{dt} = \frac{dy(t)}{dt} \tag{3-205}$$

then Eq. (3-203) is decomposed into the following two first-order differential equations:

$$\frac{dx_1(t)}{dt} = x_2(t) \tag{3-206}$$

$$\frac{dx_2(t)}{dt} = -2x_1(t) - 3x_2(t) + 2u(t) \tag{3-207}$$

where $x_1(t)$, $x_2(t)$ are the state variables, and $u(t)$ is the input, we can—at this point arbitrarily—define $y(t)$ as the output represented by

$$y(t) = x_1(t) \tag{3-208}$$

In this case, we are simply interested in state variable $x_1(t)$ to be our output. As a result,

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}; \quad \mathbf{u}(t) = u(t) \tag{3-209}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} 0 \\ 2 \end{bmatrix}; \quad \mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}; \quad \mathbf{D} = 0 \tag{3-210}$$

EXAMPLE 3-6-4 As another example the state equations in vector-matrix form:

$$\begin{bmatrix} \dfrac{dx_1(t)}{dt} \\ \dfrac{dx_2(t)}{dt} \\ \dfrac{dx_3(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ \dfrac{-(a_2 + a_3)}{1 + a_0 a_3} & -a_1 & \dfrac{1 - a_0 a_2}{1 + a_0 a_3} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} r(t) \tag{3-211}$$

The output equation may be a more complex representation of the state variables, for example,

$$y(t)=\frac{1}{1+a_0a_3}x_1(t)+\frac{a_0}{1+a_0a_3}x_3(t) \tag{3-212}$$

where

$$\mathbf{C}=\left[\frac{1}{1+a_0a_3}\quad 0\quad \frac{a_0}{1+a_0a_3}\right] \tag{3-213}$$

▲

EXAMPLE 3-6-5 Consider an accelerometer, which is a sensor used to measure the acceleration of an object it is attached to, as shown in Fig. 3-17. If the motion of the object is *u*(*t*), the equation of motion for the accelerometer seismic mass *M* shown in the free-body diagram of Fig. 3-17*b* may be written as

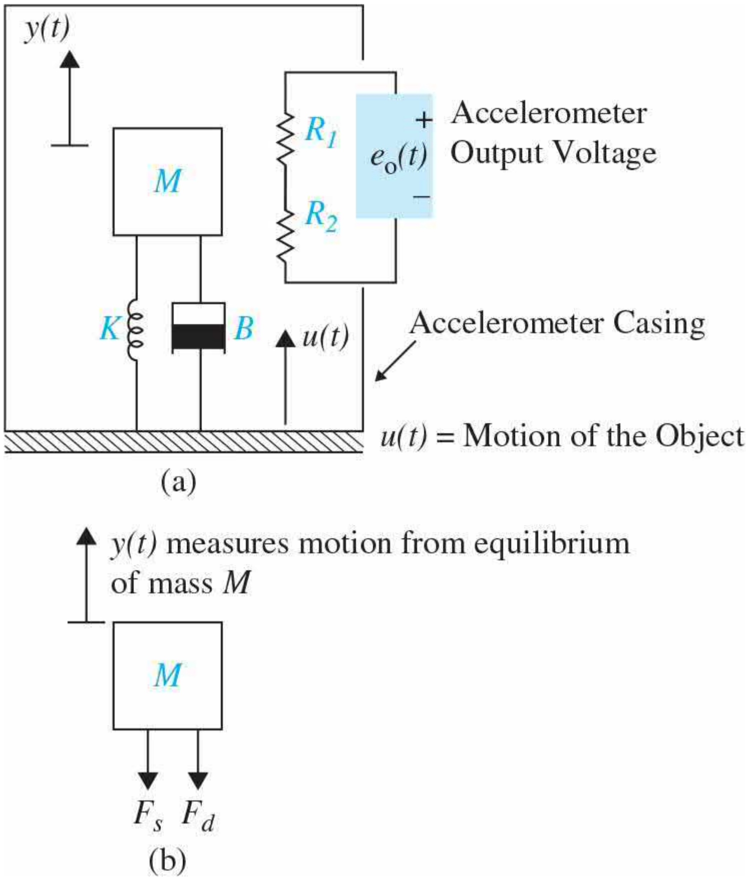


Figure 3-17 (*a*) Schematic of an accelerometer mounted on a moving object. (*b*) Free-body diagram.

$$-K(y(t)-u(t))-B(\dot{y}(t)-\dot{u}(t))=M\ddot{y}(t) \tag{3-214}$$

where *B* and *K* are the accelerometer internal material damping constant and stiffness, respectively. If we define the relative motion of the seismic mass *M* by *z*(*t*)

$$z(t)=y(t)-u(t) \tag{3-215}$$

Then Eq. (3-214) can be rewritten in terms of the object acceleration which the sensor measures:

$$M\ddot{z}(t)+B\dot{z}(t)+Kz(t)=-M\ddot{u}(t) \tag{3-216}$$

The accelerometer output is in terms of voltage, which is linearly proportional to the seismic mass relative motion through constant K_a—also known as the sensor gain. That is

$$e_o(t)=K_a z(t) \tag{3-217}$$

In state space form, if we define the state variables as

$$x_1(t)=z(t) \tag{3-218}$$

and

$$x_2(t)=\frac{dx_1(t)}{dt}=\frac{dz(t)}{dt} \tag{3-219}$$

then Eq. (3-216) is decomposed into the following two state equations:

$$\frac{dx_1(t)}{dt}=x_2(t) \tag{3-220}$$

$$\frac{dx_2(t)}{dt}=-\frac{B}{M}x_2(t)-\frac{K}{M}x_1(t)+\ddot{u}(t) \tag{3-221}$$

where $x_1(t)$, $x_2(t)$ are the state variables, and $\acute{u}(t)$ is the input. We can define $e_o(t)$ as the output.

So

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}; \quad \mathbf{u}(t) = \ddot{u}(t) \tag{3-222}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -\frac{K}{M} & -\frac{B}{M} \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}; \quad \mathbf{C} = [K_a \quad 0]; \quad \mathbf{D} = 0 \tag{3-223}$$

$$e_o(t) = K_a x_1(t) \tag{3-224}$$

We will revisit this problem later on in this chapter. As a side note, in order to better understand how an accelerometer measures acceleration, we need to understand its frequency response characteristics, which we will address in Chap. 10. For more appreciation of this topic, you may wish to refer to Ref. 14. ▲

EXAMPLE 3-6-6 Consider the differential equation

$$\frac{d^3y(t)}{dt^3} + 5\frac{d^2y(t)}{dt^2} + \frac{dy(t)}{dt} + 2y(t) = u(t) \tag{3-225}$$

Rearranging the last equation so that the highest-order derivative term is set equal to the rest of the terms, we have

$$\frac{d^3y(t)}{dt^3} = -5\frac{d^2y(t)}{dt^2} - \frac{dy(t)}{dt} - 2y(t) + u(t) \tag{3-226}$$

The state variables are defined as

$$
\begin{aligned}
x_1(t) &= y(t) \\
x_2(t) &= \frac{dy(t)}{dt} \\
x_3(t) &= \frac{d^2y(t)}{dt^2}
\end{aligned}
\tag{3-227}
$$

Then the state equations are represented by the vector-matrix equation

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \tag{3-228}$$

where x(*t*) is the 2 × 1 state vector, *u*(*t*) is the scalar input. The output equation is *arbitrarily* selected in this case to be

$$y(t) = x_1(t) = [1 \quad 0]\mathbf{x}(t) \tag{3-229}$$

Hence,

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -2 & -1 & -5 \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}; \quad \mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{3-230}$$

3-7 SOLUTION OF THE LINEAR HOMOGENEOUS STATE EQUATION

The linear time-invariant state equation

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{3-231}$$

can be solved using either the classical method of solving linear differential equations or the Laplace transform method. The Laplace transform solution is presented in the following equations.

Taking the Laplace transform on both sides of Eq. (3-231), we have

$$s\mathbf{X}(s) - \mathbf{x}(0) = \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) \tag{3-232}$$

where x(0) denotes the initial-state vector evaluated at $t = 0$. Solving for $\mathbf{X}(s)$ in Eq. (3-232) yields

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{x}(0) + (s\mathbf{I} - \mathbf{A})^{-1}[\mathbf{BU}(s)] \tag{3-233}$$

where **I** is the identity matrix, $\mathbf{X}(s) = L[\mathbf{x}(t)]$ and $\mathbf{U}(s) = L[\mathbf{u}(t)]$. The *solution of the state equation* of Eq. (3-231) is obtained by taking the inverse Laplace transform on both sides of Eq. (3-233):

$$\mathbf{x}(t) == \mathcal{L}^{-1}[(s\mathbf{I} - \mathbf{A})^{-1}]\mathbf{x}(0) + \mathcal{L}^{-1}\{(s\mathbf{I} - \mathbf{A})^{-1}[\mathbf{BU}(s)]\} \tag{3-234}$$

Once the state vector $\mathbf{x}(t)$ is found, the output is easy to obtain through

$$\mathbf{y}(t) = \mathbf{Cx}(t) + \mathbf{Du}(t) \tag{3-235}$$

EXAMPLE 3-7-1 Consider the state equations representation of the system represented by Eq. (3-203) in Example 3-6-3

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 2 \end{bmatrix} u(t) \tag{3-236}$$

The problem is to determine the solution for the state vector x(t) for $t \geq 0$ when the input is a unit step, that is, $u(t) = 1$ for $t \geq 0$. This system is the same as the second-order overdamped system in Example 3-4-1. The coefficient matrices are identified to be

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} \tag{3-237}$$

Therefore,

$$s\mathbf{I} - \mathbf{A} = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} = \begin{bmatrix} s & -1 \\ 2 & s+3 \end{bmatrix} \tag{3-238}$$

The inverse matrix of $(s\mathbf{I} - \mathbf{A})$ is

$$(s\mathbf{I}-\mathbf{A})^{-1}=\frac{1}{s^2+3s+2}\begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix} \tag{3-239}$$

The solution of the state equation is found using Eq. (3-234). Thus,

$$\mathbf{x}(t)=\begin{bmatrix} 2e^{-t}-e^{-2t} & e^{-t}-e^{-2t} \\ -2e^{-t}+2e^{-2t} & -e^{-t}+2e^{-2t} \end{bmatrix}\mathbf{x}(0)+\begin{bmatrix} 1-e^{-t}+e^{-2t} \\ e^{-t}-2e^{-2t} \end{bmatrix} \quad t\geq 0 \tag{3-240}$$

where, the second term of the solution can be obtained by taking the inverse Laplace transform of $(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{BU}(s)$. Thus, we have

$$\begin{aligned}\mathcal{L}^{-1}[(s\mathbf{I}-\mathbf{A})^{-1}]\mathbf{BU}(s)&=\mathcal{L}^{-1}\left(\frac{1}{s^2+3s+2}\begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix}\begin{bmatrix} 0 \\ 2 \end{bmatrix}\frac{1}{s}\right)\\ &=\mathcal{L}^{-1}\left(\frac{1}{s^2+3s+2}\begin{bmatrix} \frac{2}{s} \\ 2 \end{bmatrix}\right)=\begin{bmatrix} 1-e^{-t}+e^{-2t} \\ e^{-t}-2e^{-2t} \end{bmatrix} \quad t\geq 0\end{aligned} \tag{3-241}$$

Note that in the current example, the overall solution in Eq. (3-239) is a superposition of the response due to the initial conditions and the input $u(t)$. For zero initial conditions, the response in this case is identical to the solution in Eq. (3-119) obtained earlier for the overdamped system in Example 3-4-1. The solution of the state equations, however, is more powerful because it shows both states $x_1(t)$ and $x_2(t)$. Finally, having found the states, we can now find the output $\mathbf{y}(t)$ from Eq. (3-235). ▲

3-7-1 Transfer Functions (Multivariable Systems)

The definition of a transfer function is easily extended to a system with multiple inputs and outputs. A system of this type is often referred to as a multivariable system. In a multivariable system, a differential equation of the form of Eq. (3-5) may be used to describe the relationship between a pair of input and output variables, when all other inputs are set to zero. This equation is restated as

$$\begin{aligned}\frac{d^n y(t)}{dt^n} &= a_{n-1}\frac{d^{n-1}y(t)}{dt^{n-1}} + \cdots + a_1\frac{dy(t)}{dt} + a_0 y(t) \\ &= b_m\frac{d^m u(t)}{dt^m} + b_{m-1}\frac{d^{m-1}u(t)}{dt^{m-1}} + \cdots + b_1\frac{du(t)}{dt} + b_0 u(t)\end{aligned} \tag{3-242}$$

The coefficients $a_0, a_1, \ldots, a_{n-1}$ and $b_0, b_1, \ldots, b_m$ are real constants. Using the state space representation of Eq. (3-242), we have

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{Bu}(t) \tag{3-243}$$

$$\mathbf{y}(t) = \mathbf{Cx}(t) + \mathbf{Du}(t) \tag{3-244}$$

Taking the Laplace transform on both sides of Eq. (3-242) and solving for X(s), we have

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{x}(0) + (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{BU}(s) \tag{3-245}$$

The Laplace transform of Eq. (3-244) is

$$\mathbf{Y}(s) = \mathbf{CX}(s) + \mathbf{DU}(s) \tag{3-246}$$

Substituting Eq. (3-245) into Eq. (3-246), we have

$$\mathbf{Y}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{x}(0) + \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{BU}(s) + \mathbf{DU}(s) \tag{3-247}$$

Because the definition of a transfer function requires that the initial conditions be set to zero, **x**(0) = **0**; thus, Eq. (3-247) becomes

$$\mathbf{Y}(s) = [\mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}]\mathbf{U}(s) \tag{3-248}$$

We define the **transfer-function matrix** between **u**(t) and **y**(t), as

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \tag{3-249}$$

where **G**(s) is a $q \times p$. Then, Eq. (3-248) becomes

$$\mathbf{Y}(s) = \mathbf{G}(\mathbf{s})\mathbf{U}(s) \tag{3-250}$$

In general, if a linear system has *p* inputs and *q* outputs, the transfer function between the *j*th input and the *i*th output is defined as

$$G_{ij}(s) = \frac{Y_i(s)}{U_j(s)} \tag{3-251}$$

with $U_k(s) = 0$, $k = 1,2,\ldots, p$, $k \neq j$. Note that Eq. (3-251) is defined with only the *j*th input in effect, whereas the other inputs are set to zero. *Because the principle of superposition is valid for linear systems, the total effect on any output due to all the inputs acting simultaneously is obtained by adding up the outputs due to each input acting alone*. When all the *p* inputs are in action, the *i*th output transform is written as

$$Y_i(s) = G_{i1}(s)U_1(s) + G_{i2}(s)U_2(s) + \cdots + G_{ip}(s)U_p(s) \tag{3-252}$$

where

$$\mathbf{G}(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) & \cdots & G_{1p}(s) \\ G_{21}(s) & G_{22}(s) & \cdots & G_{2p}(s) \\ \cdot & \cdot & \cdots & \cdot \\ G_{q1}(s) & G_{q2}(s) & \cdots & G_{qp}(s) \end{bmatrix} \tag{3-253}$$

is the *q* × *p* transfer-function matrix.

Later in Chaps. 4 and 8 we will provide more details on treatment of differential equations using the state-space approach.

EXAMPLE 3-7-2 Consider a multivariable system, described by the following differential equations

$$\frac{d^2y_1(t)}{dt^2} + 4\frac{dy_1(t)}{dt} - 3y_2(t) = u_1(t) \tag{3-254}$$

$$\frac{dy_1(t)}{dt} + \frac{dy_2(t)}{dt} + y_1(t) + 2y_2(t) = u_2(t) \tag{3-255}$$

Using the following choice of state variables:

$$\begin{aligned} x_1(t) &= y_1(t) \\ x_2(t) &= \frac{dy_1(t)}{dt} \\ x_3(t) &= y_2(t) \end{aligned} \tag{3-256}$$

where, these state variables are defined by mere inspection of the two differential equations because no particular reasons for the definitions are given other than that these are the most convenient. Now equating the first term of each of the equations of Eqs. (3-254) and (3-255) to the rest of the terms and using the state-variable relations of Eq. (3-256), we arrive at state equations and output equations in vector-matrix form as represented by Eqs. (3-243) and (3-244). Or,

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -4 & 3 \\ -1 & -1 & -2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} \tag{3-257}$$

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} = \mathbf{Cx}(t) \tag{3-258}$$

where the output choice has been set arbitrarily. In order to determine the transfer-function matrix of the system using the state-variable formulation, we substitute the **A**, **B**, and **C** matrices into Eq. (3-249). First, we form the matrix (s**I** – **A**):

$$(s\mathbf{I} - \mathbf{A}) = \begin{bmatrix} s & -1 & 0 \\ 0 & s+4 & -3 \\ 1 & 1 & s+2 \end{bmatrix} \tag{3-259}$$

The determinant of (s**I** – **A**) is

$$|s\mathbf{I} - \mathbf{A}| = s^3 + 6s^2 + 11s + 3 \tag{3-260}$$

Thus,

$$
\begin{aligned}
(s\mathbf{I}-\mathbf{A})^{-1} &= \frac{\operatorname{adj}(s\mathbf{I}-\mathbf{A})}{\det(s\mathbf{I}-\mathbf{A})} \\
&= \frac{1}{s^3+6s^2+11s+3}\begin{bmatrix} s^2+6s+11 & s+2 & 3 \\ -3 & s(s+2) & 3s \\ -(s+4) & -(s+1) & s(s+4) \end{bmatrix}
\end{aligned}
\tag{3-261}
$$

The transfer-function matrix between **u**(*t*) and **y**(*t*) is

$$
\mathbf{G(s)} = \mathbf{C}(s\mathbf{I}-\mathbf{A})^{-1}\mathbf{B} = \frac{1}{s^3+6s^2+11s+3}\begin{bmatrix} s+2 & 3 \\ -(s+1) & s(s+4) \end{bmatrix} \tag{3-262}
$$

Alternatively, using the conventional approach, we take the Laplace transform on both sides of Eqs. (3-257) and (3-258) and assume zero initial conditions. The resulting transformed equations are written in vector-matrix form as

$$
\begin{bmatrix} s(s+4) & -3 \\ s+1 & s+2 \end{bmatrix}\begin{bmatrix} Y_1(s) \\ Y_2(s) \end{bmatrix} = \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} \tag{3-263}
$$

Solving for **Y**(*s*) from Eq. (3-263), we obtain

$$
\mathbf{Y}(s) = \mathbf{G}(s)\mathbf{U}(s) \tag{3-264}
$$

where

$$
\begin{aligned}
\mathbf{G}(s) &= \left[\begin{pmatrix} s(s+4) & -3 \\ s+1 & s+2 \end{pmatrix}\right]^{-1} \\
&= \frac{1}{(s^3+6s^2+11s+3)}\begin{bmatrix} (s+2) & 3 \\ -(s+1) & s(s+4) \end{bmatrix}
\end{aligned}
\tag{3-265}
$$

which will give the same results as in Eq. (3-262). ▲

EXAMPLE 3-7-3 For the state equation represented in Example 3-7-1, if we define the output as

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t); \quad \mathbf{C} = [1 \quad 0] \tag{3-266}$$

That is $y(t) = x_1(t)$. For zero initial conditions, the transfer-function matrix between $\mathbf{u}(t)$ and $\mathbf{y}(t)$ is

$$\mathbf{G(s)} = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} = \begin{bmatrix} 1 & 0 \end{bmatrix} \left(\frac{1}{s^2 + 3s + 2} \begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix} \begin{bmatrix} 0 \\ 2 \end{bmatrix} \right) \tag{3-267}$$

or,

$$G(s) = \frac{2}{s^2 + 3s + 2} \tag{3-268}$$

which is identical to the second-order transfer function for the overdamped system in Example 3-4-1. ▲

3-7-2 Characteristic Equation from State Equations

From the transfer function discussions in the previous section, we can write Eq. (3-249) as

$$\begin{aligned} \mathbf{G}(s) &= \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} = \mathbf{C}\frac{\text{adj}(s\mathbf{I} - \mathbf{A})}{\det(s\mathbf{I} - \mathbf{A})}\mathbf{B} + \mathbf{D} \\ &= \frac{\mathbf{C}[\text{adj}(s\mathbf{I} - \mathbf{A})]\mathbf{B} + |s\mathbf{I} - \mathbf{A}|\mathbf{D}}{|s\mathbf{I} - \mathbf{A}|} \end{aligned} \tag{3-269}$$

Setting the denominator of the transfer-function matrix $\mathbf{G}(s)$ to zero, we get the **characteristic equation**:

$$|s\mathbf{I} - \mathbf{A}| = 0 \tag{3-270}$$

which is an alternative form of the characteristic equation but should lead to the same equation as in Eq. (3-22). *An important property of the characteristic equation is that, if the coefficients of* **A** *are real, then the coefficients of* $|\mathrm{s}\mathbf{I} - \mathbf{A}|$ *are also real*. The roots of the characteristic equation are also referred to as the **eigenvalues** of the matrix **A**.

EXAMPLE 3-7-4 The matrix **A** for the state equations of the differential equation in Eq. (3-225) is given in Eq. (3-230). The characteristic equation of **A** is

$$|s\mathbf{I}-\mathbf{A}| = \begin{vmatrix} s & -1 & 0 \\ 0 & s & -1 \\ 2 & 1 & s+5 \end{vmatrix} = s^3+5s^2+s+2=0 \tag{3-271}$$

Note that the characteristic equation is a polynomial of third order while **A** is a 3 × 3 matrix. ▲

EXAMPLE 3-7-5 The matrix **A** for the state equations of Example 3-7-1 is given in Eq. (3-237). The characteristic equation of **A** is

$$|s\mathbf{I}-\mathbf{A}| = \begin{vmatrix} s & -1 \\ 2 & s+3 \end{vmatrix} = s^2+3s+2=0 \tag{3-272}$$

Note in this case the order of the characteristic equation and the **A** matrix dimension are the same. ▲

EXAMPLE 3-7-6 The characteristic equation in Example 3-7-2 is

$$|s\mathbf{I}-\mathbf{A}| = s^3+6s^2+11s+3=0 \tag{3-273}$$

Again **A** is a 3 × 3 matrix, and the characteristic equation is a third-order polynomial. ▲

3-7-3 State Equations from the Transfer Function

Based on the previous discussions, transfer function of the system can be obtained from the state space equations. However, obtaining the state space equations from the transfer function without a clear knowledge of the physical system model and its properties is not a unique process—particularly because of variety of potential choices for the state and the output variables. The process of going from the transfer function to the state diagram is called **decomposition**. In general, there are three basic ways to decompose transfer functions. These are **direct decomposition, cascade decomposition**, and **parallel decomposition**. Each of these three schemes of decomposition has its own merits and is best suited for a particular purpose. This topic will be further discussed in more detail in Chap. 8.

In this section, we demonstrate how to get state equations from the transfer function using the **direct decomposition** technique. Let us consider the transfer function between *u*(*t*) and *y*(*t*) is given by

$$G(s)=\frac{Y(s)}{U(s)}=\frac{b_m s^m+b_{m-1}s^{m-1}+\cdots+b_1 s+b_0}{s^n+a_{n-1}s^{n-1}+\cdots+a_1 s+a_0};\quad m\leq n-1 \tag{3-274}$$

where, the coefficients $a_0,a_1,\ldots,a_{n-1}$, and $b_0,b_1,\ldots,b_m$ are real constants, $U(s) = L[u(u(t)]$, and $Y(s) = L[y(t)]$. As we shall see later it is necessary to have $m \leq n - 1$. Premultiplying both sides of Eq. (3-274) by the denominator, we get

> Obtaining the state space equations from the transfer function without a clear knowledge of the physical system model and its properties is not a unique process.

$$\left(s^n+a_{n-1}s^{n-1}+\cdots+a_1 s+a_0\right)Y(s)=\left(b_m s^m+b_{m-1}s^{m-1}+\cdots+b_1 s+b_0\right)U(s) \tag{3-275}$$

Taking the inverse Laplace transform of Eq. (3-275), while recalling that the transfer function is independent of initial conditions of the system, we get the following *n*th-order differential equation with constant real coefficients:

$$\begin{aligned}\frac{d^n y(t)}{dt^n}+a_{n-1}\frac{d^{n-1}y(t)}{dt^{n-1}}+\cdots+a_1\frac{dy(t)}{dt}+a_0 y(t)\\=b_m\frac{d^m u(t)}{dt^m}+b_{m-1}\frac{d^{m-1}u(t)}{dt^{m-1}}+\cdots+b_1\frac{du(t)}{dt}+b_0 u(t)\end{aligned} \tag{3-276}$$

In this decomposition approach, our goal is to convert the transfer function in Eq. (3-274) into the state space form:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\,\mathbf{x}(t) + \mathbf{B}u(t) \tag{3-277}$$

$$y(t) = \mathbf{C}\mathbf{x}(t) + Du(t) \tag{3-278}$$

Note *y*(*t*) and *u*(*t*) are not vectors and are scalar functions. From the denominator of Eq. (3-274) the characteristic equation is an *n*th-order polynomial,

$$|s\mathbf{I} - \mathbf{A}| = s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 = 0 \tag{3-279}$$

which implies **A** to be a *n* × *n* matrix. As a result the system is expected to have *n* states. Hence,

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix} \tag{3-280}$$

We *assume* the following form for the coefficient matrix **B**:

$$\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \quad (n \times 1) \tag{3-281}$$

Hence, for Eqs. (3-277), (3-278), and (3-279), respectively, we must have

$$\begin{aligned}
\frac{dx_1(t)}{dt} &= x_2(t) \\
\frac{dx_2(t)}{dt} &= x_3(t) \\
&\vdots \\
\frac{dx_{n-1}(t)}{dt} &= x_n(t) \\
\frac{dx_n(t)}{dt} &= -a_0 x_1(t) - a_1 x_2(t) - \cdots - a_{n-2} x_{n-1}(t) - a_{n-1} x_n(t) + u(t)
\end{aligned} \tag{3-282}$$

and

$$y(t) = b_0 x_1(t) + b_1 x_2(t) + \cdots + b_{n-1} x_n(t) \tag{3-283}$$

This implies, in the output Eq. (3-278), $D = 0$ and **C** has one row and n columns. That is,

$$\mathbf{C} = [b_0 \quad b_1 \quad b_2 \quad \cdots \quad b_{n-2} \quad b_{n-1}] \quad (1 \times n) \tag{3-284}$$

where this requires m not to exceed $n - 1$ in Eq. (3-274)—that is, $m \le n - 1$.

Finally, as a result of the direct decomposition technique, the coefficient matrices **A**, **B**, **C**, and **D** are

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-2} & -a_{n-1} \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix};$$
$$\mathbf{C} = \begin{bmatrix} b_0 & b_1 & b_2 & \cdots & b_{n-2} & b_{n-1} \end{bmatrix}; \quad D = 0 \tag{3-285}$$

Again, please note that this topic will be further discussed in more detail in Chap. 8.

EXAMPLE 3-7-7 Consider the following input-output transfer function:

$$\frac{Y(s)}{U(s)} = \frac{2s^2+s+5}{s^3+6s^2+11s+4} \tag{3-286}$$

The dynamic equations of the system using direct decomposition method are

$$\begin{bmatrix} \dfrac{dx_1(t)}{dt} \\ \dfrac{dx_2(t)}{dt} \\ \dfrac{dx_3(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -4 & -11 & -6 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = [\begin{matrix} 5 & 1 & 2 \end{matrix}]x(t) \tag{3-287}$$

▲

EXAMPLE 3-7-8 Consider the accelerometer in Example 3-6-4, as shown in Fig. 3-17. If the motion of the object is *u*(*t*), the equation of motion for the accelerometer seismic mass *M* shown in the free-body diagram of Fig. 3-17*b* may be written as

$$-K(y(t)-u(t)) - B(\dot{y}(t)-\dot{u}(t)) = M\ddot{y}(t) \tag{3-288}$$

where *B* and *K* are the accelerometer internal material damping constant and stiffness, respectively. Rearranging Eq. (3-288), we have

$$M\,\ddot{y}(t) + B\,\dot{y}(t) + K\,y(t) = B\dot{u}(t) + K\,u(t) \tag{3-289}$$

In this case, we use the accelerometer absolute displacement *y*(*t*) as our output variable. The transfer function in this case represents a displacement input-output relation

$$\frac{Y(s)}{U(s)}=\frac{\frac{B}{M}s+\frac{K}{M}}{s^2+\frac{B}{M}s+\frac{K}{M}} \tag{3-290}$$

then using direct decomposition, Eq. (3-290) is decomposed into the following two state equations:

$$\frac{dx_1(t)}{dt}=x_2(t) \tag{3-291}$$

$$\frac{dx_2(t)}{dt}=-\frac{K}{M}x_1(t)-\frac{B}{M}x_2(t)+u(t) \tag{3-292}$$

where $x_1(t)$, $x_2(t)$ are the state variables, and displacement $u(t)$ is the input. From Eq. (3-283), the output is therefore

$$y(t)=\begin{bmatrix} \frac{K}{M} & \frac{B}{M} \end{bmatrix}\mathbf{x}(t)=\frac{K}{M}x_1(t)+\frac{B}{M}x_2(t) \tag{3-293}$$

In order to confirm that Eqs. (3-291) through (3-293) do indeed represent the transfer function in Eq. (3-290), let us take the Laplace transform while setting the initial conditions to zero. Hence,

$$sX_1(s)=X_2(s) \tag{3-294}$$

$$sX_2(s)=-\frac{K}{M}X_1(s)-\frac{B}{M}X_2(s)+U(s) \tag{3-295}$$

$$Y(s)=\frac{K}{M}X_1(s)+\frac{B}{M}X_2(s) \tag{3-296}$$

where $X_1(s) = L[x_1(t)]$, $X_2(s) = L[x_2(t)]$, $U(s) = L[u(t)]$ and $Y(s) = L[y(t)]$. Using Eq. (3-294) to eliminate $X_2(s)$ from Eqs. (3-295) and (3-296), we have

$$\left(s^2+\frac{B}{M}s+\frac{K}{M}\right)X_1(s)=U(s) \tag{3-297}$$

$$\frac{Y(s)}{\frac{B}{M}s+\frac{K}{M}}=X_1(s) \tag{3-298}$$

Solving Eqs. (3-297) and (3-298) in terms of *Y*(*s*) and *U*(*s*), we get

$$\frac{Y(t)}{U(t)}=\frac{\frac{B}{M}s+\frac{K}{M}}{s^2+\frac{B}{M}s+\frac{K}{M}} \tag{3-299}$$

which is the same as Eq. (3-290).

In order to reconcile this representation of the accelerometer with that of Example 3-6-5, let us introduce a new output variable

$$z(t)=y(t)-u(t)=\frac{K}{M}x_1(t)+\frac{B}{M}x_2(t)-u(t) \tag{3-300}$$

Taking the Laplace transform, assuming zero initial conditions, we have

$$Z(s)=\frac{K}{M}X_1(s)+\frac{B}{M}X_2(s)-U(s) \tag{3-301}$$

Using Eq. (3-294) to eliminate X_2(*s*) from Eq. (3-301), and solving the resulting equation and Eq. (3-297) in terms of *Z*(*s*) and *U*(*s*), we get

$$\frac{Z(s)}{U(s)}=\frac{-s^2}{s^2+\frac{B}{M}s+\frac{K}{M}} \tag{3-302}$$

which is the transfer function of Eq. (3-216). ▲

3-8 CASE STUDIES WITH MATLAB

In this section, we use state space to find the time response of simple practical examples.

EXAMPLE 3-8-1 Let us consider the *RLC* network shown in Fig. 3-18. Using the voltage law

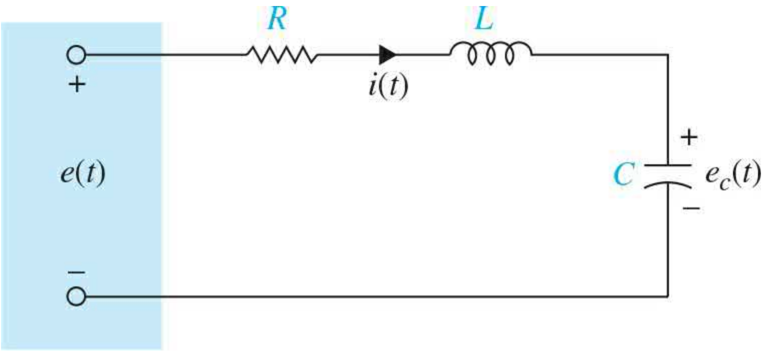


Figure 3-18 RLC network.

$$e(t) = e_R + e_L + e_c \tag{3-303}$$

where e_R = voltage across the resistor R
e_L = voltage across the inductor L
e_c = voltage across the capacitor C

or

$$e(t) = +e_c(t) + Ri(t) + L\frac{di(t)}{dt} \tag{3-304}$$

Using current in *C*,

$$C\frac{de_c(t)}{dt} = i(t) \tag{3-305}$$

and taking a derivative of Eq. (3-304) with respect to time, we get the equation of the *RLC* network as

$$L\frac{d^2i(t)}{dt^2} + R\frac{di(t)}{dt} + \frac{i(t)}{C} = \frac{de(t)}{dt} \tag{3-306}$$

A practical approach is to assign the current in the inductor *L*, *i*(*t*), and the voltage across the capacitor *C*,

$e_c(t)$, as the state variables. The reason for this choice is because the state variables are directly related to the energy-storage element of a system. The inductor stores kinetic energy, and the capacitor stores electric potential energy. By assigning $i(t)$ and $e_c(t)$ as state variables, we have a complete description of the past history (via the initial states) and the present and future states of the network. The state equations for the network in Fig. 3-18 are written by first equating the current in C and the voltage across L in terms of the state variables and the applied voltage $e(t)$. In vector-matrix form, the equations of the system are expressed as

$$\begin{bmatrix} \dfrac{de_c(t)}{dt} \\ \dfrac{di(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & \dfrac{1}{C} \\ -\dfrac{1}{L} & -\dfrac{R}{L} \end{bmatrix} \begin{bmatrix} e_c(t) \\ i(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \dfrac{1}{L} \end{bmatrix} e(t) \qquad (3\text{-}307)$$

This format is also known as the state form if we set

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} e_c(t) \\ i(t) \end{bmatrix} \qquad (3\text{-}308)$$

or

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & \dfrac{1}{C} \\ -\dfrac{1}{L} & -\dfrac{R}{L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \dfrac{1}{L} \end{bmatrix} e(t) \qquad (3\text{-}309)$$

Let us define the output as

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} e_c(t) \\ i(t) \end{bmatrix}$$
$$\mathbf{y}(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}(t) \tag{3-310}$$

That is we are interested in measuring both the current *i*(*t*) and the voltage across the capacitor, *ec*(*t*).

As a result, the coefficient matrices are

$$\mathbf{A} = \begin{bmatrix} 0 & \frac{1}{C} \\ -\frac{1}{L} & -\frac{R}{L} \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix}; \quad \mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{3-311}$$

The transfer functions between *e*(*t*) and y(*t*) are obtained by applying Eq. (3-249), when all the initial states are set to zero. Hence, from

$$\begin{aligned} \mathbf{G}(s) &= \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} \\ &= \frac{1}{LCs^2 + RCs + 1} \begin{bmatrix} 1 \\ Cs \end{bmatrix} \end{aligned} \tag{3-312}$$

More specifically, the two transfer functions are

$$\frac{E_c(s)}{E(s)} = \frac{1}{1 + RCs + LCs^2} \tag{3-313}$$

$$\frac{I(s)}{E(s)} = \frac{Cs}{1 + RCs + LCs^2} \tag{3-314}$$

Toolbox 3-8-1

Time-domain step responses for the outputs in Eqs. (3-313) and (3-314) are shown in Fig. 3-19, using *R* = 1, *L* = 1, *C* = 1:

```
R=1; L=1; C=1;
t=0:0.02:30;
num1 = [1];
den1 = [L*C R*C 1];
num2 = [C 0];
den2 = [L*C R*C 1];
G1 = tf(num1, den1);
G2 = tf(num2, den2);
y1 = step (G1, t);
y2 = step (G2, t);
plot(t,y1);
hold on
plot (t, y2, ‘--’);
xlabel(‘Time (s)’)
ylabel(‘Output’)
```

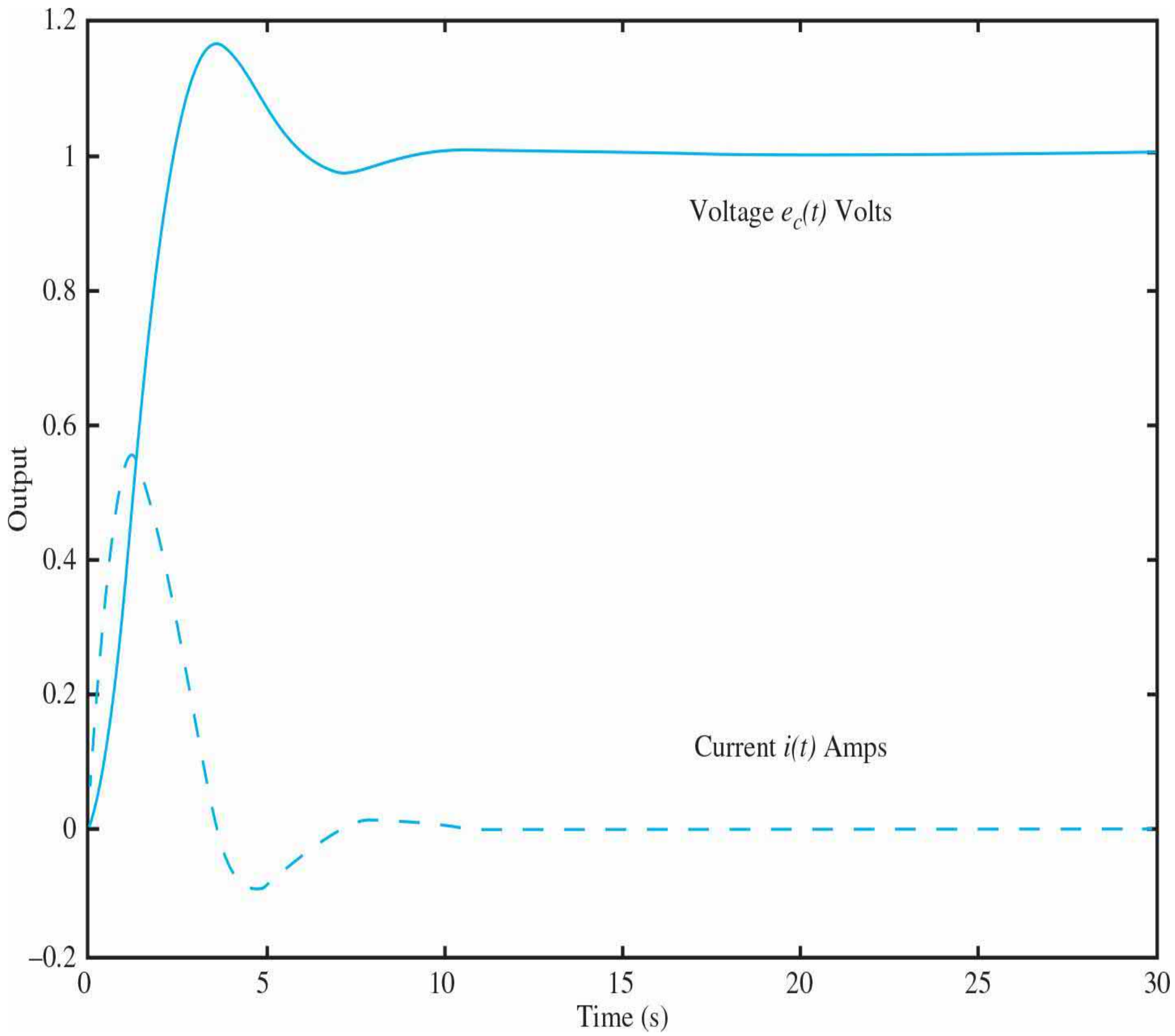


Figure 3-19 Output voltage and current step responses for Example 3-8-1.

EXAMPLE 3-8-2 As another example of writing the state equations of an electric network, consider the network shown in Fig. 3-20. According to the foregoing discussion, the voltage across the capacitor, $e_c(t)$, and the currents of the inductors, $i_1(t)$ and $i_2(t)$, are assigned as state variables, as shown in Fig. 3-20. The state equations of the network are obtained by writing the voltages across the inductors and the currents in the capacitor in terms of the three state variables. The state equations are

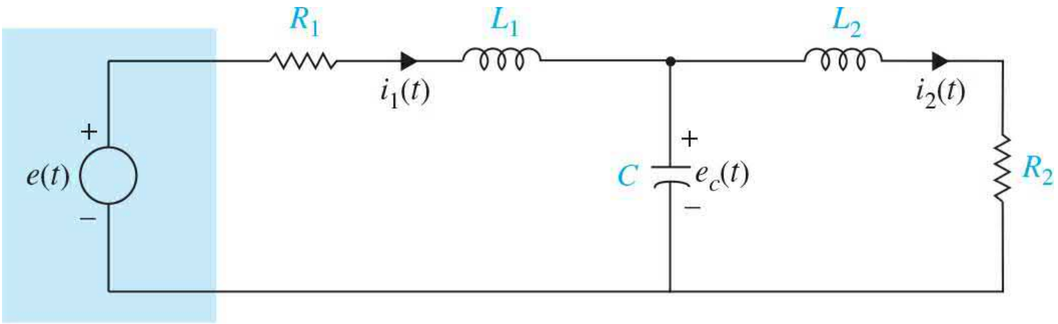


Figure 3-20 Electrical schematic of the network of Example 3-8-2.

$$L_1\frac{di_1(t)}{dt}=-R_1i_1(t)-e_c(t)+e(t) \tag{3-315}$$

$$L_2\frac{di_2(t)}{dt}=-R_2i_2(t)+e_c(t) \tag{3-316}$$

$$C\frac{de_c(t)}{dt}=i_1(t)-i_2(t) \tag{3-317}$$

In vector-matrix form, the state equations are written as

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix}=\begin{bmatrix}-\frac{R_1}{L_1} & 0 & -\frac{1}{L_1}\\ 0 & -\frac{R_2}{L_2} & \frac{1}{L_2}\\ \frac{1}{C} & -\frac{1}{C} & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}+\begin{bmatrix}\frac{1}{L_1}\\ 0\\ 0\end{bmatrix}e(t) \tag{3-318}$$

where

$$\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}=\begin{bmatrix}i_1(t)\\ i_2(t)\\ e_c(t)\end{bmatrix} \tag{3-319}$$

Similar to the procedure used in the previous example, the transfer functions between *I*1(*s*) and *E*(*s*), $I_2(s)$ and *E*(*s*), and $E_c(s)$ and *E*(*s*), respectively, appear next $i_2(t)$

$$\frac{I_1(s)}{E(s)} = \frac{L_2Cs^2 + R_2Cs + 1}{\Delta} \tag{3-320}$$

$$\frac{I_2(s)}{E(s)} = \frac{1}{\Delta} \tag{3-321}$$

$$\frac{E_c(s)}{E(s)} = \frac{L_2s + R_2}{\Delta} \tag{3-322}$$

where

$$\Delta = L_1L_2Cs^3 + (R_1L_2 + R_2L_1)Cs^2 + (L_1 + L_2 + R_1R_2C)s + R_1 + R_2 \tag{3-323}$$

Toolbox 3-8-2

Time-domain step responses for the outputs in Eqs. (3-320) to (3-322) are shown in Fig. 3-21, using $R_1 = 1$, $R_2 = 1$, $L_1 = 1$, $L_2 = 1$, $C = 1$:

```
R1=1; R2=1; L1=1; L2=1; C=1;
t=0:0.02:30;
num1 = [L2*C R2*C 1];
num2 = [1];
num3 = [L2 R2];
den = [L1*L2*C R1*L2*C+R2*L1*C L1+L2+R1*R2*C R1+R2];
G1 = tf(num1, den);
G2 = tf(num2, den);
G3 = tf(num3, den);
y1 = step (G1, t);
y2 = step (G2, t);
y3 = step (G3, t);
plot(t, y1);
hold on
plot(t, y2, ‘--’);
hold on
plot(t, y3, ‘-.’);
xlabel(‘Time (s)’)
ylabel(‘Output’)
```

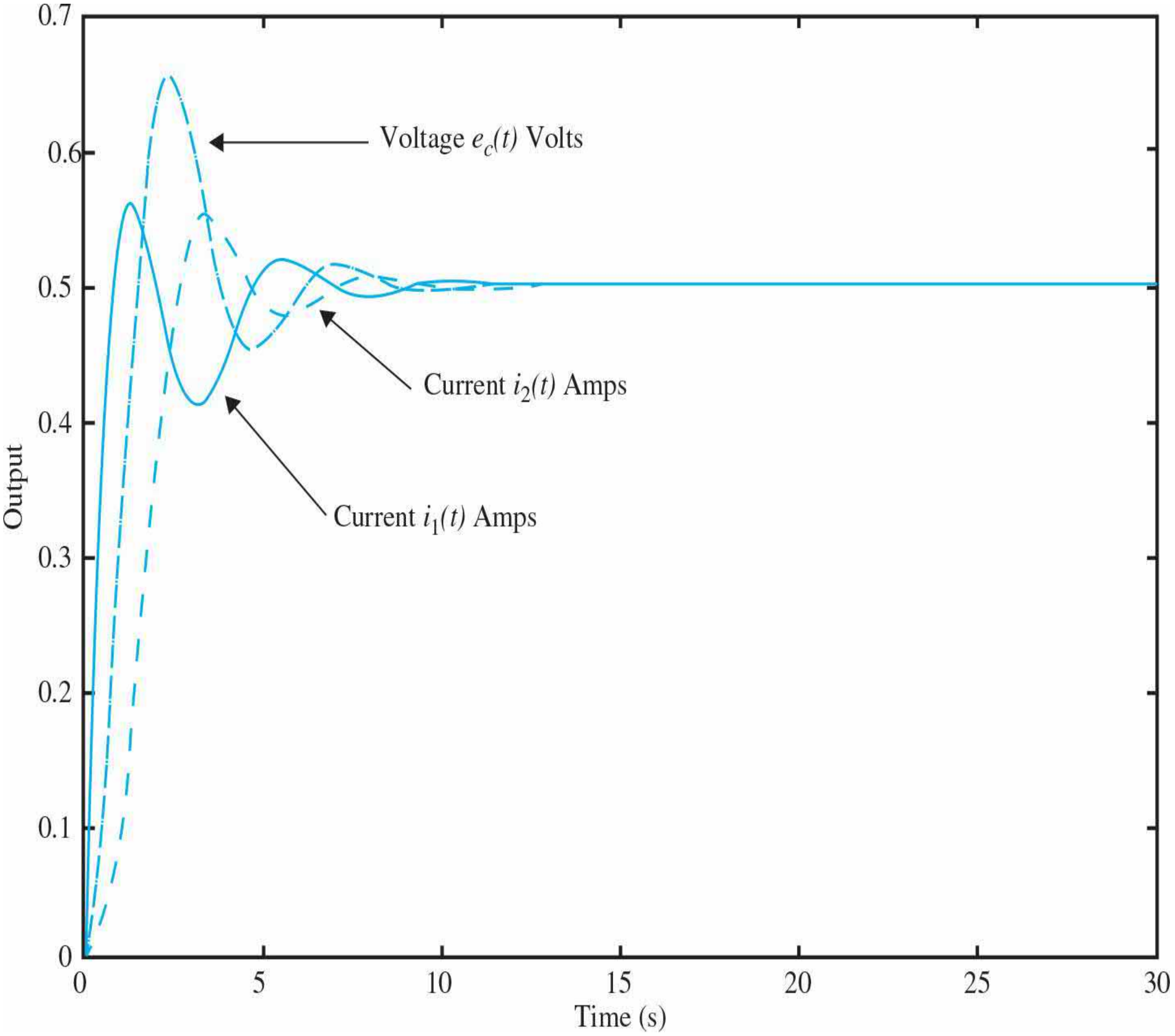


Figure 3-21 Output voltage and current step responses for Example 3-8-2.

EXAMPLE 3-8-3 Consider the accelerometer in Examples 3-6-4 and 3-7-8, as shown in Fig. 3-17. The state equations of the system using the state variables defined in Example 3-7-8 are

$$\frac{dx_1(t)}{dt} = x_2(t) \tag{3-324}$$

$$\frac{dx_2(t)}{dt} = -\frac{K}{M}x_1(t) - \frac{B}{M}x_2(t) + u(t) \tag{3-325}$$

where $x_1(t)$, $x_2(t)$ are the state variables, and displacement $u(t)$ is the input. Here we define the output equation to reflect both absolute and relative displacements of the seismic mass as outputs, that is, $y(t)$ and $z(t)$. From Eq. (3-283), the output is therefore

$$\begin{bmatrix} y(t) \\ z(t) \end{bmatrix} = \begin{bmatrix} \frac{K}{M} & \frac{B}{M} \\ \frac{K}{M} & \frac{B}{M} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix} u(t) \tag{3-326}$$

The two system transfer functions were obtained earlier as

$$\frac{Y(s)}{U(s)} = \frac{\frac{B}{M}s + \frac{K}{M}}{s^2 + \frac{B}{M}s + \frac{K}{M}} \tag{3-327}$$

$$\frac{Z(s)}{U(s)} = \frac{-s^2}{s^2 + \frac{B}{M}s + \frac{K}{M}} \tag{3-328}$$

Toolbox 3-8-3

Time-domain step responses for the outputs in Eqs. (3-327) and (3-328) are shown in Fig. 3-22, using $M = 1$, $B = 3$, and $K = 2$:

```
K=2; M=1; B=3;
t=0:0.02:30;
num1 = [B/M K/M];
num2 = [-1 0 0];
den = [1 B/M K/M];
G1 = tf(num1, den);
G2 = tf(num2, den);
y1 = step (G1, t);
y2 = step (G2, t);
plot(t, y1);
hold on
plot(t, y2, ‘--’);
xlabel(‘Time (Second)’) ; ylabel (‘Step Response’ )
```

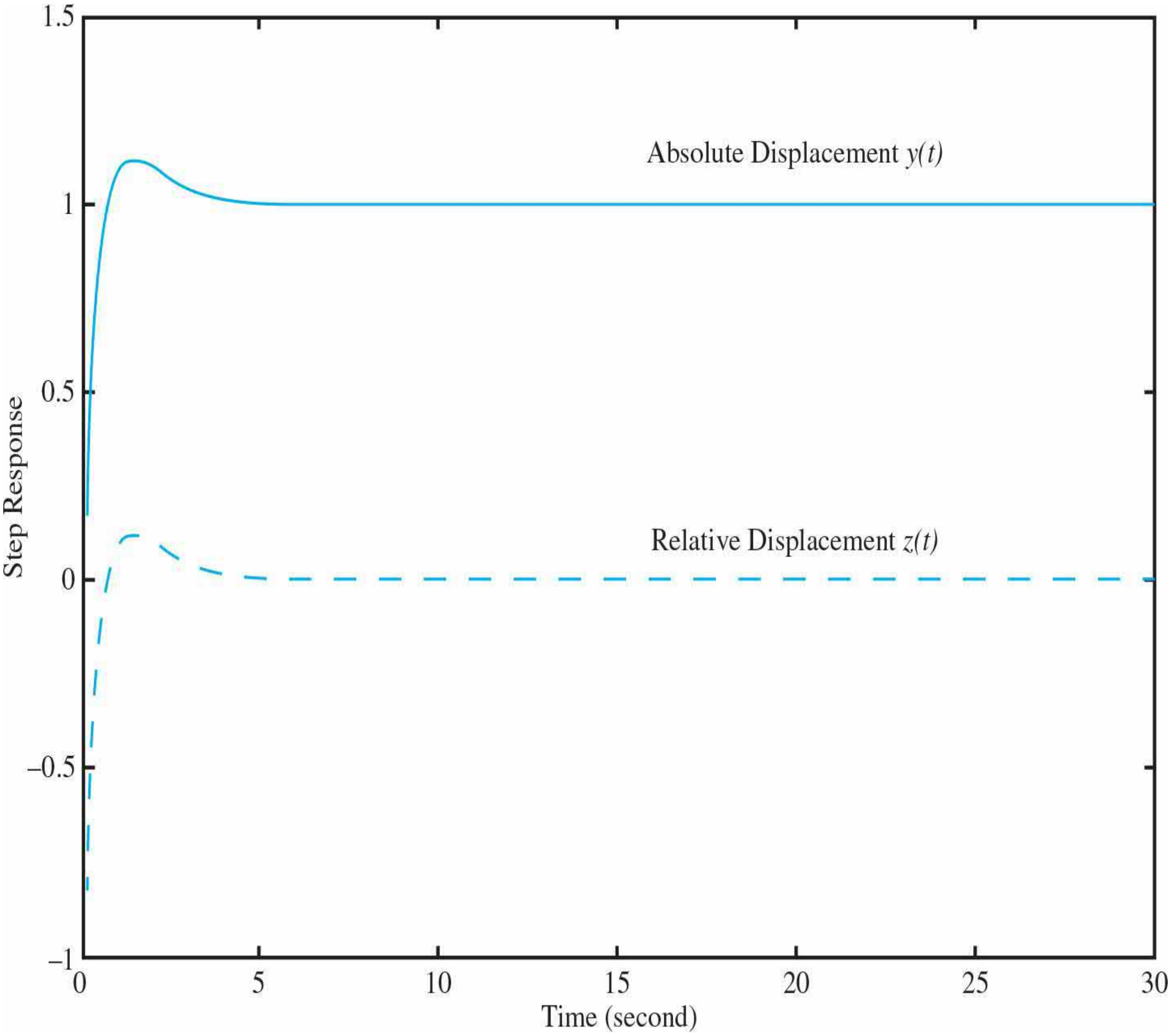


Figure 3-22 Absolute and relative displacement time responses to a step displacement input for the accelerometer seismic mass in Example 3-8-3.

Considering the time response plots in Fig. 3-22, are as expected. That is, for a unit step base movement, the absolute displacement follows the base at steady state, while the mass relative motion with respect to base, after experiencing initial acceleration, settles to zero.

3-9 LINEARIZATION REVISITED—THE STATE-SPACE APPROACH

In Sec. 2-4, we introduced the concept of linearization using the Taylor series technique. Alternatively, let us represent a nonlinear system by the following vector-matrix state equations:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}[\mathbf{x}(t), \mathbf{r}(t)] \tag{3-329}$$

where **x**(*t*) represents the *n* × 1 state vector; **r**(*t*), the *p* × 1 input vector; and **f**[**x**(*t*), **r**(*t*)], an *n* × 1 function vector. In general, f is a function of the state vector and the input vector.

Being able to represent a nonlinear and/or time-varying system by state equations is a distinct advantage of the state-variable approach over the transfer-function method, since the latter is strictly defined only for linear time-invariant systems.

As a simple example, the following nonlinear state equations are given:

$$\frac{dx_1(t)}{dt} = x_1(t) + x_2^2(t) \tag{3-330}$$

$$\frac{dx_2(t)}{dt} = x_1(t) + r(t) \tag{3-331}$$

Because nonlinear systems are usually difficult to analyze and design, it is desirable to perform a linearization whenever the situation justifies it.

A linearization process that depends on expanding the nonlinear state equations into a Taylor series about a nominal operating point or trajectory is now described. All the terms of the Taylor series of order higher than the first are discarded, and the linear approximation of the nonlinear state equations at the nominal point results.

Let the nominal operating trajectory be denoted by $\mathbf{x}_0(t)$, which corresponds to the nominal input $\mathbf{r}_0(t)$ and some fixed initial states. Expanding the nonlinear state equation of Eq. (3-329) into a Taylor series about $\mathbf{x}(t) = \mathbf{x}_0(t)$ and neglecting all the higher-order terms yields

$$x_i(t) = f_i(\mathbf{x}_0, \mathbf{r}_0) + \sum_{j=1}^{n} \left. \frac{\partial f_i(\mathbf{x}, \mathbf{r})}{\partial x_j} \right|_{x_0, r_0} (x_j - x_{0j}) + \sum_{j=1}^{p} \left. \frac{\partial f_i(\mathbf{x}, \mathbf{r})}{\partial r_j} \right|_{x_0, r_0} (r_j - r_{0j}) \tag{3-332}$$

where *i* = 1, 2,…,*n*. Let

$$\Delta x_i = x_i - x_{0i} \tag{3-333}$$

and

$$\Delta r_j = r_j - r_{0j} \tag{3-334}$$

Then

$$\Delta \dot{x}_i = \dot{x}_i - \dot{x}_{0i} \tag{3-335}$$

Since

$$\dot{x}_{0i} = f_i(\mathbf{x}_0, \mathbf{r}_0) \tag{3-336}$$

Equation (3-332) is written as

$$\Delta \dot{x}_i = \sum_{j=1}^{n} \frac{\partial f_i(\mathbf{x},\mathrm{r})}{\partial x_j}\bigg|_{x_0,r_0} \Delta x_j + \sum_{j=1}^{p} \frac{\partial f_i(\mathbf{x},\mathrm{r})}{\partial r_j}\bigg|_{x_0,r_0} \Delta r_j \tag{3-337}$$

Equation (3-337) may be written in vector-matrix form:

$$\Delta\mathbf{x} = \mathbf{A}^*\,\Delta\mathbf{x} + \mathbf{B}^*\,\Delta r \tag{3-338}$$

where

$$\mathbf{A}^* = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} & \cdots & \dfrac{\partial f_1}{\partial x_n} \\ \dfrac{\partial f_2}{\partial x_1} & \dfrac{\partial f_2}{\partial x_2} & \cdots & \dfrac{\partial f_2}{\partial x_n} \\ . & . & \cdots & . \\ \dfrac{\partial f_n}{\partial x_1} & \dfrac{\partial f_n}{\partial x_2} & \cdots & \dfrac{\partial f_n}{\partial x_n} \end{bmatrix} \tag{3-339}$$

$$\mathbf{B}^* = \begin{bmatrix} \dfrac{\partial f_1}{\partial r_1} & \dfrac{\partial f_1}{\partial r_2} & \cdots & \dfrac{\partial f_1}{\partial r_p} \\ \dfrac{\partial f_2}{\partial r_1} & \dfrac{\partial f_2}{\partial r_2} & \cdots & \dfrac{\partial f_2}{\partial r_p} \\ \cdot & \cdot & \cdots & \cdot \\ \dfrac{\partial f_n}{\partial r_1} & \dfrac{\partial f_n}{\partial r_2} & \cdots & \dfrac{\partial f_n}{\partial r_p} \end{bmatrix} \tag{3-340}$$

The following simple example illustrates the linearization procedure just described.

EXAMPLE 3-9-1 For the pendulum in Fig. 2-38, with a mass *m* and a massless rod of length *l*, if we define $x_1 = \theta$ and $x_2 = \theta$ as state variables, the state space representation of the system model becomes

$$\begin{aligned} \dot{x}_1 &= x_2(t) \\ \dot{x}_2 &= -\frac{g}{\ell}\sin x_1(t) \end{aligned} \tag{3-341}$$

Expanding the nonlinear state equation of Eq. (3-341) into a Taylor series about $\mathbf{x}(t) = \mathbf{x}_0(t) = 0$ (or $\theta = 0$) and neglecting all the higher-order terms yields, with $\mathbf{r}(t) = 0$ since there is no input (or external excitations) in this case, we get

$$\Delta\dot{x}_1(t) = \frac{\partial f_1(t)}{\partial x_2}\Delta x_2(t) = \frac{\partial x_2(t)}{\partial x_2}\Delta x_2(t) = \Delta x_2(t) \tag{3-342}$$

$$\Delta\dot{x}_2(t) = \frac{\partial f_2(t)}{\partial x_1(t)}\Delta x_1(t) = \left[\frac{\partial\left[-\frac{g}{\ell}\sin x_1(t)\right]}{\partial x_1(t)}\right]_{x_0=0} \Delta x_1(t) = -\frac{g}{\ell}\Delta x_1(t) \tag{3-343}$$

where Δx_1 and $\Delta x_2(t)$ denote nominal values of $x_1(t)$ and $x_2(t)$, respectively. Notice that the last two equations are linear and are valid only for small signals. In vector-matrix form, these linearized state equations are written as

$$\begin{bmatrix} \Delta\dot{x}_1(t) \\ \Delta\dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ a & 0 \end{bmatrix} \begin{bmatrix} \Delta x_1(t) \\ \Delta x_2(t) \end{bmatrix} \tag{3-344}$$

where

$$a = \frac{g}{\ell} = \text{constant} \tag{3-345}$$

If we let $a = \omega_n^2$, Eq. (3-344) becomes

$$\Delta\dot{x}_2(t) = \omega_n^2 \Delta x_1(t) \tag{3-346}$$

Switching back to classical representation, we get the linear system

$$\ddot{\theta} + \omega_n^2 \theta = 0 \tag{3-347}$$

▲

EXAMPLE 3-9-2 In Example 3-9-1, the linearized system turns out to be time-invariant. As mentioned earlier, linearization of a nonlinear system often results in a linear time-varying system. Consider the following nonlinear system:

$$\dot{x}_1(t) = \frac{-1}{x_2^2(t)} \tag{3-348}$$

$$\dot{x}_2(t) = u(t)x_1(t) \tag{3-349}$$

These equations are to be linearized about the nominal trajectory $[x_{01}(t), x_{02}(t)]$, which is the solution to the equations with initial conditions $x(0) = x_2(0) = 1$ and input $u(t) = 0$.

Integrating both sides of Eq. (3-349) with respect to *t*, we have

$$x_2(t) = x_2(0) = 1 \tag{3-350}$$

Then Eq. (3-348) gives

$$x_1(t) = -t + 1 \tag{3-351}$$

Therefore, the nominal trajectory about which Eqs. (3-350) and (3-351) are to be linearized is described by

$$x_{01}(t) = -t + 1 \tag{3-352}$$

$$x_{02}(t) = 1 \tag{3-353}$$

Now evaluating the coefficients of Eq. (3-337), we get

$$\frac{\partial f_1(t)}{\partial x_1(t)} = 0 \quad \frac{\partial f_1(t)}{\partial x_2(t)} = \frac{2}{x_2^3(t)} \quad \frac{\partial f_2(t)}{\partial x_1(t)} = u(t) \quad \frac{\partial f_2(t)}{\partial u(t)} = x_1(t) \tag{3-354}$$

Equation (3-337) gives

$$\Delta\dot{x}_1(t) = \frac{2}{x_{02}^3(t)}\Delta x_2(t) \tag{3-355}$$

$$\Delta\dot{x}_2(t) = u_0(t)\Delta x_1(t) + x_{01}(t)\Delta u(t) \tag{3-356}$$

By substituting Eqs. (3-353) and (3-354) into Eqs. (3-348) and (3-349), the linearized equations are

$$\begin{bmatrix} \Delta\dot{x}_1(t) \\ \Delta\dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x_1(t) \\ \Delta x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1-t \end{bmatrix} \Delta u(t) \tag{3-357}$$

which is a set of linear state equations with time-varying coefficients. ▲

EXAMPLE 3-9-3 Figure 3-23 shows the diagram of a magnetic-ball-suspension system. The objective of the system is to control the position of the steel ball by adjusting the current in the electromagnet through the input voltage *e*(*t*). The differential equations of the system are

Magnetic-Ball Suspension System

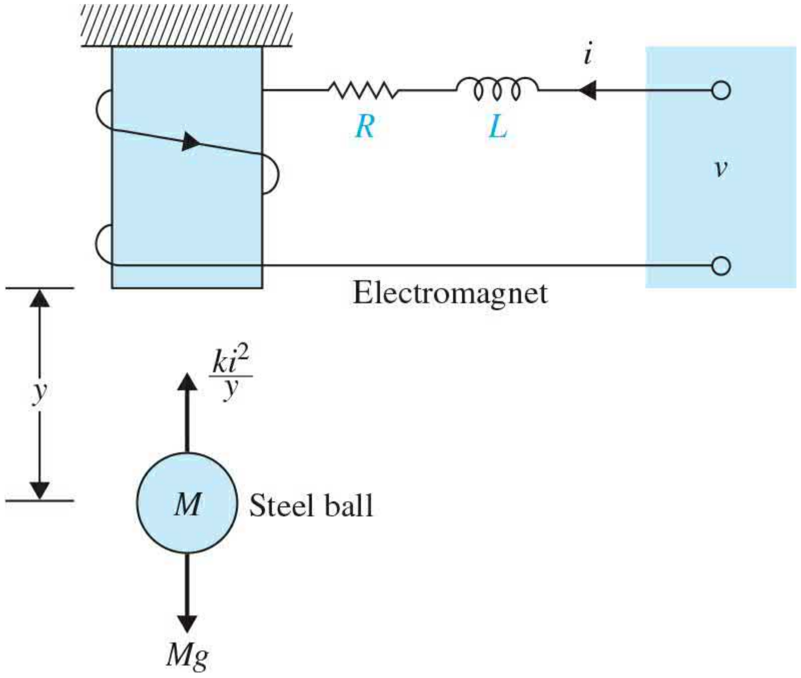


Figure 3-23 Magnetic-ball-suspension system.

$$M\frac{d^2y(t)}{dt^2} = Mg - \frac{i^2(t)}{y(t)} \tag{3-358}$$

$$e(t) = Ri(t) + L\frac{di(t)}{dt} \tag{3-359}$$

where $e(t)$ = input voltage
$y(t)$ = ball position
$i(t)$ = winding current
R = winding resistance
L = winding inductance
M = mass of ball
g = gravitational acceleration

Let us define the state variables as $x_1(t) = y(t), x_2(t) =$

dy(*t*)/*dt*, and $x_3(t) = i(t)$. The state equations of the system are

$$\frac{dx_1(t)}{dt} = x_2(t) \tag{3-360}$$

$$\frac{dx_2(t)}{dt} = g - \frac{1}{M}\frac{x_3^2(t)}{x_1(t)} \tag{3-361}$$

$$\frac{dx_3(t)}{dt} = -\frac{R}{L}x_3(t) + \frac{1}{L}e(t) \tag{3-362}$$

Let us linearize the system about the equilibrium point $y_0(t) = x_{01}$ = constant. Then,

$$x_{02}(t) = \frac{dx_{01}(t)}{dt} = 0 \tag{3-363}$$

$$\frac{d^2 y_0(t)}{dt^2} = 0 \tag{3-364}$$

The nominal value of *i*(*t*) is determined by substituting Eq. (3-364) into Eq. (3-359)

$$e(t) = Ri(t) + L\frac{di(t)}{dt} \tag{3-365}$$

Thus,

$$i_0(t) = x_{03}(t) = \sqrt{Mgx_{01}} \tag{3-366}$$

The linearized state equation is expressed in the state space form, with the coefficient matrices **A*** and **B*** evaluated as

$$\mathbf{A}* = \begin{bmatrix} 0 & 1 & 0 \\ \dfrac{x_{03}^2}{Mx_{01}^2} & 0 & \dfrac{-2x_{03}}{Mx_{01}} \\ 0 & 0 & -\dfrac{R}{L} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ \dfrac{g}{x_{01}} & 0 & -2\left(\dfrac{g}{Mx_{01}}\right)^{1/2} \\ 0 & 0 & -\dfrac{R}{L} \end{bmatrix} \qquad (3\text{-}367)$$

$$\mathbf{B}* = \begin{bmatrix} 0 \\ 0 \\ \dfrac{1}{L} \end{bmatrix} \qquad (3\text{-}368)$$

3-10 SUMMARY

Two most common tools for solving the differential equations representing dynamic systems are the transfer function and the state-variable methods. Transfer functions are based on the Laplace transform technique and are valid only for linear time-invariant systems, whereas the state equations can be applied to linear as well as nonlinear systems.

In this chapter, we started with differential equations, and how the Laplace transform is used for the solution of linear ordinary differential equations. This transform method is characterized by first transforming the real-domain equations into algebraic equations in the transform domain. The solutions are first obtained in the transform domain by using the familiar methods of solving algebraic equations. The final solution in the real domain is obtained by taking the inverse transform. For engineering problems, the transform tables and the partial-fraction expansion method are recommended for the inverse transformation. Throughout, we introduced various MATLAB toolboxes to find the solution of differential equations and to plot their corresponding time responses.

This chapter was presented the state space modeling of linear time-invariant differential equations. We further provided solution to the state equations using Laplace transform technique. The relationship between the state equations and transfer functions was also established. We finally demonstrated that given the transfer function of a linear system, the state equations of the system can be obtained by decomposition of the transfer function.

Later in Chaps. 7 to 11 we will provide more examples on modeling of physical systems that will utilize these subjects. Further in Chaps. 7 and 8, we will provide more details on solution and time response of differential equations using the Laplace transform and the state-space approaches, respectively.

REFERENCES

1. F. B. Hildebrand, *Methods of Applied Mathematics*, 2nd Ed., Prentice Hall, Englewood Cliffs, NJ, 1965.
2. B. C. Kuo, *Linear Networks and Systems*, McGraw-Hill Book Company, New York, 1967.
3. C. R. Wylie, Jr., *Advanced Engineering Mathematics*, 2nd Ed., McGraw-Hill Book Company, New York, 1960.
4. C. Pottle, “On the Partial Fraction Expansion of a Rational Function with Multiple Poles by Digital Computer,” *IEEE Trans. Circuit Theory*, Vol. CT-11, 161–162, Mar. 1964.
5. B. O. Watkins, “A Partial Fraction Algorithm,” *IEEE Trans. Automatic Control*, Vol. AC-16, 489–491, Oct. 1971.
6. William J. Palm III, *Modeling, Analysis, and Control of Dynamic Systems*, 2nd Ed., John Wiley & Sons, Inc., New York, 1999.
7. Katsuhiko Ogata, *Modern Control Engineering*, 5th Ed., Prentice Hall, ***New Jersey***, 2010.
8. Richard C. Dorf and Robert H. Bishop, *Modern Control Systems*, 12th Ed., Prentice Hall, NJ, 2011.
9. Norman S. Nise, *Control Systems Engineering*, 6th Ed., John Wiley and Sons, New York, 2011.
10. Gene F. Franklin, J. David Powell, and Abbas Emami-Naeini, *Feedback Control of Dynamic Systems*, 6th Ed., Prentice Hall, *New Jersey*, 2009.
11. J. Lowen Shearer, Bohdan T. Kulakowski, John F. Gardner, *Dynamic Modeling and Control of Engineering Systems*, 3rd Ed., Cambridge University Press, New York, 2007.

12. Robert L. Woods and Kent L. Lawrence, *Modeling and Simulation of Dynamic Systems*, Prentice Hall, *New Jersey*, 1997.
13. Benjamin C. Kuo, *Automatic Control Systems*, 7th Ed., Prentice Hall, *New Jersey*, 1995.
14. Benjamin C. Kuo and F. Golnaraghi, *Automatic Control Systems*, 8TH ED., JOHN WILEY & SONS, NEW YORK, 2003.
15. F. Golnaraghi and Benjamin C. Kuo, *Automatic Control Systems*, 9th Ed., John Wiley & Sons, New York, 2010.
16. Daniel J. Inman, *Engineering Vibration*, 3rd Ed., Prentice Hall, New York, 2007.

PROBLEMS

PROBLEMS FOR SEC. 3-2

3-1. Find the poles and zeros of the following functions (including the ones at infinity, if any). Mark the finite poles with × and the finite zeros with o in the *s*-plane.

(a) $G(s)=\dfrac{10(s+2)}{s^2(s+1)(s+10)}$

(b) $G(s)=\dfrac{10s(s+1)}{(s+2)(s^2+3s+2)}$

(c) $G(s)=\dfrac{10(s+2)}{s(s^2+2s+2)}$

(d) $G(s)=\dfrac{e^{-2s}}{10s(s+1)(s+2)}$

3-2. Poles and zeros of a function are given; find the function:

(a) Simple poles: 0, −2; poles of order 2: −3; zeros: –1, ∞

(b) Simple poles: −1, −4; zeros: 0

(c) Simple poles: −3, ∞; poles of order 2: 0, −1; zeros: ±*j*, ∞

3-3. Use MATLAB to find the poles and zeros of the functions in Prob. 2-1.

3-4. Use MATLAB to obtain $L\{\sin^2 2t\}$. Then, calculate $L\{\cos^2 2t\}$ when you know $L\{\sin^2 2t\}$. Verify your answer by calculating $L\{\cos^2 2t\}$ in MATLAB.

3-5. Find the Laplace transforms of the following functions. Use the theorems on Laplace transforms, if applicable.

(a) $g(t)=5te^{-5t}u_s(t)$

(b) $g(t)=(t\sin 2t+e^{-2t})u_s(t)$

(c) $g(t)=2e^{-2t}\sin 2tu_s(t)$

(d) $g(t)=\sin 2t\cos 2tu_s(t)$

(e) $g(t)=\sum_{k=0}^{\infty}e^{-5kT}\delta(t-kT)$, where $\delta(t)=$ unit-impulse function

3-6. Use MATLAB to solve Prob. 3-5.

3-7. Find the Laplace transforms of the functions shown in Fig. 3P-7. First, write a complete expression for $g(t)$, and then take the Laplace transform. Let $gT(t)$ be the description of the function over the basic period and then delay $gT(t)$ appropriately to get $g(t)$. Take the Laplace transform of $g(t)$ to get the following:

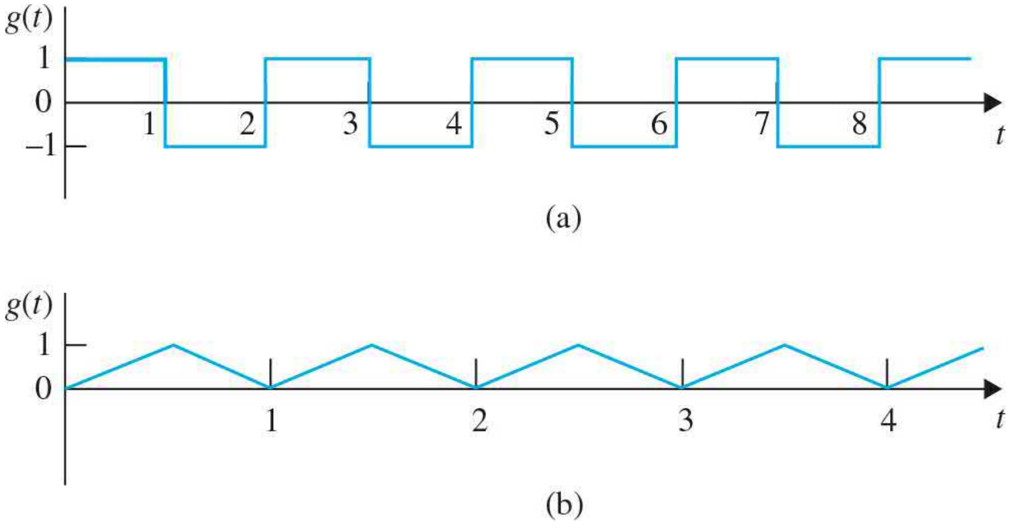


Figure 3P-7

3-8. Find the Laplace transform of the following function.

$$g(t)=\begin{cases} t+1 & 0\le t<1 \\ 0 & 1\le t<2 \\ 2-t & 2\le t<3 \\ 0 & t\ge 3 \end{cases}.$$

3-9. Find the Laplace transform of the periodic function in Fig. 3P-9.

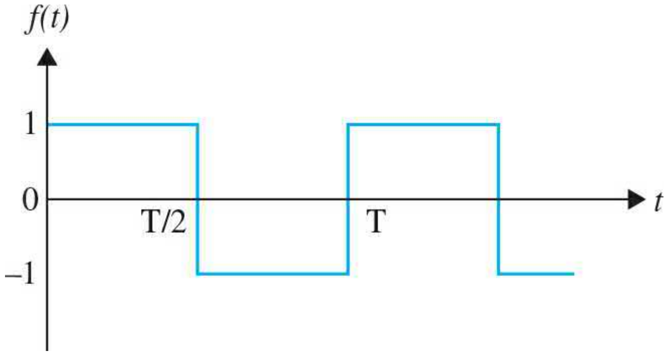


Figure 3P-9

3-10. Find the Laplace transform of the function in Fig. 3P-10.

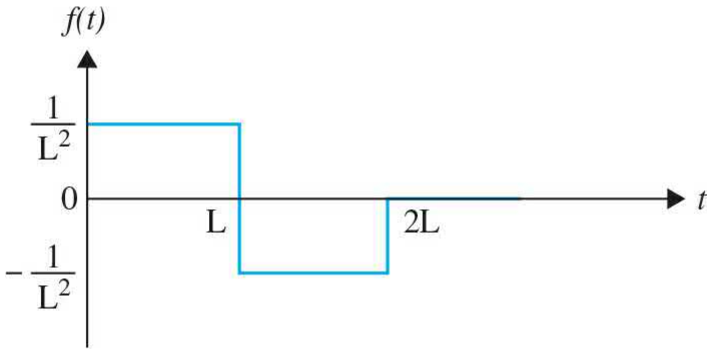


Figure 3P-10

3-11. The following differential equations represent linear time-invariant systems, where *r*(*t*) denotes the input and *y*(*t*) the output. Find the transfer function *Y*(*s*)/*R*(*s*) for each of the systems. (Assume zero initial conditions.)

(a) $$\frac{d^3y(t)}{dt^3}+2\frac{d^2y(t)}{dt^2}+5\frac{dy(t)}{dt}+6y(t)=3\frac{dr(t)}{dt}+r(t)$$

(b) $$\frac{d^4y(t)}{dt^4}+10\frac{d^2y(t)}{dt^2}+\frac{dy(t)}{dt}+5y(t)=5r(t)$$

(c) $$\frac{d^3y(t)}{dt^3}+10\frac{d^2y(t)}{dt^2}+2\frac{dy(t)}{dt}+y(t)+2\int_0^t y(\tau)d\tau=\frac{dr(t)}{dt}+2r(t)$$

(d) $$2\frac{d^2y(t)}{dt^2}+\frac{dy(t)}{dt}+5y(t)=r(t)+2r(t-1)$$

(e) $$\frac{d^2y(t+1)}{dt^2}+4\frac{dy(t+1)}{dt}+5y(t+1)=\frac{dr(t)}{dt}+2r(t)+2\int_{-\infty}^t r(\tau)d\tau$$

(f) $$\frac{d^3y(t)}{dt^2}+2\frac{d^2y(t)}{dt^2}+\frac{dy(t)}{dt}+2y(t)+2\int_{-\infty}^t y(\tau)d\tau=\frac{dr(t-2)}{dt}+2r(t-2)$$

3-12. Use MATLAB to find $Y(s)/R(s)$ for the differential equations in Prob. 2-29.

PROBLEMS FOR SEC. 3-3

3-13. Find the inverse Laplace transforms of the following functions. First, perform partial-fraction expansion on $G(s)$; then, use the Laplace transform table.

(a) $$G(s)=\frac{1}{s(s+2)(s+3)}$$

(b) $$G(s)=\frac{10}{(s+1)^2(s+3)}$$

(c) $$G(s)=\frac{100(s+2)}{s(s^2+4)(s+1)}e^{-s}$$

(d) $$G(s)=\frac{2(s+1)}{s(s^2+s+2)}$$

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

(f) $$G(s)=\frac{2(s^2+s+1)}{s(s+1.5)(s^2+5s+5)}$$

(g) $$G(s)=\frac{2+2se^{-s}+4e^{-2s}}{s^2+3s+2}$$

(h) $$G(s)=\frac{2s+1}{s^3+6s^2+11s+6}$$

(i) $$G(s)=\frac{3s^3+10s^2+8s+5}{s^4+5s^3+7s^2+5s+6}$$

3-14. Use MATLAB to find the inverse Laplace transforms of the functions in Prob. 3-13. First, perform partial-fraction expansion on *G*(*s*); then, use the inverse Laplace transform.

3-15. Use MATLAB to find the partial-fraction expansion to the following functions.

(a) $G(s)=\dfrac{10(s+1)}{s^2(s+4)(s+6)}$

(b) $G(s)=\dfrac{(s+1)}{s(s+2)(s^2+2s+2)}$

(c) $G(s)=\dfrac{5(s+2)}{s^2(s+1)(s+5)}$

(d) $G(s)=\dfrac{5e^{-2s}}{(s+1)(s^2+s+1)}$

(e) $G(s)=\dfrac{100(s^2+s+3)}{s(s^2+5s+3)}$

(f) $G(s)=\dfrac{1}{s(s^2+1)(s+0.5)^2}$

(g) $G(s)=\dfrac{2s^3+s^2+8s+6}{(s^2+4)(s^2+2s+2)}$

(h) $G(s)=\dfrac{2s^4+9s^3+15s^2+s+2}{s^2(s+2)(s+1)^2}$

3-16. Use MATLAB to find the inverse Laplace transforms of the functions in 3-15.

PROBLEMS FOR SEC. 3-4

3-17. Solve the following differential equations by means of the Laplace transform.

(a) $\dfrac{d^2f(t)}{dt^2}+5\dfrac{df(t)}{dt}+4f(t)=e^{-2t}u_s(t)$ (Assume zero initial conditions.)

(b) $$\begin{cases} \dfrac{dx_1(t)}{dt} = x_2(t) \\ \dfrac{dx_2(t)}{dt} = -2x_1(t) - 3x_2(t) + us(t) x_1(0) = 1, x_2(0) = 0 \end{cases}$$

(c) $$\begin{cases} \dfrac{d^3y(t)}{dt^2} + 2\dfrac{d^2y(t)}{dt^2} + \dfrac{dy(t)}{dt} + 2y(t) = -e^{-t}u_s(t) \\ \dfrac{d^2y}{dt^2}(0) = -1 \, \dfrac{dy}{dt}(0) = 1 \; y(0) = 0 \end{cases}$$

3-18. Use MATLAB to find the Laplace transform of the functions in Prob. 3-17.

3-19. Use MATLAB to solve the following differential equation:

$$\frac{d^2y}{dt^2} - y = e^t \quad \text{(Assuming zero initial conditions.)}$$

3-20. A series of a three-reactor tank is arranged as shown in Fig. 3P-20 for chemical reaction.

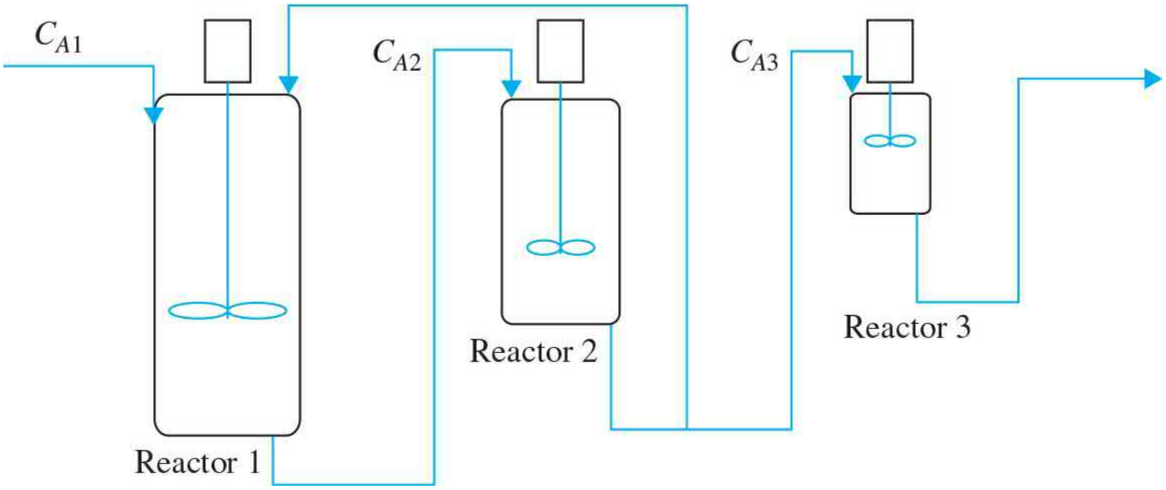


Figure 3P-20

The state equation for each reactor is defined as follows:

$$R1: \quad \frac{dC_{A1}}{dt} = \frac{1}{V_1}[1000 + 100C_{A2} - 1100C_{A1} - k_1 V_1 C_{A1}]$$

$$R2: \quad \frac{dC_{A2}}{dt} = \frac{1}{V_2}[1100C_{A1} - 1100C_{A2} - k_2 V_2 C_{A2}]$$

$$R3: \quad \frac{dC_{A3}}{dt} = \frac{1}{V_3}[1000C_{A2} - 1000C_{A3} - k_3 V_3 C_{A3}]$$

when V_i and k_i represent the volume and the temperature constant of each tank as shown in the following table:

Reactor	V_i	k_i
1	1,000	0.1
2	1,500	0.2
3	100	0.4

Use MATLAB to solve the differential equations assuming $C_{A1} = C_{A2} = C_{A3} = 0$ at $t = 0$.

PROBLEMS FOR SEC. 3-5

3-21. Figure 3P-21 shows a simple model of a vehicle suspension system hitting a bump. If the mass of wheel and its mass moment of inertia are m and J, respectively, then

(a) Find the equation of the motion.

(b) Determine the transfer function of the system.

(c) Calculate its natural frequency.

(d) Use MATLAB to plot the step response of the system.

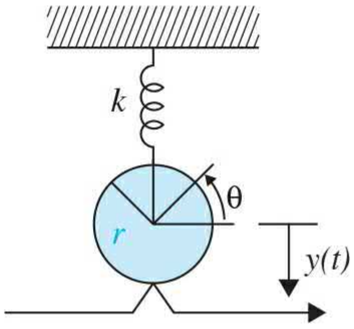


Figure 3P-21

3-22. An electromechanical system has the following system equations.

$$L\frac{di}{dt}+R1+K_1\omega=e(t)$$

$$J\frac{dw}{dt}+Bw-K_2 i=0$$

For a unit-step applied voltage $e(t)$ and zero initial conditions, find responses $i(t)$ and $\omega(t)$. Assume the following parameter values:

$$L=1HmJ=kg\ m^2,\ B=2N\ m\ s,\ R=1\ \Omega,\ K_1=1\ V\ s,\ K_2=1\ N\ m/A.$$

3-23. Consider the two-degree-of-freedom mechanical system shown in Fig. 3P-23, subjected to two applied forces, $f_1(t)$ and $f_2(t)$, and zero initial conditions. Determine system responses $x_1(t)$ and $x_2(t)$ when

(a) $f_1(t) = 0$, $f_2(t) = u_s,(t)$

(b) $f_1(t) = u_s(t)$, $f_2(t) = u_s,(t)$.

Use the following parameter values:

$$m_1=m_2=1\ kg,\ b_1=2Ns/m,\ b_2=1Ns/m,\ k_1=k_2=1N/m\ ..$$

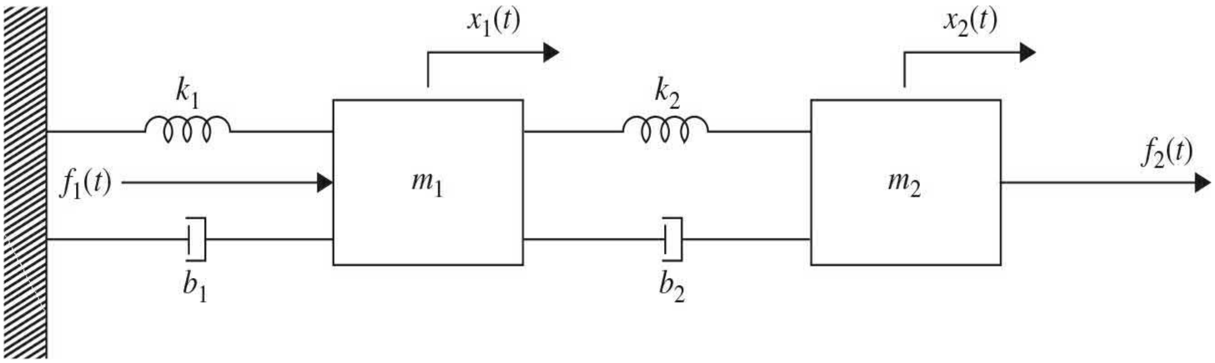


Figure 3P-23

PROBLEMS FOR SECs. 3-6 AND 3-7

3-24. Express the following set of first-order differential equations in the vector-matrix form of $\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$.

(a)
$$\frac{dx_1(t)}{dt} = -x_1(t) + 2x_2(t)$$
$$\frac{dx_2(t)}{dt} = -2x_2(t) + 3x_3(t) + u_1(t)$$
$$\frac{dx_3(t)}{dt} = -x_1(t) - 3x_2(t) - x_3(t) + u_2(t)$$

(b)
$$\frac{dx_1(t)}{dt} = -x_1(t) + 2x_2(t) + 2u_1(t)$$
$$\frac{dx_2(t)}{dt} = 2x_1(t) - x_3(t) + u_2(t)$$
$$\frac{dx_3(t)}{dt} = 3x_1(t) - 4x_2(t) - x_3(t)$$

3-25. Given the state equation of the system, convert it to the set of first-order differential equation.

(a)
$$A = \begin{bmatrix} 0 & -1 & 2 \\ 1 & 0 & 1 \\ -1 & -2 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 0 & -1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}$$

(b)
$$A = \begin{bmatrix} 3 & 1 & -2 \\ -1 & 2 & 2 \\ 0 & 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} -1 \\ 0 \\ 2 \end{bmatrix}$$

3-26. Consider a train consisting of an engine and a car, as shown in Fig. 3P-26.

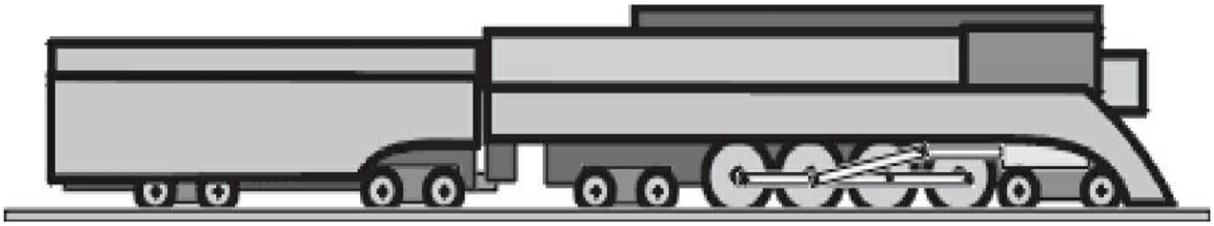


Figure 3P-26

A controller is applied to the train so that it has a smooth start and stop, along with a constant-speed ride. The mass of the engine and the car are M and m, respectively. The two are held together by a spring with the stiffness coefficient of K. F represents the force applied by the engine, and μ represents the coefficient of rolling friction. If the train only travels in one direction:

(a) Draw the free-body diagram.

(b) Find the state variables and output equations.

(c) Find the transfer function.

(d) Write the state space of the system.

3-27. A vehicle towing a trailer through a spring-damper coupling hitch is shown in Fig. 3P-27. The following parameters and variables are defined: M is the mass of the trailer; K_h, the spring constant of the hitch; B_h, the viscous-damping coefficient of the hitch; B_t, the viscous-friction coefficient of the trailer; $y_1(t)$, the displacement of the towing vehicle; $y_2(t)$, the displacement of the trailer; and $f(t)$, the force of the towing vehicle.

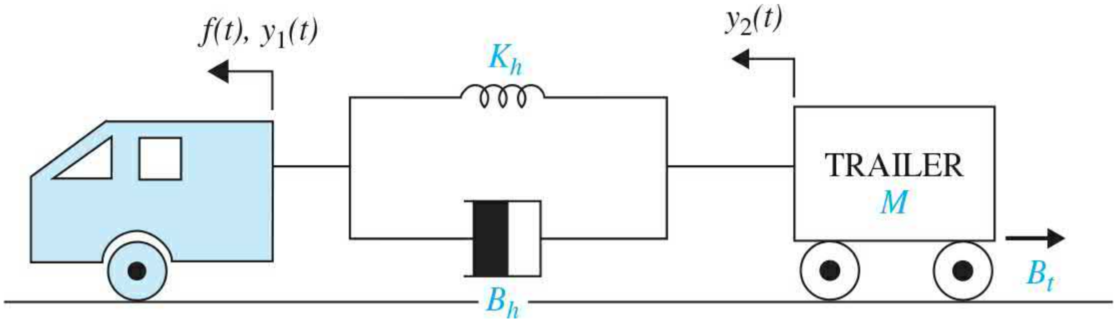


Figure 3P-27

(a) Write the differential equation of the system.

(b) Write the state equations by defining the following state variables: $x_1(t) = y_1(t) – y_2(t)$ and $x_2(t) = dy_2(t)dt$.

3-28. Figure 3P-28 shows a well-known "ball and beam" system in control systems. A ball is located on a beam to roll along the length of the beam. A lever arm is attached to the one end of the beam and a servo gear is attached to the other end of the lever arm. As the servo gear turns by an angle θ, the lever arm goes up and down, and then the angle of the beam is changed by α. The change in angle causes the ball to roll along the beam. A controller is desired to manipulate the ball's position.

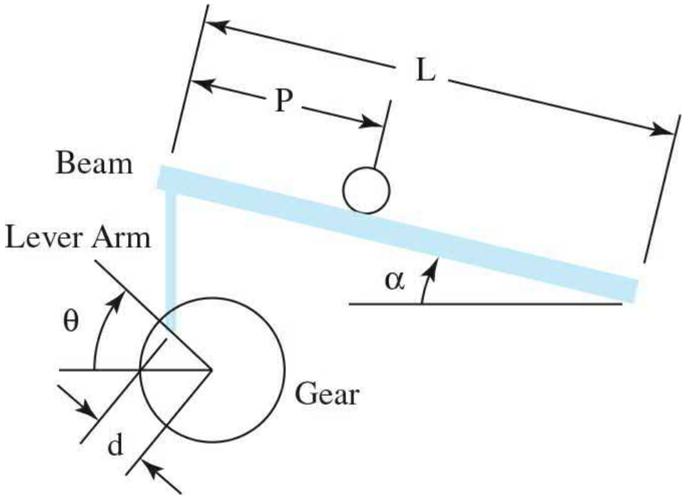


Figure 3P-28

Assuming:

$m =$ mass of the ball

$r =$ radius of the ball

$d =$ lever arm offset

$g =$ gravitational acceleration

$L =$ length of the beam

$J =$ ball's moment of inertia

$p =$ ball position coordinate

$\alpha =$ beam angle coordinate

$\theta =$ servo gear angle

(a) Determine the dynamic equation of the motion.

(b) Find the transfer function.

(c) Write the state space of the system.

(d) Find the step response of the system by using MATLAB.

3-29. Find the transfer function and state-space variables in Prob. 2-12.

3-30. Find the transfer function $Y(s)/T_m(s)$ in Prob. 2-16.

3-31. The schematic diagram of a motor-load system is shown in Fig. 3P-31. The following parameters and variables are defined: $T_m(t)$ is the motor torque; $\omega_m(t)$, the motor velocity; $\theta_m(t)$, the motor displacement; $\omega_L(t)$, the load velocity; $\theta_L(t)$, the load displacement; K, the torsional spring constant; J_m, the motor inertia; B_m, the motor viscous-friction coefficient; and B_L, the load viscous-friction coefficient.

(a) Write the torque equations of the system.

(b) Find the transfer functions $\Theta_L(s)/T_m(s)$ and $\Theta_m(s)/T_m(s)$.

(c) Find the characteristic equation of the system.

(d) Let $T_m(t) = T_m$ be a constant applied torque; show that $\omega_m = \omega_L =$ constant in the steady state. Find the steady-state speeds ω_m and ω_L.

(e) Repeat part (d) when the value of J_L is doubled, but J_m stays the same.

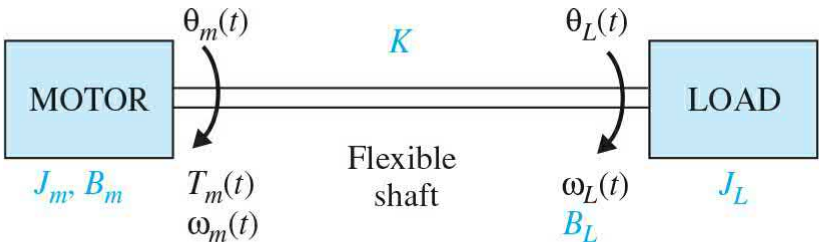


Figure 3P-31

3-32. In Prob. 2-20,

(a) Assume that T_s is a constant torque. Find the transfer function $\Theta(s)/\Delta(s)$, where $\Theta(s)$ and $\Delta(s)$ are the Laplace transforms of $\theta(t)$ and $\zeta(t)$, respectively. Assume that $\zeta(t)$ is very small.

(b) Repeat part (a) with points C and P interchanged. The d_1 in the expression of α_F should be changed to d_2.

3-33. In Prob. 2-21,

(a) Express the equations obtained in earlier as state equations by assigning the state variables as $x_1 = \theta$, $x_2 = d\theta/dt, x_3 = x$ and $x_4 = dx/dt$. Simplify these equations for small θ by making the approximations $\sin\theta \cong \theta$ and $\cos\theta \cong 1$.

(b) Obtain a small-signal linearized state-equation model for the system in the form of

$$\frac{d\Delta\mathbf{x}(t)}{dt} = \mathbf{A}^*\Delta\mathbf{x}(t) + \mathbf{B}^*\Delta\mathbf{r}(t)$$

at the equilibrium point $x_{01}(t) = 1, x_{02}(t) = 0, x_{03}(t) = 0$, and $x_{04}(t) = 0$.

3-34. Vibration absorbers are used to protect machines that work at the constant speed from steady-state harmonic disturbance. Figure 3P-34 shows a simple vibration absorber.

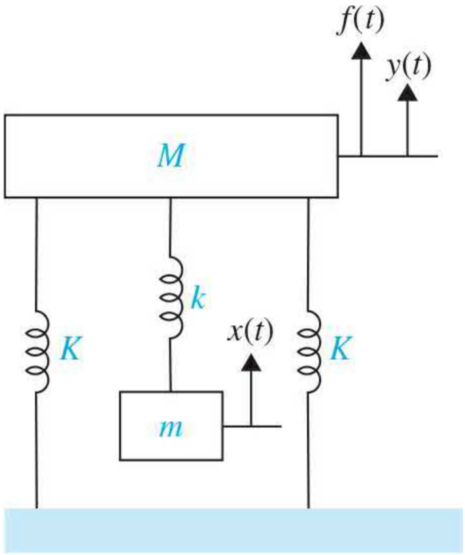


Figure 3P-34

Assuming the harmonic force $F(t) = A\sin(\omega t)$ is the disturbance applied to the mass M:

(a) Derive the state space of the system.

(b) Determine the transfer function of the system.

3-35. Figure 3P-35 represents a damping in the vibration absorption.

Assuming the harmonic force $F(t) = A\sin(\omega t)$ is the disturbance applied to the mass M:

(a) Derive the state space of the system.

(b) Determine the transfer function of the system.

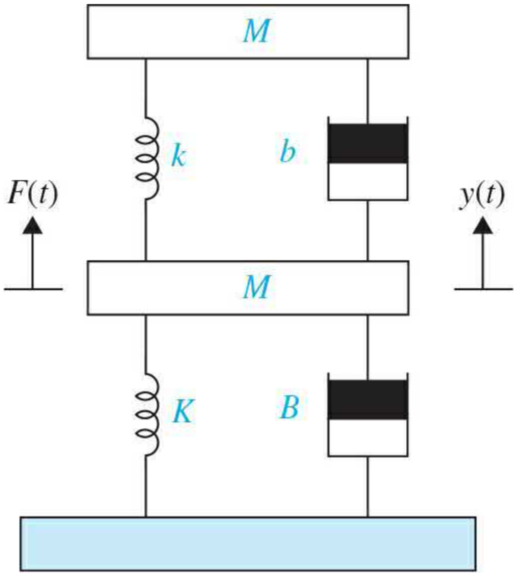


Figure 3P-35

3-36. Consider the electrical circuits shown in Fig. 3P-36*a* and *b*.

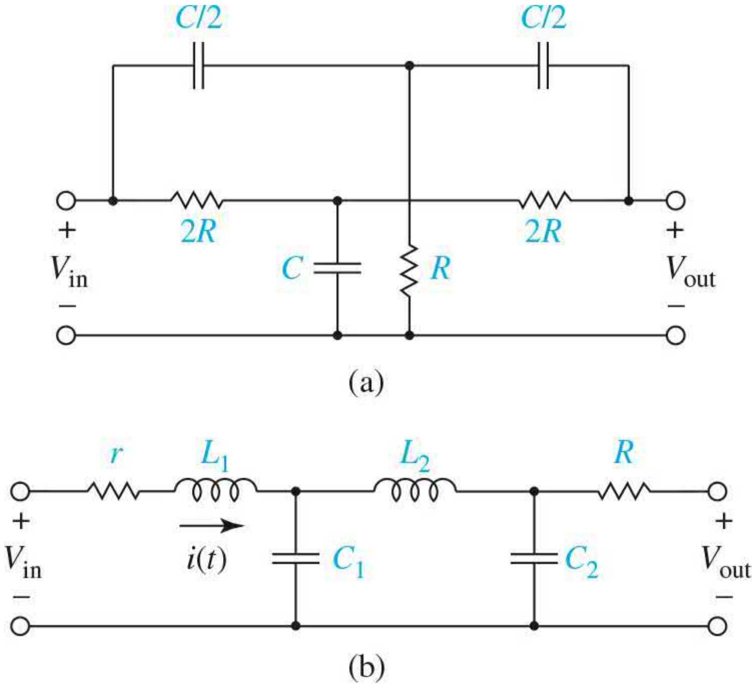


Figure 3P-36

For each circuit:

(a) Find the dynamic equations and state variables.

(b) Determine the transfer function.

(c) Use MATALB to plot the step response of the system.

3-37. The following differential equations represent linear time-invariant systems. Write the dynamic equations (state equations and output equations) in vector-matrix form.

(a) $\dfrac{d^2y(t)}{dt^2}+4\dfrac{dy(t)}{dt}+y(t)=5r(t)$

(b) $2\dfrac{d^3y(t)}{dt^3}+3\dfrac{d^2y(t)}{dt^2}+5\dfrac{dy(t)}{dt}+2y(t)=r(t)$

(c) $\dfrac{d^3y(t)}{dt^3}+5\dfrac{d^2y(t)}{dt^2}+3\dfrac{dy(t)}{dt}+y(t)+\int_0^t y(\tau)d\tau=r(\tau)$

(d) $\dfrac{d^4y(t)}{dt^4}+1.5\dfrac{d^3y(t)}{dt^3}+2.5\dfrac{dy(t)}{dt}+y(t)=2r(t)$

3-38. The following transfer functions show linear time-invariant systems. Write the dynamic equations (state equations and output equations) in vector-matrix form.

(a) $G(s)=\dfrac{s+3}{s^2+3s+2}$

(b) $G(s)=\dfrac{6}{s^3+6s^2+11s+6}$

(c) $G(s)=\dfrac{s+2}{s^2+7s+12}$

(d) $G(s)=\dfrac{s^3+11s^2+35s+250}{s^2(s^3+4s^2+39s+108)}$

3-39. Repeat Prob. 3-38 by using MATLAB.

3-40. Find the time response of the following systems:

(a) $\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\begin{bmatrix}0 & 1\\ -2 & -3\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}+\begin{bmatrix}0\\ 1\end{bmatrix}u$

(b) $\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\begin{bmatrix}-1 & -0.5\\ 1 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}+\begin{bmatrix}0.5\\ 0\end{bmatrix}uy=\begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}$

4-41. Given a system described by the dynamic equations:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \quad y(t) = \mathbf{C}\mathbf{x}(t)$$

(a) $\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -2 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$

(b) $\mathbf{A} = \begin{bmatrix} -1 & 1 \\ 0 & -1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 1 \end{bmatrix}$

(c) $\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & -2 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$

(1) Find the eigenvalues of **A**.

(2) Find the transfer-function relation between **X**(s) and $U(s)$.

(3) Find the transfer function $Y(s)/U(s)$.

3-42. Given the dynamic equations of a time-invariant system:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \quad y(t) = \mathbf{C}\mathbf{x}(t)$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -2 & -3 \end{bmatrix} \quad \mathrm{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \mathrm{C} = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$$

Find the matrices $\mathbf{A}_1$ and $\mathbf{B}_1$ so that the state equations are written as

$$\frac{d\overline{\mathrm{x}}(t)}{dt} = \mathrm{A}_1\overline{\mathbf{x}}(t) + \mathbf{B}_1 u(t)$$

where

$$\bar{\mathbf{x}}(t) = \begin{bmatrix} x_1(t) \\ y(t) \\ \dfrac{dy(t)}{dt} \end{bmatrix}$$

3-43. Figure 3P-43*a* shows a well-known "broom-balancing" system in control systems. The objective of the control system is to maintain the broom in the upright position by means of the force $u(t)$ applied to the car as shown. In practical applications, the system is analogous to a one-dimensional control problem of the balancing of a unicycle or a missile immediately after launching. The free-body diagram of the system is shown in Fig. 3P-43*b*, where

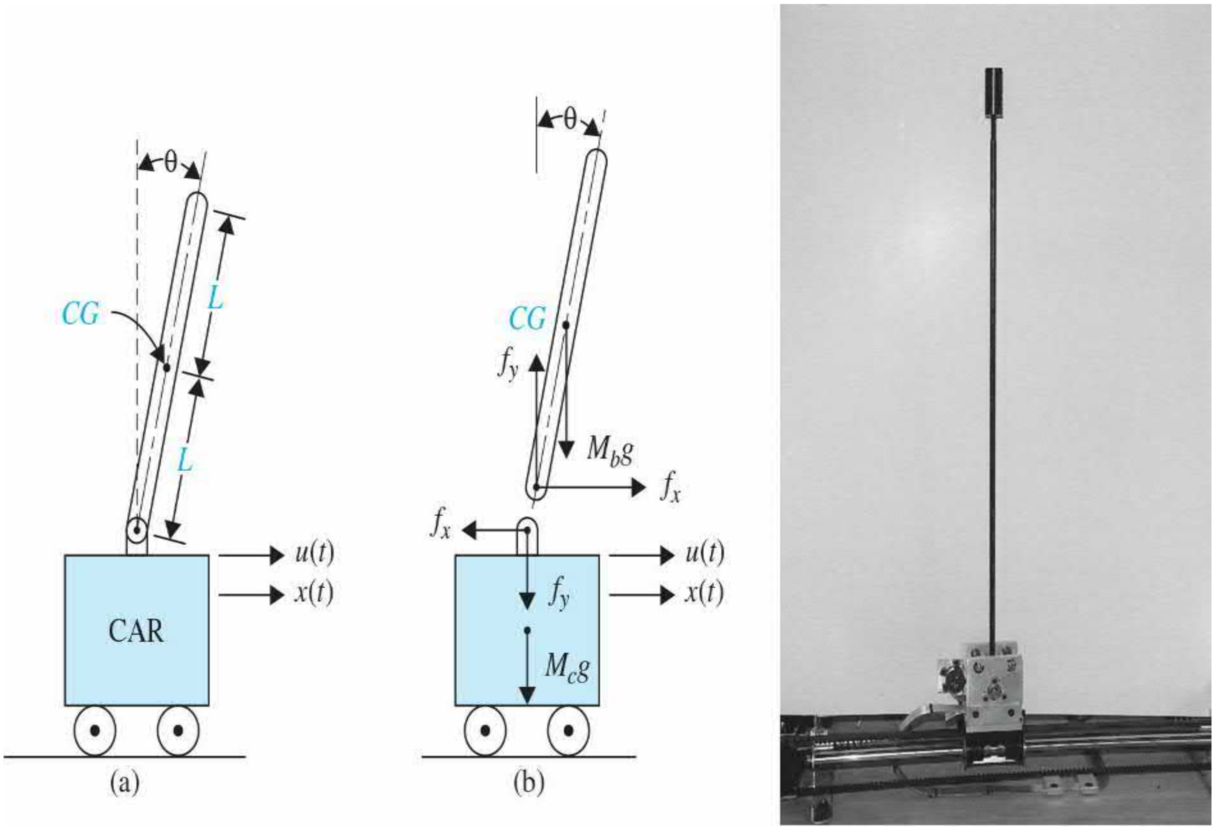


Figure 3P-43

f_x = force at broom base in horizontal direction

f_y = force at broom base in vertical direction

M_b = mass of broom

g = gravitational acceleration

M_c = mass of car

J_b = moment of inertia of broom about center of gravity $CG = M_b L_2/3$

(a) Write the force equations in the x and the y directions at the pivot point of the broom. Write the torque equation about the center of gravity CG of the broom. Write the force equation of the car in the horizontal direction.

(b) Express the equations obtained in part (a) as state equations by assigning the state variables as $x_1 = \theta$, $x_2 = d\theta/dt$,$x_3 = x$ and $x_4 = dx/dt$. Simplify these equations for small θ by making the approximations $\theta \cong \theta$ and $\cos\theta \cong 1$.

(c) Obtain a small-signal linearized state-equation model for the system in the form of

$$\frac{d\Delta\mathbf{x}(t)}{dt} = \mathbf{A}^*\,\Delta\mathbf{x}(t) + \mathbf{B}^*\,\Delta\mathbf{r}(t)$$

at the equilibrium point $x_{01}(t) = 1$,$x_{02}(t) = 0$,$x_{03}(t) = 0$, and $x_{04}(t) = 0$.

3-44. The “broom-balancing” control system described in Prob. 3-43 has the following parameters:

$$M_b = 1\,\text{kg} \quad M_c = 10\,\text{kg} \quad L = 1\,\text{m} \quad g = 32.2\,\text{ft/s}^2$$

The small-signal linearized state equation model of the system is

$$\Delta\dot{\mathbf{x}}(t) = \mathbf{A}^*\,\Delta\mathbf{x}(t) + \mathbf{B}^*\,\Delta r(t)$$

where

$$\mathbf{A}^* = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 25.92 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -2.36 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{B}^* = \begin{bmatrix} 0 \\ -0.0732 \\ 0 \\ 0.0976 \end{bmatrix}$$

Find the characteristic equation of **A*** and its roots.

3-45. Figure 3P-45 shows the schematic diagram of a ball-suspension control system. The steel ball is suspended in the air by the electromagnetic force generated by the electromagnet. The objective of the control is to keep the metal ball suspended at the nominal equilibrium position by controlling the current in the magnet with the voltage $e(t)$. The practical application of this system is the magnetic levitation of trains or magnetic bearings in high-precision control systems. The resistance of the coil is R, and the inductance is $L(y) = L/y(t)$, where L is a constant. The applied voltage $e(t)$ is a constant with amplitude E.

(a) Let E_{eq} be a nominal value of E. Find the nominal values of $y(t)$ and $dy(t)/dt$ at equilibrium.

(b) Define the state variables at $x_1(t) = i(t), x_2(t) = y(t)$, and $x_3(t) = dy(t)/dt$. Find the nonlinear state equations in the form of $\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(\mathbf{x}, e)$.

(c) Linearize the state equations about the equilibrium point and express the linearized state equations as

$$\frac{d\Delta\mathbf{x}(t)}{dt} = \mathbf{A}^* \,\Delta\mathbf{x}(t) + \mathbf{B}^* \,\Delta e(t)$$

The force generated by the electromagnet is $Ki^2(t)/y(t)$, where K is a proportional constant, and the gravitational force on the steel ball is Mg.

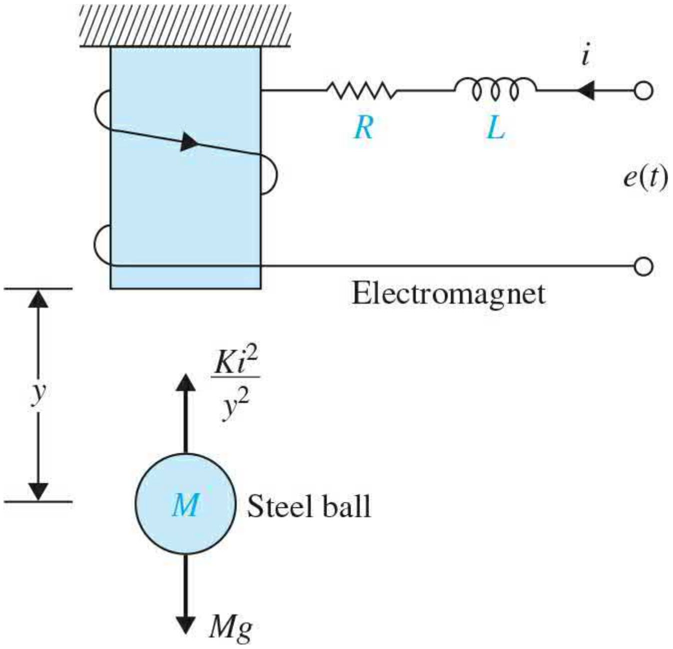


Figure 3P-45

3-46. The linearized state equations of the ball-suspension control system described in Prob. 3-45 are expressed as

$$\Delta\dot{\mathbf{x}}(t) = \mathbf{A}^* \, \Delta\mathbf{x}(t) + \mathbf{B}^* \, \Delta i(t)$$

where

$$\mathbf{A}^* = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 115.2 & -0.05 & -18.6 & 0 \\ 0 & 0 & 0 & 1 \\ -37.2 & 0 & 37.2 & -0.1 \end{bmatrix} \quad \mathbf{B}^* = \begin{bmatrix} 0 \\ -6.55 \\ 0 \\ -6.55 \end{bmatrix}$$

Let the control current $\Delta i(t)$ be derived from the state feedback $\Delta i(t) = -\mathbf{K}\Delta\mathbf{x}(t)$, where

$$\mathbf{K} = \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \end{bmatrix}$$

(a) Find the elements of **K** so that the eigenvalues of $\mathbf{A}^* - \mathbf{B}^*\mathbf{K}$ are at $-1 + j$, $-1 - j$, -10, and -10.

(b) Plot the responses of $\Delta x_1(t) = \Delta y_1(t)$ (magnet displacement) and $\Delta x_3(t) = \Delta y_2(t)$ (ball displacement) with the initial condition

$$\Delta \mathbf{x}(0) = \begin{bmatrix} 0.1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

(c) Repeat part (b) with the initial condition

$$\Delta \mathbf{x}(0) = \begin{bmatrix} 0 \\ 0 \\ 0.1 \\ 0 \end{bmatrix}$$

Comment on the responses of the closed-loop system with the two sets of initial conditions used in (b) and (c).

[1] Strictly speaking, the one-sided Laplace transform should be defined from $t = 0^-$ to $t = \infty$. The symbol $t = 0^-$ implies the limit of $t \to 0$ is taken from the left side of $t = 0$. For simplicity, we shall use $t = 0$ or $t = t_0$ (≥ 0) as the initial time in all subsequent discussions. A Laplace transform table is given in App. C.

[2] The transfer function in Eq. (3-20) is said to be strictly proper if the order of the denominator polynomial is greater than that of the numerator polynomial (i.e., $n > m$). If $n = m$, the transfer function is called *proper*. The transfer function is improper if $m > n$.

[3] The definition of a **pole** can be stated as: *If a function* G(s) *is analytic and single-valued in the neighborhood of point* p_i, *it is said to have a pole of order* r *at* $s = p_i$ *if the limit* $\lim_{s \to p_i}[(s - p_i)^r G(s)]$ *has a finite, nonzero value*. In other words, the denominator of $G(s)$ must include the factor $(s - p_i)^r$, so when $s = p_i$, the function becomes infinite. If r =1, the pole at $s = p_i$ is called a **simple pole**.

[4] The definition of a **zero** of a function can be stated as: *If the function* G(s) *is analytic at* $s = Z_i$, *it is said to have a zero of order* r *at* $s = Z_i$ *if the limit* $\lim_{s \to z_i}[(s - z_i)^{-r} G(s)]$ *has a finite, nonzero value. Or, simply,* G(s) *has a zero of order* r *at* $s = Z_i$ *if* 1/G(s) *has an* r*th-order pole at* $s = Z_i$.

CHAPTER 4

Block Diagrams and Signal-Flow Graphs

In Chap. 2, we studied the modeling of basic dynamic systems, and later in Chap. 3 we utilized transfer function and state space methods to convert these models from differential equation representation into formats more suitable for control system analysis. In this chapter, we introduce block diagrams as graphical alternatives for modeling control systems and their underlying mathematics. Block diagrams are popular in the study of control systems because they provide better understanding of the composition and interconnection of the components of a dynamic system. A signal-flow graph (SFG) may also be used as an alternative graphical representation of a control system model. SFGs may be regarded as an alternative representation of a block diagram.

Learning Outcomes

After successful completion of this chapter, you will be able to

1. Utilize block diagrams, its components, and their underlying mathematics to obtain transfer function of a control system.
2. Establish a parallel between block diagrams and signal-flow graphs.
3. Utilize signal-flow graphs and Mason’s gain formula to find transfer function of a control system.
4. Obtain the state diagrams, an extension of the SFG to portray state equations and differential equations.

In this chapter, we utilize the block diagrams and SFGs and the Mason’s gain formula to find the transfer function of the overall control system. Through case studies at the end of the chapter, we apply these techniques to the modeling of various dynamic systems that we already studied in Chaps. 2

and 3.

4-1 BLOCK DIAGRAMS

Block diagram modeling together with transfer function models describe the cause-and-effect (input-output) relationships throughout the system. For example, consider a simplified block diagram representation of the heating system in your lecture room, shown in Fig. 4-1, where by setting a desired temperature, also defined as the **input**, one can set off the furnace to provide heat to the room. The process is relatively straightforward. The actual room temperature is also known as the **output** and is measured by a **sensor** within the thermostat. A simple electronic circuit within the thermostat compares the actual room temperature to the desired room temperature (**comparator**). If the room temperature is below the desired temperature, an **error** voltage will be generated. The error voltage acts as a switch to open the gas valve and turn on the furnace (or the **actuator**). Opening the windows and the door in the classroom would cause heat loss and, naturally, would disturb the heating process (**disturbance**). The room temperature is constantly monitored by the output sensor. The process of sensing the output and comparing it with the input to establish an error signal is known as **feedback**. Note that the error voltage here causes the furnace to turn on, and the furnace would finally shut off when the error reaches zero.

> Block diagrams provide better understanding of the composition and interconnection of the components of a dynamic system.

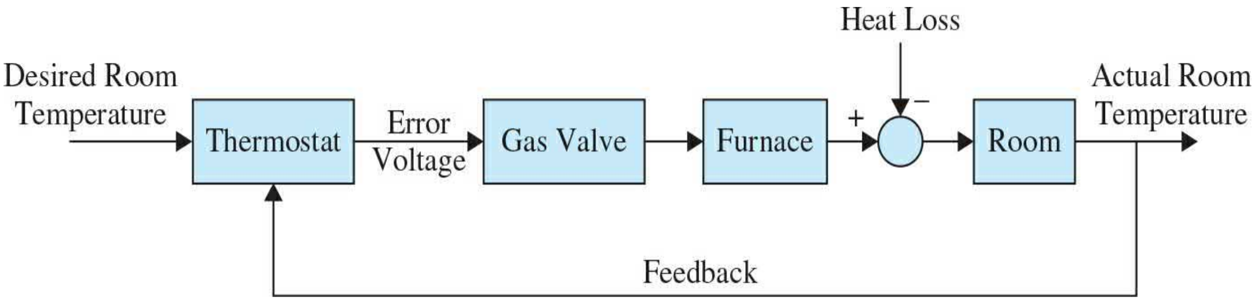


Figure 4-1 A simplified block diagram representation of a heating

system.

The block diagram in this case simply shows how the system components are interconnected, and no mathematical details are given. If the mathematical and functional relationships of all the system elements are known, the block diagram can be used as a tool for the analytic or computer solution of the system.

In general, block diagrams can be used to model linear as well as nonlinear systems. For nonlinear systems, the block diagram variables are in time domain, and **for linear systems, the Laplace transform variables are used**.

So in this case, assuming linear models for all system components, the system dynamics can be represented, in the Laplace domain, by a transfer function

$$\frac{Ti(s)}{To(s)} \tag{4-1}$$

where *Ti*(*s*) is the Laplace representation of the Desired Room Temperature and *To*(*s*) is the Actual Room Temperature, as shown in Fig. 4-1.

Alternatively, we can use signal flow graphs or state diagrams to provide a graphical representation of a control system. These topics are discussed later in this chapter.

4-1-1 Modeling of Typical Elements of Block Diagrams in Control Systems

The common elements in block diagrams of most control systems include

- Comparators
- Blocks representing individual component transfer functions, including
- Reference sensor (or input sensor)
- Output sensor
- Actuator
- Controller
- Plant (the component whose variables are to be controlled)

- Input or reference signals[1]
- Output signals
- Disturbance signal
- Feedback loops

Figure 4-2 shows one configuration where these elements are interconnected. You may wish to compare Figs. 4-1 and 4-2 to find the control terminology for each system. As a rule, each block represents an element in the control system, and each element can be modeled by one or more equation. These equations are normally in the Laplace domain (because of ease in manipulation using transfer functions), but the time representation may also be used. Once the block diagram of a system is fully constructed, one can study individual components or the overall system behavior. The key components of a block diagram are discussed next.

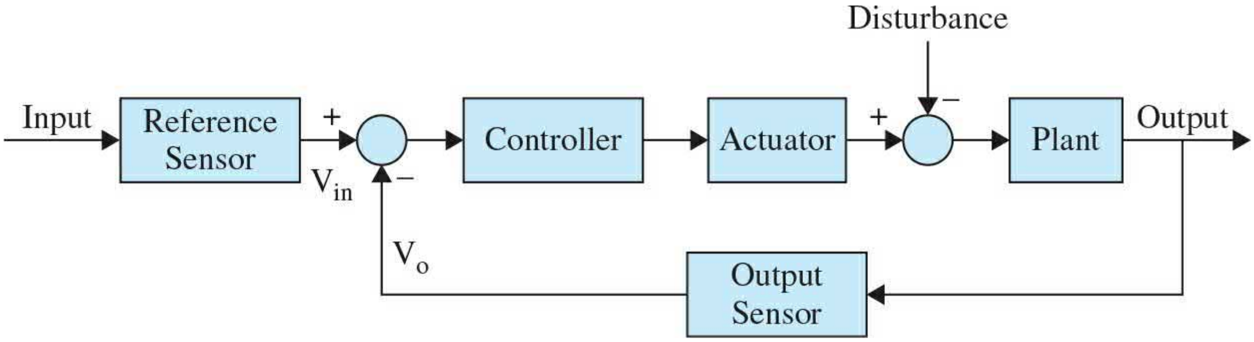
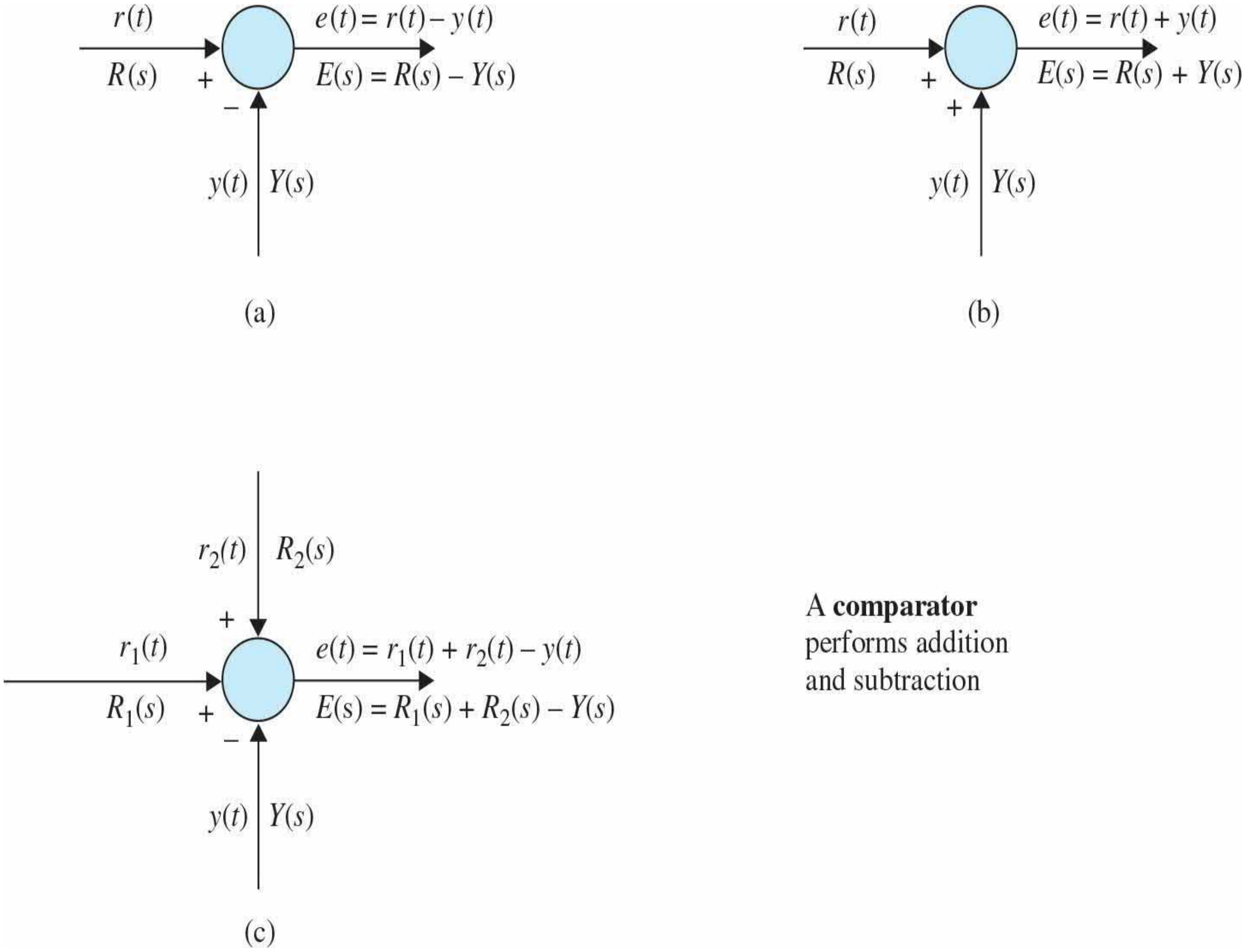


Figure 4-2 Block diagram representation of a general control system.

Comparators

One of the important components of a control system is the sensing and the electronic device that acts as a junction point for signal comparisons—otherwise known as a **comparator**. In general, these devices possess sensors and perform simple mathematical operations such as addition and subtraction (such as the thermostat in Fig. 4-1). Three examples of comparators are illustrated in Fig. 4-3. Note that the addition and subtraction operations in Fig. 4-3*a* and *b* are linear, so the input and output variables of these block diagram elements can be time-domain variables or Laplace-transform variables. Thus, in Fig. 4-3*a*, the block diagram implies



A **comparator** performs addition and subtraction

Figure 4-3 Block diagram elements of typical sensing devices of control systems. (*a*) Subtraction. (*b*) Addition. (*c*) Addition and subtraction.

$$e(t) = r(t) - y(t) \tag{4-2}$$

or

$$E(s) = R(s) - Y(s) \tag{4-3}$$

Blocks

As mentioned earlier, **blocks** represent the equations of the system in time domain or the **transfer function** of the system in the Laplace domain, as demonstrated in Fig. 4-4.

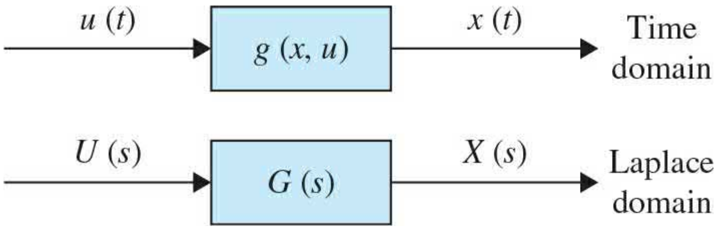


Figure 4-4 Time and Laplace domain block diagrams.

In Laplace domain, the following input-output relationship can be written for the system in Fig. 4-4:

$$E(s) = R(s) - Y(s) \tag{4-3}$$

If signal $X(s)$ is the output and signal $U(s)$ denotes the input, the transfer function of the block in Fig. 4-4 is

$$G(s) = \frac{X(s)}{U(s)} \tag{4-5}$$

Typical block elements that appear in the block diagram representation of most control systems include **plant**, **controller**, **actuator**, and **sensor**.

EXAMPLE 4-1-1 Consider the block diagram of a cascade system with transfer functions $G\mathbf{1}(s)$ and $G\mathbf{2}(s)$ that are connected in series, as shown in Fig. 4-5. The transfer function $G(s)$ of the overall system can be obtained by combining individual block equations. Hence, for variables $A(s)$ and $X(s)$, we have

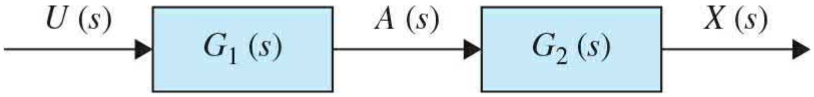


Figure 4-5 Block diagrams $G\mathbf{1}(s)$ and $G\mathbf{2}(s)$ connected in series—a cascade system.

$$X(s) = A(s)G_2(s)$$

$$A(s) = U(s)G_1(s)$$

$$X(s) = G_1(s)G_2(s)$$

$$G(s) = \frac{X(s)}{U(s)}$$

Or,

$$G(s) = G_1(s)G_2(s) \tag{4-6}$$

Using Eq. (4-6), the system in Fig. 4-5 can be represented by the system in Fig. 4-4.

EXAMPLE 4-1-2 Consider a more complicated system of two transfer functions $G_1(s)$ and $G_2(s)$ that are connected in parallel, as shown in Fig. 4-6. The transfer function $G(s)$ of the overall system can be obtained by combining individual block equations. Note for the two blocks, $G_1(s)$ and $G_2(s)$, $A_1(s)$ act as the input, and $A_2(s)$ and $A_3(s)$ are the outputs, respectively. Further, note that signal $U(s)$ goes through a **branch point P** and is renamed as $A_1(s)$. Hence, for the overall system, we combine the equations as follows:

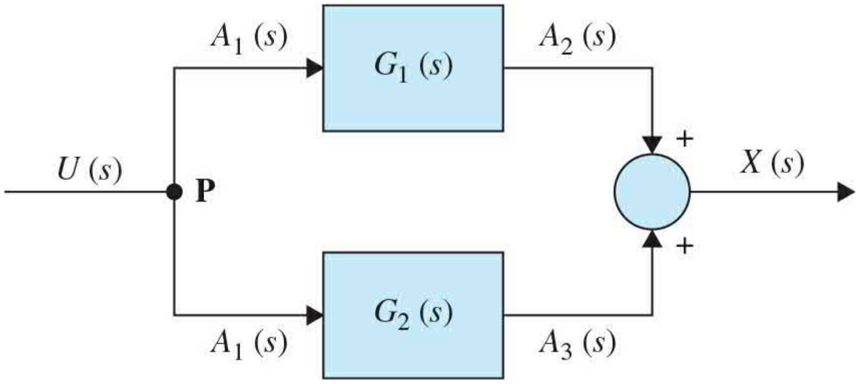


Figure 4-6 Block diagrams $G_1(s)$ and $G_2(s)$ connected in parallel.

$$
\begin{aligned}
A_1(s) &= U(s) \\
A_2(s) &= A_1(s)G_1(s) \\
A_3(s) &= A_1(s)G_2(s) \\
X(s) &= A_2(s) + A_3(s) \\
X(s) &= U(s)(G_1(s) + G_2(s)) \\
G(s) &= \frac{X(s)}{U(s)}
\end{aligned}
$$

Or,

$$G(s) = G_1(s) + G_2(s) \tag{4-7}$$

Using Eq. (4-7), the system in Fig. 4-6 can be represented by the system in Fig. 4-4.

Feedback

For a system to be classified as a **feedback control system**, it is necessary that the controlled variable be fed back and **compared** with the reference input. After the comparison, an **error** signal is generated, which is used to **actuate** the control system. As a result, the actuator is activated in the presence of the error to minimize or eliminate that very error. A necessary component of every feedback control system is an **output sensor**, which is used to convert the output signal to a quantity that has the same units as the reference input. A feedback control system is also known a **closed-loop** system. A system may have multiple feedback loops. Figure 4-7 shows the block diagram of a linear feedback control system with a single-feedback loop. The following terminology is defined with reference to the diagram:

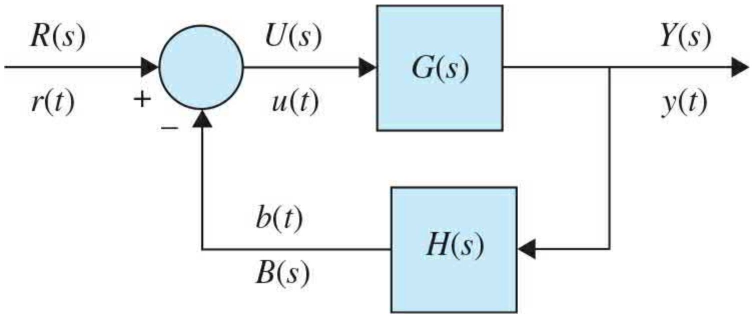


Figure 4-7 Block diagram of a basic negative feedback control system.

$r(t), R(s) =$ reference input (command)

$y(t), Y(s) =$ output (controlled variable)

$b(t), B(s) =$ feedback signal

$u(t), U(s) =$ actuating signal, also known as error signal $e(t), E(s)$ when $H(s) = 1$. But in most textbooks $E(s)$ is used regardless of value of the feedback transfer function

$H(s) =$ feedback transfer function

$G(s)H(s) = L(s) =$ loop transfer function

$G(s) =$ forward-path transfer function

$M(s) = Y(s)/R(s) =$ closed-loop transfer function or system transfer function

The closed-loop transfer function $M(s)$ can be expressed as a function of $G(s)$ and $H(s)$. From Fig. 4-7, we write

$$Y(s) = G(s)U(s) \tag{4-8}$$

and

$$B(s) = H(s)Y(s) \tag{4-9}$$

The actuating signal is written as

$$U(s) = R(s) - B(s) \tag{4-10}$$

Substituting Eq. (4-10) into Eq. (4-8) yields

$$Y(s)=G(s)R(s)-G(s)H(s)Y(s) \tag{4-11}$$

Substituting Eq. (4-9) into Eq. (4-7) and then solving for *Y*(*s*)/*R*(*s*) gives the closed-loop transfer function

$$M(s)=\frac{Y(s)}{R(s)}=\frac{G(s)}{1+G(s)H(s)} \tag{4-12}$$

The feedback system in Fig. 4-7 is said to have a **negative feedback loop** because the comparator **subtracts**. When the comparator **adds** the feedback, it is called **positive feedback,** and the transfer function Eq. (4-12) becomes

$$M(s)=\frac{Y(s)}{R(s)}=\frac{G(s)}{1-G(s)H(s)} \tag{4-13}$$

If *G* and *H* are constants, they are also called **gains**. If *H* = 1 in Fig. 4-7, the system is said to have a **unity feedback loop**, and if *H* = 0, the system is said to be **open loop**.

4-1-2 Relation between Mathematical Equations and Block Diagrams

Consider the second-order prototype system that we have studied in Chaps. 2 and 3:

$$\ddot{x}(t)+2\zeta\omega_n\dot{x}(t)+\omega_n^2x(t)=\omega_n^2u(t) \tag{4-14}$$

which has Laplace representation (assuming zero initial conditions $x(0)=\dot{x}(0)=0$):

$$X(s)s^2+2\zeta\omega_nX(s)s+\omega_n^2X(s)=\omega_n^2U(s) \tag{4-15}$$

Equation (4-15) consists of constant damping ratio *ζ*, constant natural frequency ω_n, input *U*(*s*), and output *X*(*s*). If we rearrange Eq. (4-15) to

$$\omega_n^2U(s)-2\zeta\omega_nX(s)s-\omega_n^2X(s)=X(s)s^2 \tag{4-16}$$

it can graphically be shown as in Fig. 4-8.

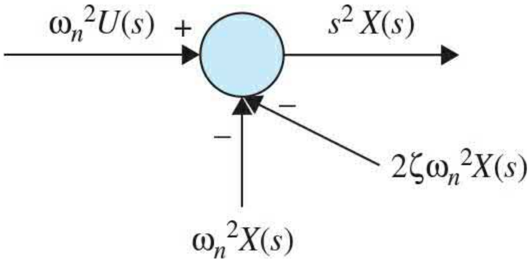


Figure 4-8 Graphical representation of Eq. (4-16) using a comparator.

The signals $2\zeta\omega_n sX(s)$ and $\omega_n^2X(s)$ may be conceived as the signal $X(s)$ going into blocks with transfer functions $2\zeta\omega_n s$ and ω_n^2, respectively, and the signal $X(s)$ may be obtained by integrating $s^2\ X(s)$ twice or by postmultiplying by 1/***s*2**, as shown in Fig. 4-9.

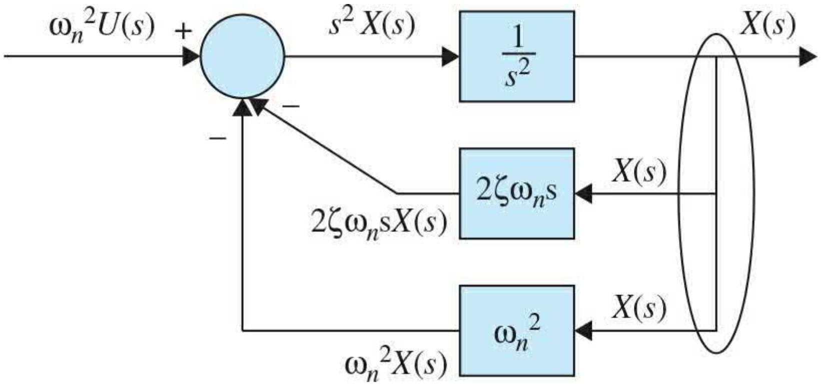


Figure 4-9 Addition of blocks $1/s_2$, $2\zeta\omega_n s$, and ω_n^2 to the graphical representation of Eq. (4-16).

Because the signals $X(s)$ in the right-hand side of Fig. 4-9 are the same, they can be connected, leading to the block diagram representation of the system Eq. (4-16), as shown in Fig. 4-10. If you wish, you can further dissect the block diagram in Fig. 4-10 by factoring out the term $1/s$ as in Fig. 4-11*a* to obtain Fig. 4-11*b*.

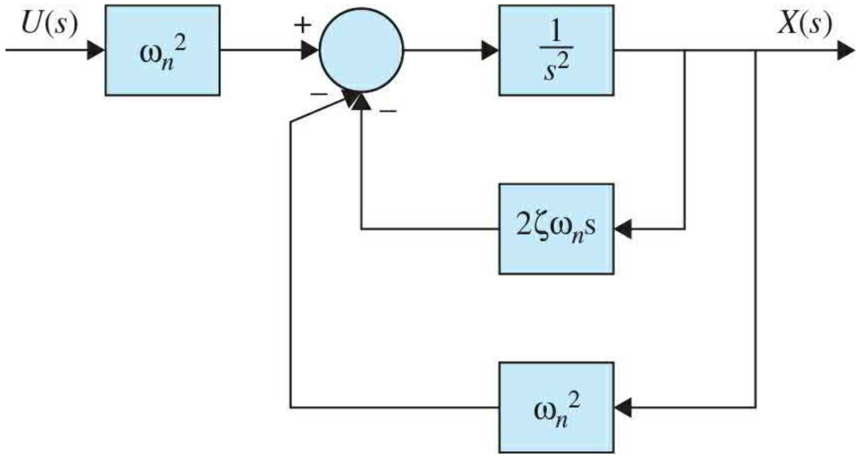


Figure 4-10 Block diagram representation of Eq. (4-16) in Laplace domain.

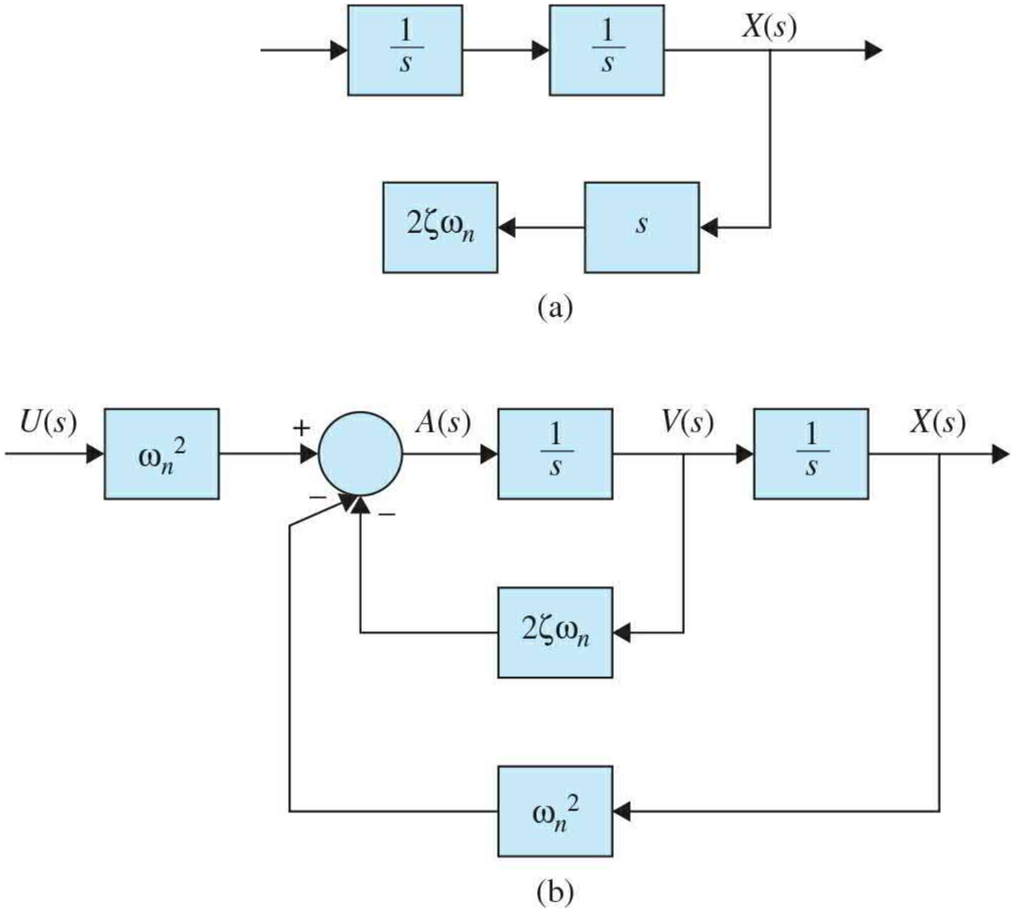


Figure 4-11 (*a*) Factorization of 1/*s* term in the internal feedback loop of Fig. 4-10. (*b*) Final block diagram representation of Eq. (4-16) in Laplace domain.

From Chap. 2, we know that the second-order prototype system in Eq. (4-14) can represent various dynamic systems. If the system studied here, for example, corresponds to the spring-mass-damper seen in Fig. 2-2, then internal variables $A(s)$ and $V(s)$ representing acceleration and velocity of the system, respectively, may also be incorporated in the block diagram model. The best way to see this is by recalling that $1/s$ is equivalent of integration in Laplace domain. Hence, if $A(s)$ is integrated once, we get $V(s)$, and after integrating $V(s)$, we get the $X(s)$ signal, as shown in Fig. 4-11*b*.

It is evident that there is no unique way of representing a system model with block diagrams. We may use different block diagram forms for different purposes. As long as the overall transfer function of the system is not altered. For example, to obtain the transfer function $V(s)/U(s)$, we may yet rearrange Fig. 4-11 to get $V(s)$ as the system output, as shown in Fig. 4-12. This enables us to determine the behavior of velocity signal with input $U(s)$.

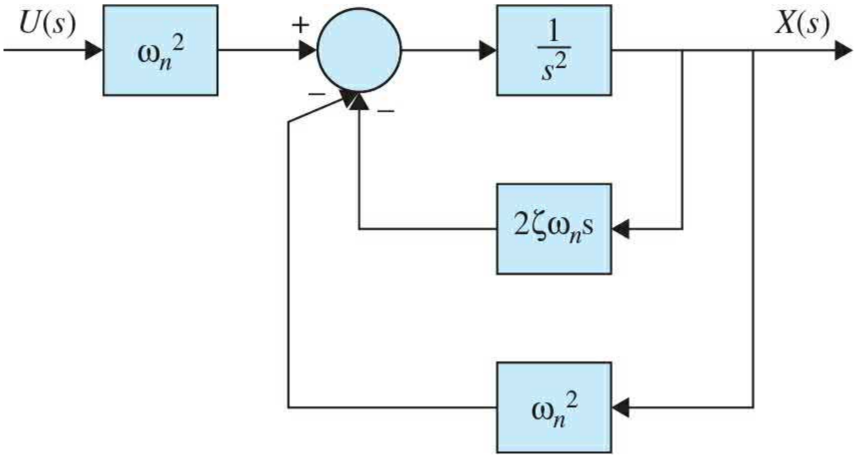


Figure 4-12 Block diagram of Eq. (4-16) in Laplace domain with $V(s)$ represented as the output.

EXAMPLE 4-1-3 Find the transfer function of the system in Fig. 4-11*b* and compare that to Eq. (4-15).

SOLUTION The ω_n^2 block at the input and feedback signals in Fig. 4-11*b* may be moved to the right-hand side of the comparator, as shown in Fig. 4-13*a*. This is the same as factorization of ω_n^2 as shown below:

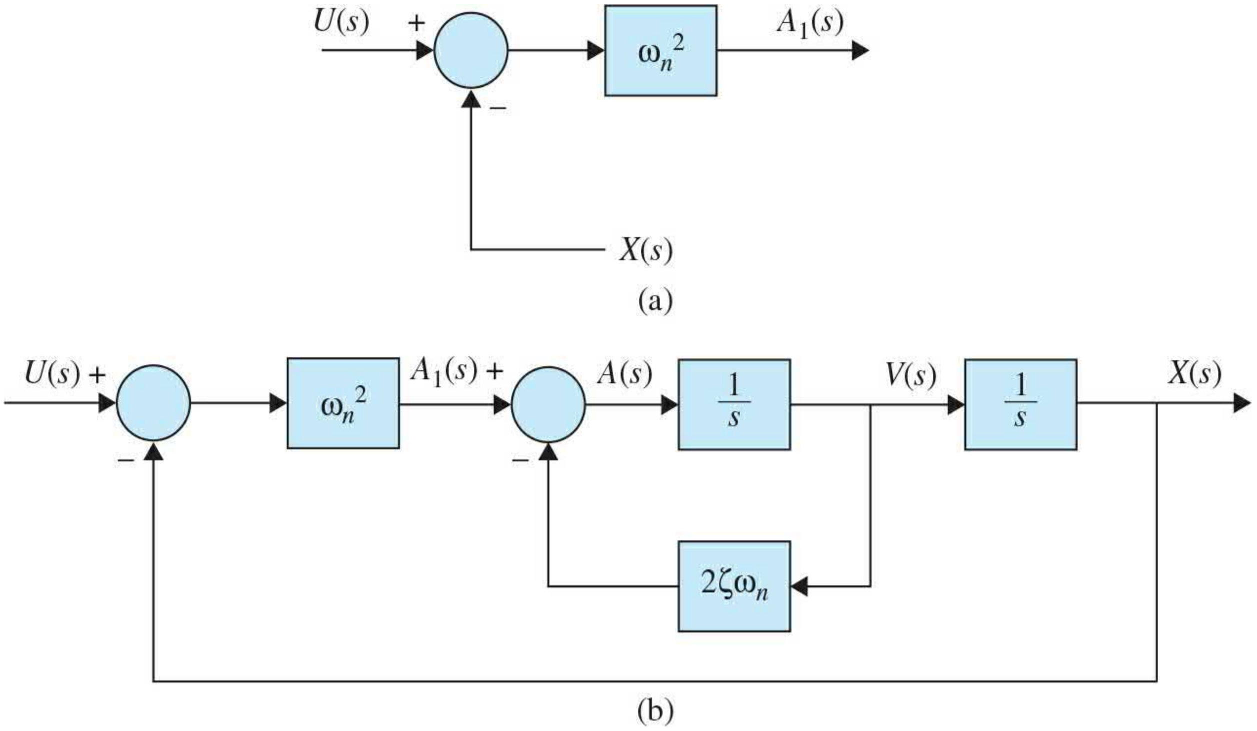


Figure 4-13 (*a*) Factorization of ω_n^2. (*b*) Alternative block diagram representation of Eq. (4-16) in Laplace domain.

$$\omega_n^2 U(s) - \omega_n^2 X(s) = \omega_n^2 (U(s) - X(s)) \tag{4-17}$$

The factorization operation on Eq. (4-16) results in a simpler block diagram representation of the system shown in Fig. 4-13*b*. Note that Figs. 4-11*b* and 4-13*b* are equivalent systems. Considering Fig. 4-11*b*, it is easy to identify the internal feedback loop, which in turn can be simplified using Eq. (4-12), or

$$\frac{V(s)}{A_1(s)} = \frac{\frac{1}{s}}{1 + \frac{2\zeta\omega_n}{s}} = \frac{1}{s + 2\zeta\omega_n} \tag{4-18}$$

After pre- and postmultiplication by ω_n^2 and 1/s, respectively, the block diagram of the system is simplified to what is shown in Fig. 4-14, which ultimately results in

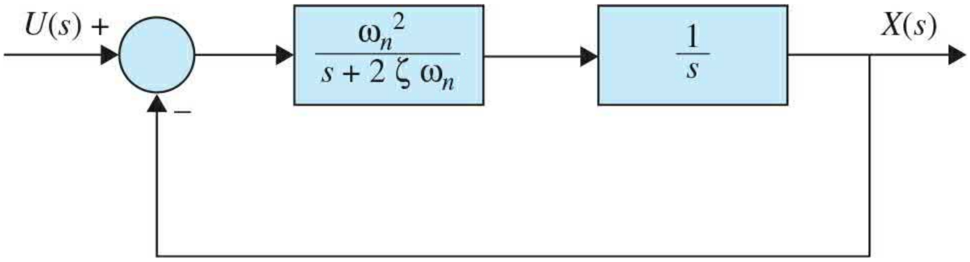


Figure 4-14 A block diagram representation of $\frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$.

$$\frac{X(s)}{U(s)} = \frac{\frac{\omega_n^2}{s(s+2\zeta\omega_n)}}{1+\frac{\omega_n^2}{s(s+2\zeta\omega_n)}} = \frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2} \tag{4-19}$$

Equation (4-19) is the transfer function of system Eq. (4-15). ▲

EXAMPLE 4-1-4 Find the velocity transfer function using Fig. 4-12 and compare that to the derivative of Eq. (4-19).

SOLUTION Simplification of the two feedback loops in Fig. 4-12, starting with the internal loop first, we have

$$\frac{V(s)}{U(s)} = \frac{\frac{\frac{1}{s}}{1+\frac{2\zeta\omega_n}{s}}\omega_n^2}{1+\frac{\frac{1}{s}}{1+\frac{2\zeta\omega_n}{s}}\frac{\omega_n^2}{s}}$$

$$\frac{V(s)}{U(s)} = \frac{\omega_n^2 s}{s^2+2\zeta\omega_n s+\omega_n^2} \tag{4-20}$$

Equation (4-20) is the same as the derivative of Eq. (4-

19), which is nothing but multiplying Eq. (4-19) by an s term. Try to find the $A(s)/U(s)$ transfer function. Obviously you must get $s^2X(s)/U(s)$. ▲

4-1-3 Block Diagram Reduction

As you might have noticed from the examples in the previous section, the transfer function of a control system may be obtained by manipulation of its block diagram and by its ultimate reduction into one block. For complicated block diagrams, it is often necessary to move a **comparato**r or a **branch point** to make the block diagram reduction process simpler. The two key operations in this case are

1. **Moving a branch point** from **P** to **Q**, as shown in Figs. 4-15*a* and *b*. This operation must be done such that the signals $Y(s)$ and $B(s)$ are unaltered. In Fig. 4-15*a*, we have the following relations:

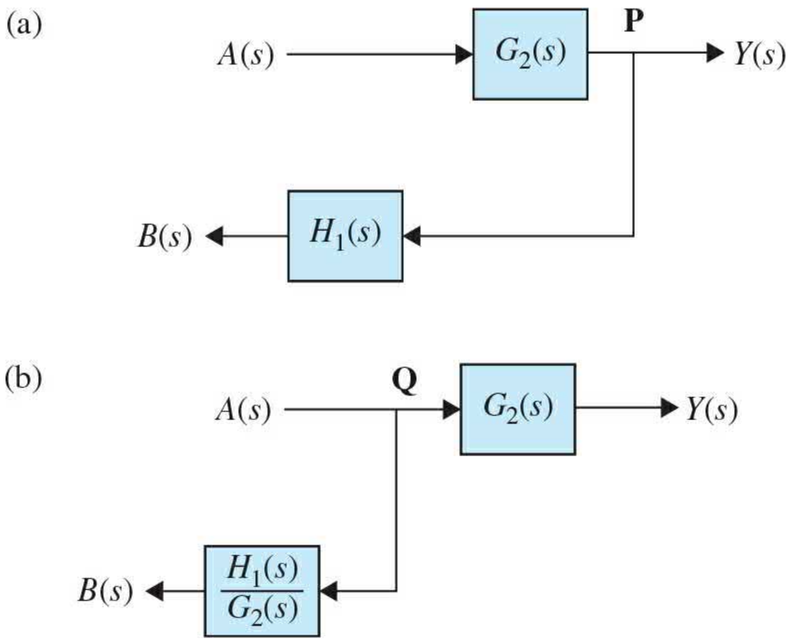


Figure 4-15 (*a*) Branch point relocation from point **P** to (*b*) point **Q**.

$$\begin{aligned} Y(s) &= A(s)G_2(s) \\ B(s) &= Y(s)H_1(s) \end{aligned} \tag{4-21}$$

In Fig. 4-15*b*, we have the following relations:

$$Y(s) = A(s)G_2(s)$$
$$B(s) = A(s)\frac{H_1(s)}{G_2(s)} \tag{4-22}$$

But

$$G_2(s) = \frac{A(s)}{Y(s)} \tag{4-23}$$
$$\Rightarrow B(s) = Y(s)H_1(s)$$

2. Moving **a comparator**, as shown in Figs. 4-16*a* and *b*, should also be done such that the output *Y*(*s*) is unaltered. In Fig. 4-16*a*, we have the following relations:

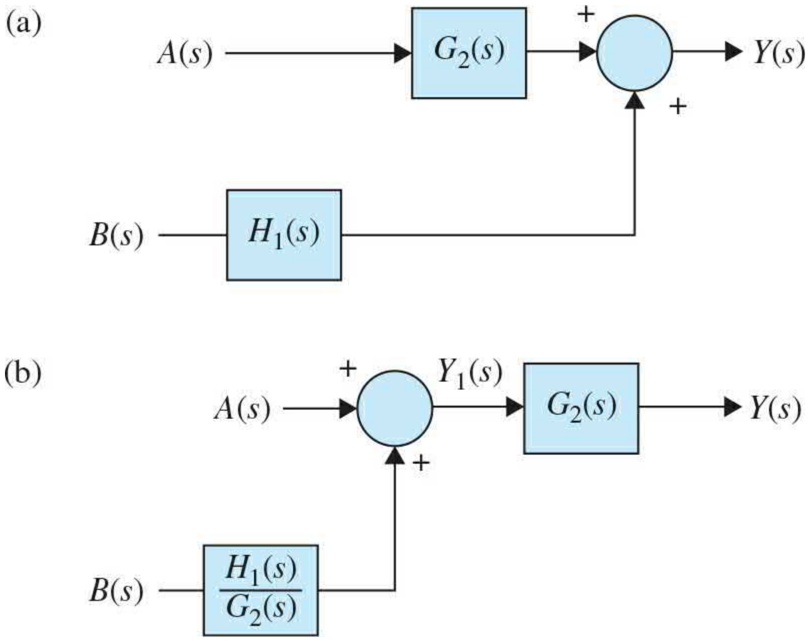


Figure 4-16 (*a*) Comparator relocation from the right-hand side of block $G_2(s)$ to (*b*) the left-hand side of block $G_2(s)$.

$$Y(s) = A(s)G_2(s) + B(s)H_1(s) \tag{4-24}$$

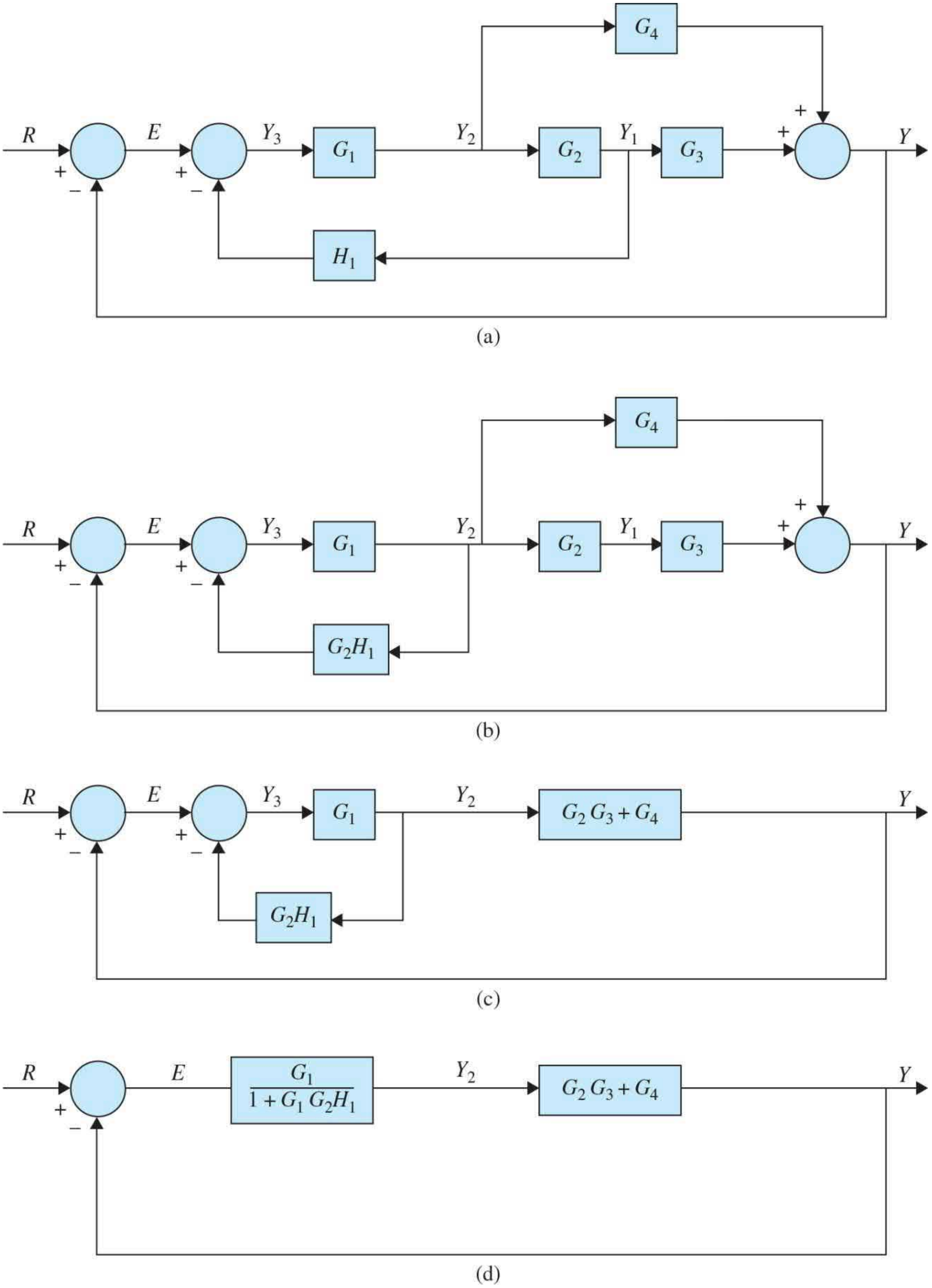
In Fig. 4-16*b*, we have the following relations:

$$\begin{aligned} Y_1(s) &= A(s) + B(s)\frac{H_1(s)}{G_2(s)} \\ Y(s) &= Y_1(s)G_2(s) \end{aligned} \tag{4-25}$$

So

$$\begin{aligned} Y(s) &= A(s)G_2(s) + B(s)\frac{H_1(s)}{G_2(s)}G_2(s) \\ \Rightarrow Y(s) &= A(s)G_2(s) + B(s)H_1(s) \end{aligned} \tag{4-26}$$

EXAMPLE 4-1-5 Find the input-output transfer function of the system shown in Fig. 4-17*a*.


R
E
Y_3
G_1
Y_2
G_4
G_2
Y_1
G_3
Y
H_1
+
−
+
−
+
+
R
E
Y_3
G_1
Y_2
G_4
G_2
Y_1
G_3
Y
G_2H_1
+
−
+
−
+
+
R
E
Y_3
G_1
Y_2
$G_2 G_3 + G_4$
Y
G_2H_1
+
−
+
−
R
E
$\frac{G_1}{1 + G_1 G_2 H_1}$
Y_2
$G_2 G_3 + G_4$
Y
+
−

(a)

(b)

(c)

(d)

Figure 4-17 (*a*) Original block diagram. (*b*) Moving the branch point at Y_1 to the left of block G_2. (*c*) Combining the blocks G_1, G_2, and G_3. (*d*) Eliminating the inner feedback loop.

SOLUTION To perform the block diagram reduction, one approach is to move the branch point at Y_1 to the left of block G_2, as shown in Fig. 4-17*b*. After that, the reduction becomes trivial, first by combining the blocks G_2, G_3, and G_4 as shown in Fig. 4-17*c*, and then by eliminating the two feedback loops. As a result, the transfer function of the final system after the reduction in Fig. 4-17*d* becomes

$$\frac{Y(s)}{E(s)} = \frac{G_1G_2G_3 + G_1G_4}{1 + G_1G_2H_1 + G_1G_2G_3 + G_1G_4} \tag{4-27}$$

4-1-4 Block Diagrams of Multi-Input Systems: Special Case—Systems with a Disturbance

An important case in the study of control systems is when a disturbance signal is present. Disturbance (such as heat loss in the example in Fig. 4-1) usually adversely affects the performance of the control system by placing a burden on the controller/actuator components. A simple block diagram with two inputs is shown in Fig. 4-18. In this case, one of the inputs, $D(s)$, is known as disturbance, while $R(s)$ is the input. Before designing a proper controller for the system, it is always important to learn the effects of $D(s)$ on the system.

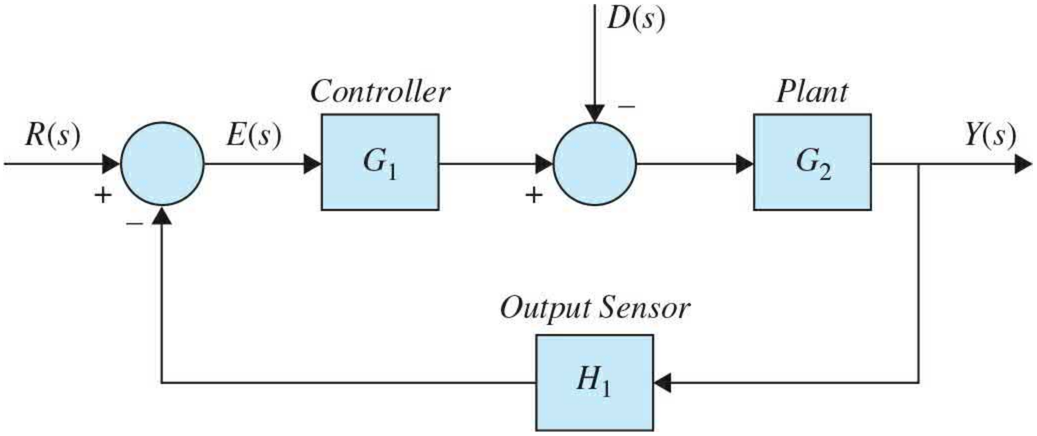


Figure 4-18 Block diagram of a system undergoing disturbance.

We use the method of superposition in modeling a multi-input system.

Super Position

For linear systems, the overall response of the system under multi-inputs is the summation of the responses due to the individual inputs, that is, in this case,

$$Y_{\text{total}} = Y_R \,|_{D=0} + Y_D \,|_{R=0} \tag{4-28}$$

When $D(s) = 0$, the block diagram is simplified (Fig. 4-19) to give the transfer function:

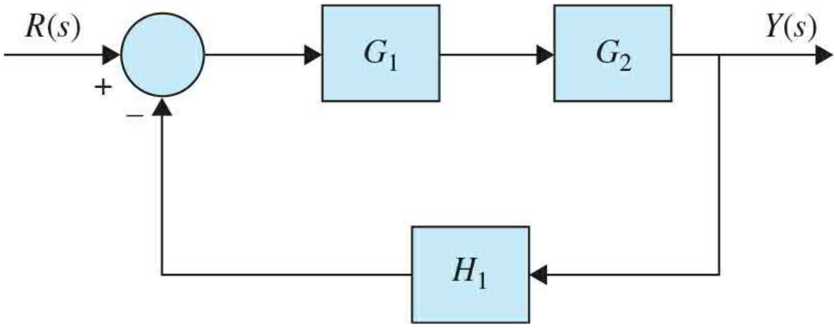


Figure 4-19 Block diagram of the system in Fig. 4-18, when $D(s) = 0$.

$$\frac{Y(s)}{R(s)} = \frac{G_1(s)G_2(s)}{1+G_1(s)G_2(s)H_1(s)} \tag{4-29}$$

When $R(s) = 0$, the block diagram is rearranged to give (Fig. 4-20):

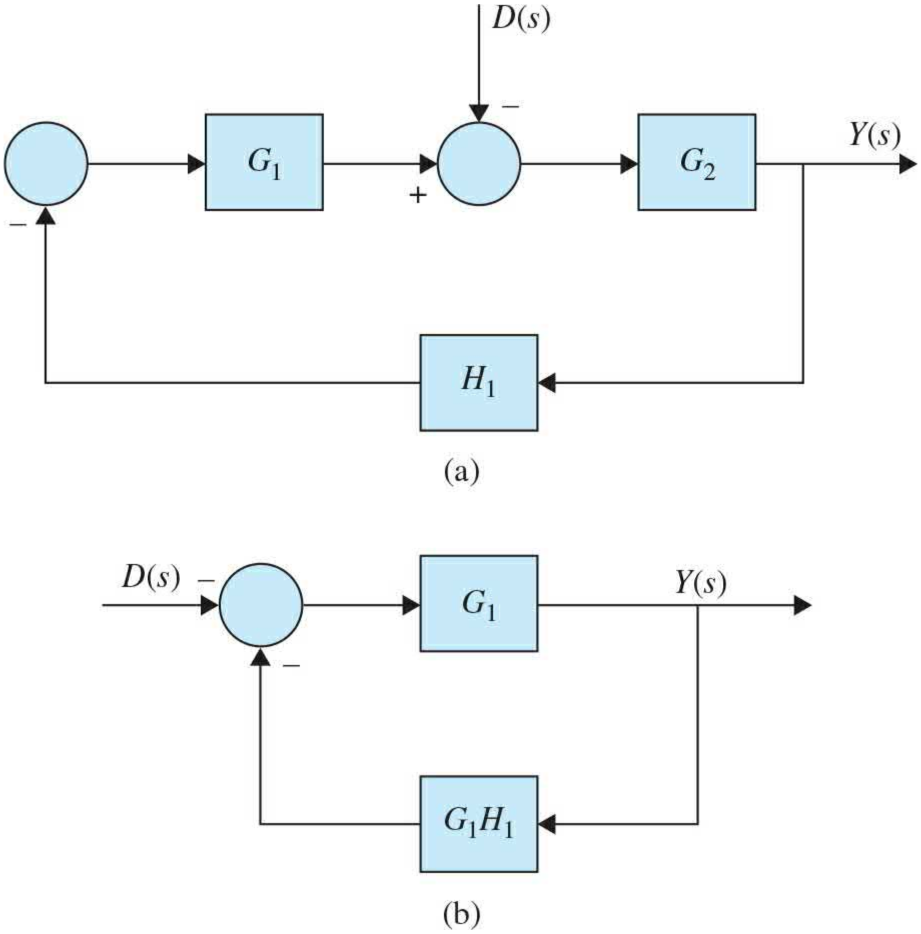


Figure 4-20 Block diagram of the system in Fig. 4-18, when *R*(*s*) = 0.

$$\frac{Y(s)}{D(s)} = \frac{-G_2(s)}{1+G_1(s)G_2(s)H_1(s)} \tag{4-30}$$

As a result, from Eq. (4-28) to Eq. (4-32), we ultimately get

$$\begin{aligned} Y_{\text{total}} &= \left.\frac{Y(s)}{R(s)}\right|_{D=0} R(s) + \left.\frac{Y(s)}{D(s)}\right|_{R=0} D(s) \\ Y(s) &= \frac{G_1G_2}{1+G_1G_2H_1} R(s) + \frac{-G_2}{1+G_1G_2H_1} D(s) \end{aligned} \tag{4-31}$$

Observations

$\left.\frac{Y}{R}\right|_{D=0}$ and $\left.\frac{Y}{D}\right|_{R=0}$ have the same denominators if the disturbance signal goes to the forward path. The negative sign in the numerator of $\left.\frac{Y}{D}\right|_{R=0}$ shows that the disturbance signal interferes with the controller signal, and, as a result, it adversely affects the performance of the system. Naturally, to compensate, there will be a higher burden on the controller.

4-1-5 Block Diagrams and Transfer Functions of Multivariable Systems

In this section, we illustrate the block diagram and matrix representations (see App. A) of multivariable systems. Two block diagram representations of a multivariable system with p inputs and q outputs are shown in Figs. 4-21*a* and *b*. In Fig. 4-21*a*, the individual input and output signals are designated, whereas in the block diagram of Fig. 4-21*b*, the multiplicity of the inputs and outputs is denoted by vectors. The case of Fig. 4-21*b* is preferable in practice because of its simplicity.

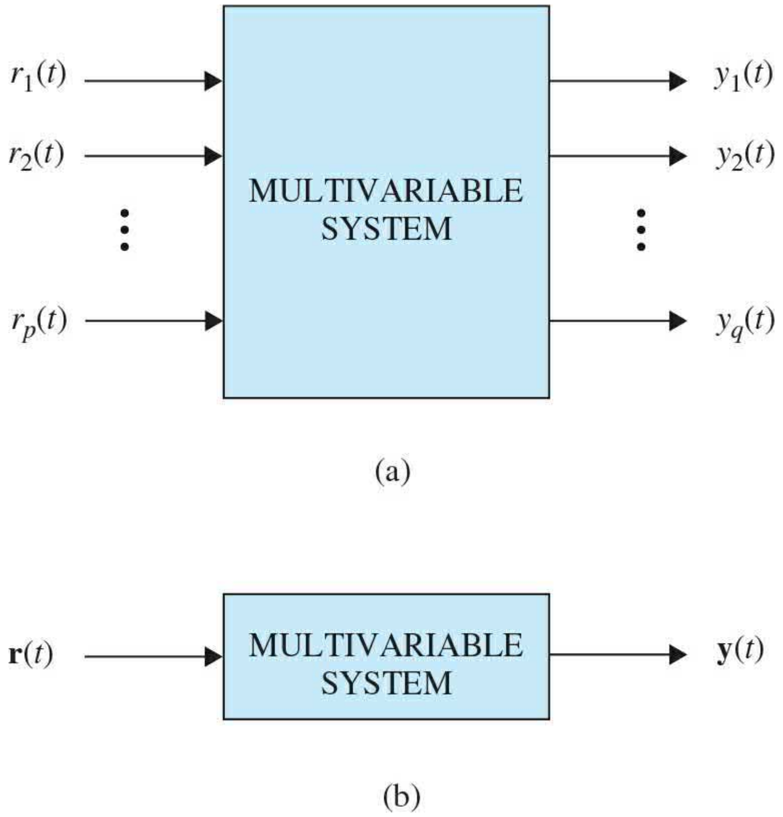


Figure 4-21 Block diagram representations of a multivariable system.

Figure 4-22 shows the block diagram of a multivariable feedback control system. The transfer function relationships of the system are expressed in vector-matrix form (see App. A):

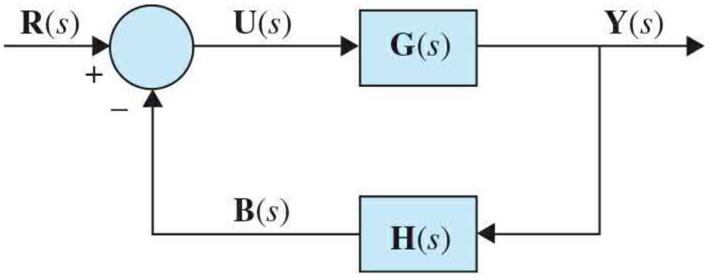


Figure 4-22 Block diagram of a multivariable feedback control system.

$$\mathbf{Y}(s)=\mathbf{G}(s)\mathbf{U}(s) \tag{4-32}$$

$$\mathbf{U}(s)=\mathbf{R}(s)-\mathbf{B}(s) \tag{4-33}$$

$$\mathbf{B}(s)=\mathbf{H}(s)\mathbf{Y}(s) \tag{4-34}$$

where **Y**(*s*) is the *q* × 1 output vector; **U**(*s*), **R**(*s*), and **B**(*s*) are all *p* × 1 vectors; and **G**(*s*) and **H**(*s*) are *q* × *p* and *p* × *q* transfer-function matrices, respectively. Substituting Eq. (4-11) into Eq. (4-10) and then from Eq. (4-10) to Eq. (4-9), we get

$$\mathbf{Y}(s)=\mathbf{G}(s)\mathbf{R}(s)-\mathbf{G}(s)\mathbf{H}(s)\mathbf{Y}(s) \tag{4-35}$$

Solving for **Y**(*s*) from Eq. (4-12) gives

$$\mathbf{Y}(s)=[\mathbf{I}+\mathbf{G}(s)\mathbf{H}(s)]^{-1}\mathbf{G}(s)\mathbf{R}(s) \tag{4-36}$$

provided that **I** + **G**(*s*)**H** (*s*) is nonsingular. The closed-loop transfer matrix is defined as

$$\mathbf{M}(s)=[\mathbf{I}+\mathbf{G}(s)\mathbf{H}(s)]^{-1}\mathbf{G}(s) \tag{4-37}$$

Then Eq. (4-14) is written as

$$\mathbf{Y}(s)=\mathbf{M}(s)\mathbf{R}(s) \tag{4-38}$$

EXAMPLE 4-1-6 Consider that the forward-path transfer function matrix and the feedback-path transfer function matrix of the system shown in Fig. 4-22 are

$$\mathbf{G}(s)=\begin{bmatrix} \dfrac{1}{s+1} & -\dfrac{1}{s} \\ 2 & \dfrac{1}{s+2} \end{bmatrix} \mathbf{H}(s)=\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4-39}$$

respectively. The closed-loop transfer function matrix of the system is given by Eq. (4-15), and is evaluated as follows:

$$\mathbf{I}+\mathbf{G}(s)\mathbf{H}(s)=\begin{bmatrix} 1+\dfrac{1}{s+1} & -\dfrac{1}{s} \\ 2 & 1+\dfrac{1}{s+2} \end{bmatrix}=\begin{bmatrix} \dfrac{s+2}{s+1} & -\dfrac{1}{s} \\ 2 & \dfrac{s+3}{s+2} \end{bmatrix} \tag{4-40}$$

The closed-loop transfer function matrix is

$$\mathbf{M}(s)=[\mathbf{I}+\mathbf{G}(s)\mathbf{H}(s)]^{-1}\mathbf{G}(s)=\frac{1}{\Delta}\begin{bmatrix} \dfrac{s+3}{s+2} & \dfrac{1}{s} \\ -2 & \dfrac{s+2}{s+1} \end{bmatrix}\begin{bmatrix} \dfrac{1}{s+1} & -\dfrac{1}{s} \\ 2 & \dfrac{1}{s+2} \end{bmatrix} \tag{4-41}$$

where

$$\Delta=\frac{s+2}{s+1}\frac{s+3}{s+2}+\frac{2}{s}=\frac{s^2+5s+2}{s(s+1)} \tag{4-42}$$

Thus,

$$\mathbf{M}(s)=\frac{s(s+1)}{s^2+5s+2}\begin{bmatrix} \dfrac{3s^2+9s+4}{s(s+1)(s+2)} & -\dfrac{1}{s} \\ 2 & \dfrac{3s+2}{s(s+1)} \end{bmatrix} \tag{4-43}$$

4-2 SIGNAL-FLOW GRAPHS

A SFG may be regarded as an alternative representation of a block diagram. The SFG was introduced by S. J. Mason [2, 3] for the cause-and-effect (input-output) representation of linear systems that are modeled by algebraic equations. An SFG may be defined as a graphical means of portraying the input-output relationships among the variables of a set of linear algebraic equations.

The relation between block diagrams and SFGs are tabulated for four important cases, as shown in Fig. 4-23.

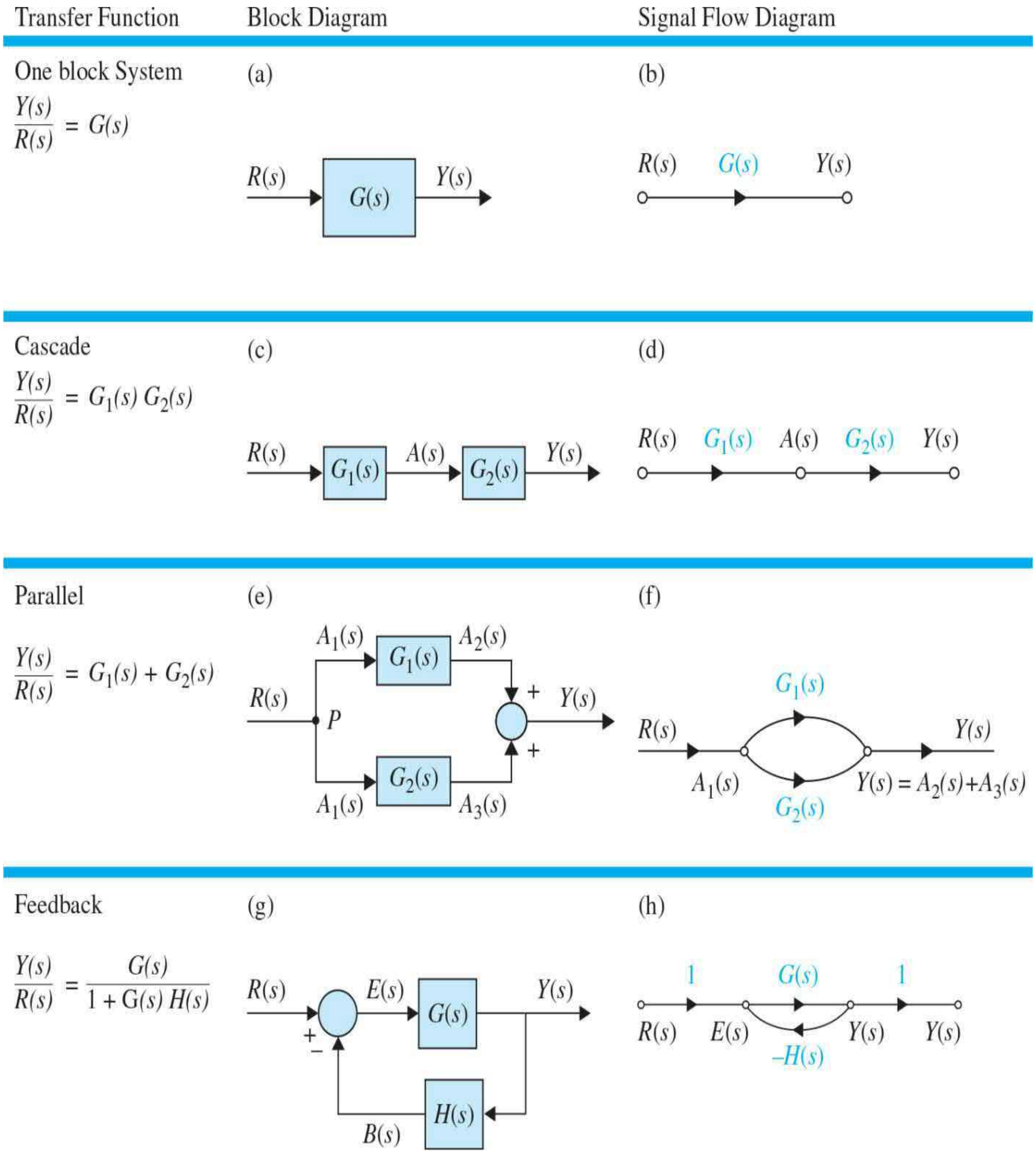


Figure 4-23 Block diagrams and their SFG equivalent representations. (*a*) Input-output representation in block diagram form. (*b*) Equivalent input-output representation in SFG form. (*c*) A cascade block diagram representation. (*d*) Equivalent cascade SFG representation. (*e*) A parallel block diagram representation. (*f*) Equivalent parallel SFG representation. (*g*) A negative feedback block diagram representation. (*h*) Equivalent negative

feedback SFG representation.

Considering Fig. 4-23*b*, when constructing an SFG, junction points, or **nodes**, are used to represent variables—in this case *U*(*s*) is the input variable and *Y*(*s*) is the output variable. The nodes are connected by line segments called **branches**, according to the cause-and-effect equations. The branches have associated branch gains and directions—in this case the branch represents the transfer function *G*(*s*). *A signal can transmit through a branch only in the direction of the arrow*. In general, the construction of the SFG is basically a matter of following through the input-output relations of each variable in terms of itself and the others. As a result, in Fig. 4-23*b*, the SFG represents the transfer function:

In an SFG, signals can transmit through a branch only in the direction of the arrow.

$$\frac{Y(s)}{U(s)}=G(s) \tag{4-44}$$

where *U*(*s*) is the input, *Y(s)* is the output, and *G*(*s*) is the gain, or transmittance, between the two variables. The branch between the input node and the output node should be interpreted as a unilateral amplifier with gain *G*(*s*), so when a signal of one unit is applied at the input *U*(*s*), a signal of strength *G*(*s*)*U(s)* is delivered at node *Y(s)*. Although algebraically Eq. (4-44) can be written as

$$U(s)=\frac{1}{G(s)}Y(s) \tag{4-45}$$

the SFG of Fig. 4-23*b* does not imply this relationship. If Eq. (4-45) is valid as a cause-and-effect equation, a new SFG should be drawn with *Y(s)* as the input and *U*(*s*) as the output.

Comparing Fig. 4-23*c* with Fig. 4-23*d*, or Fig. 4-23*e* with Fig. 4-23*g*, it is easy to see that the nodes in SFGs represent the variables in the block diagrams—that is, input, output, and intermediate variables such as *A*(*s*). The nodes are then connected through branches with gains that represent the transfer functions $G_1(s)$ and $G_2(s)$, respectively.

The SFG representation of cascade and parallel forms and the feedback system, shown in Figs. 4-23*e* and *f*, are discussed in more detail in the next section.

4-2-1 SFG Algebra

Let us outline the following manipulation rules and algebra for the SFGs:

1. The value of the variable represented by a node is equal to the sum of all the signals entering the node. For the SFG of Fig. 4-24, the value of y_1 is equal to the sum of the signals transmitted through all the incoming branches; that is,

$$y_1 = a_{21}y_2 + a_{31}y_3 + a_{41}y_4 + a_{51}y_5 \tag{4-46}$$

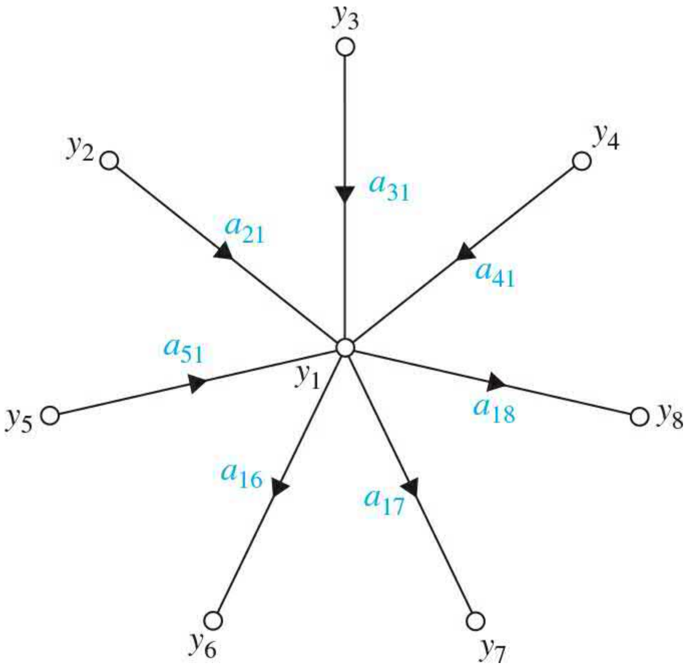


Figure 4-24 Node as a summing point and as a transmitting point.

2. The value of the variable represented by a node is transmitted through all branches leaving the node. In the SFG of Fig. 4-24, we have

$$
\begin{aligned}
y_6 &= a_{17}y_1 \\
y_7 &= a_{17}y_1 \\
y_8 &= a_{18}y_1
\end{aligned}
\tag{4-47}
$$

3. **Parallel branches** in the same direction connecting two nodes can be replaced by a single branch with gain equal to the sum of the gains of the parallel branches. An example of this case is illustrated in Figs. 4-23*f* and 4-25.

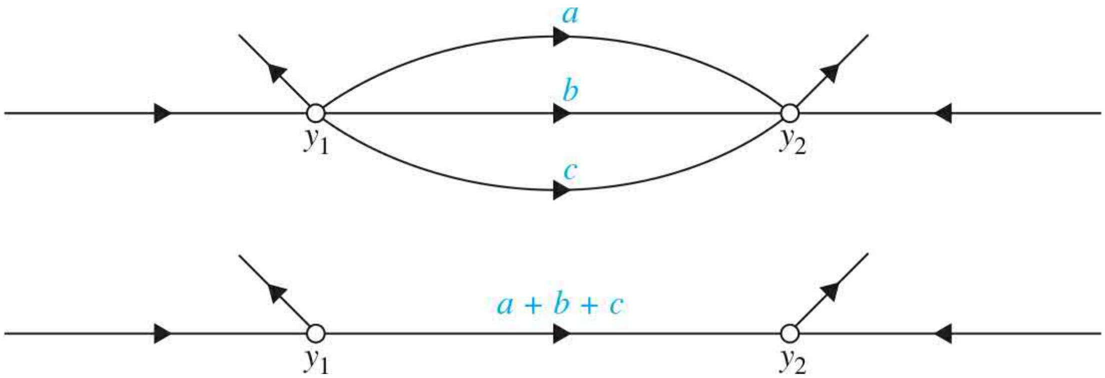


Figure 4-25 Signal-flow graph with parallel paths replaced by one with a single branch.

4. A **series (cascade) connection** of unidirectional branches, as shown in Fig. 4-23*d* or 4-26, can be replaced by a single branch with gain equal to the product of the branch gains.

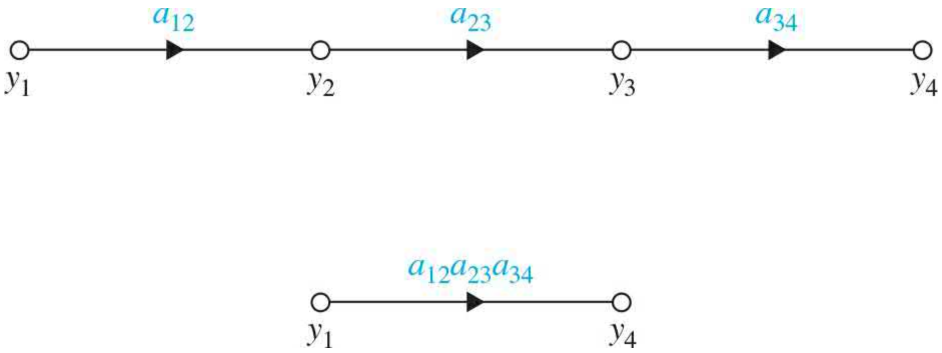


Figure 4-26 Signal-flow graph with cascade unidirectional branches replaced by a single branch.

5. A **feedback system** as shown in Fig. 4-23*g* is subject to the

following algebraic equations:

$$E(s) = R(s) - H(s)Y(s) \tag{4-48}$$

and

$$Y(s) = G(s)E(s) \tag{4-49}$$

Substituting Eq. (4-49) into Eq. (4-48), while eliminating the intermediate variable *E*(*s*), we get

$$Y(s) = G(s)R(s) - G(s)H(s)Y(s) \tag{4-50}$$

Solving for *Y*(*s*)/*R*(*s*), we get the closed-loop transfer function

$$M(s) = \frac{Y(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)} \tag{4-51}$$

EXAMPLE 4-2-1 Convert the block diagram in Fig. 4-27*a* to an SFG format.

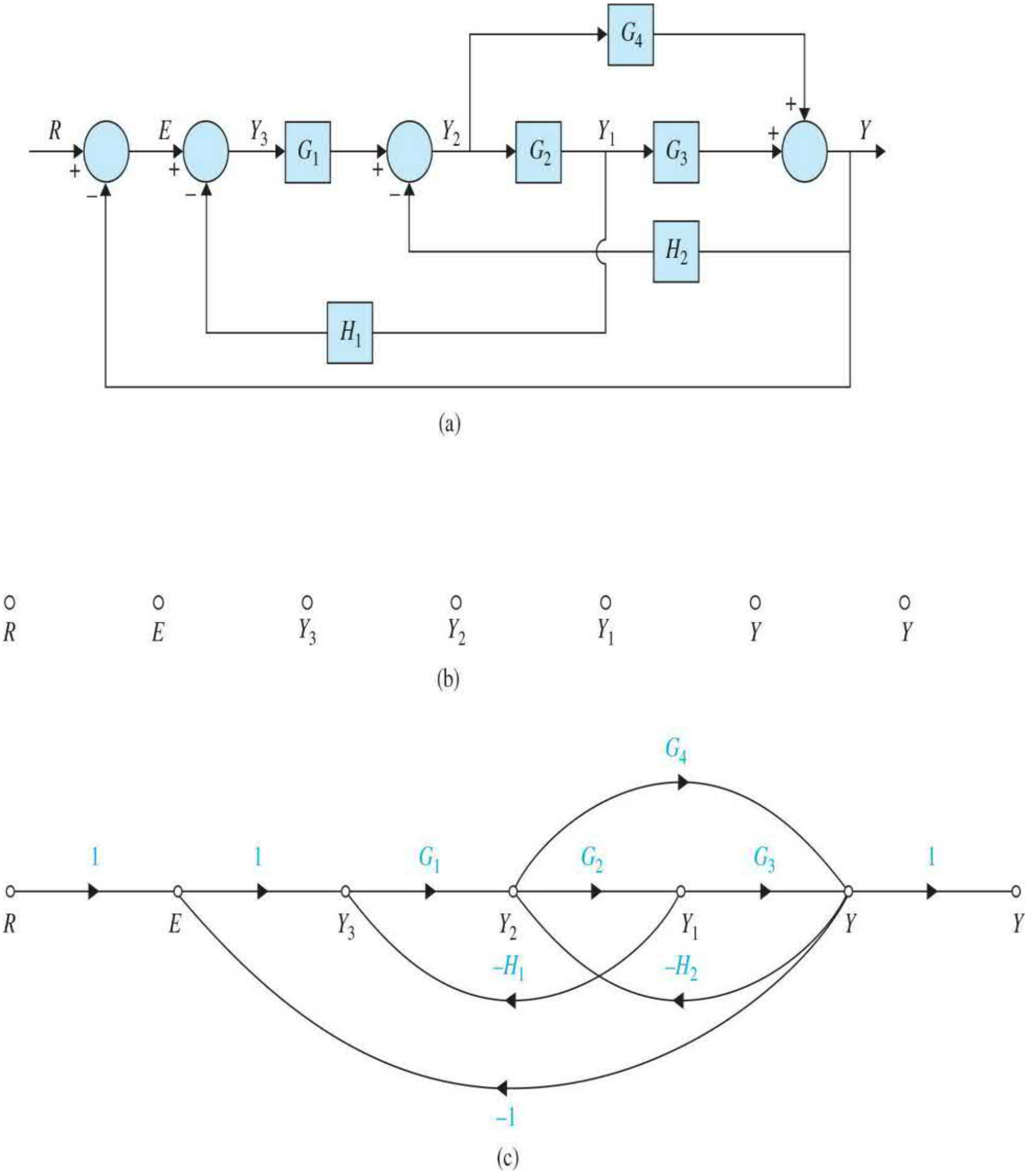


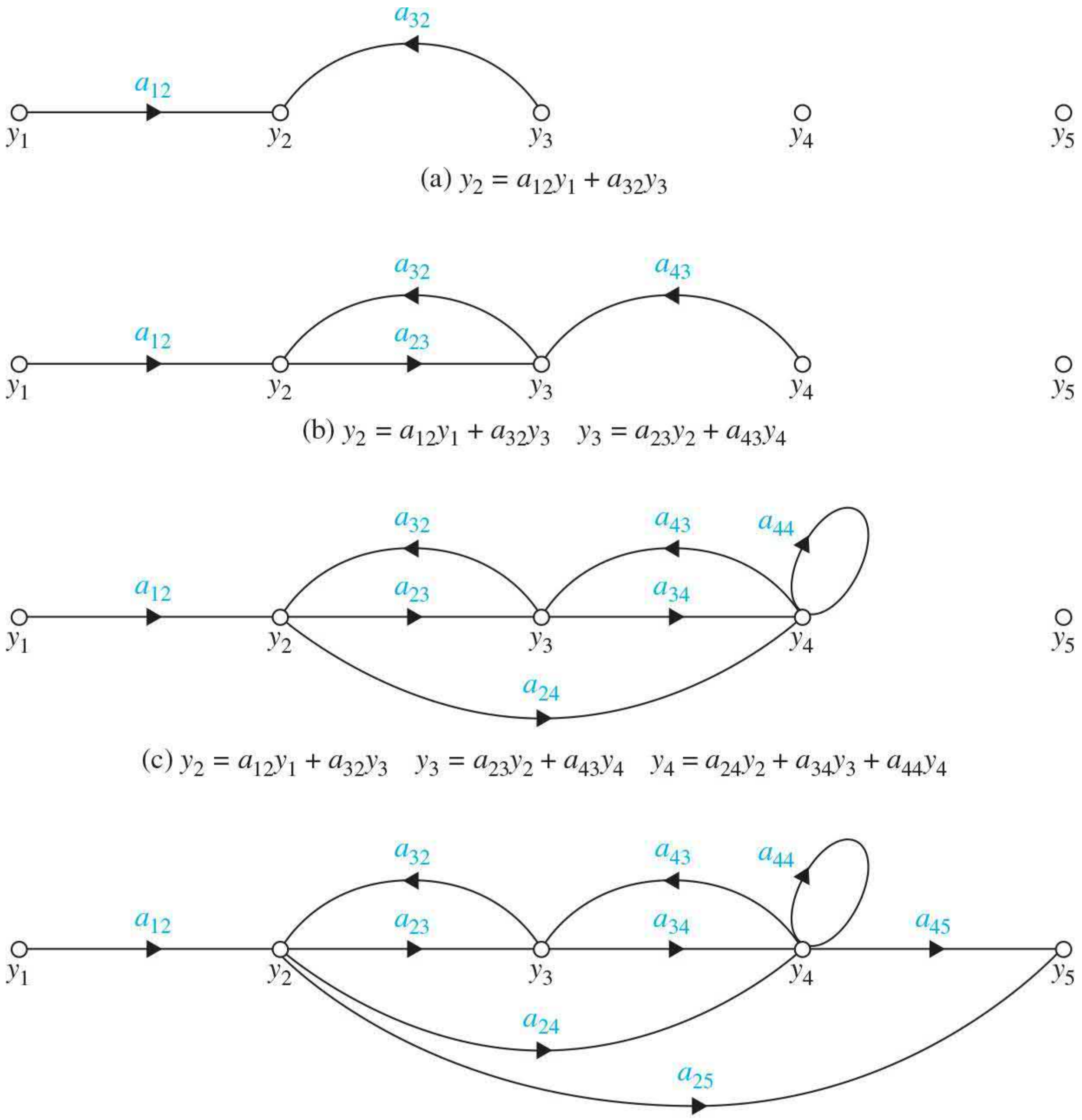
Figure 4-27 (*a*) Block diagram of a control system. (*b*) Signal nodes. (*c*) Equivalent signal-flow graph.

SOLUTION First identify all block diagram variables—in this case, *R*, *E*, Y_3, Y_2, Y_1, and *Y*. Next, associate each variable to a node, as shown in Fig. 4-27*b*. Note that it is important to clearly identify the input and output nodes *R* and *Y*, respectively, as shown in Fig. 4-27*b*. Use branches to interconnect the nodes while ensuring the branch directions match the signal directions in the block diagram. Label each branch with the appropriate gain corresponding to a transfer function in Fig. 4-27*a*. Make sure to incorporate the negative feedback signs into the gains (i.e., $-G_1(s)$,$-G_2(s)$ and – 1)—see Fig. 4-27*c*.

EXAMPLE 4-2-2 As an example on the construction of an SFG, consider the following set of algebraic equations:

$$\begin{aligned} y_2 &= a_{12}y_1 + a_{32}y_3 \\ y_3 &= a_{23}y_2 + a_{43}y_4 \\ y_4 &= a_{24}y_2 + a_{34}y_3 + a_{44}y_4 \\ y_5 &= a_{25}y_2 + a_{45}y_4 \end{aligned} \tag{4-52}$$

The SFG for these equations is constructed, step by step, in Fig. 4-28.



(a) $y_2 = a_{12}y_1 + a_{32}y_3$

(b) $y_2 = a_{12}y_1 + a_{32}y_3 \quad y_3 = a_{23}y_2 + a_{43}y_4$

(c) $y_2 = a_{12}y_1 + a_{32}y_3 \quad y_3 = a_{23}y_2 + a_{43}y_4 \quad y_4 = a_{24}y_2 + a_{34}y_3 + a_{44}y_4$

(d) Complete signal-flow graph

Figure 4-28 Step-by-step construction of the signal-flow graph in Eq. (4-52).

4-2-2 Definitions of SFG Terms

In addition to the branches and nodes defined earlier for the SFG, the following terms are useful for the purpose of identification and execution of the SFG algebra.

Input Node (Source)

An input node *is a node that has only outgoing branches* (example: node $U(s)$ in Fig. 4-23*b*).

Output Node (Sink)

An output node *is a node that has only incoming branches* (example: node $Y(s)$ in Fig. 4-23*b*). However, this condition is not always readily met by an output node. For instance, the SFG in Fig. 4-29*a* does not have a node that satisfies the condition of an output node. It may be necessary to regard y_2 and/or y_3 as output nodes to find the effects at these nodes due to the input. To make y_2 an output node, we simply connect a branch with unity gain from the existing node y_2 to a new node also designated as y_2, as shown in Fig. 4-29*b*. The same procedure is applied to y_3. Notice that, in the modified SFG of Fig. 4-29*b*, the equations $y_2 = y_2$ and $y_3 = y_3$ are added to the original equations. In general, we can make any noninput node of an SFG an output by the procedure just illustrated. However, we **cannot** convert a noninput node into an input node by reversing the branch direction of the procedure described for output nodes. For instance, node y_2 of the SFG in Fig. 4-29*a* is not an input node. If we attempt to convert it into an input node by adding an incoming branch with unity gain from another identical node y_2, the SFG of Fig. 4-30 would result. The equation that portrays the relationship at node y_2 now reads

An **input node** has only outgoing branches.

An **output node** has only incoming branches.

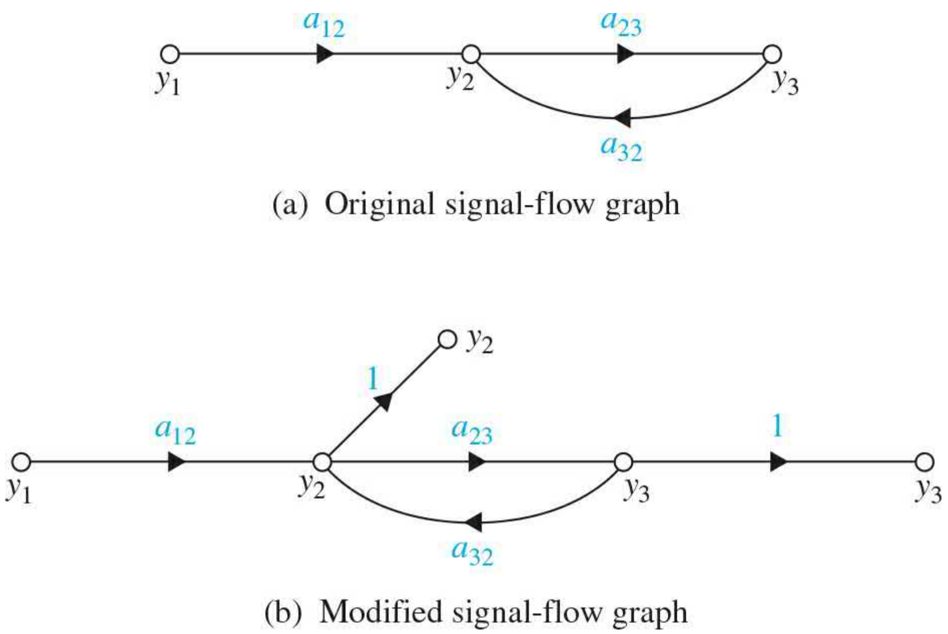


Figure 4-29 Modification of a signal-flow graph so that y_2 and y_3 satisfy the condition as output nodes.

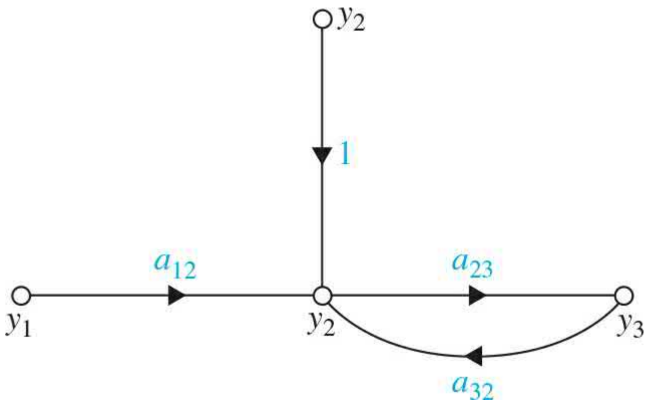


Figure 4-30 Erroneous way to make node y_2 an input node.

$$y_2 = y_2 + a_{12}y_1 + a_{32}y_3 \tag{4-53}$$

which is different from the original equation given in Fig. 4-29*a*.

Path

A path *is any collection of a continuous succession of branches traversed in the same direction*. The definition of a path is entirely general, since it does not prevent any node from being traversed more than once. Therefore, as simple as the SFG of Fig. 4-29*a* is, it may have numerous paths just by traversing the branches a_{23} and a_{32} continuously.

Forward Path

A forward path *is a path that starts at an input node and ends at an output node and along which no node is traversed more than once*. For example, in the SFG of Fig. 4-28*d*, y_1 is the input node, and the rest of the nodes are all possible output nodes. The forward path between y_1 and y_2 is simply the connecting branch between the two nodes. There are two forward paths between y_1 and y_3: One contains the branches from y_1 to y_2 to y_3, and the other one contains the branches from y_1 to y_2 to y_4 (through the branch with gain a_{24}) and then back to y_3 (through the branch with gain a_{43}). The reader should try to determine the two forward paths between y_1 and y_4. Similarly, there are three forward paths between y_1 and y_5.

Path Gain

The product of the branch gains encountered in traversing a path is called the path gain. For example, the path gain for the path $y_1 - y_2 - y_3 - y_4$ in Fig. 4-28*d* is $a_{12}a_{23}a_{34}$

Loop

A loop *is a path that originates and terminates on the same node and along which no other node is encountered more than once*. For example, there are four loops in the SFG of Fig. 4-28*d*. These are shown in Fig. 4-31.

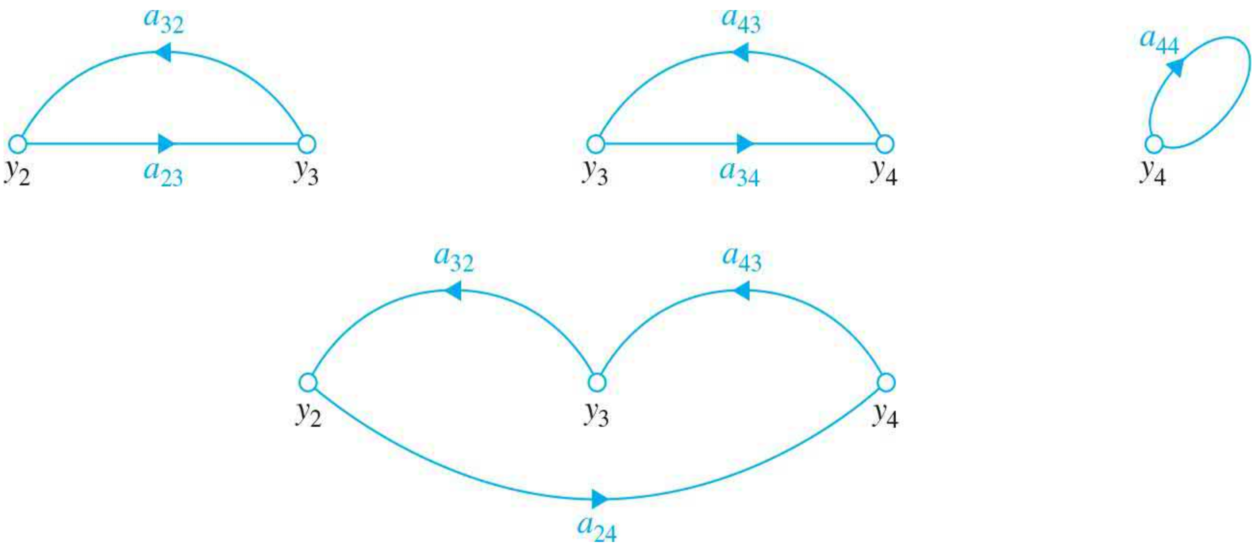


Figure 4-31 Four loops in the signal-flow graph of Fig. 4-28*d*.

The SFG **gain formula** can only be applied between an **input** node and an **output** node.

Δ is the same regardless of which output node is chosen.

Forward-Path Gain

The forward-path gain *is the path gain of a forward path*.

Loop Gain

The loop gain *is the path gain of a loop*. For example, the loop gain of the loop $y_2 - y_4 - y_3 - y_2$ in Fig. 4-31 is $a_{24}a_{43}a_{32}$.

Nontouching Loops

Two parts of an SFG are nontouching *if they do not share a common node*. For example, the loops $y_2 - y_3 - y_2$ and $y_4 - y_4$ of the SFG in Fig. 4-28*d* are nontouching loops.

Two parts of an SFG are **nontouching** if they do not share a common node.

4-2-3 Gain Formula for SFG

Given an SFG or block diagram, the task of solving for the input-output relations by algebraic manipulation could be quite tedious. Fortunately, there is a general gain formula available that allows the determination of the input-output relations of an SFG by inspection.

Given an SFG with *N* forward paths and *K* loops, the gain between the input node y_{in} and output node y_{out} is [3]

$$M = \frac{y_{\text{out}}}{y_{\text{in}}} = \sum_{k=1}^{N} \frac{M_k \Delta_k}{\Delta} \tag{4-54}$$

where

$$
\begin{aligned}
y_{\text{in}} &= \text{input-node variable}\\
y_{\text{out}} &= \text{output-node variable}\\
M &= \text{gain between } y_{\text{in}} \text{ and } y_{\text{out}}\\
N &= \text{total number of forward paths between } y_{\text{in}} \text{ and } y_{\text{out}}\\
M_k &= \text{gain of the } k\text{th forward paths between } y_{\text{in}} \text{ and } y_{\text{out}}
\end{aligned}
$$

$$\Delta = 1 - \sum_{i=1} L_{i1} + \sum_{j=1} L_{j2} - \sum_{k=1} L_{k3} + \cdots \tag{4-55}$$

or

Δ = 1 – (sum of the gains of **all individual** loops) + (sum of products of gains of all possible combinations of **two** nontouching loops) – (sum of products of gains of all possible combinations of **three** nontouching loops) + (sum of products of gains of all possible combinations of **four** nontouching loops) –...

Δ_k = the Δ for that part of the SFG that is nontouching with the *k*th forward path.

The gain formula in Eq. (4-54) may seem formidable to use at first glance. However, Δ and Δ_k are the only terms in the formula that could be complicated if the SFG has a large number of loops and nontouching loops.

Care must be taken when applying the gain formula to ensure that it is applied between an **input node** and an **output node**.

EXAMPLE 4-2-3 Consider that the closed-loop transfer function *Y*(*s*)/*R*(*s*) of the SFG in Fig. 4-23*f* is to be determined by use of the gain formula, Eq. (4-54). The following results are obtained by inspection of the SFG:

1. There is only one forward path between *R*(*s*) and *Y*(*s*), and the forward-path gain is

$$M_1 = G(s) \tag{4-56}$$

2. There is only one loop; the loop gain is

$$L_{11} = -G(s)H(s) \tag{4-57}$$

3. There are no nontouching loops since the forward path is in touch with the loop L_{11}. Thus, Δ1 = 1, and

$$\Delta = 1 - L_{11} = 1 + G(s)H(s) \tag{4-58}$$

Using Eq. (4-54), the closed-loop transfer function is written as

$$\frac{Y(s)}{R(s)} = \frac{M_1\Delta_1}{\Delta} = \frac{G(s)}{1+G(s)H(s)} \tag{4-59}$$

which agrees with Eq. (4-12) or (4-51). ▲

EXAMPLE 4-2-4 Consider the SFG shown in Fig. 4-28*d*. Let us first determine the gain between y_1 and y_5 using the gain formula.

The three forward paths between y_1 and y_5 and the **forward-path gains** are

Forward path	Gain
$y_1 - y_2 - y_3 - y_4 - y_5$	$M_1 = a_{12}a_{23}a_{34}a_{45}$
$y_1 - y_2 - y_5$	$M_2 = a_{12}a_{25}$
$y_1 - y_2 - y_4 - y_5$	$M_3 = a_{12}a_{24}a_{45}$

The **four loops** of the SFG are shown in Fig. 4-28. The loop gains are

Loop	Gain
$y_2 - y_3 - y_2$	$L_{11} = a_{23}a_{32}$
$y_3 - y_4 - y_3$	$L_{21} = a_{34}a_{43}$
$y_2 - y_4 - y_3 - y_2$	$L_{31} = a_{24}a_{43}a_{32}$
$y_4 - y_4$	$L_{41} = a_{44}$

There are two **nontouching loops**; that is,

$$y_2 - y_3 - y_2 \quad \text{and} \quad y_4 - y_4$$

Thus, the product of the gains of the two nontouching loops is

$$L_{12} = a_{23}a_{32}a_{44} \tag{4-60}$$

All the loops are in touch with forward paths M_1 and M_3. Thus, $\Delta_1 = \Delta_3 = 1$. Two of the loops are not in touch with forward path M**2**. These loops are $y_3 - y_4 - y_3$ and $y_4 - y_4$. Thus,

$$\Delta_2 = 1 - a_{34}a_{43} - a_{44} \tag{4-61}$$

Substituting these quantities into Eq. (4-54), we have

$$\begin{aligned}\frac{y_5}{y_1} &= \frac{M_1\Delta_1 + M_2\Delta_2 + M_3\Delta_3}{\Delta} \\ &= \frac{(a_{12}a_{23}a_{34}a_{45}) + (a_{12}a_{25})(1 - a_{34}a_{43} - a_{44}) + a_{12}a_{24}a_{45}}{1 - (a_{23}a_{32} + a_{34}a_{43} + a_{24}a_{32}a_{43} + a_{44}) + a_{23}a_{32}a_{44}}\end{aligned} \tag{4-62}$$

where

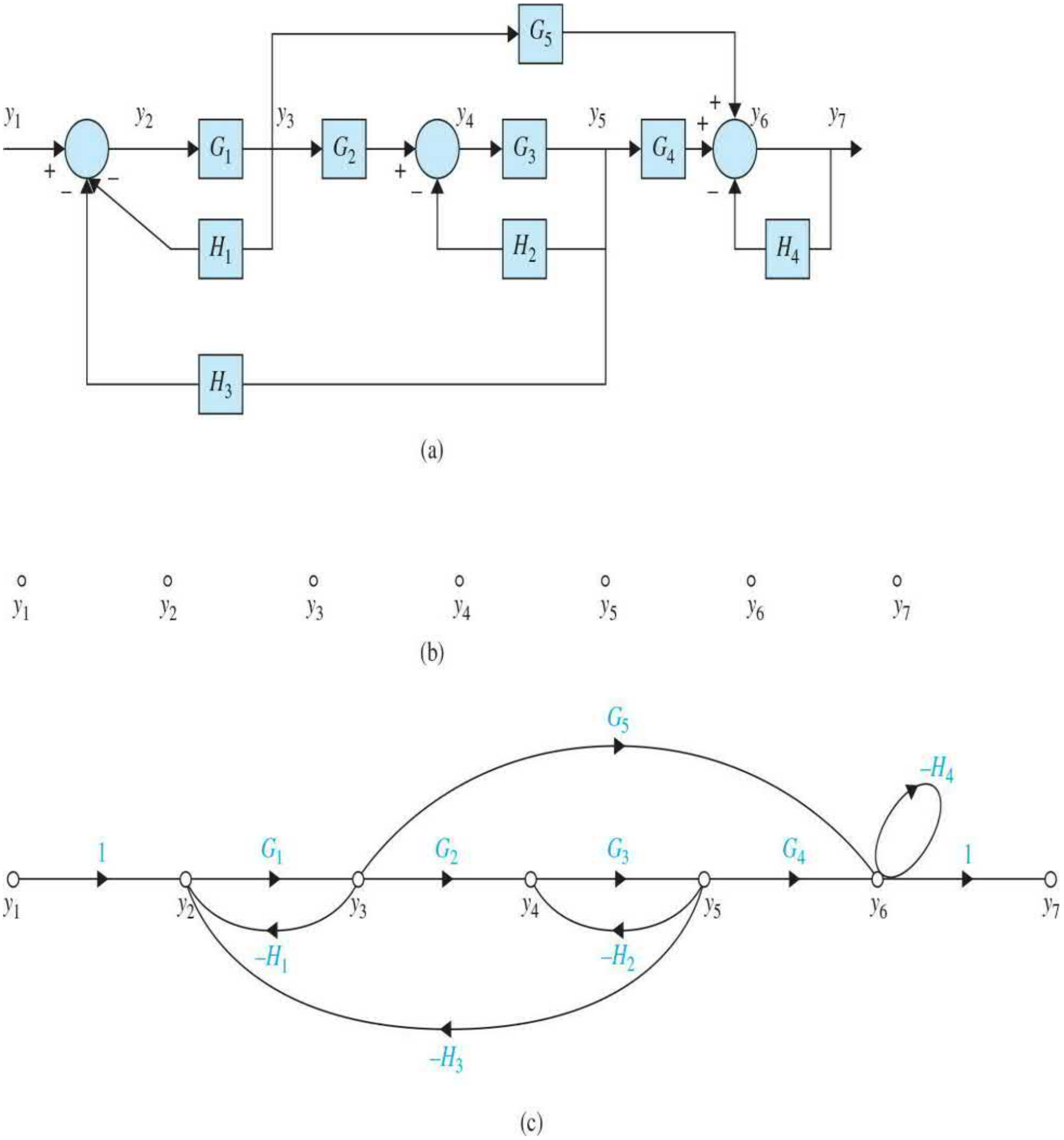
$$\begin{aligned}\Delta &= 1 - (L_{11} + L_{21} + L_{31} + L_{41}) + L_{12} \\ &= 1 - (a_{23}a_{32} + a_{34}a_{43} + a_{24}a_{32}a_{43} + a_{44}) + a_{23}a_{32}a_{44}\end{aligned} \tag{4-63}$$

The reader should verify that choosing y**2** as the output,

$$\frac{y_2}{y_1} = \frac{a_{12}(1 - a_{34}a_{43} - a_{44})}{\Delta} \tag{4-64}$$

where Δ is given in Eq. (4-63). ▲

EXAMPLE 4-2-5 We can convert the block diagram in Fig. 4-32*a* to an SFG format in Fig. 4-32*c*, by first associating all block diagram variables $y_1 - y_7$ to a node as in Fig. 4-32*b*. Next using branches we interconnect the nodes while ensuring the branch directions match the signal directions in the block diagram. Then we label each branch with the appropriate gain corresponding to a transfer function in Fig. 4-32*a*. Make sure to incorporate the negative feedback signs into the gains (i.e., – $H_1(s)$, – $H_2(s)$, – $H_3(s)$, and – $H_4(s)$)—see Fig. 4-32*c*.



(a)

(b)

(c)

Figure 4-32 (*a*) Block diagram of a control system. (*b*) Signal nodes representing the variables. (*c*) Equivalent signal-flow graph.

The two forward paths between y_1 and y_7 and the **forward-path gains** are

Forward path	Gain
$y_1 - y_2 - y_3 - y_4 - y_5 - y_6 - y_7$	$M_1 = G_1G_2G_3G_4$
$y_1 - y_2 - y_3 - y_6 - y_7$	$M_2 = G_1G_5$

The **four loops** of the SFG are shown in Fig. 4-32. The loop gains are

Loop	Gain
$y_2 - y_3 - y_2$	$L_{11} = -G_1H_1$
$y_4 - y_5 - y_4$	$L_{21} = -G_3H_2$
$y_2 - y_3 - y_4 - y_5 - y_2$	$L_{31} = -G_1G_2G_3H_3$
$y_6 - y_7 - y_6$	$L_{41} = -H_4$

The three following loops are **nontouching**

$$y_2 - y_3 - y_2, \quad y_4 - y_5 - y_4 \quad \text{and} \quad y_6 - y_7 - y_6$$

Thus, the products of the gains of two of the three nontouching loops are

$$L_{12} = G_1G_3H_1H_2, \quad L_{22} = G_1H_1H_4 \quad \text{and} \quad L_{32} = G_3H_2H_4 \tag{4-65}$$

Also the following two loops are **nontouching**

$$y_2 - y_3 - y_4 - y_5 - y_2 \quad \text{and} \quad y_6 - y_7 - y_6$$

Thus, the product of the gains of the nontouching loops is

$$L_{42} = G_1G_2G_3H_3H_4 \tag{4-66}$$

Further, the product of the three nontouching loop gains is

$$L_{13} = -G_1G_3H_1H_2H_4 \tag{4-67}$$

Hence,

$$\begin{aligned}\Delta = 1 &+ G_1H_1 + G_3H_2 + G_1G_2G_3H_3 + H_4 \\ &+ G_1G_3H_1H_2 + G_1H_1H_4 + G_3H_2H_4 + G_1G_2G_3H_3H_4 + G_1G_3H_1H_2H_4\end{aligned} \tag{4-68}$$

All the loops are in touch with forward paths M_1. Thus, $\Delta_1 = 1$. Loop $y_4 - y_5 - y_4$ is not in touch with forward path M_2. Thus,

$$\Delta_2 = 1 + G_3H_2 \tag{4-69}$$

Substituting these quantities into Eq. (4-54), we have

$$\frac{y_6}{y_1} = \frac{y_7}{y_1}\frac{M_1\Delta_1 + M_2\Delta_2}{\Delta} = \frac{G_1G_2G_3G_4 + G_1G_5(1 + G_3H_2)}{\Delta} \tag{4-70}$$

The following input-output relations may also be obtained by use of the gain formula:

$$\frac{y_2}{y_1} = \frac{1 + G_3H_2 + H_4 + G_3H_2H_4}{\Delta} \tag{4-71}$$

$$\frac{y_4}{y_1} = \frac{G_1G_2(1 + H_4)}{\Delta} \tag{4-72}$$

▲

4-2-4 Application of the Gain Formula between Output Nodes and Noninput Nodes

It was pointed out earlier that the gain formula can only be applied between a pair of input and output nodes. Often, it is of interest to find the relation between an output-node variable and a noninput-node variable. For example, in the SFG of Fig. 4-32, it may be of interest to find the relation y_7/y_2, which represents the dependence of y_7 upon y_2; the latter is not an input.

We can show that, by including an input node, the gain formula can still be applied to find the gain between a noninput node and an output node. Let y_{in}

be an input and y_{out} be an output node of an SFG. The gain, y_{out}/y_2, where y_2 is not an input, may be written as

$$\frac{y_{\text{out}}}{y_2} = \frac{\dfrac{y_{\text{out}}}{y_{\text{in}}}}{\dfrac{y_2}{y_{\text{in}}}} = \frac{\dfrac{\Sigma M_k \Delta_k \big|_{\text{from } y_{\text{in}} \text{ to } y_{\text{out}}}}{\Delta}}{\dfrac{\Sigma M_k \Delta_k \big|_{\text{from } y_{\text{in}} \text{ to } y_2}}{\Delta}} \tag{4-73}$$

Because Δ is independent of the inputs and the outputs, the last equation is written as

$$\frac{y_{\text{out}}}{y_2} = \frac{\Sigma M_k \Delta_k \big|_{\text{from } y_{\text{in}} \text{ to } y_{\text{out}}}}{\Sigma M_k \Delta_k \big|_{\text{from } y_{\text{in}} \text{ to } y_2}} \tag{4-74}$$

Notice that Δ does not appear in the last equation.

EXAMPLE 4-2-6 From the SFG in Fig. 4-32, the gain between y_2 and y_7 is written as

$$\frac{y_7}{y_2} = \frac{y_7/y_1}{y_2/y_1} = \frac{G_1G_2G_3G_4 + G_1G_5(1+G_3H_2)}{1+G_3H_2+H_4+G_3H_2H_4} \tag{4-75}$$

EXAMPLE 4-2-7 Consider the block diagram shown in Fig. 4-27*a*. The equivalent SFG of the system is shown in Fig. 4-27*c*. Notice that since a node on the SFG is interpreted as the summing point of all incoming signals to the node, the negative feedback on the block diagram is represented by assigning negative gains to the feedback paths on the SFG. First, we can identify the forward paths and loops in the system and their corresponding gains. That is,

Forward path	**Gain**
$R-E-Y_3-Y_2-Y_1-Y$	$M_1 = G_1G_2G_3$
$R-E-Y_3-Y_2-Y$	$M_2 = G_1G_4$

The **four loops** of the SFG are shown in Fig. 4-28. The loop gains are

Loop	Gain
$y_2 - y_3 - y_2$	$L_{11} = -G_1G_2H_1$
$y_4 - y_5 - y_4$	$L_{21} = -G_2G_3H_2$
$y_2 - y_3 - y_4 - y_5 - y_2$	$L_{31} = -G_1G_2G_3$
$y_6 - y_7 - y_6$	$L_{41} = -G_1G_4$

Note that all loops touch. Hence, the closed-loop transfer function of the system is obtained by applying Eq. (4-54) to either the block diagram or the SFG in Fig. 4-27. That is,

$$\frac{Y(s)}{R(s)} = \frac{G_1G_2G_3 + G_1G_4}{\Delta} \tag{4-76}$$

where

$$\Delta = 1 + G_1G_2H_1 + G_2G_3H_2 + G_1G_2G_3 + G_4H_2 + G_1G_4 \tag{4-77}$$

Similarly,

$$\frac{E(s)}{R(s)} = \frac{1 + G_1G_2H_1 + G_2G_3H_2 + G_4H_2}{\Delta} \tag{4-78}$$

$$\frac{Y(s)}{E(s)} = \frac{G_1G_2G_3 + G_1G_4}{1 + G_1G_2H_1 + G_2G_3H_2 + G_4H_2} \tag{4-79}$$

The last expression is obtained using Eq. (4-74).

4-2-5 Simplified Gain Formula

From Example 4-2-7, we can see that *all loops and forward paths are touching* in this case. As a general rule, if there are no nontouching loops and forward paths (e.g., $y_2 - y_3 - y_2$ and $y_4 - y_4$ in Example 4-2-3) in the block diagram or SFG of the system, then Eq. (4-54) takes a far simpler look, as shown next.

$$M = \frac{y_{\text{out}}}{y_{\text{in}}} = \sum \frac{\text{Forward Path Gains}}{1 - \text{Loop Gains}} \tag{4-80}$$

EXAMPLE 4-2-8 For Example 4-2-5, where there are nontouching loops, as seen in Fig. 4-33, the simplified gain formula can be used by eliminating the nontouching loops after some block diagram manipulations.

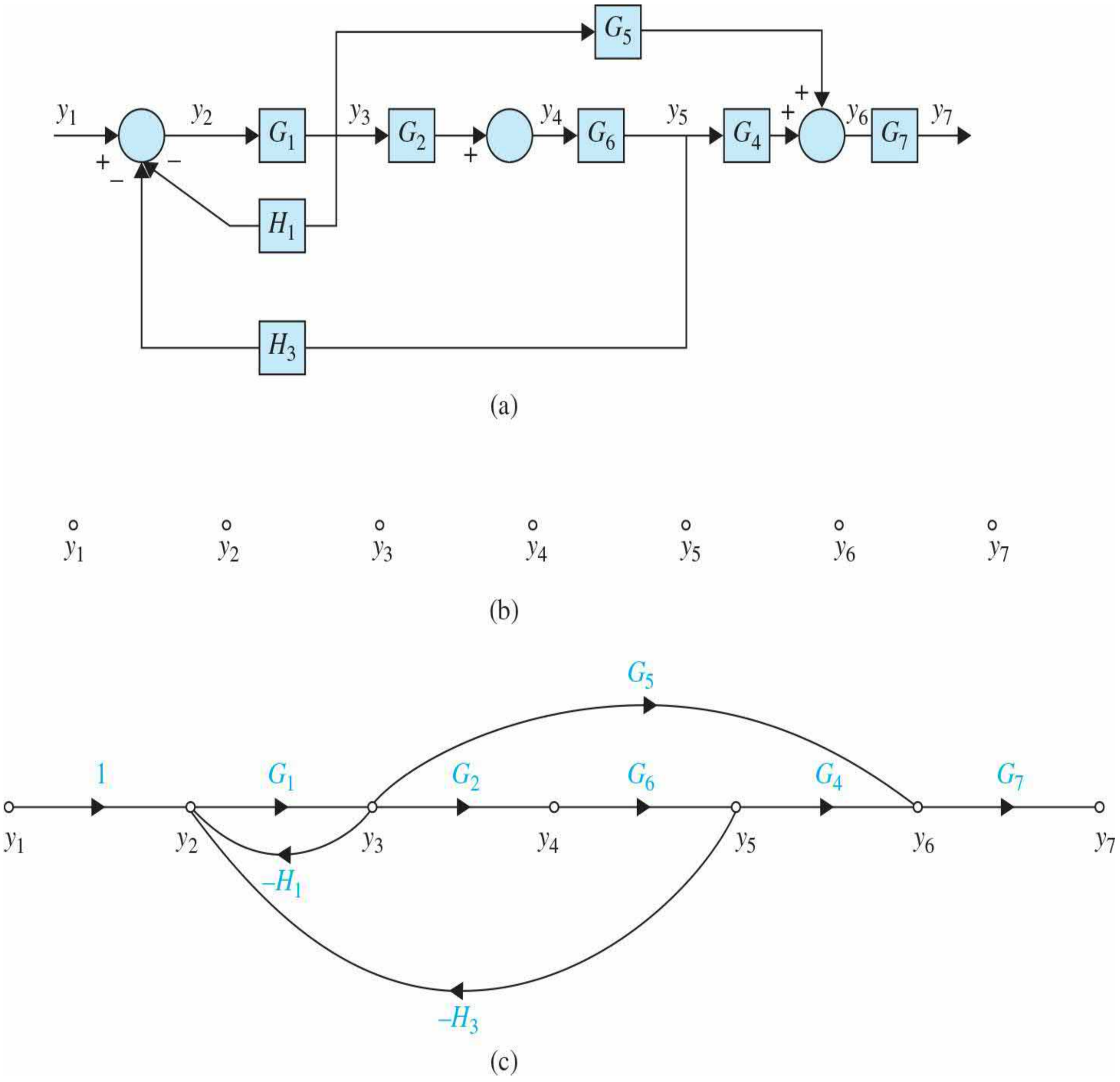


Figure 4-33 (*a*) Modified block diagram of the control system in Fig. 4-32 to eliminate the nontouching loops. (*b*) Signal nodes representing the

variables. (*c*) Equivalent signal-flow graph.

The two forward paths between y_1 and y_7 and the **forward-path gains** are now

Forward path	Gain
$y_1 - y_2 - y_3 - y_4 - y_5 - y_6 - y_7$	$M_1 = G_1G_2G_6G_4G_7$
$y_1 - y_2 - y_3 - y_6 - y_7$	$M_2 = G_1G_5G_7$

The two **touching loops** of the SFG are shown in Fig. 4-33. The loop gains are

Loop	Gain
$y_2 - y_3 - y_2$	$L_{11} = -G_1H_1$
$y_2 - y_3 - y_4 - y_5 - y_2$	$L_{21} = -G_1G_2G_6H_3$

Note in this case

$$G_6 = \frac{G_3}{1+G_3H_2} \quad \text{and} \quad G_7 = \frac{1}{1+H_4} \tag{4-81}$$

Hence,

$$\begin{aligned}\Delta = 1 &+ G_1H_1 + G_3H_2 + G_1G_2G_3H_3 + H_4 \\ &+ G_1G_3H_1H_2 + G_1H_1H_4 + G_3H_2H_4 + G_1G_2G_3H_3H_4 + G_1G_3H_1H_2H_4\end{aligned} \qquad (4\text{-}82)$$

As a result,

$$\frac{Y(s)}{R(s)} = \frac{G_1G_2G_3 + G_1G_4}{\Delta} \qquad (4\text{-}83$$

4-3 STATE DIAGRAM

In this section, we introduce the state diagram, which is an extension of the SFG to portray state equations and differential equations. A state diagram is constructed following all the rules of the SFG using the Laplace-transformed state equations. The basic elements of a state diagram are similar to the conventional SFG, except for the **integration** operation.

Let the variables $x_1(t)$ and $x_2(t)$ be related by the first-order differentiation:

$$\frac{dx_1(t)}{dt} = x_2(t) \tag{4-84}$$

Integrating both sides of the last equation with respect to t from the initial time t_0, we get

$$x_1(t) = \int_{t_0}^{t} x_2(\tau)\,d\tau + x_1(t_0) \tag{4-85}$$

Because the SFG algebra does not handle integration in the time domain, we must take the Laplace transform on both sides of Eq. (4-85). We have

$$X_1(s) = \mathcal{L}\left[\int_{t_0}^{t} x_2(\tau)\,d\tau\right] + \frac{x_1(t_0)}{s} = \frac{X_2(s)}{s} - \left[\int_{0}^{t_0} x_2(\tau)\,d\tau\right] + \frac{x_1(t_0)}{s} \tag{4-86}$$

Because the past history of the integrator is represented by $x_1(t_0)$, and the state transition is assumed to start at $\tau = t_0$, $x_2(\tau) = 0$ for $0 < \tau < t_0$. Thus, Eq. (4-86) becomes

$$X_1(s) = \frac{X_2(s)}{s} + \frac{x_1(t_0)}{s} \qquad \tau \geq t_0 \tag{4-87}$$

Equation (4-83) is now algebraic and can be represented by an SFG, as shown in Fig. 4-34, where *the output of the integrator is equal to* s^{-1} *times the input, plus the initial condition* $x_1(t_0)/s$. An alternative SFG with fewer elements for Eq. (4-87) is shown in Fig. 4-35.

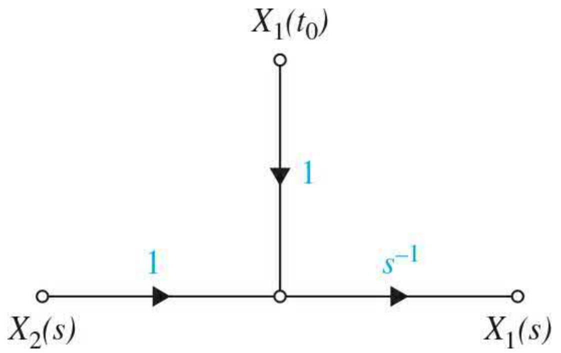


Figure 4-34 Signal-flow graph representation of $X_1(s)=[X_2(s)/s]+[x_1(t_0)/s]$.

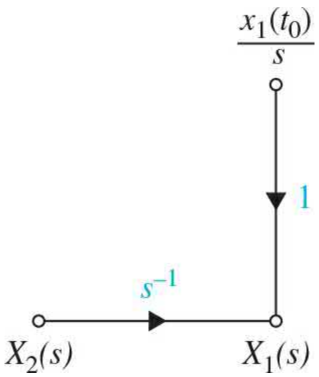


Figure 4-35 San alternative signal-flow graph representation of $X_1(s)=[X_2(s)/s]+[x_1(t_0)/s]$.

4-3-1 From Differential Equations to State Diagrams

When a linear system is described by a high-order differential equation, a state diagram can be constructed from these equations, although a direct approach is not always the most convenient. Consider the following differential equation:

$$\frac{d^n y(t)}{dt^n}+a_n\frac{d^{n-1}y(t)}{dt^{n-1}}+\cdots+a_2\frac{dy(t)}{dt}+a_1y(t)=r(t) \tag{4-88}$$

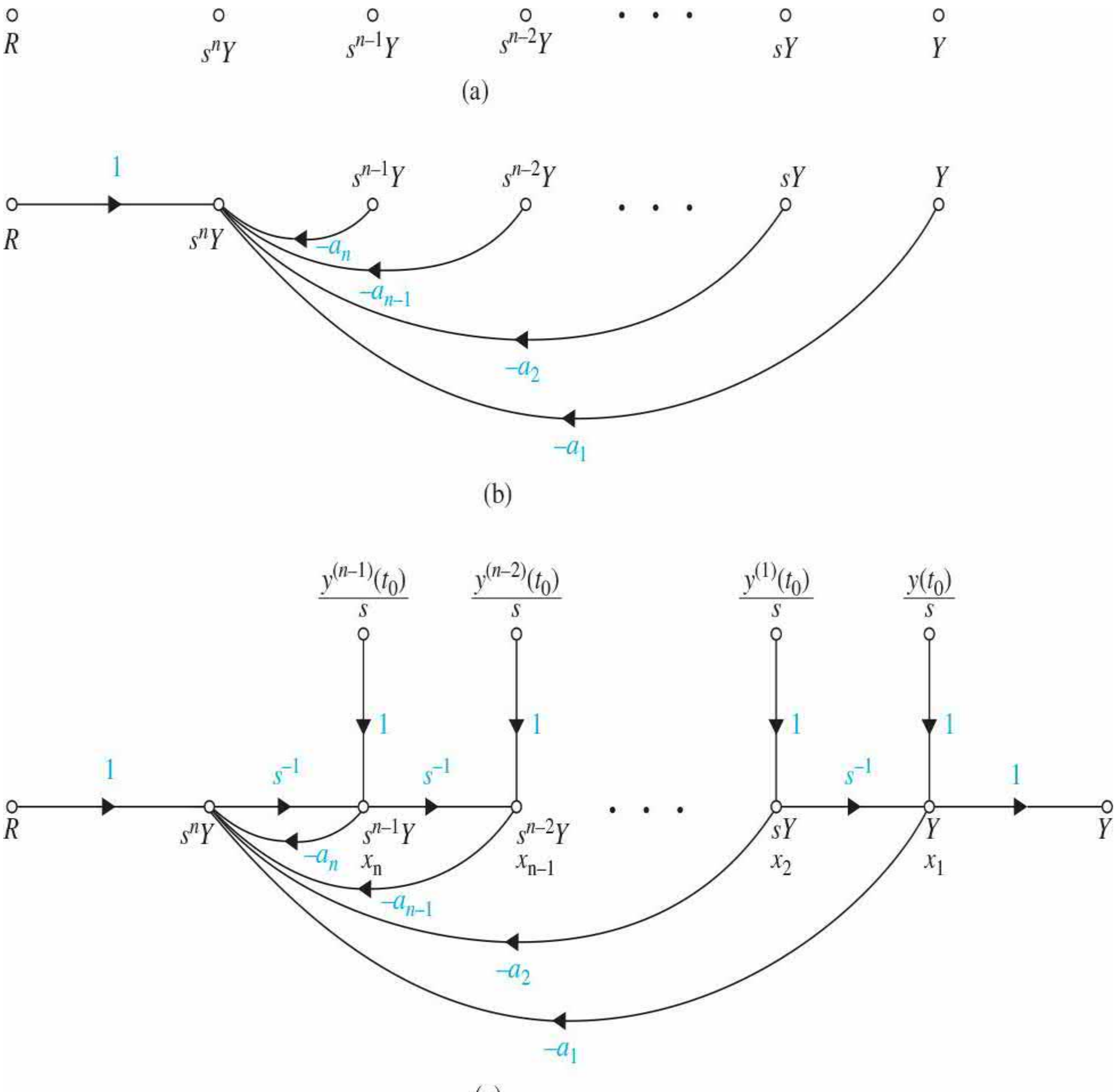
To construct a state diagram using this equation, we rearrange the equation as

$$\frac{d^n y(t)}{dt^n}=-a_n\frac{d^{n-1}y(t)}{dt^{n-1}}-\cdots-a_2\frac{dy(t)}{dt}-a_1y(t)+r(t) \tag{4-89}$$

The outputs of the integrators in the state diagram are usually defined as the state variable.

The process is highlighted next.

1. The nodes representing $R(s), s^nY(s), s^{n-1}Y(s), \ldots, sY(s),$ and $Y(s)$ are arranged from left to right, as shown in Fig. 4-36*a*.
2. Because $s^iY(s)$ corresponds to $d^iy(t)/dt^i$, $i = 0, 1, 2, \ldots, n$, in the Laplace domain, the nodes in Fig. 4-36*a* are connected by branches to portray Eq. (4-85), resulting in Fig. 4-36*b*.



(a)

(b)

(c)

Figure 4-36 State-diagram representation of the differential equation of Eq. (4-89).

3. Finally, the integrator branches with gains of s^{-1} are inserted, and the initial conditions are added to the outputs of the integrators, according to the basic scheme in Fig. 4-35.

The complete state diagram is drawn as shown in Fig. 4-36*c*. *The outputs of the integrators are defined as the state variables*, $x_1, x_2, \ldots, x_n$. This is usually the natural choice of state variables once the state diagram is drawn.

When the differential equation has derivatives of the input on the right side, the problem of drawing the state diagram directly is not as straightforward as just illustrated. We will show that, in general, it is more convenient to obtain the transfer function from the differential equation first and then arrive at the state diagram through decomposition (Sec. 8-10).

The outputs of the integrators in the state diagram are usually defined as the state variable.

EXAMPLE 4-3-1 Consider the differential equation

$$\frac{d^2y(t)}{dt^2} + 3\frac{dy(t)}{dt} + 2y(t) = r(t) \tag{4-90}$$

Equating the highest-ordered term of the last equation to the rest of the terms, we have

$$\frac{d^2y(t)}{dt^2} = -3\frac{dy(t)}{dt} - 2y(t) + r(t) \tag{4-91}$$

Following the procedure just outlined, the state diagram of the system is drawn as shown in Fig. 4-37. The state variables x_1 and x_2 are assigned as shown. ▲

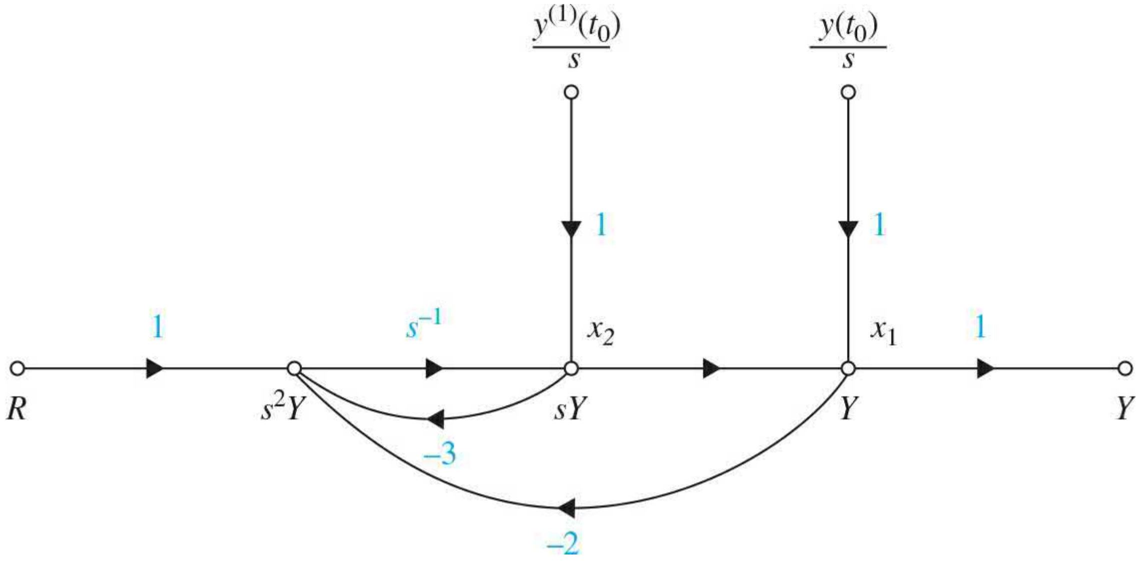


Figure 4-37 State diagram for Eq. (4-89).

4-3-2 From State Diagrams to Transfer Functions

The transfer function between an input and an output is obtained from the state diagram by using the gain formula and setting all other inputs and initial states to zero. The following example shows how the transfer function is obtained directly from a state diagram.

EXAMPLE 4-3-2 Consider the state diagram of Fig. 4-37. The transfer function between $R(s)$ and $Y(s)$ is obtained by applying the gain formula between these two nodes and setting the initial states to zero. We have

$$\frac{Y(s)}{R(s)}=\frac{1}{s^2+3s+2} \tag{4-92}$$

▲

4-3-3 From State Diagrams to State and Output Equations

The state equations and the output equations can be obtained directly from the state diagram by using the SFG gain formula. The general form of a state equation and the output equation for a linear system is described in Chap. 3 and presented here.

State equation:

$$\frac{dx(t)}{dt} = ax(t) + br(t) \tag{4-93}$$

Output equation:

$$y(t) = cx(t) + dr(t) \tag{4-94}$$

where $x(t)$ is the state variable; $r(t)$ is the input; $y(t)$ is the output; and a, b, c, and d are constant coefficients. Based on the general form of the state and output equations, the following procedure of deriving the state and output equations from the state diagram are outlined:

1. Delete the initial states and the integrator branches with gains s^{-1} from the state diagram, since the state and output equations do not contain the Laplace operator s or the initial states.
2. For the state equations, regard the nodes that represent the derivatives of the state variables as output nodes, since these variables appear on the left-hand side of the state equations. The output $y(t)$ in the output equation is naturally an output node variable.
3. Regard the state variables and the inputs as input variables on the state diagram, since these variables are found on the right-hand side of the state and output equations.
4. Apply the SFG gain formula to the state diagram.

EXAMPLE 4-3-3 Figure 4-38 shows the state diagram of Fig. 4-37 with the integrator branches and the initial states eliminated. Using $dx_1(t)/dt$ and $dx_2(t)/dt$ as the output nodes and $x_1(t)$, $x_2(t)$, and $r(t)$ as input nodes, and applying the gain formula between these nodes, the state equations are obtained as

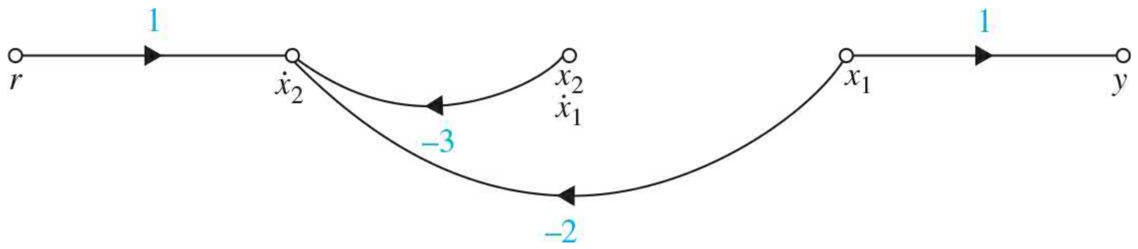


Figure 4-38 State diagram of Fig. 4-37 with the initial states and the

integrator branches left out.

$$\frac{dx_1(t)}{dt} = x_2(t) \tag{4-95}$$

$$\frac{dx_2(t)}{dt} = -2x_1(t) - 3x_2(t) + r(t) \tag{4-96}$$

Applying the gain formula with $x_1(t)$, $x_2(t)$, and $r(t)$ as input nodes and $y(t)$ as the output node, the output equation is written as

$$y(t) = x_1(t) \tag{4-97}$$

Note that for the complete state diagram, shown in Fig. 4-37 with t_0 as the initial time. The outputs of the integrators are assigned as state variables. Applying the gain formula to the state diagram in Fig. 4-37, with $X_1(s)$ and $X_2(s)$ as output nodes and $x_1(t_0)$, $x_2(t_0)$, and $R(s)$ as input nodes, we have

$$X_1(s) = \frac{s^{-1}(1+3s^{-1})}{\Delta} x_1(t_0) + \frac{s^{-2}}{\Delta} x_2(t_0) + \frac{s^{-2}}{\Delta} R(s) \tag{4-98}$$

$$X_2(s) = \frac{-2s^{-2}}{\Delta} x_1(t_0) + \frac{s^{-1}}{\Delta} x_2(t_0) + \frac{s^{-1}}{\Delta} R(s) \tag{4-99}$$

where

$$\Delta = 1 + 3s^{-1} + 2s^{-2} \tag{4-100}$$

After simplification, Eqs. (4-98) and (4-99) are presented in vector-matrix form:

$$\begin{bmatrix} X_1(s) \\ X_2(s) \end{bmatrix} = \frac{1}{(s+1)(s+2)} \begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix} \begin{bmatrix} x_1(t_0) \\ x_2(t_0) \end{bmatrix} + \frac{1}{(s+1)(s+2)} \begin{bmatrix} 1 \\ s \end{bmatrix} R(s) \tag{4-101}$$

Note that Eq. (4-100) may also be obtained by taking the Laplace transform of Eqs. (4-95) and (4-96). For zero initial conditions, and since $Y(s) = X_1(s)$, the output-input transfer function is

$$\frac{Y(s)}{R(s)} = \frac{1}{s^2 + 3s + 2} \tag{4-102}$$

which is the same as Eq. (4-88). ▲

EXAMPLE 4-3-4 As another example on the determination of the state equations from the state diagram, consider the state diagram shown in Fig. 4-39*a*. This example will also emphasize the importance of applying the gain formula. Figure 4-39*b* shows the state diagram with the initial states and the integrator branches deleted. Notice that, in this case, the state diagram in Fig. 4-39*b* still contains a loop. By applying the gain formula to the state diagram in Fig. 4-39*b* with $\dot{x}_1(t)$, $\dot{x}_2(t)$, and $\dot{x}_3(t)$ as output-node variables and $r(t)$, $x_1(t)$, $x_2(t)$, and $x_3(t)$ as input nodes, the state equations are obtained as follows in vector-matrix form:

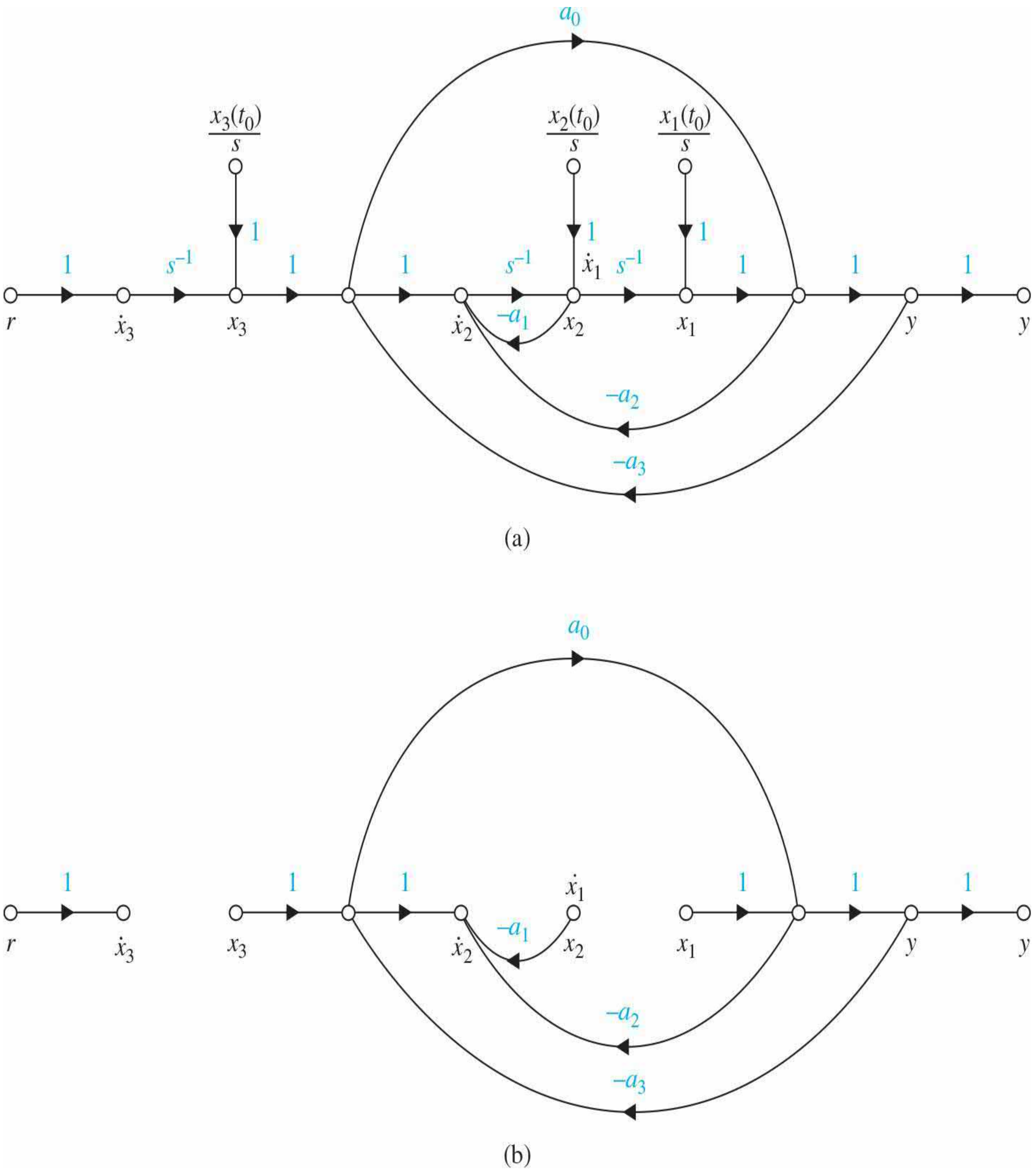


Figure 4-39 (*a*) State diagram. (*b*) State diagram in part (*a*) with all initial states and integrators left out.

$$\begin{bmatrix} \dfrac{dx_1(t)}{dt} \\ \dfrac{dx_2(t)}{dt} \\ \dfrac{dx_3(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ \dfrac{-(a_2+a_3)}{1+a_0a_3} & -a_1 & \dfrac{1-a_0a_2}{1+a_0a_3} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} r(t) \tag{4-103}$$

The output equation is

$$y(t) = \frac{1}{1+a_0a_3}x_1(t) + \frac{a_0}{1+a_0a_3}x_3(t) \tag{4-104}$$

4-4 CASE STUDIES

EXAMPLE 4-4-1 Consider the mass-spring-damper system shown in Fig. 4-40*a*. The linear motion concerned is in the horizontal direction. The free-body diagram of the system is shown in Fig. 4-40*b*. Following the procedure outlined in Sec. 2-1-1, the equation of motion may be written into an input-output form as

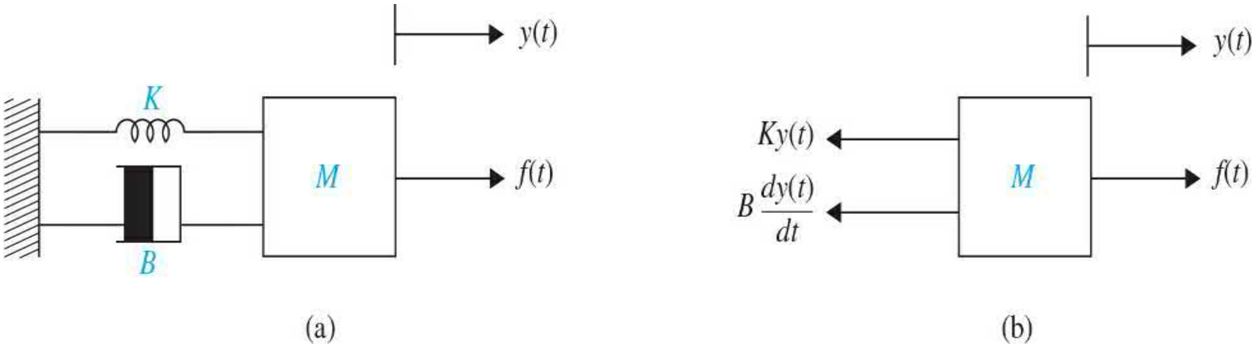


Figure 4-40 (*a*) Mass-spring-friction system. (*b*) Free-body diagram.

$$\ddot{y}(t) + \frac{B}{M}y(t) + \frac{K}{M}\dot{y}(t) = \frac{1}{M}f(t) \tag{4-105}$$

where *y*(*t*) is the output, $\frac{f(t)}{M}$ is considered the input, and $\dot{y}(t)=\left(\frac{dy(t)}{dt}\right)$ and $\ddot{y}(t)=\left(\frac{d^2y(t)}{dt^2}\right)$ represent velocity and acceleration, respectively.

For **zero initial conditions**, the transfer function between *Y*(*s*) and *F*(*s*) is obtained by taking the Laplace transform on both sides of Eq. (4-105):

$$Y(s)\left(s^2+\frac{B}{M}s+\frac{K}{M}\right)=\frac{F(s)}{M} \tag{4-106}$$

Hence,

$$\frac{Y(s)}{F(s)}=\frac{1}{Ms^2+Bs+K} \tag{4-107}$$

The same result is obtained by applying the gain formula to the block diagram, which is shown in Fig. 4-41.

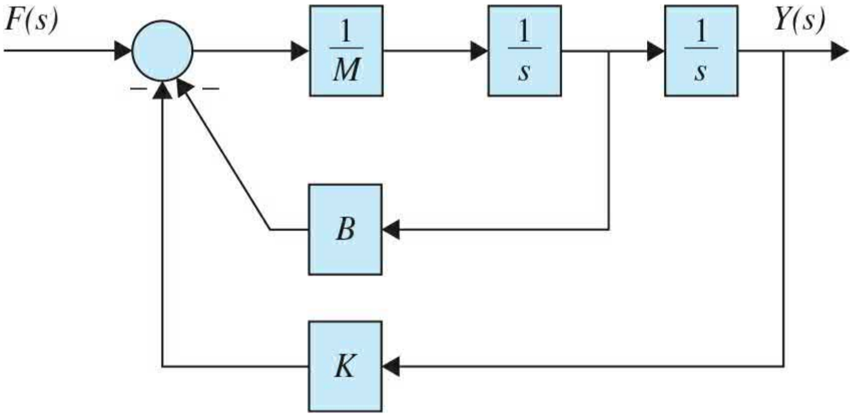


Figure 4-41 Block diagram representation of mass-spring-damper system of Eq. (4-106).

Equation (4-105) may also be represented in the space state form

$$\dot{\mathbf{x}}(t)=\mathbf{A}\mathbf{x}(t)+\mathbf{B}\mathbf{u}(t) \tag{4-107}$$

where

$$\mathbf{x}(t)=\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \tag{4-108}$$

and

$$\mathbf{u}(t)=\frac{f(t)}{M} \tag{4-109}$$

The output equation is

$$y(t)=x_1(t) \tag{4-110}$$

So Eq. (4-107) is rewritten as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix}=\begin{bmatrix} 0 & 1 \\ -\dfrac{K}{M} & -\dfrac{B}{M} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}+\frac{f(t)}{M} \tag{4-111}$$

The state Eq. (4-111) may also be written as a set of first-order differential equations:

$$\begin{aligned} \frac{dx_1(t)}{dt} &= x_2(t) \\ \frac{dx_2(t)}{dt} &= -\frac{K}{M}x_1(t)-\frac{B}{M}x_2(t)+\frac{1}{M}f(t) \\ y(t) &= x_1(t) \end{aligned} \tag{4-112}$$

For zero initial conditions, the transfer function between *Y*(*s*) and *F*(*s*) is obtained by taking the Laplace transform on both sides of Eq. (4-112):

$$\begin{aligned} sX_1(s) &= X_2(s) \\ sX_2(s) &= -\frac{B}{M}X_2(s)-\frac{K}{M}X_1(s)+\frac{1}{M}F(s) \\ Y(s) &= X_1(s) \end{aligned} \tag{4-113}$$

resulting in

$$\frac{Y(s)}{F(s)} = \frac{1}{Ms^2 + Bs + K} \tag{4-114}$$

The block diagram associated with Eq. (4-113) is shown in Fig. 4-42. Note that this block diagram may also be obtained directly from the block diagram in Fig. 4-41 by factoring out the 1/*M* term. The transfer function in Eq. (4-114) may also be obtained by applying the gain formula to the block diagram in Fig. 4-42.

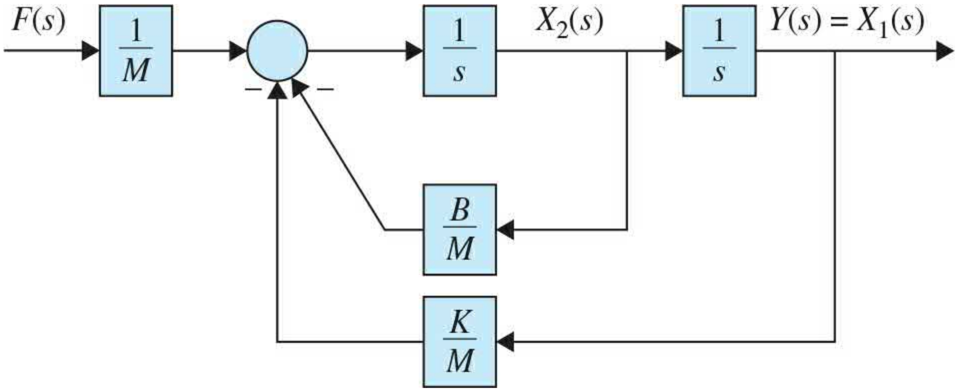


Figure 4-42 Block diagram representation of mass-spring-damper system of shown in Fig. 4-41.

For nonzero initial conditions, Eq. (4-112) has a different Laplace transform representation that may be written as

$$\begin{aligned} sX_1(s) - x_1(0) &= X_2(s) \\ sX_2(s) - x_2(0) &= -\frac{B}{M}X_2(s) - \frac{K}{M}X_1(s) + \frac{1}{M}F(s) \\ Y(s) &= X_1(s) \end{aligned} \tag{4-115}$$

The corresponding SFG representation for Eq. (4-115) is shown in in Fig. 4-43.

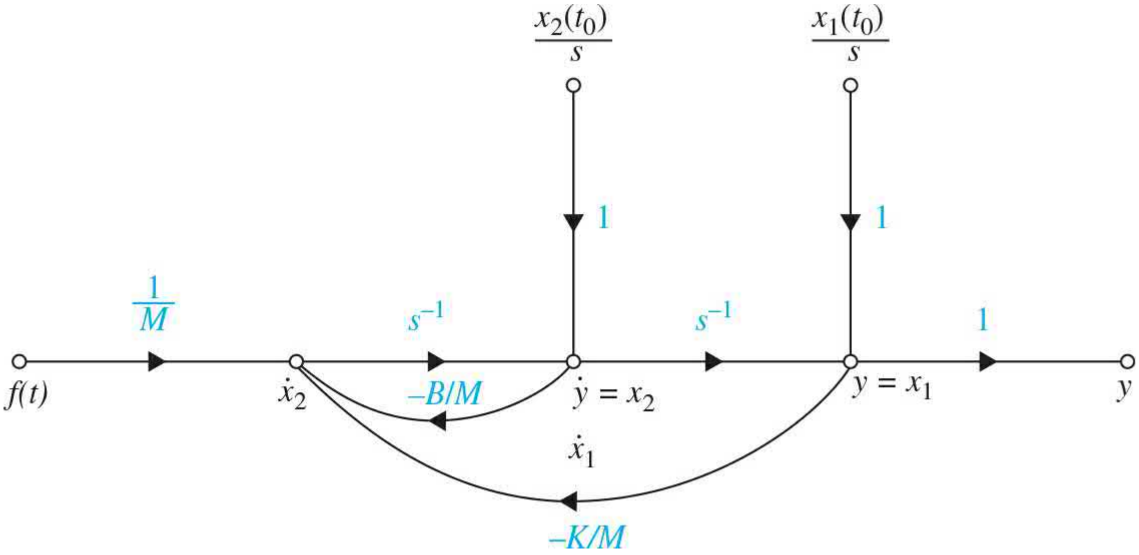


Figure 4-43 SFG representation of mass-spring-damper system of Eq. (4-115) with nonzero initial conditions $x_1(t_0)$ and $x_2(t_0)$.

Upon simplifying Eq. (4-115) or by applying the gain formula to the SFG representations of the system, the output becomes

$$Y(s) = \frac{1}{Ms^2 + Bs + K} F(s) + \frac{Ms}{Ms^2 + Bs + K} x_1(t_0) + \frac{M}{Ms^2 + Bs + K} x_2(t_0) \tag{4-116}$$

Toolbox 4-4-1

Time domain step response for Eq. (4-114) is calculated using MATLAB for $K = 1$, $M = 1$, $B = 1$:

```
K=1; M=1; B=1;
t=0 : 0.02: 30;
num = [1];
den = [M B K];
G = tf(num, den);
y1 = step (G, t);
plot(t, y1);
xlabel(‘Time (Second)’) ; ylabel (‘Step Response’)
title (‘Response of the system in Eq. (4-114) to step input’)
```

The step response of the system in Eq. (4-114) is shown in Fig. 4-44.

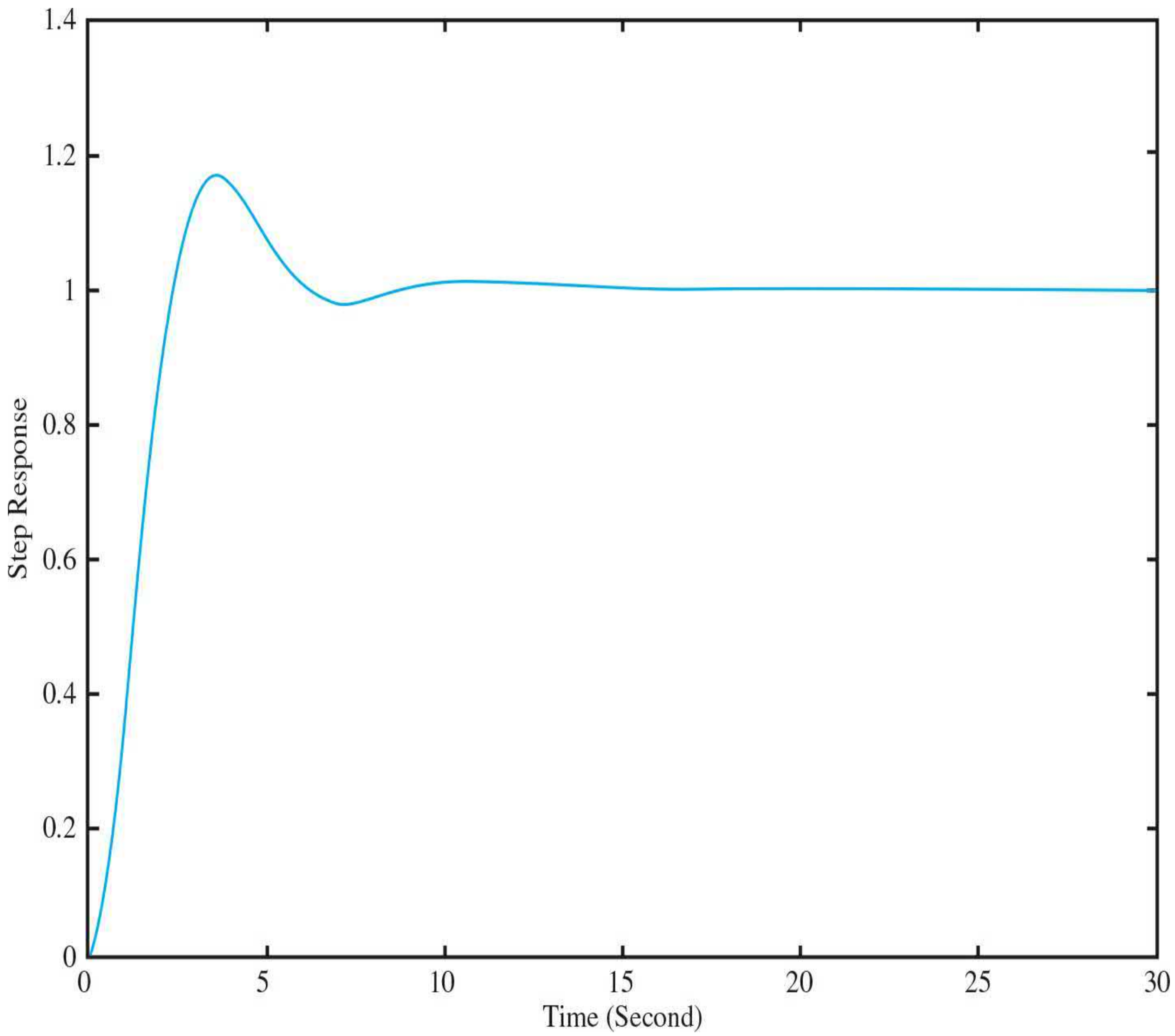


Figure 4-44 Time response of Eq. (4-114) for a unit step input.

EXAMPLE 4-4-2 Consider the system shown in Fig. 4-45*a*. Because the spring is deformed when it is subject to a force $f(t)$, two displacements, y_1 and y_2, must be assigned to the end points of the spring. The free-body diagrams of the system are shown in Fig. 4-45*b*. The force equations are

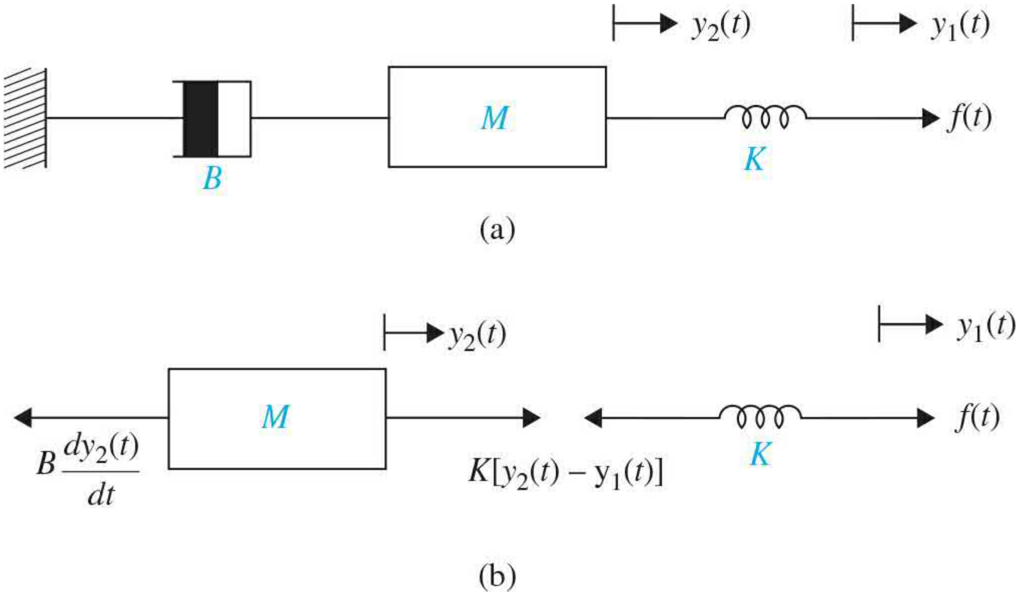


Figure 4-45 Mechanical system for Example 4-4-2. (*a*) Mass-spring-damper system. (*b*) Free-body diagram.

$$f(t) = K[y_1(t) - y_2(t)] \tag{4-117}$$

$$\frac{dy(t)}{dt} + \frac{1}{\tau}y(t) = \frac{1}{\tau}u(t) \tag{3-83}$$

These equations are rearranged in input-output form as

$$\frac{d^2y_2(t)}{dt^2} + \frac{B}{M}\frac{dy_2(t)}{dt} + \frac{K}{M}y_2(t) = \frac{K}{M}y_1(t) \tag{4-119}$$

For zero initial conditions, the transfer function between $Y_1(s)$ and $F_2(s)$ is obtained by taking the Laplace transform on both sides of Eq. (4-118):

$$\frac{Y_1(s)}{Y_2(s)} = \frac{K}{Ms^2 + Bs + K} \tag{4-120}$$

For state representation, the equations may be rearranged as

$$\begin{aligned} y_1(t) &= y_2(t) + \frac{1}{K} f(t) \\ \frac{d^2 y_2(t)}{dt^2} &= -\frac{B}{M}\frac{dy_2(t)}{dt} + \frac{K}{M}[y_1(t) - y_2(t)] \end{aligned} \tag{4-121}$$

The transfer function in Eq. (4-120) may also be obtained by applying the gain formula to the block diagram representation of the system, which is from Eq. (4-121) and is shown in Fig. 4-46. Note that in Fig. 4-46, $F(s)$, $Y_1(s)$, $X_1(s)$, $Y_2(s)$, and $X_2(s)$ are Laplace transforms of $f(t)$, $y_1(t)$, $x_1(t)$, $y_2(t)$, and $x_2(t)$, respectively. For zero initial conditions, the transfer function of Eq. (4-121) is the same as that of Eq. (4-119). By using the last two equations, the state variables are defined as $x_1(t) = y_2(t)$ and $x_2(t) = dy_2(t)/dt$, and ***t*** state equations are therefore written as

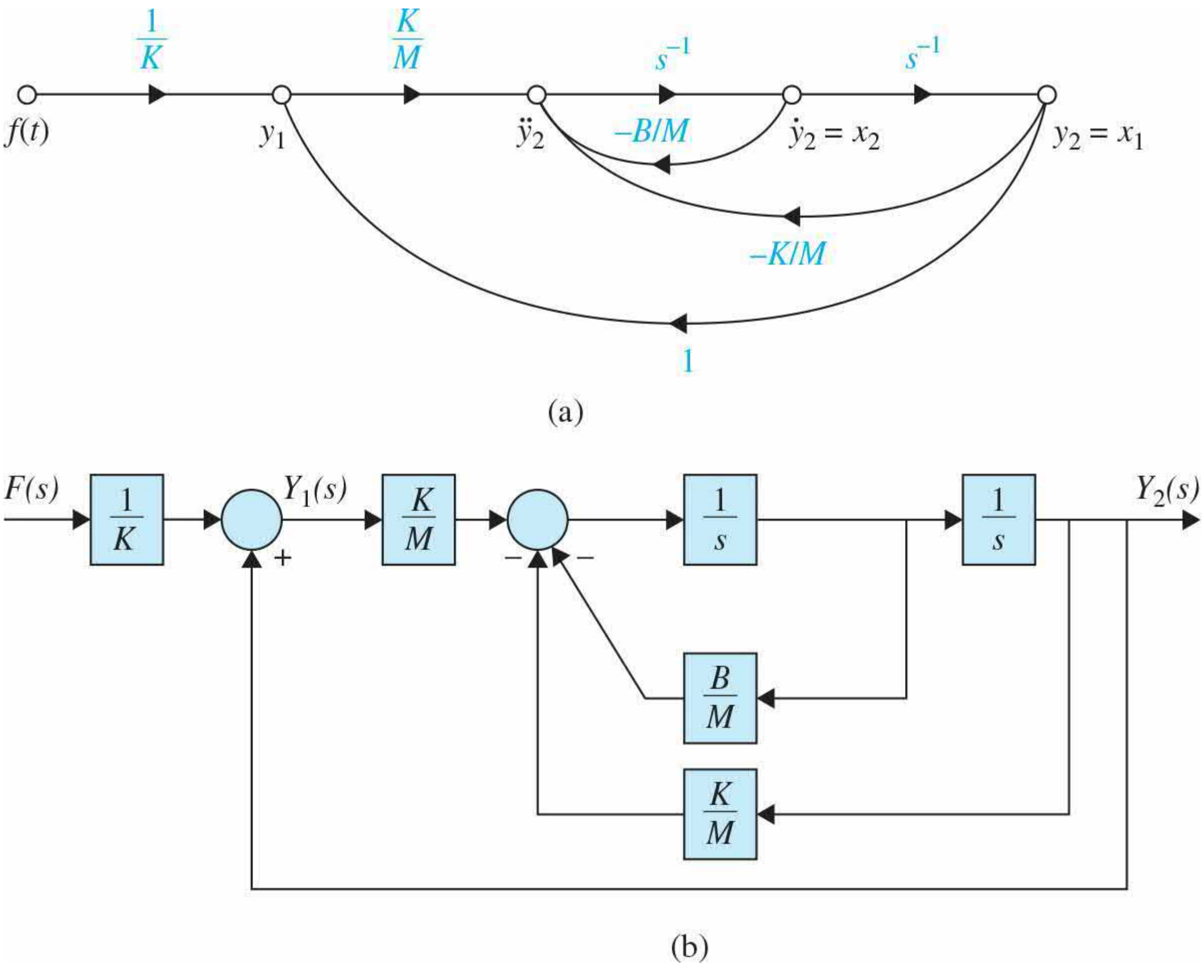


Figure 4-46 Mass-spring-damper system of Eq. (4-121). (*a*) The signal-flow graph representation. (*b*) Block diagram representation.

$$
\begin{aligned}
\frac{dx_1(t)}{dt} &= x_2(t) \\
\frac{dx_2(t)}{dt} &= -\frac{B}{M}x_2(t) + \frac{1}{M}f(t) \\
y_2(t) &= x_1(t)
\end{aligned}
\tag{4-122}
$$

EXAMPLE 4-4-3 Figure 4-47*a* shows the diagram of a motor coupled to an inertial load through a shaft with a spring constant *K*. A nonrigid coupling between two mechanical components in a control system often causes torsional resonances that can be transmitted to all parts of the system. The system variables and parameters are defined as follows:

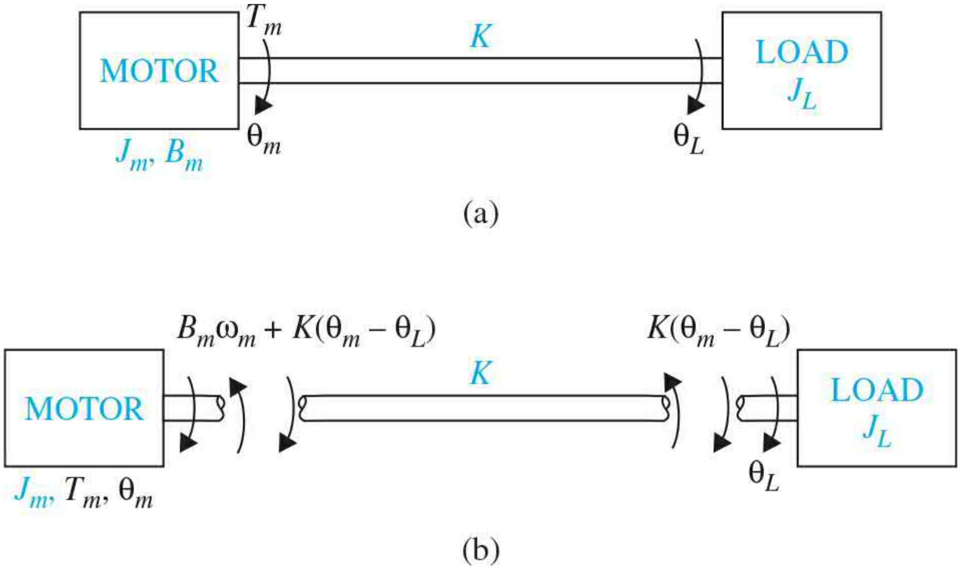


Figure 4-47 (*a*) Motor-load system. (*b*) Free-body diagram.

$T_m(t)$ = motor torque

B_m = motor viscous-friction coefficient

K = spring constant of the shaft

$\theta_m(t)$ = motor displacement

$\omega_m(t)$ = motor velocity

J_m = motor inertia

$\theta_L(t)$ = load displacement

$\omega_L(t)$ = load velocity

J_L = load inertia

The free-body diagrams of the system are shown in Fig. 4-47*b*. The torque equations of the system are

$$\frac{d^2\theta_m(t)}{dt^2} = -\frac{B_m}{J_m}\frac{d\theta_m(t)}{dt} - \frac{K}{J_m}[\theta_m(t) - \theta_L(t)] + \frac{1}{J_m}T_m(t) \tag{4-123}$$

$$K[\theta_m(t) - \theta_L(t)] = J_L\frac{d^2\theta_L(t)}{dt^2} \tag{4-124}$$

In this case, the system contains three energy-storage elements in J_m, J_L, and K. Thus, there should be three state variables. Care should be taken in constructing the state diagram and assigning the state variables so that a minimum number of the latter are incorporated. Equations (4-123) and (4-124) are rearranged as

$$\frac{d^2\theta_m(t)}{dt^2} = -\frac{B_m}{J_m}\frac{d\theta_m(t)}{dt} - \frac{K}{J_m}[\theta_m(t) - \theta_L(t)] + \frac{1}{J_m}T_m(t) \tag{4-125}$$

$$\frac{d^2\theta_L(t)}{dt^2} = \frac{K}{J_L}[\theta_m(t) - \theta_L(t)] \tag{4-126}$$

The state variables in this case are defined as $x_1(t) = \theta_m(t) - \theta_L(t)$, $x_2(t) = d\theta_L(t)/dt$, and $x_3(t) = d\theta_m(t)/dt$. The state equations are

$$\begin{aligned}\frac{dx_1(t)}{dt} &= x_3(t) - x_2(t)\\ \frac{dx_2(t)}{dt} &= \frac{K}{J_L}x_1(t)\\ \frac{dx_3(t)}{dt} &= -\frac{K}{J_m}x_1(t) - \frac{B_m}{J_m}x_3(t) + \frac{1}{J_m}T_m(t)\end{aligned} \tag{4-127}$$

The SFG representation is shown in Fig. 4-48.

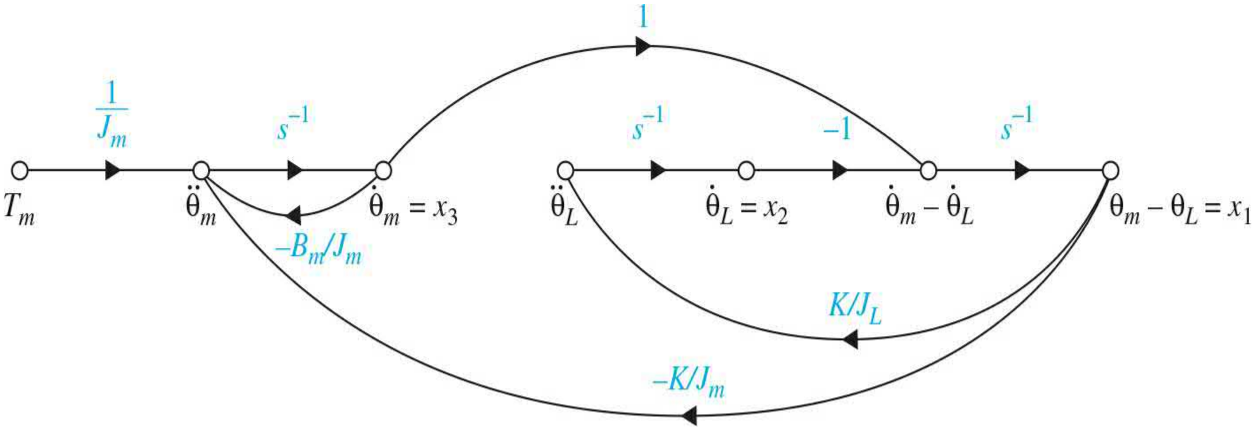
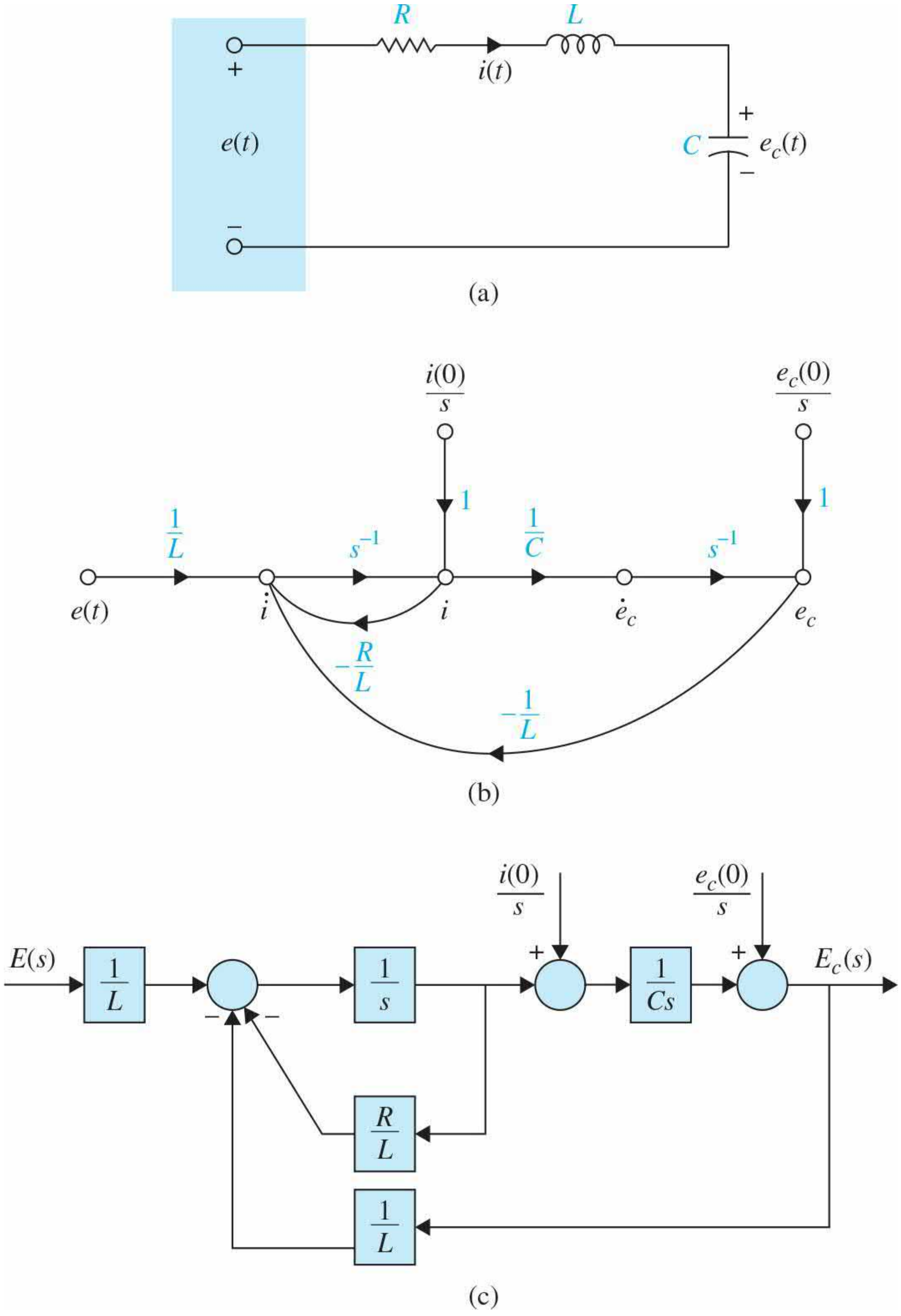


Figure 4-48 Rotational system of Eq. (4-123) signal-flow graph representation.

EXAMPLE 4-4-4 Let us consider the **RLC** network shown in Fig. 4-49*a*. Using the voltage law



(a)

(b)

(c)

Figure 4-49 RLC network. (*a*) Electrical schematics. (*b*) Signal-flow graph representation. (*c*) Block diagram representation.

$$e(t) = e_R + e_L + e_c \tag{4-128}$$

where e_R = voltage across the resistor R
e_L = voltage across the inductor L
e_c = voltage across the capacitor C

Then

$$e(t) = +e_c(t) + Ri(t) + L\frac{di(t)}{dt} \tag{4-129}$$

Taking a derivative of Eq. (4-129) with respect to time, and using the relation for current in *C:*

$$C\frac{de_c(t)}{dt} = i(t) \tag{4-130}$$

we get the equation of the RLC network as

$$L\frac{d^2i(t)}{dt^2} + R\frac{di(t)}{dt} + \frac{i(t)}{C} = \frac{de(t)}{dt} \tag{4-131}$$

A practical approach is to assign the current in the inductor *L*, *i*(*t*), and the voltage across the capacitor *C*, $e_c(t)$, as the state variables. The reason for this choice is because the state variables are directly related to the energy-storage element of a system. The inductor stores kinetic energy, and the capacitor stores electric potential energy. By assigning *i*(*t*) and $e_c(t)$ as state variables, we have a complete description of the past history (via the initial states) and the present and future states of the network. The state equations for the network in Fig. 4-49*b* are written by first equating the current in *C* and the voltage across *L* in terms of the state variables and the applied voltage *e*(*t*). In vector-matrix form, the equations of the system are expressed as

$$\begin{bmatrix} \dfrac{de_c(t)}{dt} \\ \dfrac{di(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & \dfrac{1}{C} \\ -\dfrac{1}{L} & -\dfrac{R}{L} \end{bmatrix} \begin{bmatrix} e_c(t) \\ i(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \dfrac{1}{L} \end{bmatrix} e(t) \tag{4-132}$$

This format is also known as the state form if we set

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} e_c(t) \\ i(t) \end{bmatrix} \tag{4-133}$$

Then

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & \dfrac{1}{C} \\ -\dfrac{1}{L} & -\dfrac{R}{L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \dfrac{1}{L} \end{bmatrix} e(t) \tag{4-134}$$

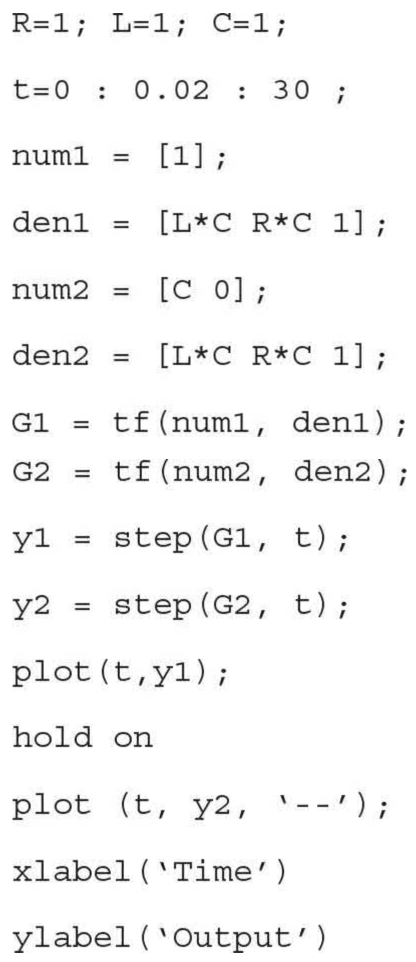
The transfer functions of the system are obtained by applying the gain formula to the SFG or block diagram of the system in Fig. 4-49*c* when all the initial states are set to zero.

$$\frac{E_c(s)}{E(s)} = \frac{(1/LC)s^{-2}}{1+(R/L)s^{-1}+(1/LC)s^{-2}} = \frac{1}{1+RCs+LCs^2} \tag{4-135}$$

$$\frac{I(s)}{E(s)} = \frac{(1/L)s^{-1}}{1+(R/L)s^{-1}+(1/LC)s^{-2}} = \frac{Cs}{1+RCs+LCs^2} \tag{4-136}$$

Toolbox 4-4-2

Time domain unit step responses using Eqs. (4-135) and (4-136) are shown using MATLAB for *R* = 1, *L* = 1, and *C* = 1:



```
R=1; L=1; C=1;

t=0 : 0.02 : 30 ;

num1 = [1];

den1 = [L*C R*C 1];

num2 = [C 0];

den2 = [L*C R*C 1];

G1 = tf(num1, den1);
G2 = tf(num2, den2);

y1 = step(G1, t);

y2 = step(G2, t);

plot(t,y1);

hold on

plot (t, y2, ‘--’);

xlabel(‘Time’)

ylabel(‘Output’)
```

The results are shown in Fig. 4-50 where unit step responses for $e_c(t)$ and $i(t)$ are obtained from Eq. (4-135) and $i(t)$ using Eq. (4-136) for $R = 1$, $L = 1$, and $C = 1$.

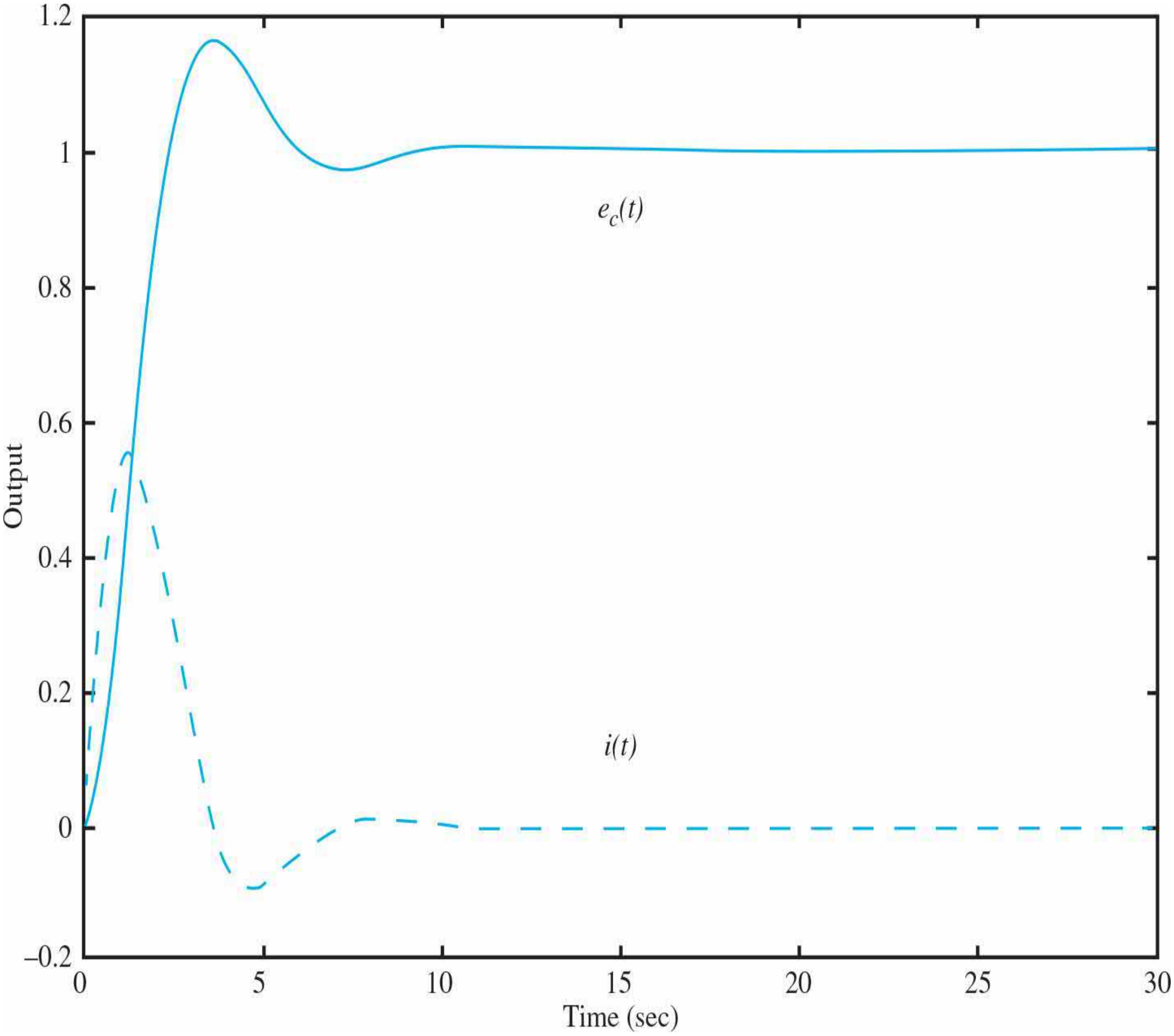


Figure 4-50 RLC network time domain unit step responses for $e_c(t)$ using Eq. (4-135) and $i(t)$ using Eq. (4-136) for $R = 1$, $L = 1$, and $C = 1$.

EXAMPLE 4-4-5 As another example of writing the state equations of an electric network, consider the network shown in Fig. 4-51*a*. According to the foregoing discussion, the voltage across the capacitor, $e_c(t)$, and the currents of the inductors, $i_1(t)$ and $i_2(t)$, are assigned as state variables, as shown in Fig. 4-51*a*. The state equations of the network are obtained by writing the voltages across the inductors and the currents in the capacitor in terms of the three state variables. The state equations are

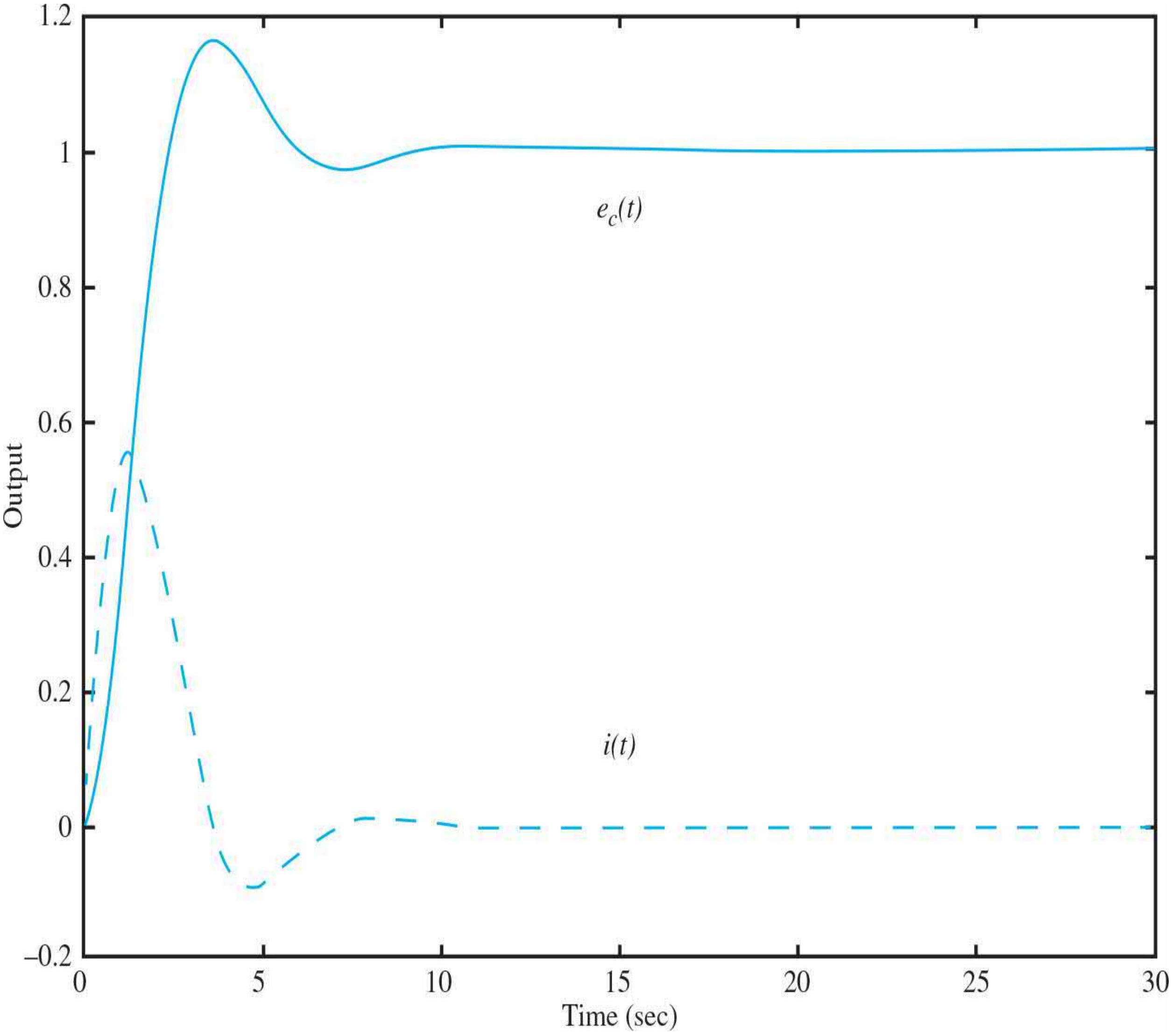


Figure 4-51 Network of Example 4-4-5. (*a*) Electrical schematic. (*b*) SFG representation.

$$L_1 \frac{di_1(t)}{dt} = -R_1 i_1(t) - e_c(t) + e(t) \tag{4-137}$$

$$L_2 \frac{di_2(t)}{dt} = -R_2 i_2(t) + e_c(t) \tag{4-138}$$

$$C \frac{de_c(t)}{dt} = i_1(t) - i_2(t) \tag{4-139}$$

In vector-matrix form, the state equations are written as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -\frac{R_1}{L_1} & 0 & -\frac{1}{L_1} \\ 0 & -\frac{R_2}{L_2} & \frac{1}{L_2} \\ \frac{1}{C} & -\frac{1}{C} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} \frac{1}{L_1} \\ 0 \\ 0 \end{bmatrix} e(t) \tag{4-140}$$

where $x_1 = i_1(t)$, $x_2 = i_2(t)$, and $x_3 = e_c(t)$. The signal-flow diagram of the network, without the initial states, is shown in Fig. 4-51*b*. The transfer functions between $I_1(s)$ and $E(s)$, $I_2(s)$ and $E(s)$, and $E_c(s)$ and $E(s)$, respectively, are written from the state diagram

$$\frac{I_1(s)}{E(s)} = \frac{L_2Cs^2 + R_2Cs + 1}{\Delta} \tag{4-141}$$

$$\frac{I_2(s)}{E(s)} = \frac{1}{\Delta} \tag{4-142}$$

$$\frac{E_c(s)}{E(s)} = \frac{L_2s + R_2}{\Delta} \tag{4-143}$$

where

$$\Delta = L_1L_2Cs^3 + (R_1L_2 + R_2L_1)Cs^2 + (L_1 + L_2 + R_1R_2C)s + R_1 + R_2 \tag{4-144}$$

The the unit step responses are shown in Fig. 4-52.

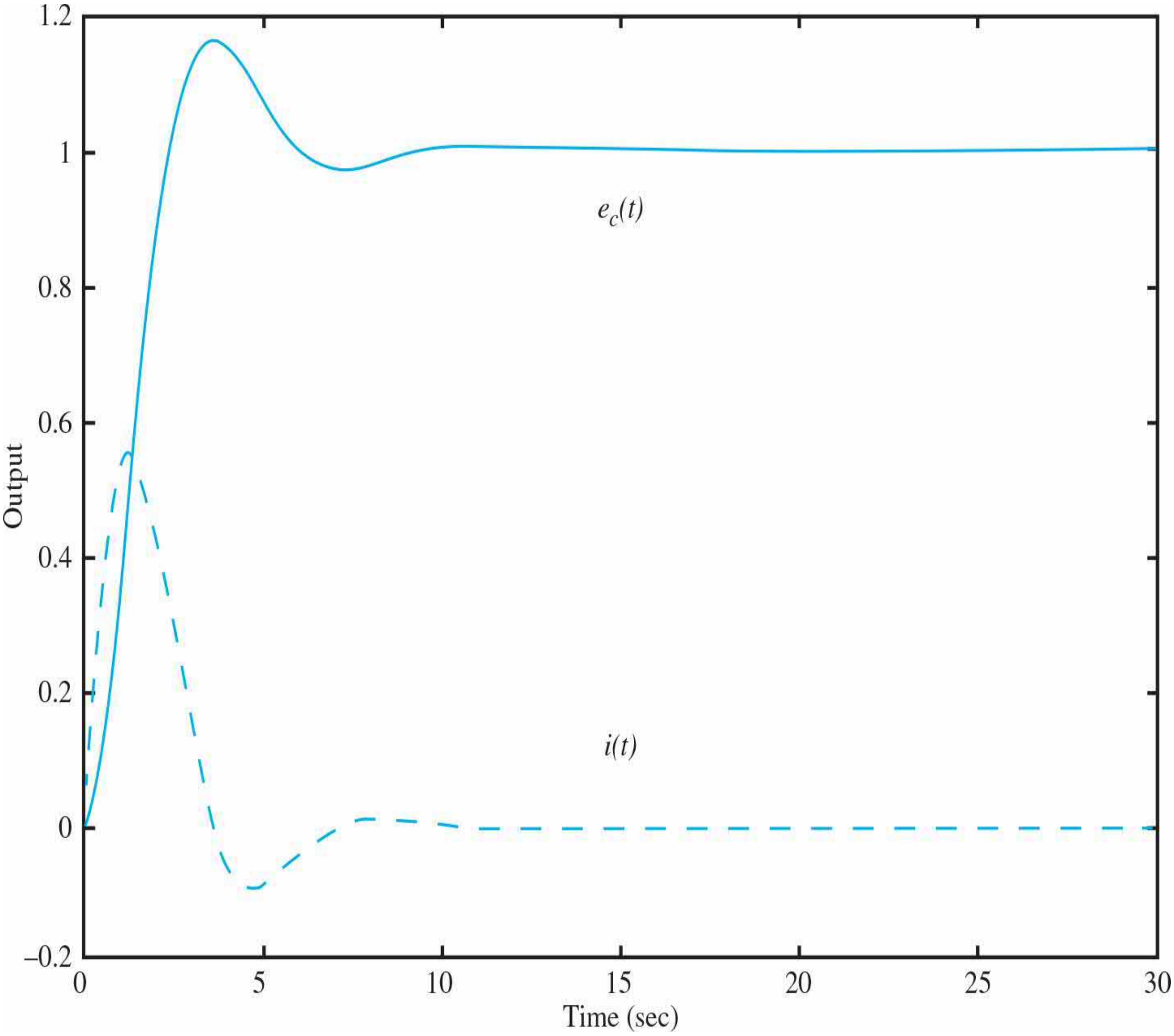
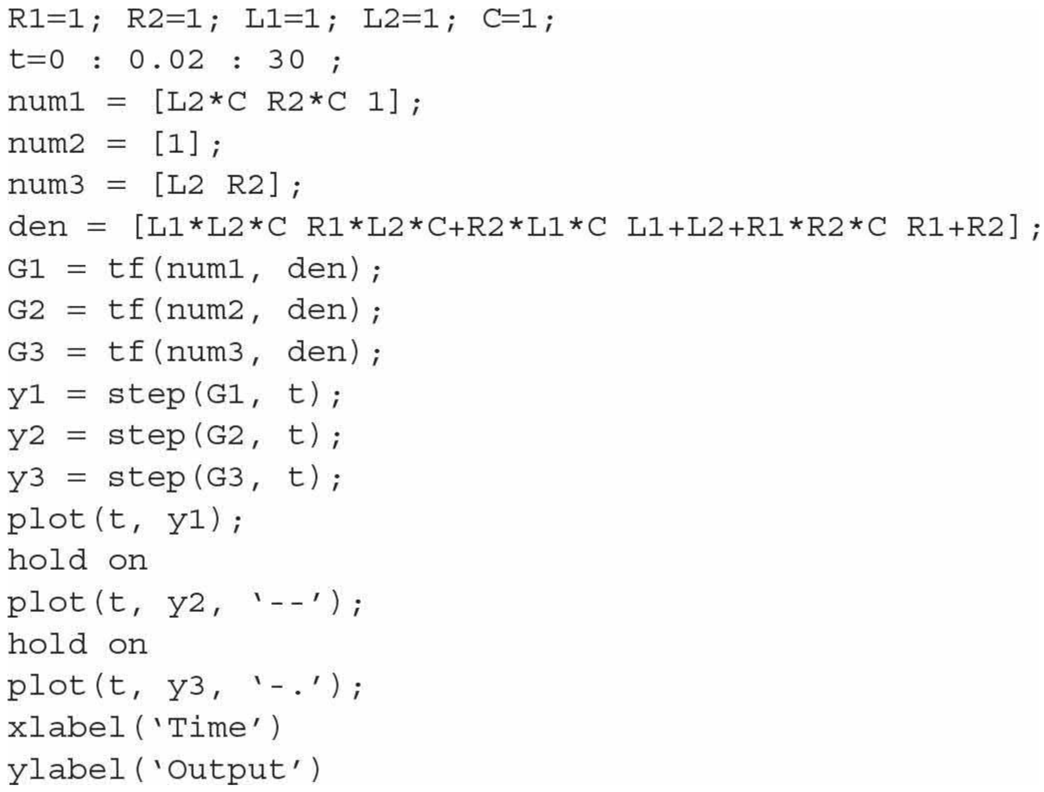


Figure 4-52 Example 4-4-5 network time domain unit step responses for $i_1(t)$ using Eq. (4-141), $i_2(t)$ using Eq. (4-142) and $e_c(t)$ using Eq. (4-143) for $R_1 = 1$, $R_2 = 1$, $L_1 = 1$, $L_2 = 1$, and $C = 1$.

Toolbox 4-4-3

Time domain unit step responses using Eqs. (4-141) to (4-143) are shown using MATLAB for $R_1 = 1$, $R_2 = 1$, $L_1 = 1$, $L_2 = 1$, and $C = 1$:

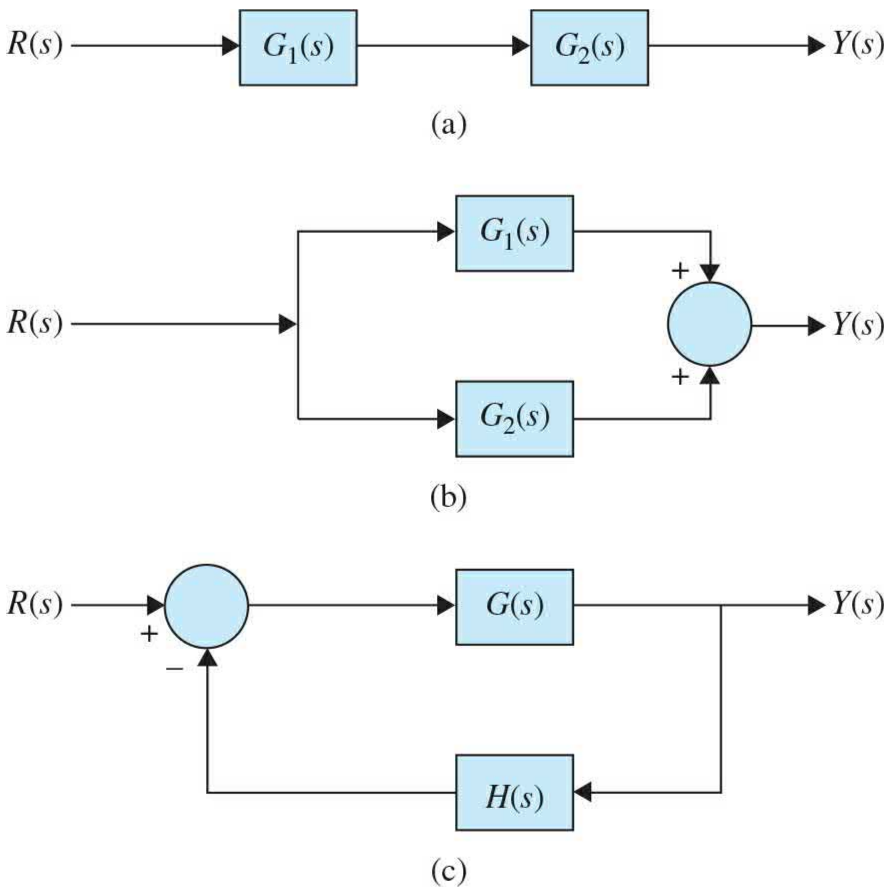


```
R1=1; R2=1; L1=1; L2=1; C=1;
t=0 : 0.02 : 30 ;
num1 = [L2*C R2*C 1];
num2 = [1];
num3 = [L2 R2];
den = [L1*L2*C R1*L2*C+R2*L1*C L1+L2+R1*R2*C R1+R2];
G1 = tf(num1, den);
G2 = tf(num2, den);
G3 = tf(num3, den);
y1 = step(G1, t);
y2 = step(G2, t);
y3 = step(G3, t);
plot(t, y1);
hold on
plot(t, y2, ‘--’);
hold on
plot(t, y3, ‘-.’);
xlabel(‘Time’)
ylabel(‘Output’)
```

4-5 MATLAB TOOLS

There is no specific software developed for this chapter. Although the MATLAB Controls Toolbox offers functions for finding the transfer functions from a given block diagram, it was felt that students may master this subject without referring to a computer. For simple operations, however, MATLAB may be used, as shown in the following example.

EXAMPLE 4-5-1 Consider the following transfer functions, which correspond to the block diagrams shown in Fig. 4-53.



(a)

(b)

(c)

Figure 4-53 Basic block diagrams used for Example 4-3-1.

$$G_1(s) = \frac{1}{s+1}, \quad G_2(s) = \frac{s+1}{s+2}, \quad G(s) = \frac{1}{s(s+1)}, \quad H(s) = 10 \tag{4-145}$$

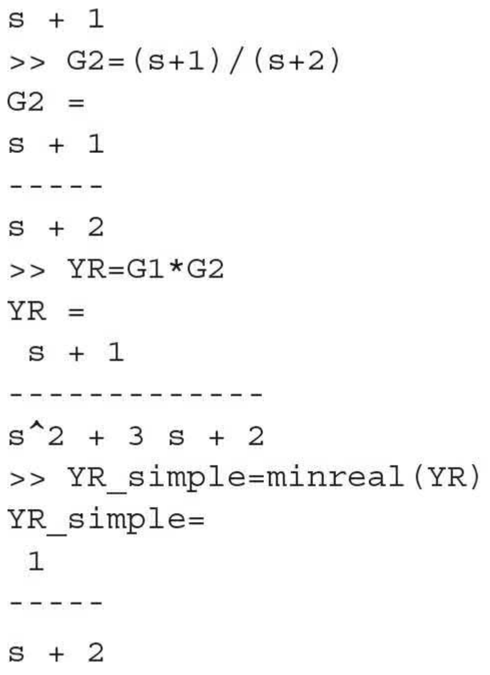
Use MATLAB to find the transfer function $Y(s)/R(s)$ for each case. The results are as follows.

Toolbox 4-5-1

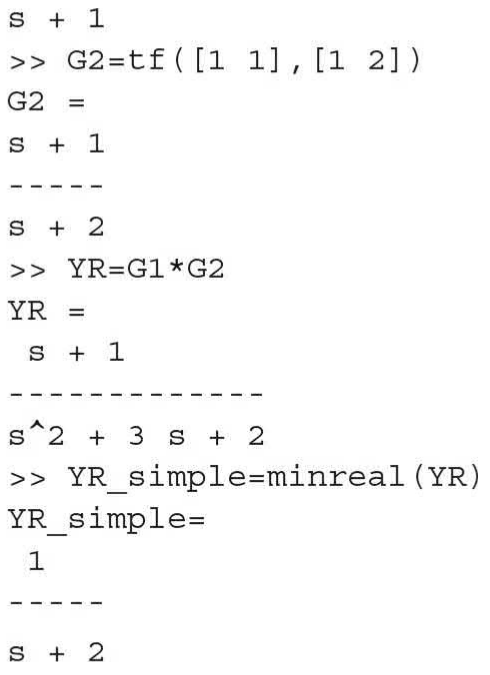
Case (a): Use MATLAB to find G_1*G_2.

```
Approach 1
>> clear all
>> s = tf(‘s’);
>> G1=1/(s+1)
G1 =
 1
----
```

```
Approach 2
>> clear all
>> G1=tf([1],[1 1])

G1 =
 1
----
```



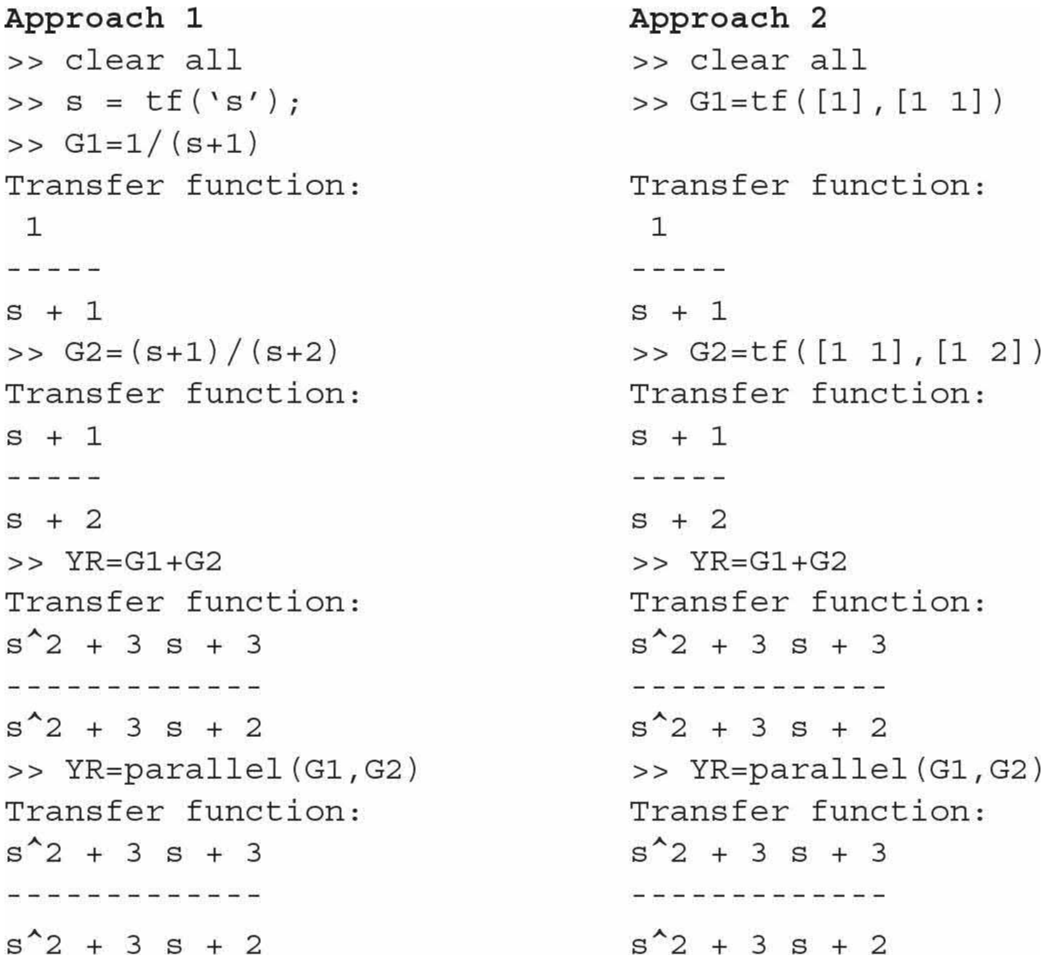
```
s + 1
>> G2=(s+1)/(s+2)
G2 =
s + 1
-----
s + 2
>> YR=G1*G2
YR =
 s + 1
-------------
s^2 + 3 s + 2
>> YR_simple=minreal(YR)
YR_simple=
 1
-----
s + 2
```



```
s + 1
>> G2=tf([1 1],[1 2])
G2 =
s + 1
-----
s + 2
>> YR=G1*G2
YR =
 s + 1
-------------
s^2 + 3 s + 2
>> YR_simple=minreal(YR)
YR_simple=
 1
-----
s + 2
```

Use “minreal(YR)” for pole zero cancellation, if necessary.
Alternatively use “YR=series(G1,G2)” instead of “YR=G1*G2”.
Case (b): Use MATLAB to find G_1 + G_2.

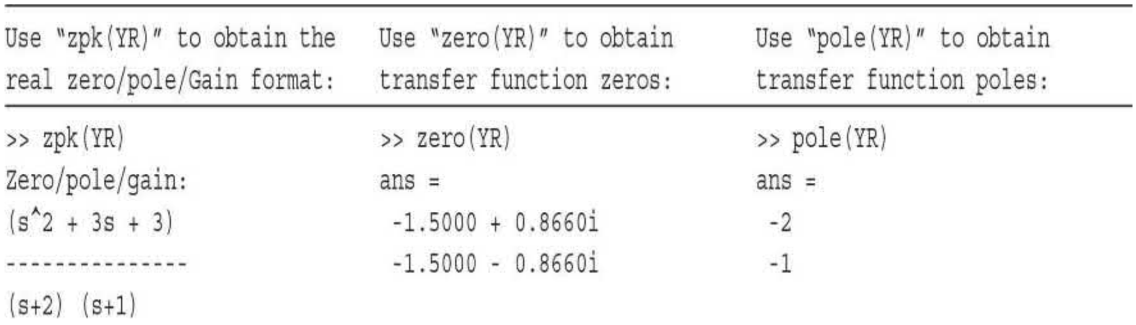
$$\frac{Y(s)}{R(s)} = \frac{2s+3}{s^2+3s+2} = \frac{2(s+1.5)}{(s+1)(s+2)}$$



```
Approach 1
>> clear all
>> s = tf('s');
>> G1=1/(s+1)
Transfer function:
 1
-----
s + 1
>> G2=(s+1)/(s+2)
Transfer function:
s + 1
-----
s + 2
>> YR=G1+G2
Transfer function:
s^2 + 3 s + 3
-------------
s^2 + 3 s + 2
>> YR=parallel(G1,G2)
Transfer function:
s^2 + 3 s + 3
-------------

s^2 + 3 s + 2
```

```
Approach 2
>> clear all
>> G1=tf([1],[1 1])

Transfer function:
 1
-----
s + 1
>> G2=tf([1 1],[1 2])
Transfer function:
s + 1
-----
s + 2
>> YR=G1+G2
Transfer function:
s^2 + 3 s + 3
-------------
s^2 + 3 s + 2
>> YR=parallel(G1,G2)
Transfer function:
s^2 + 3 s + 3
-------------

s^2 + 3 s + 2
```

Use “minreal(YR)” for pole zero cancellation, if necessary. Alternatively use “YR=parallel(G1,G2)” instead of “YR=G1+G2”.

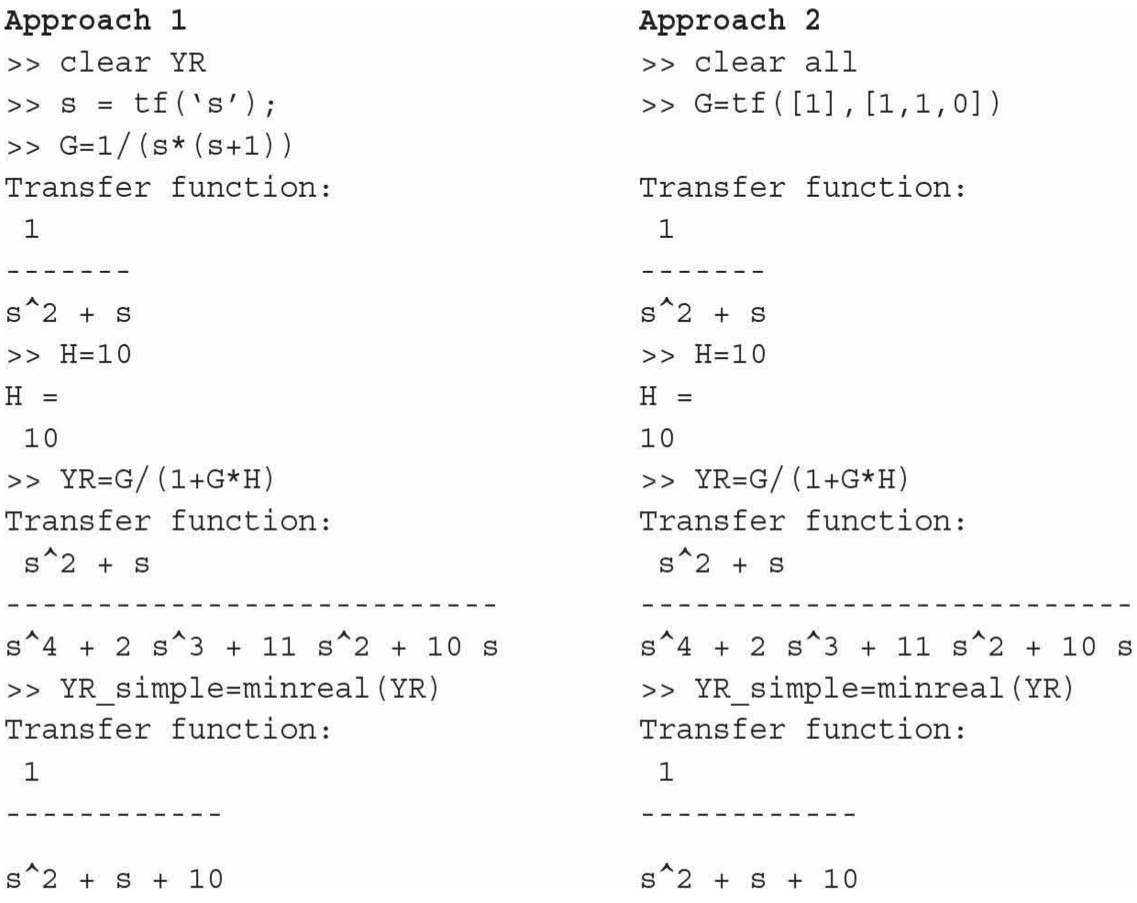


Use "zpk(YR)" to obtain the real zero/pole/Gain format:	Use "zero(YR)" to obtain transfer function zeros:	Use "pole(YR)" to obtain transfer function poles:
>> zpk(YR) Zero/pole/gain: (s^2 + 3s + 3) -------------- (s+2) (s+1)	>> zero(YR) ans = -1.5000 + 0.8660i -1.5000 - 0.8660i	>> pole(YR) ans = -2 -1

Toolbox 4-5-2

Case (c): Use MATLAB to find the closed-loop feedback function $\frac{G}{1+GH}$.

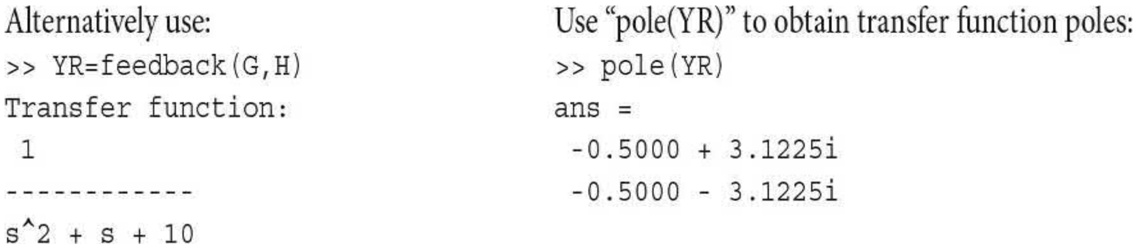
$$\frac{Y(s)}{R(s)}=\frac{1}{s^2+s+10}$$



```
Approach 1
>> clear YR
>> s = tf('s');
>> G=1/(s*(s+1))
Transfer function:
 1
-------
s^2 + s
>> H=10
H =
 10
>> YR=G/(1+G*H)
Transfer function:
 s^2 + s
---------------------------
s^4 + 2 s^3 + 11 s^2 + 10 s
>> YR_simple=minreal(YR)
Transfer function:
 1
------------

s^2 + s + 10
```

```
Approach 2
>> clear all
>> G=tf([1],[1,1,0])

Transfer function:
 1
-------
s^2 + s
>> H=10
H =
10
>> YR=G/(1+G*H)
Transfer function:
 s^2 + s
---------------------------
s^4 + 2 s^3 + 11 s^2 + 10 s
>> YR_simple=minreal(YR)
Transfer function:
 1
------------

s^2 + s + 10
```

Use “minreal (YR)” for pole zero cancellation, if necessary.



Alternatively use:

```
>> YR=feedback(G,H)
Transfer function:
 1
------------
s^2 + s + 10
```

Use “pole(YR)” to obtain transfer function poles:

```
>> pole(YR)
ans =
  -0.5000 + 3.1225i
  -0.5000 - 3.1225i
```

4-6 SUMMARY

This chapter was devoted to the mathematical modeling of physical systems. Transfer functions, block diagrams, and signal-flow graphs were defined. The block diagram representation was shown to be a versatile method of portraying linear and nonlinear systems. A powerful method of representing the interrelationships between the signals of a linear system is the SFG. When applied properly, an SFG allows the derivation of the transfer functions between input and output variables of a linear system using the gain formula. A state diagram is an SFG that is applied to dynamic systems that are represented by differential equations.

At the end of the chapter, various practical examples were given, which complete the modeling aspects of dynamic and control systems already studied in Chaps. 2 and 3. MATLAB was also used to calculate transfer functions and time responses of simple block diagram systems.

REFERENCES

Block Diagrams and Signal-Flow Graphs

1. T. D. Graybeal, “Block Diagram Network Transformation,” *Elec. Eng.*, Vol. 70, pp. 985–990 1951.
2. S. J. Mason, “Feedback Theory—Some Properties of Signal Flow Graphs,” *Proc. IRE*, Vol. 41, No. 9, pp. 1144–1156, Sep. 1953.
3. S. J. Mason, “Feedback Theory—Further Properties of Signal Flow Graphs,” *Proc. IRE*, Vol. 44, No. 7, pp. 920–926, July 1956.
4. L. P. A. Robichaud, M. Boisvert, and J. Robert, *Signal Flow Graphs and Applications*, Prentice Hall, Englewood Cliffs, NJ, 1962.
5. B. C. Kuo, *Linear Networks and Systems*, McGraw-Hill, New York, 1967.

State-Variable Analysis of Electric Networks

6. B. C. Kuo, *Linear Circuits and Systems*, McGraw-Hill, New York, 1967.

PROBLEMS

PROBLEMS FOR SEC. 4-1

4-1. Consider the block diagram shown in Fig. 4P-1.

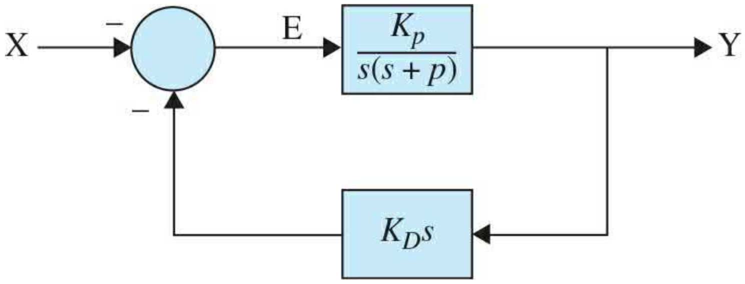


Figure 4P-1

Find:

(a) The loop transfer function.

(b) The forward path transfer function.

(c) The error transfer function.

(d) The feedback transfer function.

(e) The closed loop transfer function.

4-2. Reduce the block diagram shown in Fig. 4P-2 to unity feedback form and find the system characteristic equation.

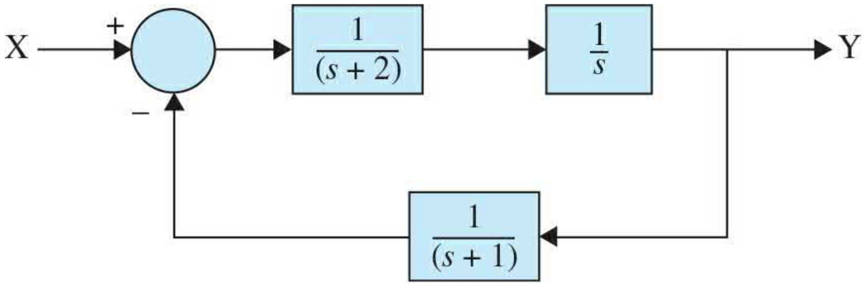


Figure 4P-2

4-3. Reduce the block diagram shown in Fig. 4P-3 and find the Y/X.

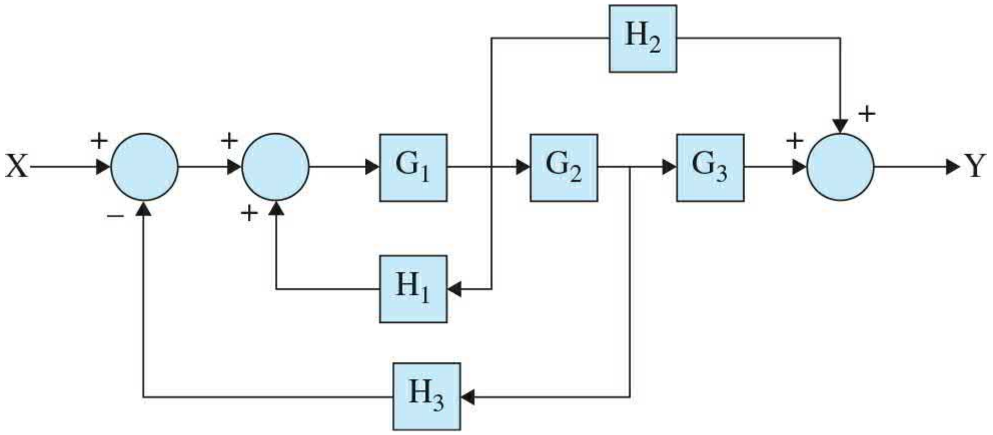


Figure 4P-3

4-4. Reduce the block diagram shown in Fig. 4P-4 to unity feedback form and find the Y/X.

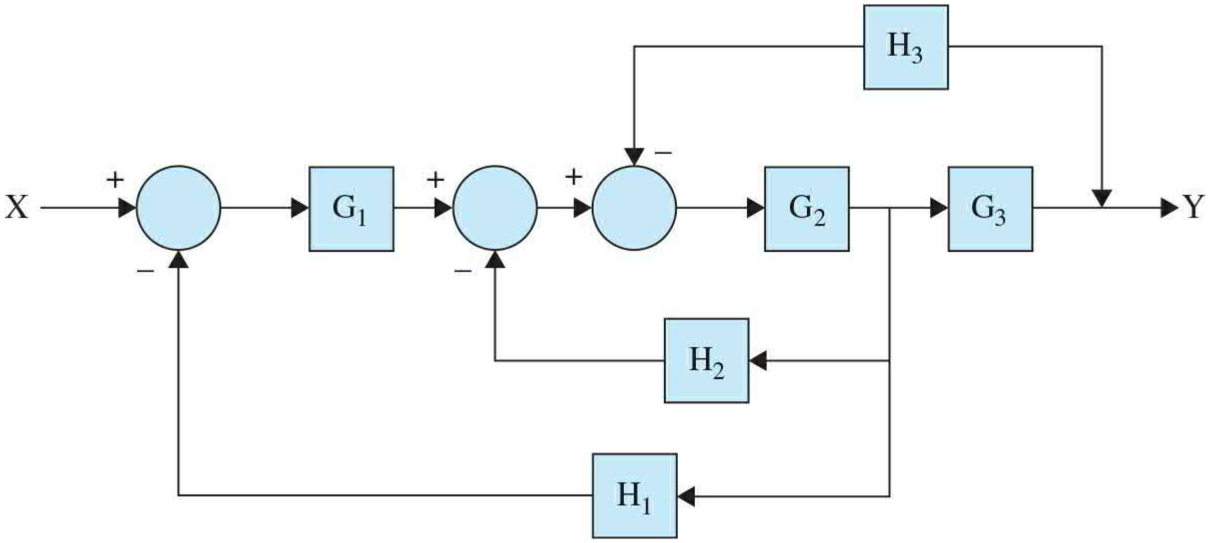


Figure 4P-4

4-5. The aircraft turboprop engine shown in Fig. 4P-5*a* is controlled by a closed-loop system with block diagram shown in Fig. 4P-5*b*. The engine is modeled as a multivariable system with input vector **E**(*s*), which contains the fuel rate and propeller blade angle, and output vector **Y**(*s*), consisting of the engine speed and turbine-inlet temperature. The transfer function matrices are given as

$$\mathbf{G}(s)=\begin{bmatrix} \dfrac{2}{s(s+2)} & 10 \\ \dfrac{5}{s} & \dfrac{1}{s+1} \end{bmatrix} \quad \mathbf{H}(s)=\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Find the closed-loop transfer function matrix $[\mathbf{I}+\mathbf{G}(s)]^{-1}\mathbf{G}(s)$.

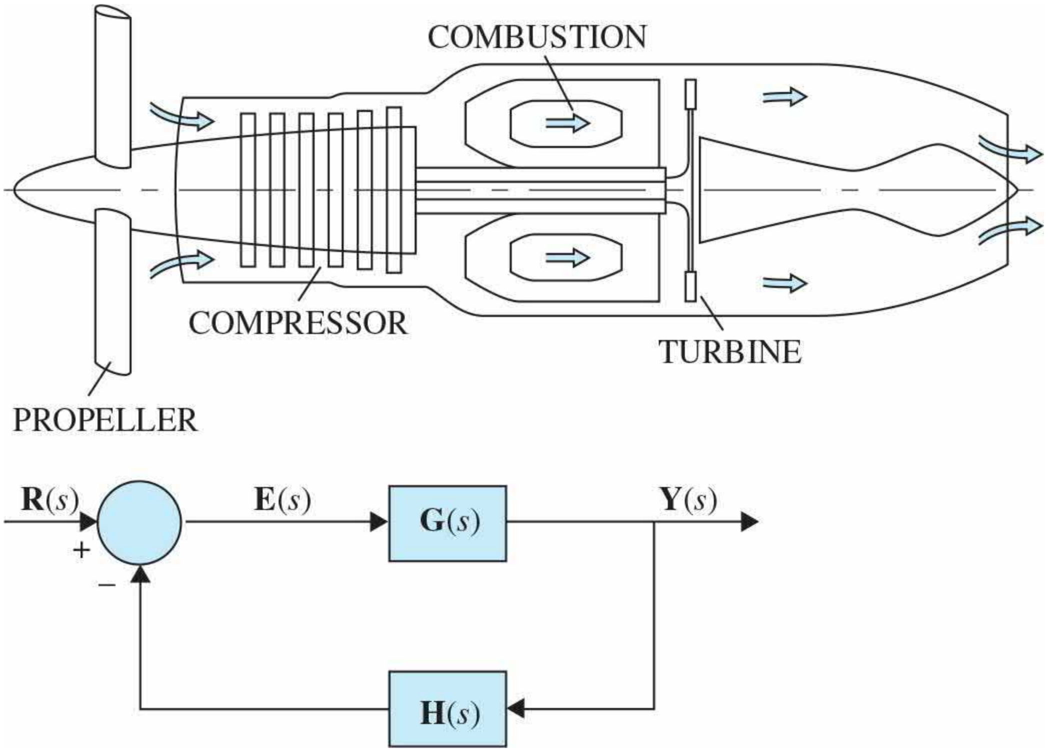


Figure 4P-5

4-6. Use MATLAB to solve Prob. 4-5.

4-7. The block diagram of the position-control system of an electronic word processor is shown in Fig. 4P-7.

(a) Find the loop transfer function $\Theta_o(s)/\Theta_e(s)$ (the outer feedback path is open).

(b) Find the closed-loop transfer function $\Theta_o(s)/\Theta_r(s)$.

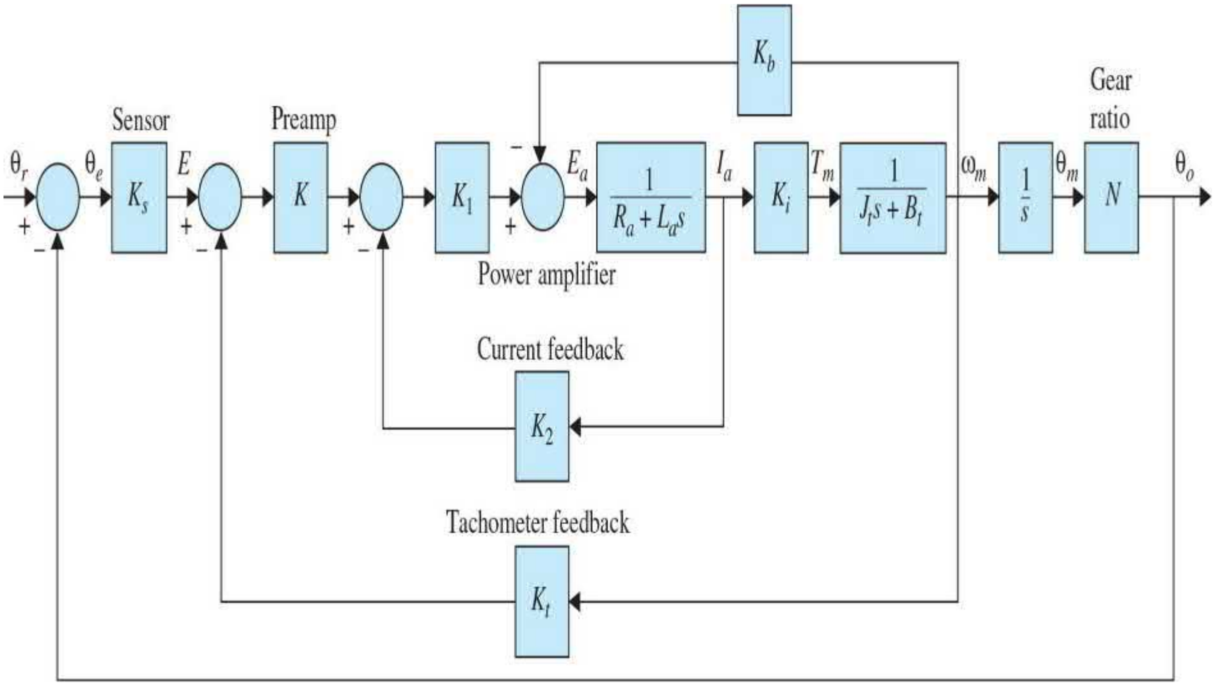


Figure 4P-7

4-8. The block diagram of a feedback control system is shown in Fig. 4P-8. Find the following transfer functions:

(a) $\left.\frac{Y(s)}{R(s)}\right|_{N=0}$

(b) $\left.\frac{Y(s)}{E(s)}\right|_{N=0}$

(c) $\left.\frac{Y(s)}{N(s)}\right|_{R=0}$

(d) Find the output $Y(s)$ when $R(s)$ and $N(s)$ are applied simultaneously.

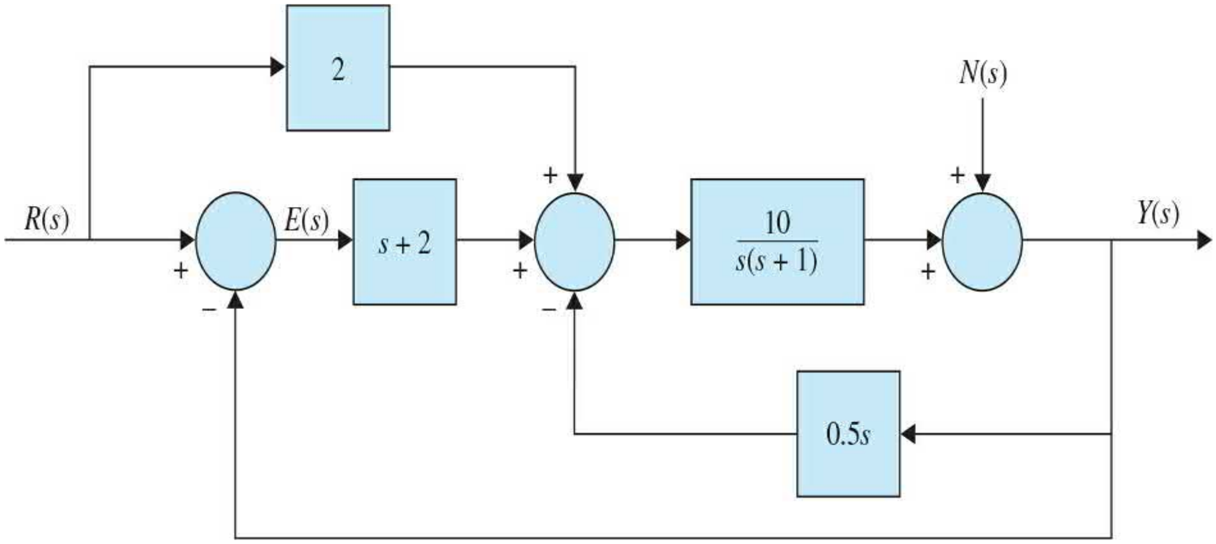


Figure 4P-8

4-9. The block diagram of a feedback control system is shown in Fig. 4P-9.

(a) Apply the SFG gain formula directly to the block diagram to find the transfer functions:

$$\left.\frac{Y(s)}{R(s)}\right|_{N=0} \quad \left.\frac{Y(s)}{N(s)}\right|_{R=0}$$

Express $Y(s)$ in terms of $R(s)$ and $N(s)$ when both inputs are applied simultaneously.

(b) Find the desired relation among the transfer functions $G_1(s)$, $G_2(s)$, $G_3(s)$, $G_4(s)$, $H_1(s)$ and $H_2(s)$ so that the output $Y(s)$ is not affected by the disturbance signal $N(s)$ at all.

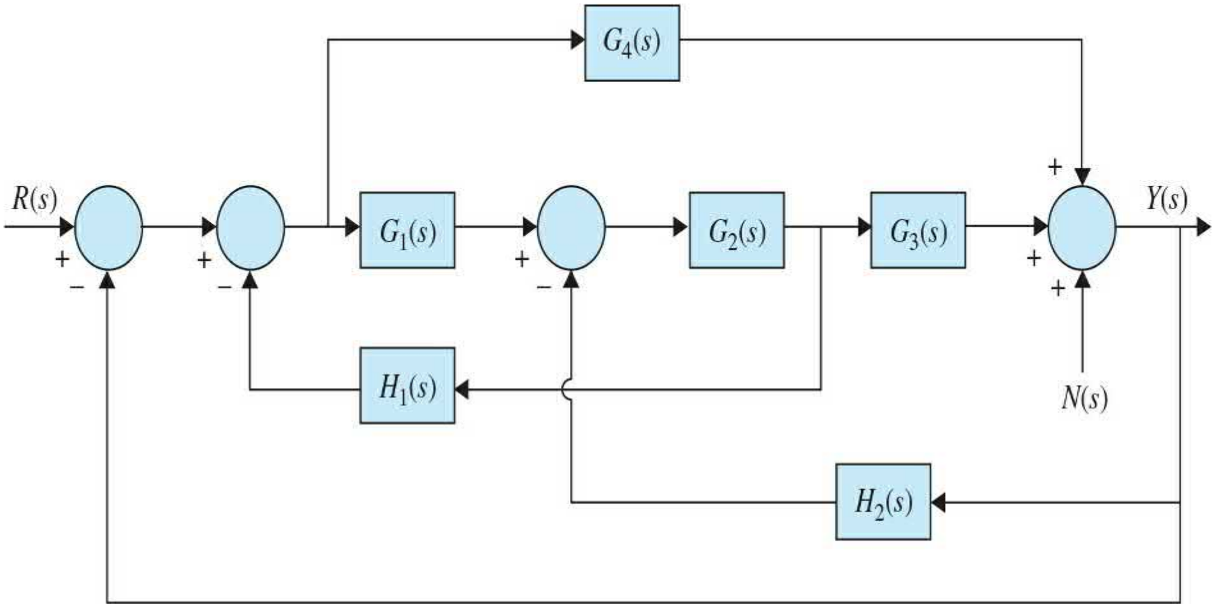


Figure 4P-9

4-10. Figure 4P-10 shows the block diagram of the antenna control system of the solar-collector field shown in Fig. 1-5. The signal $N(s)$ denotes the wind gust disturbance acted on the antenna. The feedforward transfer function $G_d s)$ is used to eliminate the effect of $N(s)$ on the output $Y(s)$. Find the transfer function $Y(s)/N(s)|_{R=0}$. Determine the expression of $G_d(s)$ so that the effect of $N(s)$ is entirely eliminated.

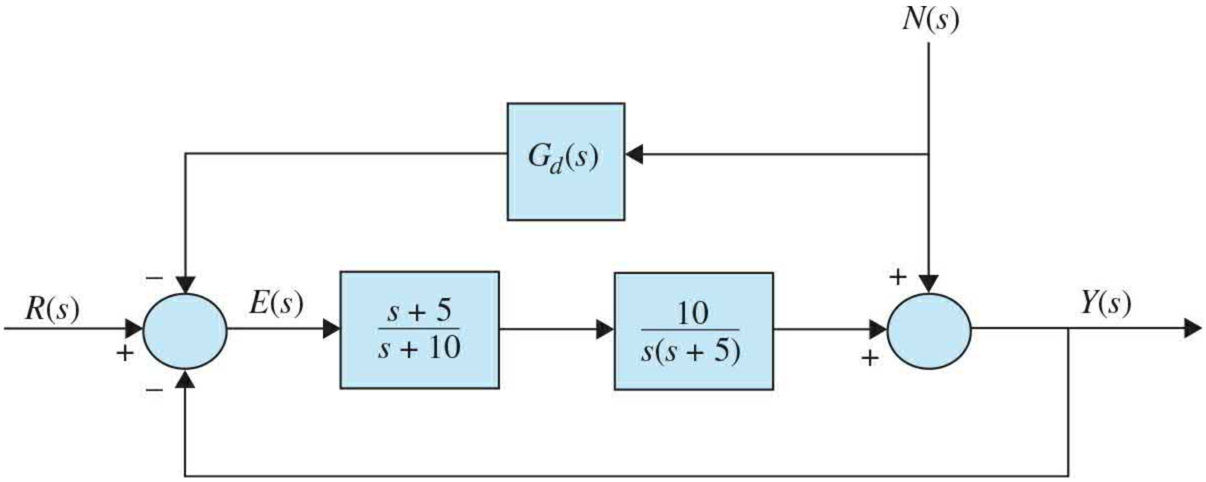
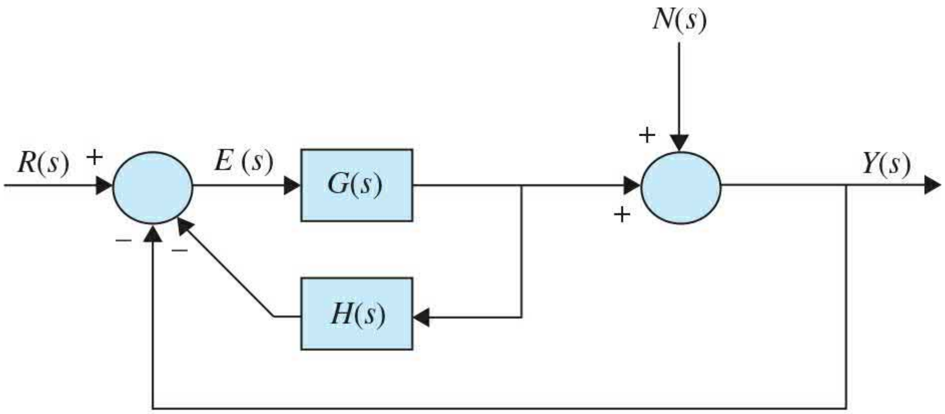


Figure 4P-10

4-11. Figure 4P-11 shows the block diagram of a dc-motor control system. The signal $N(s)$ denotes the frictional torque at the motor shaft.

(a) Find the transfer function $H(s)$ so that the output $Y(s)$ is not affected by the disturbance torque $N(s)$.

(b) With $H(s)$ as determined in part (a), find the value of K so that the steady-state value of $e(t)$ is equal to 0.1 when the input is a unit-ramp function, $r(t) = tu_s(t)$, $R(s) = 1/s^2$, and $N(s) = 0$. Apply the final-value theorem.



$$G(s) = \frac{K(s+3)}{s(s+1)(s+2)}$$

Figure 4P-11

4-12. The block diagram of an electric train control is shown in Fig. 4P-12. The system parameters and variables are

$e_r(t)$ = voltage representing the desired train speed,

$v(t)$ = speed of train, ft/sec

M =Mass of train = 30,000 lb/sec

K = amplifier gain

K_t = gain of speed indicator = 0.15 V/ft/sec

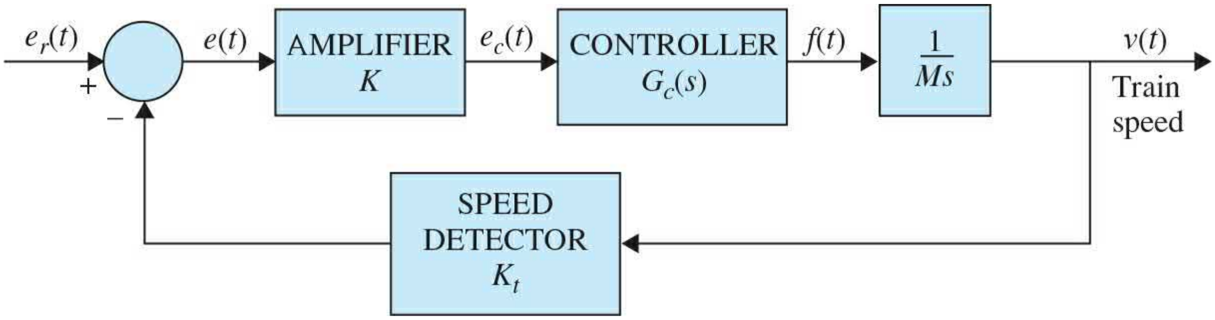


Figure 4P-12

To determine the transfer function of the controller, we apply a step function of 1 V to the input of the controller, that is, $e_c(t) = u_s(t)$. The output of the controller is measured and described by the following equation:

$$f(t) = 100(1 - 0.3e^{-6t} - 0.7e^{-10t})u_s(t)$$

(a) Find the transfer function $G_c(s)$ of the controller.

(b) Derive the forward-path transfer function $V(s)/E(s)$ of the system. The feedback path is opened in this case.

(c) Derive the closed-loop transfer function $V(s)/E_r(s)$ of the system.

(d) Assuming that K is set at a value so that the train will not run away (unstable), find the steady-state speed of the train in feet per second when the input is $e_r(t) = us(t)$V.

4-13. Use MATLAB to solve Prob. 4-12.

4-14. Repeat Prob. 4-12 when the output of the controller is measured and described by the following expression:

$$f(t) = 100(1 - 0.3e^{-6(t-0.5)})u_s(t - 0.5)$$

when a step input of 1 V is applied to the controller.

4-15. Use MATLAB to solve Prob. 4-14.

4-16. A linear time-invariant multivariable system with inputs $r_1(t)$ and $r_2(t)$ and outputs $y_1(t)$ and $y_2(t)$ is described by the following set of differential equations.

$$\frac{d^2y_1(t)}{dt^2}+2\frac{dy_1(t)}{dt}+3y_2(t)=r_1(t)+r_2(t)$$
$$\frac{d^2y_2(t)}{dt^2}+3\frac{dy_1(t)}{dt}+y_1(t)-y_2(t)=r_2(t)+\frac{dr_1(t)}{dt}$$

Find the following transfer functions:

$$\left.\frac{Y_1(s)}{R_1(s)}\right|_{R_2=0}\quad \left.\frac{Y_2(s)}{R_1(s)}\right|_{R_2=0}\quad \left.\frac{Y_1(s)}{R_2(s)}\right|_{R_1=0}\quad \left.\frac{Y_2(s)}{R_2(s)}\right|_{R_1=0}$$

PROBLEMS FOR SEC. 4-2

4-17. Find the state-flow diagram for the system shown in Fig. 4P-4.

4-18. Draw a signal-flow diagram for the system with the state-space of

$$\dot{X}=\begin{bmatrix} -5 & -6 & 3 \\ 1 & 0 & -1 \\ -0.5 & 1.5 & 0.5 \end{bmatrix}\mathbf{X}+\begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}\mathbf{U}$$
$$\mathbf{Z}=\begin{bmatrix} 0.5 & 0.5 & 0 \\ 0.5 & 0 & 0.5 \end{bmatrix}\mathbf{X}$$

4-19. Find the state-space of a system with the following transfer function:

$$G(s)=\frac{B_1s+B_0s}{s^2+A_1s+A_0s}$$

4-20. Draw signal-flow graphs for the following sets of algebraic equations. These equations should first be arranged in the form of cause-and-effect relations before SFGs can be drawn. Show that there are many possible SFGs for each set of equations.

$$x_1=-x_2-3x_3+3$$
$$x_2=5x_1-2x_2+x_3$$

(a) $x_3=4x_1+x_2-5x_3+5$

$$2x_1 + 3x_2 + x_3 = -1$$
$$x_1 - 2x_2 - x_3 = 1$$

(b) $$3x_2 + x_3 = 0$$

4-21. The block diagram of a control system is shown in Fig. 4P-21.

(a) Draw an equivalent SFG for the system.

(b) Find the following transfer functions by applying the gain formula of the SFG directly to the block diagram.

$$\left.\frac{Y(s)}{R(s)}\right|_{N=0} \quad \left.\frac{Y(s)}{N(s)}\right|_{R=0} \quad \left.\frac{E(s)}{R(s)}\right|_{N=0} \quad \left.\frac{E(s)}{N(s)}\right|_{R=0}$$

(c) Compare the answers by applying the gain formula to the equivalent SFG.

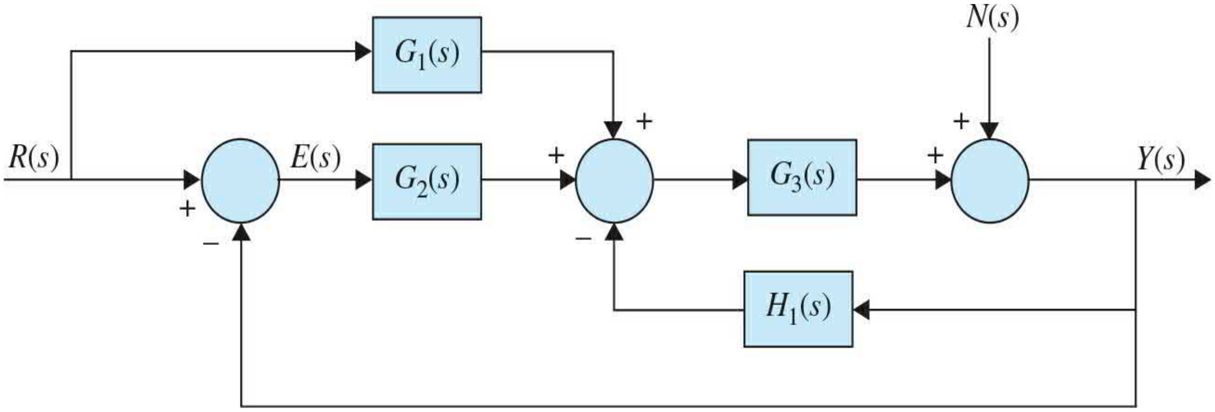
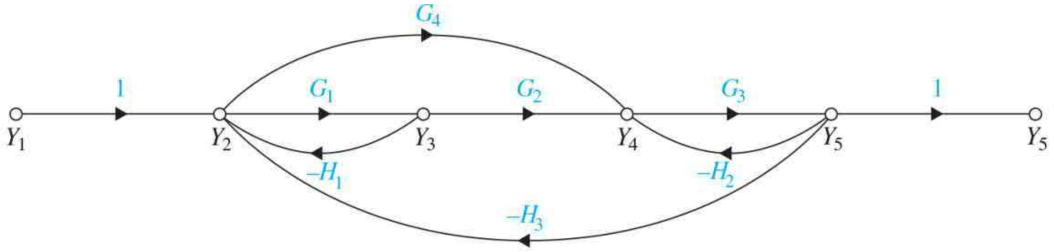
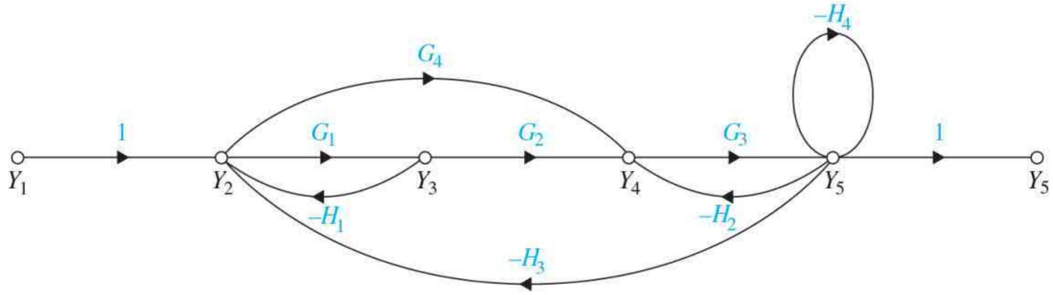


Figure 4P-21

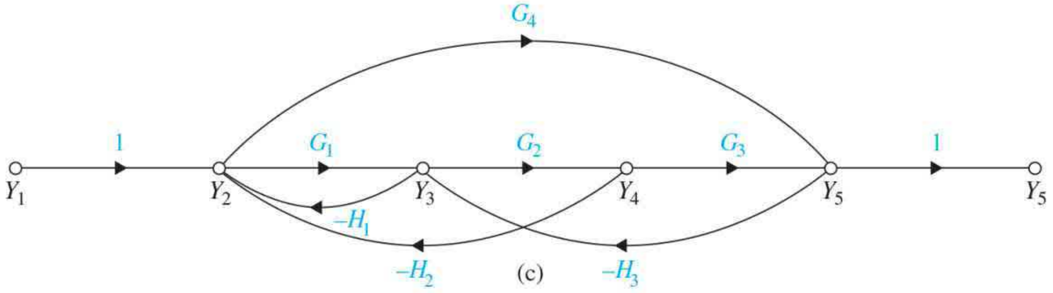
4-22. Apply the gain formula to the SFGs shown in Fig. 4P-22 to find the following transfer functions: $\frac{Y_5}{Y_1}\;\frac{Y_4}{Y_1}\;\frac{Y_2}{Y_1}\;\frac{Y_5}{Y_2}$.


G4
1
G1
G2
G3
1
Y1
Y2
Y3
Y4
Y5
Y5
–H1
–H2
–H3

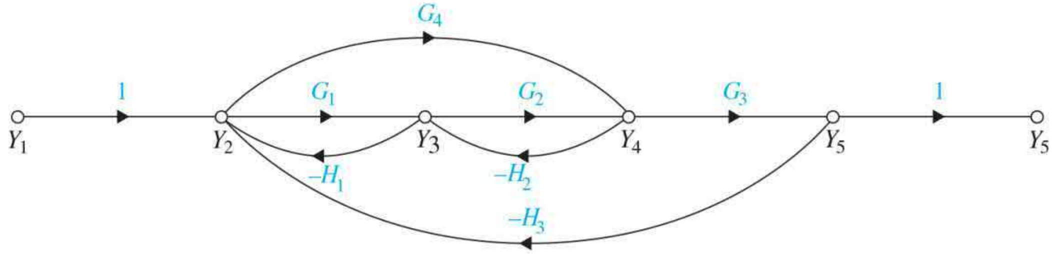
(a)


–H4
G4
1
G1
G2
G3
1
Y1
Y2
Y3
Y4
Y5
Y5
–H1
–H2
–H3

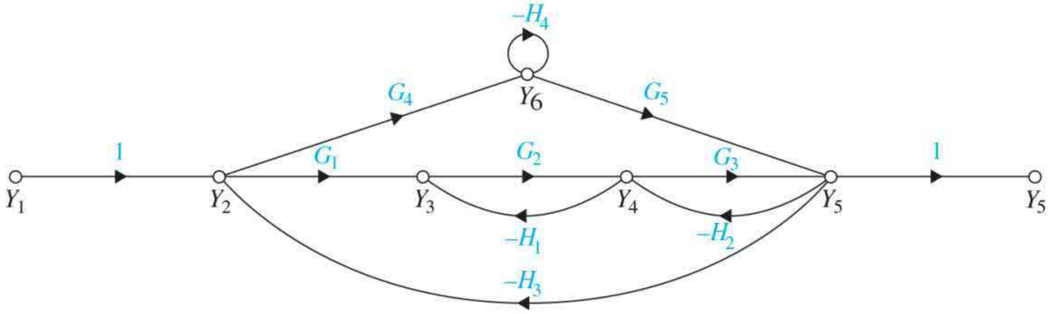
(b)


G4
1
G1
G2
G3
1
Y1
Y2
Y3
Y4
Y5
Y5
–H1
–H2
–H3

(c)


G4
1
G1
G2
G3
1
Y1
Y2
Y3
Y4
Y5
Y5
–H1
–H2
–H3

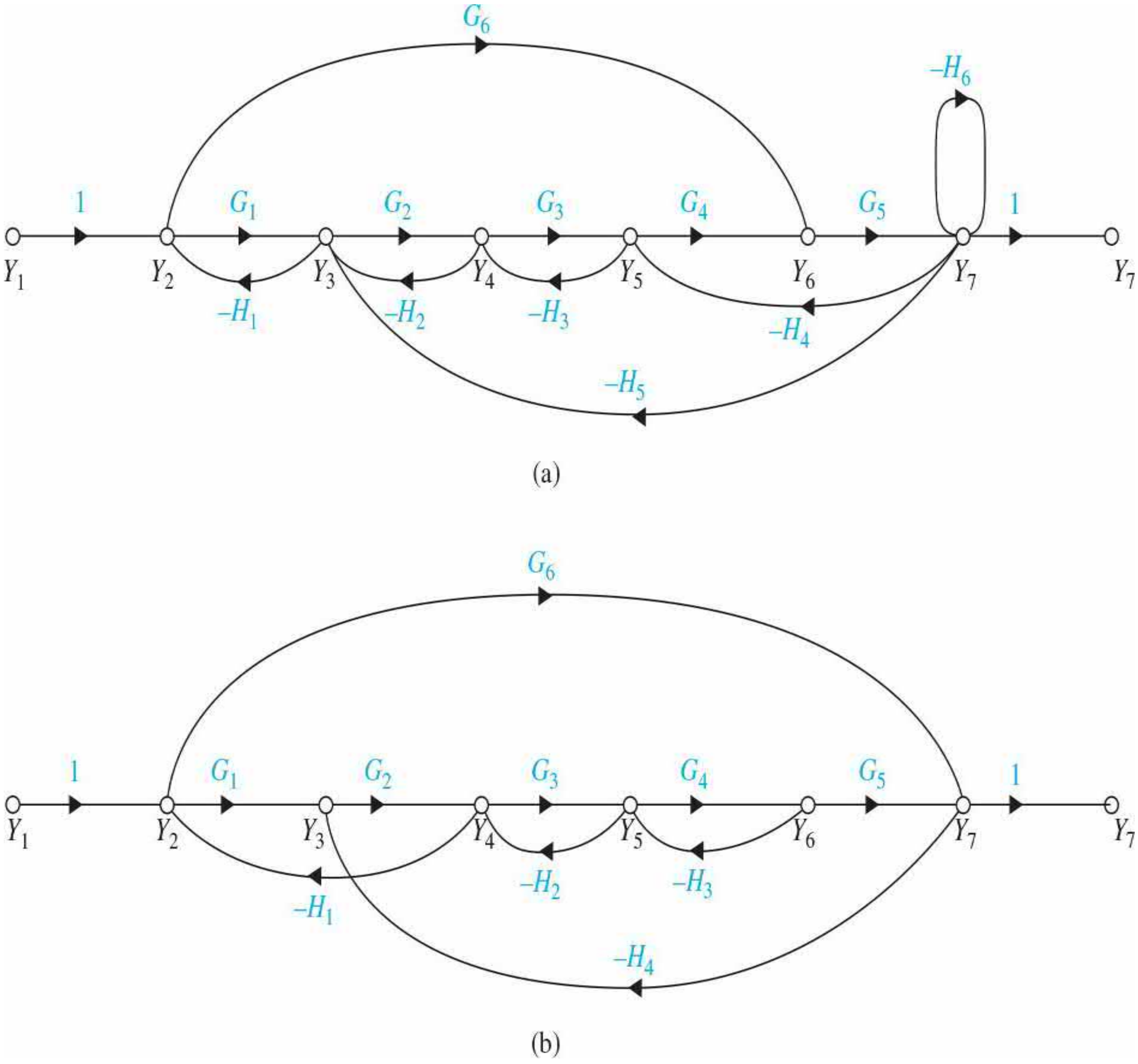
(d)


–H4
Y6
G4
G5
1
G1
G2
G3
1
Y1
Y2
Y3
Y4
Y5
Y5
–H1
–H2
–H3

(e)

Figure 4P-22

4-23. Find the transfer functions Y_7/Y_1 and Y_2/Y_1 of the SFGs shown in Fig. 4P-23.



(a)

(b)

Figure 4P-23

4-24. Signal-flow graphs may be used to solve a variety of electric network problems. Shown in Fig. 4P-24 is the equivalent circuit of an electronic circuit. The voltage source $e_d(t)$ represents a disturbance voltage. The objective is to find the value of the constant k so that the output voltage

$e_o(t)$ is not affected by $e_d(t)$. To solve the problem, it is best to first write a set of cause-and-effect equations for the network. This involves a combination of node and loop equations. Then construct an SFG using these equations. Find the gain e_o/e_d with all other inputs set to zero. For e_d not to affect e_o, set e_o/e_d to zero.

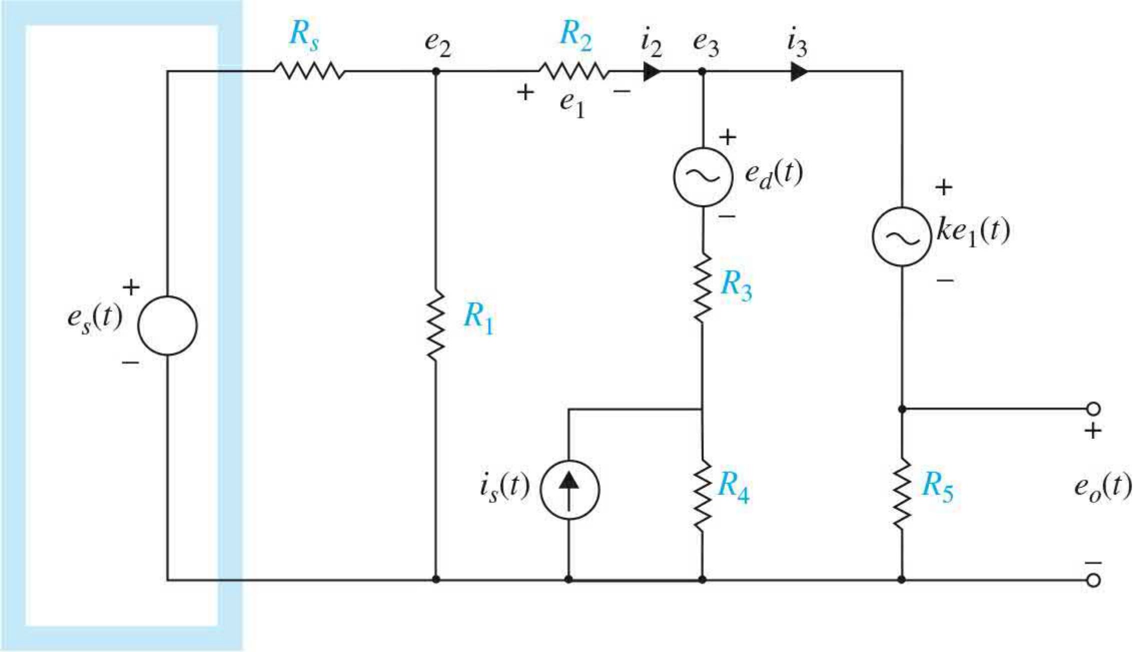


Figure 4P-24

4-25. Show that the two systems shown in Fig. 4P-25*a* and *b* are equivalent.

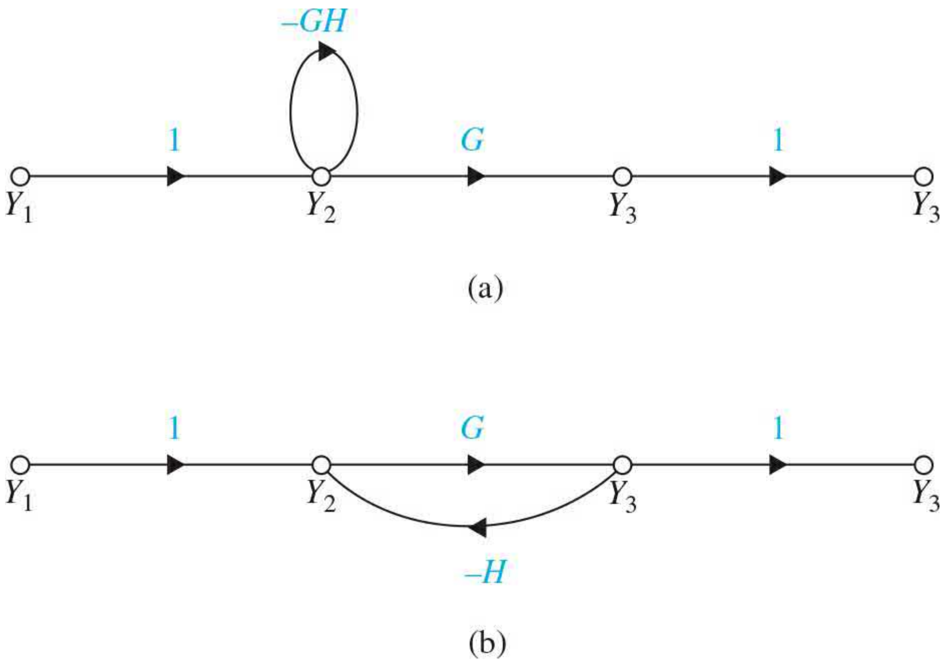


Figure 4P-25

4-26. Show that the two systems shown in Fig. 4P-26*a* and *b* are not equivalent.

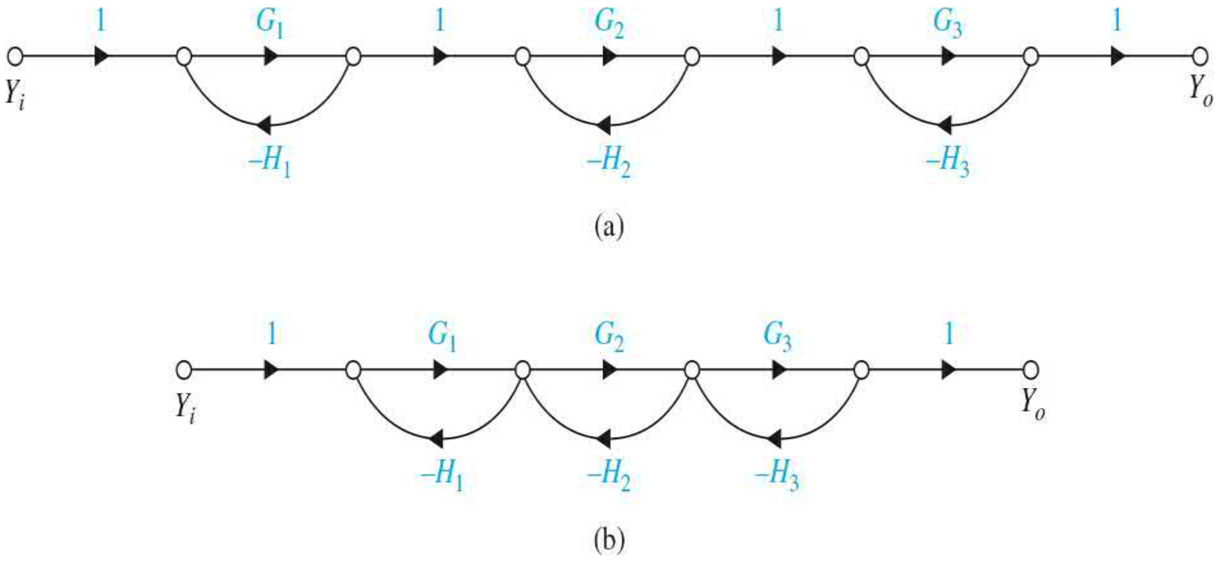


Figure 4P-26

4-27. Find the following transfer functions for the SFG shown in Fig. 4P-27.

$$\left.\frac{Y_6}{Y_1}\right|_{Y_7=0} \quad \left.\frac{Y_6}{Y_7}\right|_{Y_1=0}$$

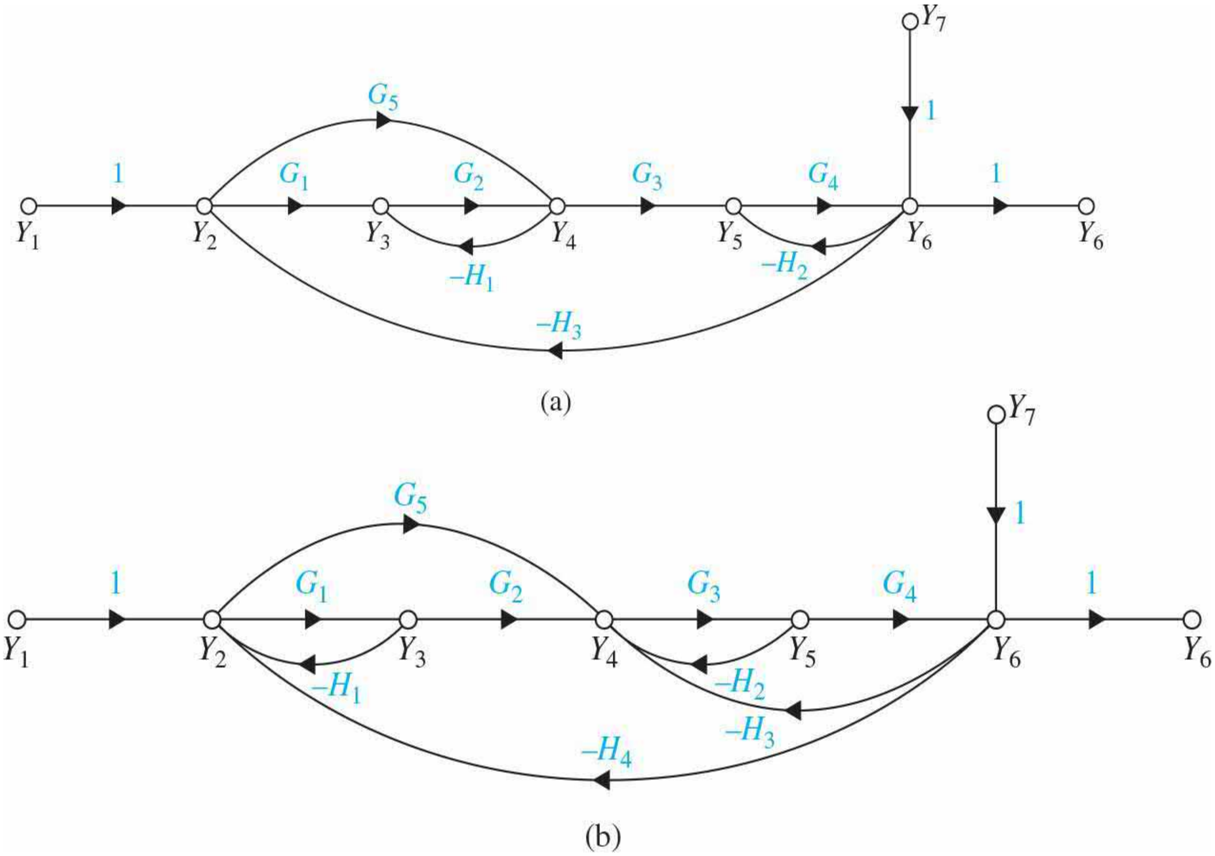


Figure 4P-27

4-28. Find the following transfer functions for the SFG shown in Fig. 4P-28. Comment on why the results for parts (c) and (d) are not the same.

(a) $\left.\dfrac{Y_7}{Y_1}\right|_{Y_8=0}$

(b) $\left.\dfrac{Y_7}{Y_8}\right|_{Y_1=0}$

(c) $\left.\dfrac{Y_7}{Y_4}\right|_{Y_8=0}$

(d) $\left.\dfrac{Y_7}{Y_4}\right|_{Y_1=0}$

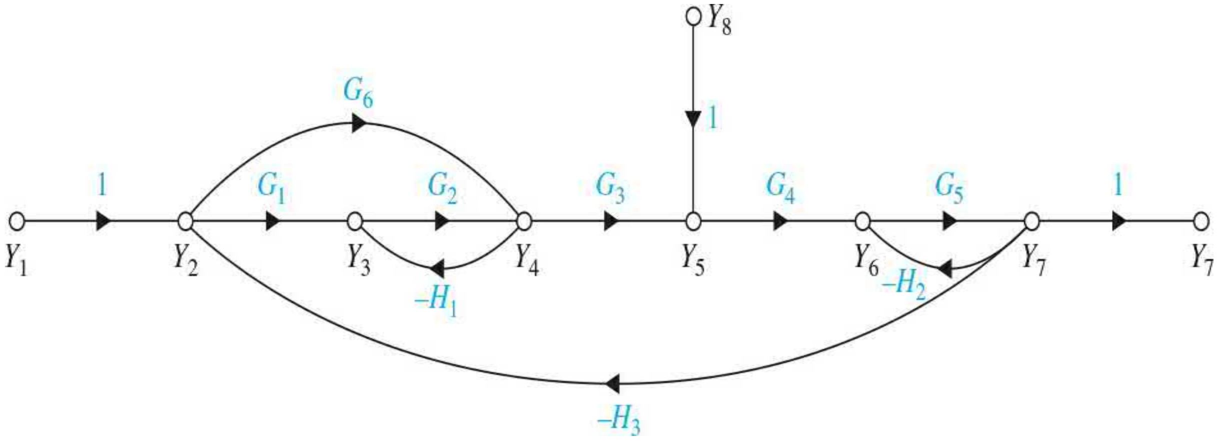


Figure 4P-28

4-29. The coupling between the signals of the turboprop engine shown in Fig. 4P-4*a* is shown in Fig. 4P-29. The signals are defined as

$R_1(s)$ = fuel rate

$R_2(s)$ = propeller blade angle

$Y_1(s)$ = engine speed

$Y_2(s)$ = turbine inlet temperature

(a) Draw an equivalent SFG for the system.

(b) Find the Δ of the system using the SFG gain formula.

(c) Find the following transfer functions:

$$\left.\frac{Y_1(s)}{R_1(s)}\right|_{R_2=0} \quad \left.\frac{Y_1(s)}{R_2(s)}\right|_{R_1=0} \quad \left.\frac{Y_2(s)}{R_1(s)}\right|_{R_2=0} \quad \left.\frac{Y_2(s)}{R_2(s)}\right|_{R_1=0}$$

(d) Express the transfer functions in matrix form, $\mathbf{Y}(s) = \mathbf{G}(s)\mathbf{R}(s)$.

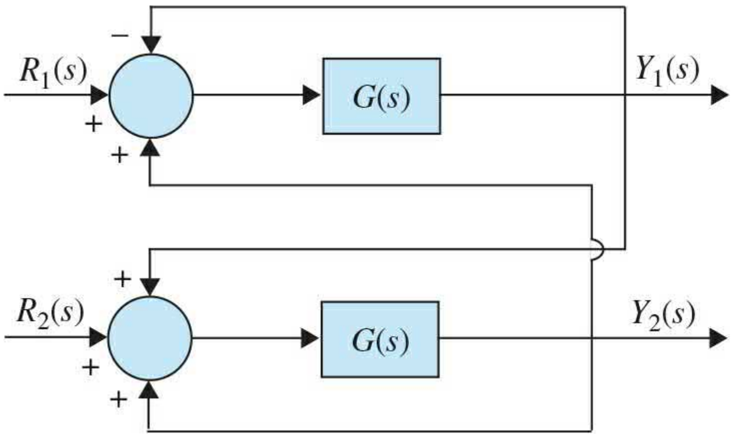


Figure 4P-29

4-30. Figure 4P-30 shows the block diagram of a control system with conditional feedback. The transfer function $G_p(s)$ denotes the controlled process, and $G_c(s)$ and $H(s)$ are the controller transfer functions.

(a) Derive the transfer functions $Y(s)/R(s)|_{N=0}$ and $Y(s)/N(s)|_{R=0}$. Find $Y(s)/R(s)|N=0$ when $G_c(s) = G_p(s)$.

(b) Let

$$G_p(s) = G_c(s) = \frac{100}{(s+1)(s+5)}$$

Find the output response $y(t)$ when $N(s) = 0$ and $r(t) = u_s(t)$.

(c) With $G_p(s)$ and $G_c(s)$ as given in part (b), select $H(s)$ among the following choices such that when $n(t) = u_s(t)$ and $r(t) = 0$, the steady-state value of $y(t)$ is equal to zero. (There may be more than one answer.)

$$H(s) = \frac{10}{s(s+1)} \qquad H(s) = \frac{10}{(s+1)(s+2)}$$

$$H(s) = \frac{10(s+1)}{s+2} \quad H(s) = \frac{K}{s^n} \ (n = \text{positive integer}) \text{ Select } n.$$

Keep in mind that the poles of the closed-loop transfer function must all be in the left-half s-plane for the final-value theorem to be valid.

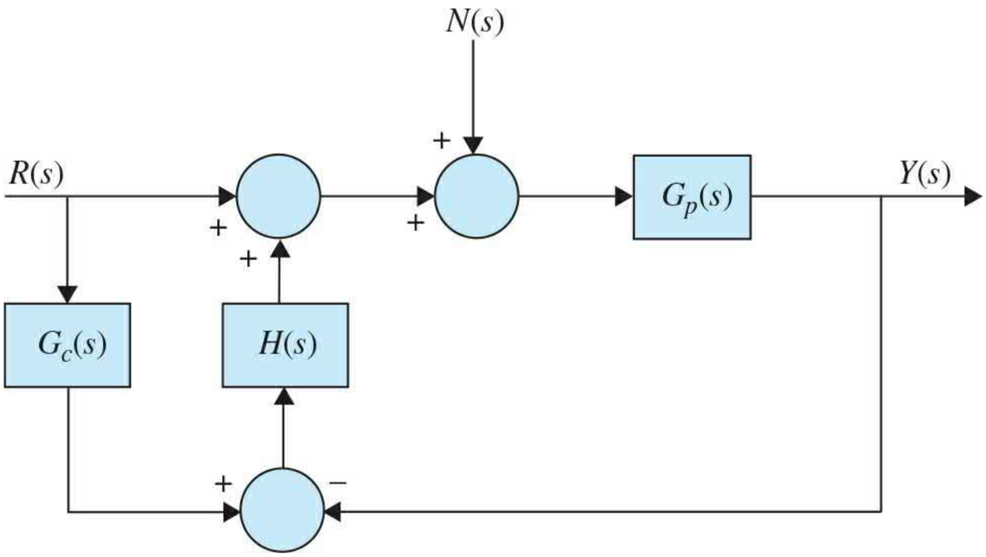


Figure 4P-30

4-31. Use MATLAB to solve Prob. 4-30.

PROBLEMS FOR SEC. 4-3

4-32. Consider the following differential equations of a system:

$$\frac{dx_1(t)}{dt} = -2x_1(t) + 3x_2(t)$$

$$\frac{dx_2(t)}{dt} = -5x_1(t) - 5x_2(t) + 2r(t)$$

(a) Draw a state diagram for the following state equations.
(b) Find the characteristic equation of the system.
(c) Find the transfer functions $X_1(s)/R(s)$ and $X_2(s)/R(s)$.

4-33. The differential equation of a linear system is

$$\frac{d^3y(t)}{dt^3} + 5\frac{d^2y(t)}{dt^2} + 6\frac{dy(t)}{dt} + 10y(t) = r(t)$$

where $y(t)$ is the output, and $r(t)$ is the input.
(a) Draw a state diagram for the system.
(b) Write the state equation from the state diagram. Define the state

variables from right to left in ascending order.

(c) Find the characteristic equation and its roots. Use any computer program to find the roots.

(d) Find the transfer function $Y(s)/R(s)$.

(e) Perform a partial-fraction expansion of $Y(s)/R(s)$.

(f) Find the output $y(t)$ for $t \geq 0$ when $r(t) = u_s(t)$.

(g) Find the final value of $y(t)$ by using the final-value theorem.

4-34. Consider the differential equation given in Prob. 4-33. Use MATLAB to

(a) Perform a partial-fraction expansion of $Y(s)/R(s)$.

(b) Find the Laplace transform of the system.

(c) Find the output $y(t)$ for $t \geq 0$ when $r(t) = u_s(t)$.

(d) Plot the step response of the system.

(e) Verify the final value that you obtained in Prob. 4-33 part (g).

4-35. Repeat Prob. 4-33 for the following differential equation:

$$\frac{d^4 y(t)}{dt^4} + 4\frac{d^3 y(t)}{dt^3} + 3\frac{d^2 y(t)}{dt^2} + 5\frac{dy(t)}{dt} + y(t) = r(t)$$

4-36. Repeat Prob. 4-34 for the differential equation given in Prob. 4-35.

4-37. The block diagram of a feedback control system is shown in Fig. 4P-37.

(a) Derive the following transfer functions:

$$\left.\frac{Y(s)}{R(s)}\right|_{N=0} \quad \left.\frac{Y(s)}{N(s)}\right|_{R=0} \quad \left.\frac{E(s)}{R(s)}\right|_{N=0}$$

(b) The controller with the transfer function $G_4(s)$ is for the reduction of the effect of the noise $N(s)$. Find $G_4(s)$ so that the output $Y(s)$ is totally independent of $N(s)$.

(c) Find the characteristic equation and its roots when $G_4(s)$ is as determined in part (b).

(d) Find the steady-state value of $e(t)$ when the input is a unit-step function. Set $N(s) = 0$.

(e) Find $y(t)$ for $t \geq 0$ when the input is a unit-step function. Use $G_4(s)$ as determined in part (b).

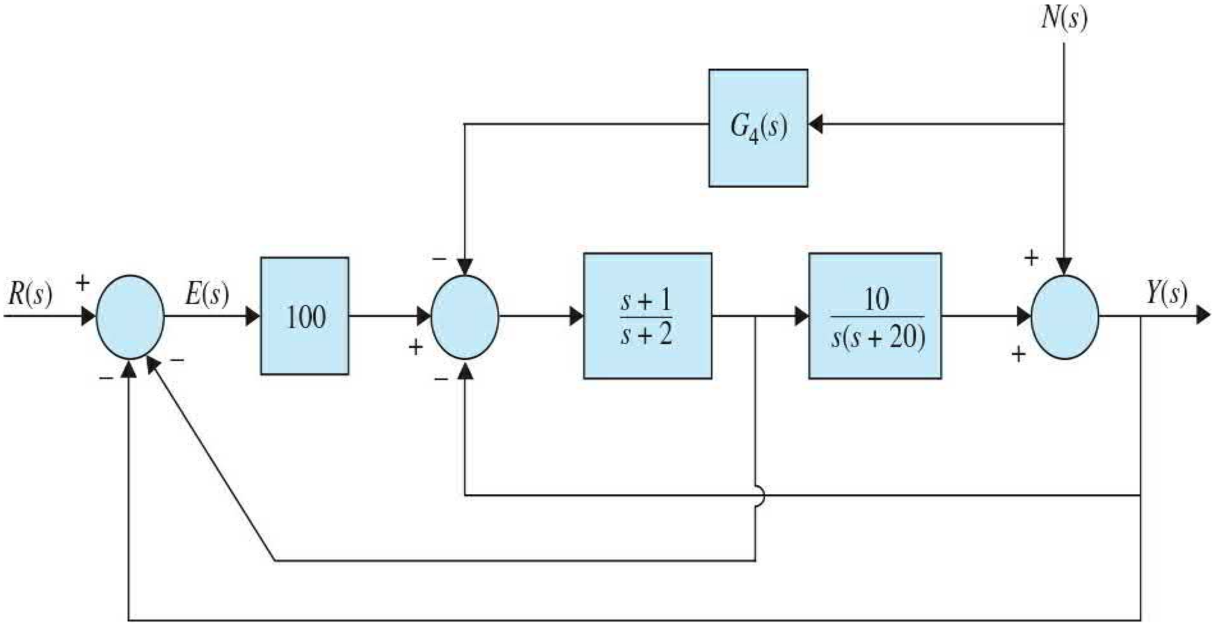


Figure 4P-37

4-38. Use MATLAB to solve Prob. 4-37.

ADDITIONAL PROBLEMS

4-39. Assuming

$$P_1 = 2S^6 + 9S^5 + 15S^4 + 25S^3 + 25S^2 + 14s + 6$$
$$P_2 = S^6 + 8S^5 + 23S^4 + 36S^3 + 38S^2 + 28s + 16$$

(a) Use MATLAB to find roots of P_1 and P_2.

(b) Use MATLAB to calculate $P_3 = P_2 - P_1$, $P_4 = P_2 + P_1$, and $P_5 = (P_1 - P_2)*P_1$.

4-40. Use MATLAB to calculate the polynomial.

(a) $P_6 = (s+1)(s^2+2)(s+3)(2s^2+s+1)$

(b) $P_7 = (s^2+1)(s+2)(s+4)(s^2+2s+1)$

4-41. Use MATLAB to perform partial-fraction expansion to the following functions:

(a) $$G_1(s) = \frac{(s+1)(s^2+2)(s+4)(s+10)}{s(s+2)(s^2+2s+5)(2s^2+s+4)}$$

(b) $$G_2(s) = \frac{s^3+12s^2+47s+60}{4s^6+28s^5+83s^4+135s^3+126s^2+62s+12}$$

4-42. Use MATLAB to calculate unity feedback closed loop transfer function for Prob. 4-41.

4-43. Use MATLAB to calculate

(a) $G_3(s) = G_1(s) + G_2(s)$

(b) $G_4(s) = G_1(s) - G_2(s)$

(c) $$G_5(s) = \frac{G_4(s)}{G_3(s)}$$

(d) $$G_6(s) = \frac{G_4(s)}{G_1(s)*G_2(s)}$$

[1] See Chap. 7 for the difference between an input or a reference input.

CHAPTER 5

Stability of Linear Control Systems

Among the many forms of performance specifications used in design of control systems, **stability** is the most important one. When all types of systems are considered—linear, nonlinear, time-invariant, and time-varying—the definition of stability can be given in many different forms. In this book, we deal only with the stability of linear single-input single-output (SISO) time-invariant systems.

Learning Outcomes

After successful completion of this chapter, you will be able to

1. Assess stability of linear SISO time-invariant systems in the Laplace domain, or in the state space form.
2. Use the Routh-Hurwitz criterion to investigate the stability of the system.
3. Use MATLAB to conduct your stability study.

In this chapter, we introduce the concept of stability, and utilize the Routh-Hurwitz criterion to investigate the stability of SISO time-invariant systems. Through various examples, we investigate the stability of transfer functions and state space systems. We further use MATLAB tools to help solve various problems.

5-1 INTRODUCTION TO STABILITY

In Chap. 3, from the studies of linear differential equations with constant coefficients, we learned that the time response of a linear time-invariant system is usually divided into two parts: the transient response and the steady-state response. Let $y(t)$ denote the time response of a continuous-data system; then, in general, it can be written as

$$y(t)=y_t(t)+y_{ss}(t) \tag{5-1}$$

where $y_t(t)$ denotes the transient response and $y_{ss}(t)$ denotes the steady-state response.

In **stable** control systems, *transient response* corresponds to the homogeneous solution of the governing differential equations and is defined as the part of the time response that goes to zero as time becomes very large. Thus, $y_t(t)$ has the property

$$\lim_{t\to\infty} y_t(t)=0 \tag{5-2}$$

The steady-state response is simply the part of the total response that remains after the transient has died out.

As discussed in Sec. 3-4-3, **stability** of the system is directly dependent on the roots of the system characteristic equation—or the poles of the system. For a bounded input, the total response of the system generally follows the input (bounded-output), if the roots of characteristic equation have negative real components.

> For BIBO stability, the roots of the characteristic equation must all lie in the left-half s-plane.
>
> A system that is BIBO stable is simply called stable; otherwise, it is unstable.

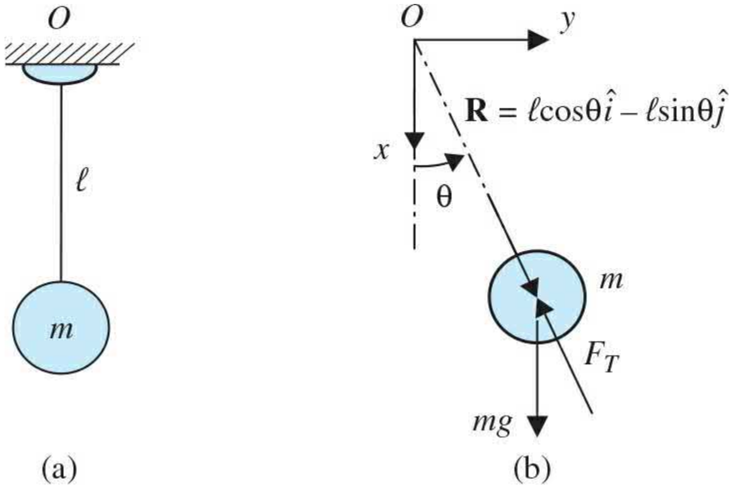
With zero initial conditions, the system is said to be BIBO (bounded-input, bounded-output) **stable**, or **simply stable**, if its output is bounded to a bounded input. For BIBO stability, the roots of the characteristic equation, or the poles of $G(s)$, cannot be located in the right-half s-plane or on the $j\omega$-axis; in other words, they must all lie in the left-half s-plane. A system is said to be *unstable* if it is not BIBO stable. When a system has roots on the $j\omega$-axis, say, at $s = J\omega_0$ and $s = -J\omega_0$, if the input is a sinusoid, (sin $\omega_0 t$), then the output will be of the form of (t sin $\omega_0 t$), which is unbounded, and the system is unstable.

For analysis and design purposes, we can classify stability as **absolute stability** and **relative stability**.[1] Absolute stability refers to whether the system is stable or unstable; it is a *yes* or *no* answer. Once the system is found to be stable, it is of interest to determine how stable it is, and this degree of stability is a measure of relative stability.

We start by demonstrating the concept of stability through a simple example.

EXAMPLE 5-1-1 In Chap. 2, we obtained the equation of motion of a simple (ideal) pendulum with a mass *m* and a massless rod of length *l*, *hinged at point O*, as shown in Fig. 5-1. This equation was simplified to

$$\ddot{\theta}+\frac{g}{\ell}\sin\theta=0 \tag{5-3}$$



O
ℓ
m
(a)
O
y
$\mathbf{R} = \ell\cos\theta\hat{i} - \ell\sin\theta\hat{j}$
x
θ
m
F_T
mg
(b)

Figure 5-1 (*a*) A simple pendulum. (*b*) Free-body diagram of mass *m*.

From the free body diagram of mass *m* in Fig. 5-1*b*, we note the simple pendulum to have two static equilibrium positions, where

$$\begin{aligned}\sum F_x &= -F_T\cos\theta+mg=0\\ \sum F_y &= -F_T\sin\theta=0\end{aligned} \tag{5-4}$$

Upon solving Eq. (5-4), we identify θ =0 and π as the pendulum static equilibria, where it can be fully at rest in the absence of any external forces, initial conditions, or other disturbances. In this case, the weight of the pendulum is completely balanced by the reaction force at the hinge *O*.

Using static equilibrium position θ =0 as the operating point, the linearization of the system implies $\theta \approx \theta$. Hence, the linear representation of the system is

$$\ddot{\theta} + \frac{g}{\ell}\theta = 0 \tag{5-5}$$

or

$$\ddot{\theta} + \omega_n^2 \theta = 0 \tag{5-6}$$

where $\omega_n = \sqrt{\frac{g}{\ell}}\ \text{rad/s}$ rad/s is the natural frequency of the linearized model.

Equation (5-6) describes the pendulum behavior in the neigborhood of the static equilibrium position θ = 0, for small motions (just give the pendulum a slight tap). The system response in this case, after application of a small initial condition $\theta(0) = \theta_o$, is sinusoidal. Further if we account for the hinge *O* friction and the air resistance facing mass *m*, given a small initial condition, the pendulum swings a few times until it comes to rest at θ =0. Note in this case, the **transient response** of the system due to the initial conditions approaches zero with time. As a result, the motion of the pendulum about θ =0 is **stable**.

If we add an external force $F(t)$ to the mass *m* and add viscous damping, which approximates all mentioned frictional forces in the system, Eq. (5-6) can be modified to

$$\ddot{\theta} + 2\zeta\omega_n\dot{\theta} + \omega_n^2\theta = f(t) \tag{5-7}$$

where $f(t)=\frac{F(t)}{m\ell}$. In this case, Eq. (5-7) has the following transfer function

$$\frac{\Theta(s)}{F(s)}=\frac{1}{s^2+2\zeta\omega_n s+\omega_n^2} \tag{5-8}$$

From Chap. 3, for $\zeta > 0$ the **poles** of the system in Eq. (5-8)—that is, roots of the characteristic equation—**have negative real parts**. Note that to obtain the time response of the system, you need to clearly identify and include the initial conditions in the Laplace transform of the system in Eq. (5-7)—see Example 3-4-2. However, because the motion of the damped pendulum about $\theta = 0$ is stable, its transient response diminishes with time, and the pendulum follows the input (**steady-state response**)—say a step input—moving to a constant angle.

Looking back at Eq. (5-3), upon using $\theta = \pi$ as the operating point, the linearized equation of the system is (verify):

$$\ddot{\theta}-\frac{g}{\ell}\theta=0 \tag{5-9}$$

Upon a small disturbance, the pendulum starts swinging about $\theta = 0$; not $\theta = \pi$! As a result, $\theta = \pi$ is considered as an **unstable** equilibrium, and the solution of Eq. (5-9) reflects the pendulum to be moving away from this equilibrium point. The time response of the linearized Eq. (5-9) includes an exponential growth (unbounded motion), regardless of the initial conditions, which implies the mass to be moving away from the point $\theta = \pi$. Obviously the linear approximation range is small and this solution is only valid while the mass stays within this region.

In this case using a similar approach as in Eq. (5-8), we can obtain the transfer function of the system to be

$$\frac{\Theta(s)}{F(s)} = \frac{1}{s^2 + 2\zeta\omega_n s - \omega_n^2} \tag{5-10}$$

where for $\zeta > 0$ one of the **poles** of the system in Eq. (5-10) is **real and positive**, while the other is negative. ▲

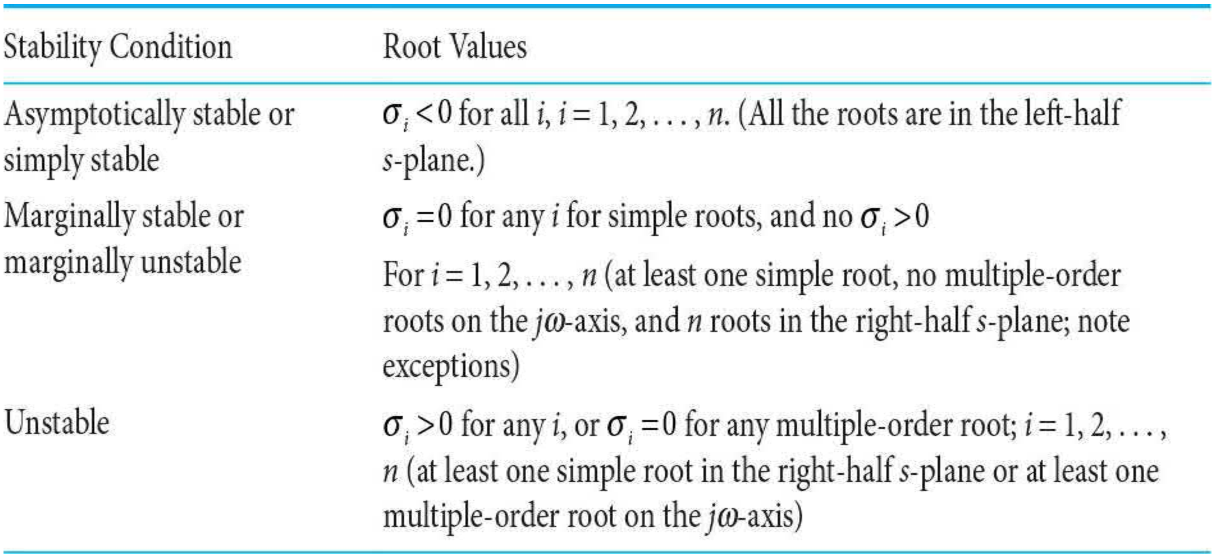
See MATLAB tools in Sec. 5-4.

From the preceding discussions, we see that, *for linear time-invariant systems*, stability of the system can be determined by the poles of the system, and *for a stable system the roots of the characteristic equation must all be located in the left-half* s*-plane. Further*, *if a system is stable*, *it must also be zero-input stable*—that is, its response solely due to initial conditions converges to zero. For this reason, we shall simply refer to the stability condition of a linear system as **stable** or **unstable**. The latter condition refers to the condition that at least one of the characteristic equation roots is not in the left-half *s*-plane. For practical reasons, we often refer to the situation in which the characteristic equation has simple roots on the *jω*-axis and none in the right-half plane as **marginally stable** or **marginally unstable**. *An exception to this is if the system were intended to be an integrator* (*or, in the case of control systems, a velocity control system*); *then the system would have root*(*s*) at *s* =0 and would be considered stable. Similarly, if the system were designed to be an oscillator, the characteristic equation would have simple roots on the *jω*-axis, and the system would be regarded as stable.

As discussed in Sec. 3-7-2, because the roots of the characteristic equation are the same as the **eigenvalues** of matrix **A** of the state equations, the stability condition places the same restrictions on the eigenvalues—see Example 5-4-4. We further discuss the stability analysis of state space equations in Chap. 8.

Let the characteristic equation roots or eigenvalues of matrix **A** of a linear continuous-data time-invariant SISO system be $s_t = \sigma_t + j\omega_t$, i = 1, 2, ..., n. If any of the roots is complex, it is in complex-conjugate pairs. The possible stability conditions of the system are summarized in Table 5-1 with respect to the roots of the characteristic equation.

TABLE 5-1 STABILITY CONDITIONS OF LINEAR CONTINUOUS-DATA TIME-INVARIANT SISO SYSTEMS



Stability Condition	Root Values
Asymptotically stable or simply stable	$\sigma_i < 0$ for all i, $i = 1, 2, \ldots, n$. (All the roots are in the left-half s-plane.)
Marginally stable or marginally unstable	$\sigma_i = 0$ for any i for simple roots, and no $\sigma_i > 0$ For $i = 1, 2, \ldots, n$ (at least one simple root, no multiple-order roots on the $j\omega$-axis, and n roots in the right-half s-plane; note exceptions)
Unstable	$\sigma_i > 0$ for any i, or $\sigma_i = 0$ for any multiple-order root; $i = 1, 2, \ldots, n$ (at least one simple root in the right-half s-plane or at least one multiple-order root on the $j\omega$-axis)

In conclusion, **the stability of linear time-invariant SISO systems can be determined by checking on the location of the roots of the characteristic equation of the system**. For all practical purposes, there is no need to compute the complete system response to determine stability. The regions of stability and instability in the *s*-plane are illustrated in Fig. 5-2.

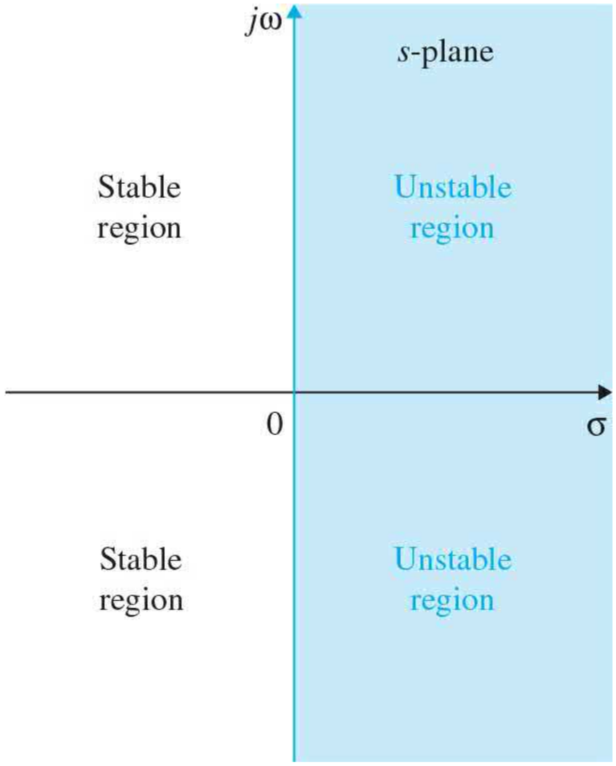


Figure 5-2 Stable and unstable regions in the *s*-plane.

The following example illustrates the stability conditions of systems with reference to the poles of the system transfer functions that are also the roots of the characteristic equation.

EXAMPLE 5-1-2 The following closed-loop transfer functions and their associated stability conditions are given.

$M(s)=\dfrac{20}{(s+1)(s+2)(s+3)}$	BIBO stable (or, asymptotically stable)
$M(s)=\dfrac{20(s+1)}{(s-1)(s^2+2s+2)}$	Unstable due to the pole at $s=1$
$M(s)=\dfrac{20(s-1)}{(s+2)(s^2+4)}$	Marginally stable or marginally unstable due to $s=\pm j2$
$M(s)=\dfrac{10}{(s^2+4)^2(s+10)}$	Unstable due to the multiple-order pole at $s=\pm j2$

5-2 METHODS OF DETERMINING STABILITY

When the system parameters are all known, the roots of the characteristic equation can be found using MATLAB as demonstrated in various MATLAB toolbox windows discussed earlier in Chap. 3 (also see Toolbox 5-3-1). The methods outlined in the following list are well known for determining the stability of linear continuous-data systems without involving root solving.

1. *Routh-Hurwitz criterion.* This criterion is an algebraic method that provides information on the absolute stability of a linear time-invariant system that has a characteristic equation with constant coefficients. The criterion tests whether any of the roots of the characteristic equation lie in the right-half s-plane. The number of roots that lie on the $j\omega$-axis and in the right-half s-plane is also indicated.
2. *Nyquist criterion.* This criterion is a semigraphical method that gives information on the difference between the number of poles and zeros of the closed-loop transfer function that are in the right-half s-plane by observing the behavior of the Nyquist plot of the loop transfer function. This topic is discussed in detail in Chap. 10.
3. *Bode diagram.* This diagram is a plot of the magnitude of the loop transfer function $G(j\omega)H(j\omega)$ in dB and the phase of $G(j\omega)H(j\omega)$ in degrees, all versus frequency ω· The stability of the closed-loop system can be determined by observing the behavior of these plots. This topic is discussed in detail in Chap. 10.

Thus, as will be evident throughout the text, most of the analysis and design techniques on control systems represent alternate methods of solving the same problem. The designer simply has to choose the best analytical tool, depending on the particular situation.

Details of the Routh-Hurwitz stability criterion are presented in the following section.

5-3 ROUTH-HURWITZ CRITERION

The Routh-Hurwitz criterion represents a method of determining the location of zeros of a polynomial with constant real coefficients with respect to the left half and right half of the *s*-plane, without actually solving for the zeros. *Because root-finding computer programs can solve for the zeros of a polynomial with ease, the value of the Routh-Hurwitz criterion is at best limited to equations with at least one unknown parameter*.

Consider that the characteristic equation of a linear time-variant SISO system is of the form

$$F(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0 = 0 \tag{5-11}$$

where all the coefficients are real. To ensure the last equation does not have roots with positive real parts, it is *necessary (but not sufficient)* that the following conditions hold:

1. All the coefficients of the equation have the same sign.
2. None of the coefficients vanish.

These conditions are based on the laws of algebra, which relate the coefficients of Eq. (5-11) as follows:

$$\frac{a_{n-1}}{a_n} = -\sum \text{all roots} \tag{5-12}$$

$$\frac{a_{n-2}}{a_n} = \sum \text{products of the roots taken two at a time} \tag{5-13}$$

$$\frac{a_{n-3}}{a_n} = -\sum \text{product of the roots taken three at a time} \tag{5-14}$$

$$\vdots$$

$$\frac{a_0}{a_n} = (-1)^n \text{ products of all the roots} \tag{5-15}$$

Thus, all these ratios must be positive and nonzero unless at least one of the roots has a positive real part.

The two necessary conditions for Eq. (5-11) to have no roots in the right-half *s*-plane can easily be checked by inspection of the equation. However, these conditions are not sufficient, for it is quite possible that an equation with all its coefficients nonzero and of the same sign still may not have all the roots in the left half of the *s*-plane.

5-3-1 Routh's Tabulation

The Hurwitz criterion gives the necessary and sufficient condition for all roots of Eq. (5-11) to lie in the left half of the *s*-plane. The criterion requires that the equation's *n* Hurwitz determinants must all be positive.

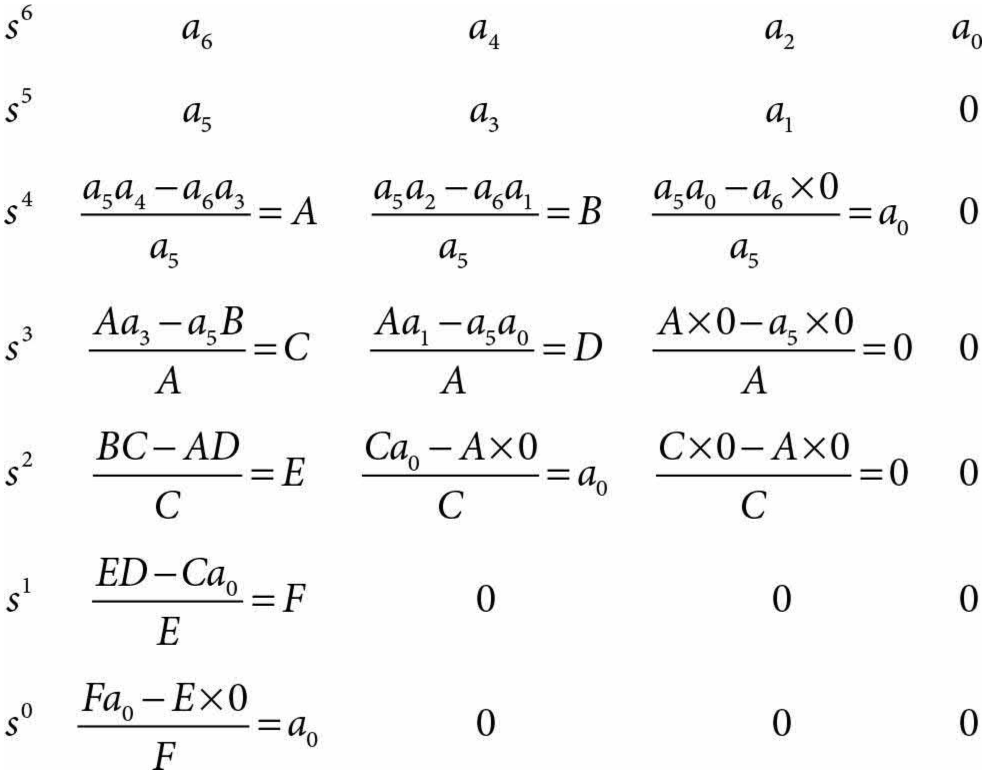
However, the evaluation of the *n* Hurwitz determinants is tedious to carry out. But Routh simplified the process by introducing a tabulation method in place of the Hurwitz determinants.

The first step in the simplification of the Hurwitz criterion, now called the Routh-Hurwitz criterion, is to arrange the coefficients of the equation in Eq. (5-11) into two rows. The first row consists of the first, third, fifth, …, coefficients, and the second row consists of the second, fourth, sixth, …, coefficients, all counting from the highest-order term, as shown in the following tabulation:

$$\begin{array}{lllll} a_n & a_{n-2} & a_{n-4} & a_{n-6} & \cdots \\ a_{n-1} & a_{n-3} & a_{n-5} & a_{n-7} & \cdots \end{array}$$

The next step is to form the following array of numbers by the indicated operations, illustrated here for a sixth-order equation:

$$a_6 s^6 + a_5 s^5 + \cdots + a_1 s + a_0 = 0 \tag{5-16}$$



s^6	a_6	a_4	a_2	a_0
s^5	a_5	a_3	a_1	0
s^4	$\frac{a_5a_4-a_6a_3}{a_5}=A$	$\frac{a_5a_2-a_6a_1}{a_5}=B$	$\frac{a_5a_0-a_6\times 0}{a_5}=a_0$	0
s^3	$\frac{Aa_3-a_5B}{A}=C$	$\frac{Aa_1-a_5a_0}{A}=D$	$\frac{A\times 0-a_5\times 0}{A}=0$	0
s^2	$\frac{BC-AD}{C}=E$	$\frac{Ca_0-A\times 0}{C}=a_0$	$\frac{C\times 0-A\times 0}{C}=0$	0
s^1	$\frac{ED-Ca_0}{E}=F$	0	0	0
s^0	$\frac{Fa_0-E\times 0}{F}=a_0$	0	0	0

This array is called the **Routh’s tabulation** or **Routh’s array**. The column of *s*’s on the left side is used for identification purposes. The reference column keeps track of the calculations, and the last row of the Routh’s tabulation should always be the s^0 row.

Once the Routh’s tabulation has been completed, the last step in the application of the criterion is to investigate the *signs* of the coefficients in the *first column* of the tabulation, which contains information on the roots of the equation. The following conclusions are made:

> The roots of the equation are all in the left half of the *s*-plane if all the elements of the first column of the Routh’s tabulation are of the same sign. The number of changes of signs in the elements of the first column equals the number of roots with positive real parts, or those in the right-half *s*-plane.

The following examples illustrate the applications of the Routh-Hurwitz criterion when the tabulation terminates without complications.

EXAMPLE 5-3-1 Consider the equation

$$2s^4+s^3+3s^2+5s+10=0 \tag{5-17}$$

Because the equation has no missing terms and the coefficients are all of the same sign, it satisfies the necessary condition for not having roots in the right-half or on the imaginary axis of the *s*-plane. However, the sufficient condition must still be checked. Routh's tabulation is made as follows:

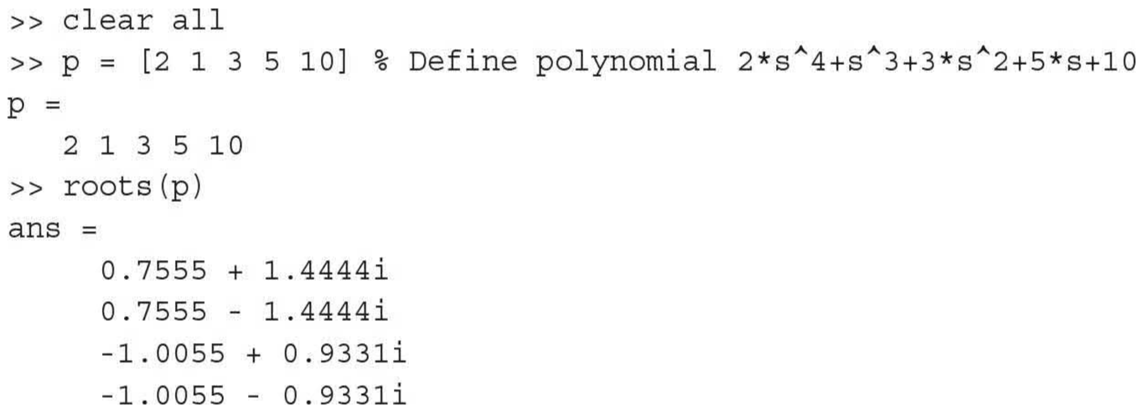
See MATLAB tools in Sec. 5-4.

	s^4	2	3	10
	s^3	1	5	0
Sign change	s^2	$\dfrac{(1)(3)-(2)(5)}{1}=-7$	10	0
Sign change	s^1	$\dfrac{(-7)(5)-(1)(10)}{-7}=6.43$	0	0
	s^0	10	2	0

Because there are two sign changes in the first column of the tabulation, the equation has two roots in the right half of the *s*-plane. Solving for the roots of Eq. (5-17), we have the four roots at $s=-1.005\pm j0.933$ and $s=0.755\pm j1.444$. Clearly, the last two roots are in the right-half *s*-plane, which cause the system to be unstable.

Toolbox 5-3-1

The roots of the polynomial in Eq. (5-17) are obtained using the following sequence of MATLAB functions.



```
>> clear all
>> p = [2 1 3 5 10] % Define polynomial 2*s^4+s^3+3*s^2+5*s+10
p =
   2 1 3 5 10
>> roots(p)
ans =
     0.7555 + 1.4444i
     0.7555 - 1.4444i
    -1.0055 + 0.9331i
    -1.0055 - 0.9331i
```

5-3-2 Special Cases When Routh’s Tabulation Terminates Prematurely

The equations considered in the two preceding examples are designed so that Routh’s tabulation can be carried out without any complications. Depending on the coefficients of the equation, the following difficulties may occur, which prevent Routh’s tabulation from completing properly:

The coefficients of the auxiliary equation are those of the row just above the row of zeros in Routh’s tabulation.

The roots of the auxiliary equation must also satisfy the original equation.

1. The first element in any one row of Routh’s tabulation is zero, but the others are not.
2. The elements in one row of Routh’s tabulation are all zero.

In the first case, if a zero appears in the first element of a row, the elements in the next row will all become infinite, and Routh’s tabulation cannot continue.

To remedy the situation, we *replace the zero element in the first column by an arbitrary small positive number ε, and then proceed with Routh's tabulation*. This is illustrated by the following example.

EXAMPLE 5-3-2 Consider the characteristic equation of a linear system

$$s^4 + s^3 + 2s^2 + 2s + 3 = 0 \tag{5-18}$$

Because all the coefficients are nonzero and of the same sign, we need to apply the Routh-Hurwitz criterion. Routh's tabulation is carried out as follows:

$$\begin{array}{llll}
s^4 & 1 & 2 & 3 \\
s^3 & 1 & 2 & 0 \\
s^2 & 0 & 3 &
\end{array}$$

See MATLAB tools in Sec. 5-4.

Because the first element of the *s2* row is zero, the elements in the *s1* row would all be infinite. To overcome this difficulty, we replace the zero in the *s2* row with a small positive number *ε*, and then proceed with the tabulation. Starting with the *s2* row, the results are as follows:

$$\begin{array}{llcc}
 & s^2 & \varepsilon & 3 \\
\text{Sign change} & s^1 & \dfrac{2\varepsilon - 3}{\varepsilon} \cong -\dfrac{3}{\varepsilon} & 0 \\
\text{Sign change} & s^0 & 3 & 0
\end{array}$$

Because there are two sign changes in the first column of Routh's tabulation, the equation in Eq. (5-18) has two roots in the right-half s-plane. Solving for the roots of Eq. (5-18), we get $s=-0.091\pm j0.902$ and $s=0.406\pm j1.293$; the last two roots are clearly in the right-half s-plane.

It should be noted that the ε-method described may not give correct results if the equation has pure imaginary roots. ▲

In the second special case, when all the elements in one row of Routh's tabulation are zeros before the tabulation is properly terminated, it indicates that one or more of the following conditions may exist:

1. The equation has at least one pair of real roots with equal magnitude but opposite signs.
2. The equation has one or more pairs of imaginary roots.
3. The equation has pairs of complex-conjugate roots forming symmetry about the origin of the s-plane; for example, $s=-1\pm j1$, $s=1\pm j1$.

The situation with the entire row of zeros can be remedied by using the **auxiliary equation** $A(s)=0$, which is formed from the coefficients of the row just above the row of zeros in Routh's tabulation. The auxiliary equation is always an even polynomial; that is, only even powers of s appear. *The roots of the auxiliary equation also satisfy the original equation*. Thus, by solving the auxiliary equation, we also get some of the roots of the original equation. To continue with Routh's tabulation when a row of zero appears, we conduct the following steps:

1. Form the auxiliary equation $A(s)=0$ by using the coefficients from the row just preceding the row of zeros.
2. Take the derivative of the auxiliary equation with respect to s; this gives $dA(s)/ds=0$.
3. Replace the row of zeros with the coefficients of $dA(s)/ds=0$.
4. Continue with Routh's tabulation in the usual manner with the newly formed row of coefficients replacing the row of zeros.
5. Interpret the change of signs, if any, of the coefficients in the first column of the Routh's tabulation in the usual manner.

EXAMPLE 5-3-3 Consider the following equation, which may be the characteristic equation of a linear control system:

$$s^5 + 4s^4 + 8s^3 + 8s^2 + 7s + 4 = 0 \tag{5-19}$$

Routh's tabulation is

$$\begin{array}{llll} s^5 & 1 & 8 & 7 \\ s^4 & 4 & 8 & 4 \\ s^3 & 6 & 6 & 0 \\ s^2 & 4 & 4 & \\ s^1 & 0 & 0 & \end{array}$$

See MATLAB tools in Sec. 5-4.

Because a row of zeros appears prematurely, we form the auxiliary equation using the coefficients of the *s2* row:

$$A(s) = 4s^2 + 4 = 0 \tag{5-20}$$

The derivative of $A(s)$ with respect to s is

$$\frac{dA(s)}{ds} = 8s = 0 \tag{5-21}$$

from which the coefficients 8 and 0 replace the zeros in the *s1* row of the original tabulation. The remaining portion of the Routh's tabulation is

$$\begin{array}{llll} s^1 & 8 & 0 & \text{coefficients of } dA(s)/ds \\ s^0 & 4 & & \end{array}$$

Because there are no sign changes in the first column of the entire Routh's tabulation, the equation in Eq. (5-21) does not have any root in the right-half s-plane. Solving the auxiliary equation in Eq. (5-20), we get the two roots at $s = j$ and $s =-j$, which are also two of the roots of Eq. (5-19). Thus, the equation has two roots on the $j\omega$-axis, and the system is marginally stable. These imaginary roots caused the initial Routh's tabulation to have the entire row of zeros in the s^1 row.

Because all zeros occurring in a row that corresponds to an odd power of s creates an auxiliary equation that has only even powers of s, the roots of the auxiliary equation may all lie on the $j\omega$-axis. For design purposes, we can use the all-zero-row condition to solve for the marginal value of a system parameter for system stability. The following example illustrates the realistic value of the Routh-Hurwitz criterion in a simple design problem.

EXAMPLE 5-3-4 Consider that a third-order control system has the characteristic equation

$$s^3 + 3408.3s^2 + 1{,}204{,}000s + 1.5\times10^7 K = 0 \tag{5-22}$$

See MATLAB tools in Sec. 5-4.

The Routh-Hurwitz criterion is best suited to determine the critical value of K for stability, that is, the value of K

for which at least one root will lie on the *j*ω-axis and none in the right-half *s*-plane. Routh's tabulation of Eq. (5-22) is made as follows:

$$\begin{array}{lcc} s^3 & 1 & 1{,}204{,}000 \\ s^2 & 3408.3 & 1.5\times 10^7 K \\ s^1 & \dfrac{410.36\times 10^7 - 1.5\times 10^7 K}{3408.3} & 0 \\ s^0 & 1.5\times 10^7 K & \end{array}$$

For the system to be stable, all the roots of Eq. (5-22) must be in the left-half *s*-plane, and, thus, all the coefficients in the first column of Routh's tabulation must have the same sign. This leads to the following conditions:

$$\frac{410.36\times 10^7 - 1.5\times 10^7 K}{3408.3} > 0 \tag{5-23}$$

and

$$1.5\times 10^7 K > 0 \tag{5-24}$$

From the inequality of Eq. (5-23), we have $K < 273.57$, and the condition in Eq. (5-24) gives $K > 0$. Therefore, the condition of *K* for the system to be stable is

$$0 < K < 273.57 \tag{5-25}$$

If we let $K = 273.57$, the characteristic equation in Eq. (5-22) will have two roots on the *j*ω-axis. To find these roots, we substitute $K = 273.57$ in the auxiliary equation, which is obtained from Routh's tabulation by using the coefficients of the s^2 row. Thus,

$$A(s) = 3408.3s^2 + 4.1036\times 10^9 = 0 \tag{5-26}$$

which has roots at $s = j1097$ and $s = -j1097$, and the corresponding value of K at these roots is 273.57. Also, if the system is operated with K = 273.57, the zero-input response of the system will be an undamped sinusoid with a frequency of 1097.27 rad/s. ▲

EXAMPLE 5-3-5 As another example of using the Routh-Hurwitz criterion for simple design problems, consider that the characteristic equation of a closed-loop control system is

$$s^3 + 3Ks^2 + (K+2)s + 4 = 0 \tag{5-27}$$

See MATLAB tools in Sec. 5-4.

It is desired to find the range of K so that the system is stable. Routh's tabulation of Eq. (5-27) is

$$\begin{array}{lcc}
s^3 & 1 & K+2 \\
s^2 & 3K & 4 \\
s^1 & \dfrac{3K(K+2)-4}{3K} & 0 \\
s^0 & 4 &
\end{array}$$

From the s^2 row, the condition of stability is $K < 0$, and from the s^1 row, the condition of stability is

$$3K^2 + 6K - 4 > 0 \tag{5-28}$$

or

$$K < -2.528 \quad \text{or} \quad K > 0.528 \tag{5-29}$$

When the conditions of $K > 0$ and $K > 0.528$ are compared, it is apparent that the latter requirement is more stringent. Thus, for the closed-loop system to be stable, K must satisfy

$$K > 0.528 \tag{5-30}$$

The requirement of $K < -2.528$ is disregarded because K cannot be negative. ▲

It should be reiterated that the Routh-Hurwitz criterion is valid only if the characteristic equation is algebraic with real coefficients. If any one of the coefficients is complex, or if the equation is not algebraic, for example, containing exponential functions or sinusoidal functions of s, the Routh-Hurwitz criterion simply cannot be applied.

Another limitation of the Routh-Hurwitz criterion is that it is valid only for the determination of roots of the characteristic equation with respect to the left half or the right half of the s-plane. The stability boundary is the $j\omega$-axis of the s-plane. The criterion *cannot* be applied to any other stability boundaries in a complex plane, such as the unit circle in the z-plane, which is the stability boundary of discrete-data systems (App. H).

5-4 MATLAB TOOLS AND CASE STUDIES

The easiest way to assess stability of known transfer functions is to find the location of the poles. For that purpose, the MATLAB code that appears in Toolbox 5-3-1 is the easiest way for finding the roots of the characteristic equation polynomial—that is, the poles of the system.

In this section, we introduce the **tfrouth** stability tool, which may be used to find the Routh array, and more importantly it may be utilized for controller design applications where it is important to assess the stability of a system for a controller gain, say k.

The steps involved in setting up and then solving a given stability problem using tfrouth are as follows:

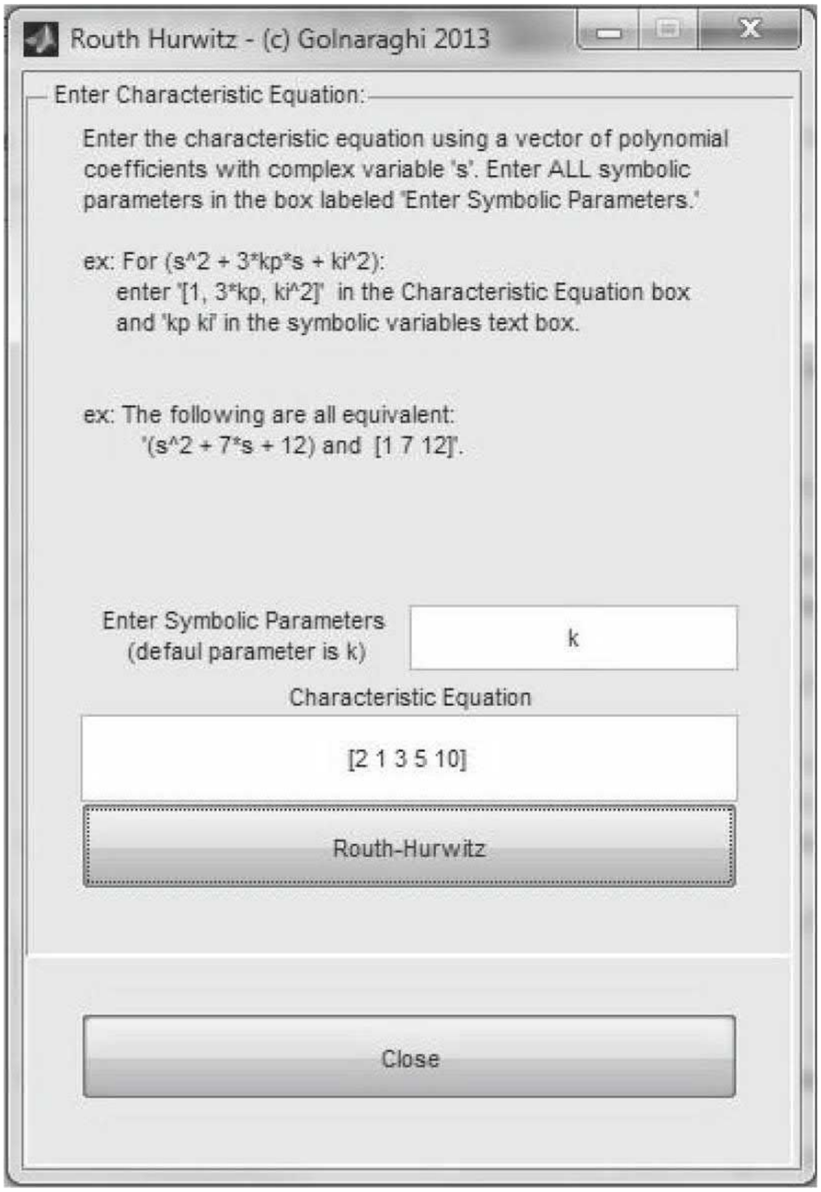
1. In order to use the tfrouth tool, you first need to download the **ACSYS** software, available at www.mhprofessional.com/golnaraghi.
2. Load MATLAB, and use the folder browser at the top of the MATLAB command window to go to your ACSYS directory—for example, for PCs, C:\documents\ACSYS2013—if you have put ACSYS2013 directory in your documents folder.[2]
3. At the MATLAB command window type “dir” and identify the “TFSymbolic” directory.
4. Move to “TFSymbolic” directory by typing in “cd TFSymbolic.”
5. Type “tfrouth” in the MATLAB command window within the “TFSymbolic” directory.
6. The “Routh-Hurwitz” window will appear. Enter the characteristic polynomial as a coefficient row vector (e.g., for $s^3 + s^2 + s + 1$, enter: [1 1 1 1]).
7. Press the “Routh-Hurwitz” button and check the results in the MATLAB command window.
8. In case you wish to assess the stability of the system for a design parameter, enter it in the box designated as “Enter Symbolic Parameters.” For example, for $s^3 + k_1 s^2 + k_2 s + 1$, you need to enter “k1 k2” in the “Enter Symbolic Parameters” box, followed by entering the polynomial as [1 k1 k2 1] in the “Characteristic Equation” box. Note that the default coefficient is “k,” which you can change to other character(s)—in this case k1 and k2.
9. Press the “Routh-Hurwitz” button to form the Routh’s array and conduct the Routh-Hurwitz stability test.

To better illustrate how to use tfrouth, let us solve some of the earlier examples in this chapter.

EXAMPLE 5-4-1 Recall Example 5-3-1; let’s use tfrouth for the following polynomial:

$$2s^4 + s^3 + 3s^2 + 5s + 10 = 0 \tag{5-31}$$

Following the instructions in the beginning of the section, in the MATLAB command window, type in "tfrouth" and enter the characteristic Eq. (5-31) in coefficient vector form [2 1 3 5 10], followed by clicking the "Routh-Hurwitz" button to get the Routh-Hurwitz matrix, as shown in Fig. 5-3.


Routh Hurwitz - (c) Golnaraghi 2013
Enter Characteristic Equation:

Enter the characteristic equation using a vector of polynomial coefficients with complex variable 's'. Enter ALL symbolic parameters in the box labeled 'Enter Symbolic Parameters.'

ex: For (s^2 + 3*kp*s + ki^2):
enter '[1, 3*kp, ki^2]' in the Characteristic Equation box
and 'kp ki' in the symbolic variables text box.

ex: The following are all equivalent:
'(s^2 + 7*s + 12) and [1 7 12]'.

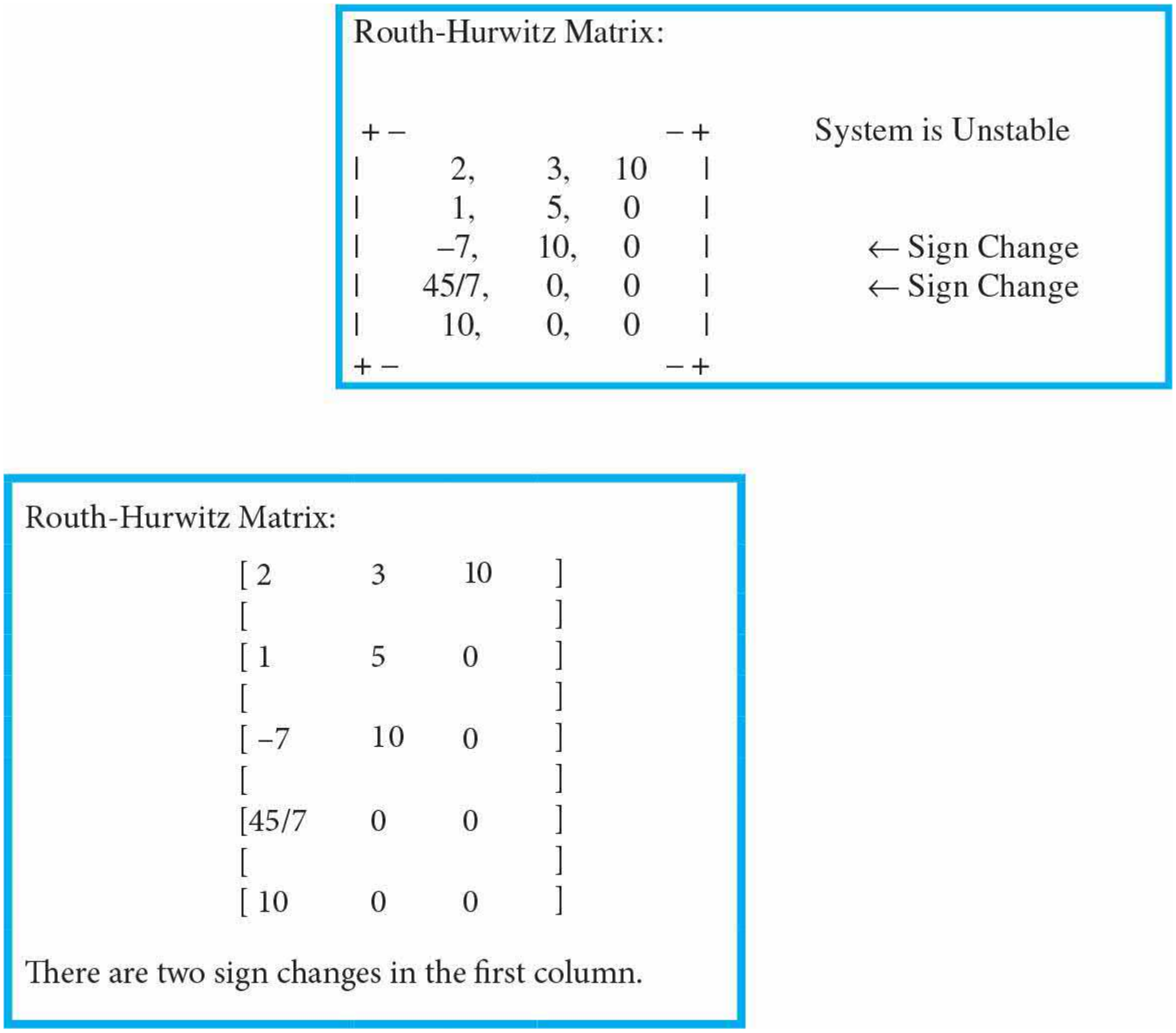
Enter Symbolic Parameters
(defaul parameter is k): k

Characteristic Equation: [2 1 3 5 10]

Figure 5-3 Entering characteristic polynomial for Example 5-3-1 using the tfrouth module.

The results match with Example 5-3-1. The system is therefore unstable because of two positive poles. The Routh's array first column also shows two sign changes to confirm this result. To see the complete Routh table, the user must refer to the MATLAB command window, as shown in Fig. 5-4.

The following box shows where the sign changes occur.



```
Routh-Hurwitz Matrix:


 +-                     -+         System is Unstable
 |      2,      3,   10  |
 |      1,      5,   0   |
 |     -7,     10,   0   |           ← Sign Change
 |    45/7,     0,   0   |           ← Sign Change
 |     10,      0,   0   |
 +-                     -+
```

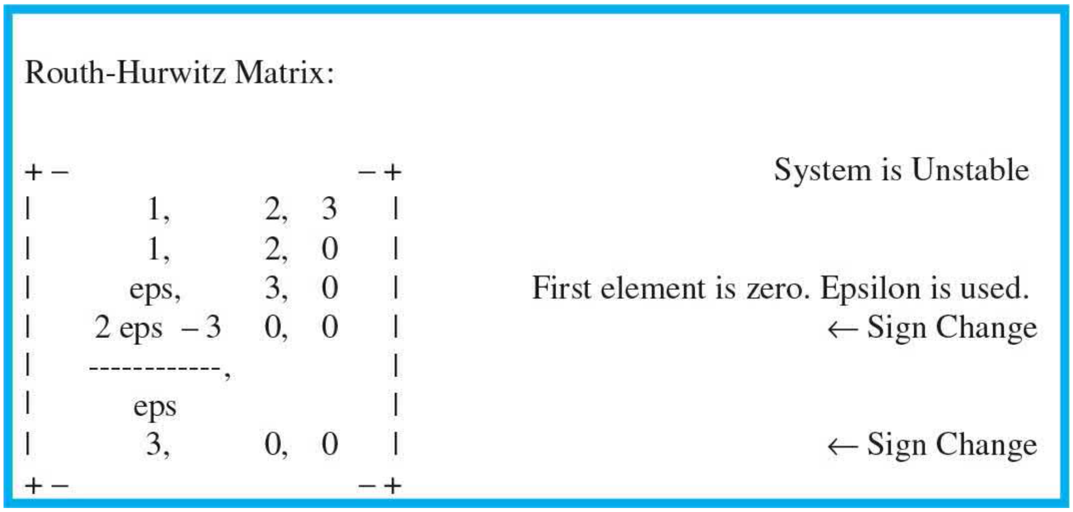
```
Routh-Hurwitz Matrix:
                 [ 2       3      10    ]
                 [                      ]
                 [ 1       5      0     ]
                 [                      ]
                 [ -7      10     0     ]
                 [                      ]
                 [45/7     0      0     ]
                 [                      ]
                 [ 10      0      0     ]

There are two sign changes in the first column.
```

Figure 5-4 Stability results for Example 5-4-1, after using the Routh-Hurwitz test.

EXAMPLE 5-4-2 Consider Example 5-3-2 for characteristic equation of a linear system:

$$s^4+s^3+2s^2+2s+3=0 \qquad (5\text{-}32)$$

After entering the transfer function characteristic equation using tfrouth as [1 1 2 3] and pressing the “Routh-Hurwitz” button, we get the following output:



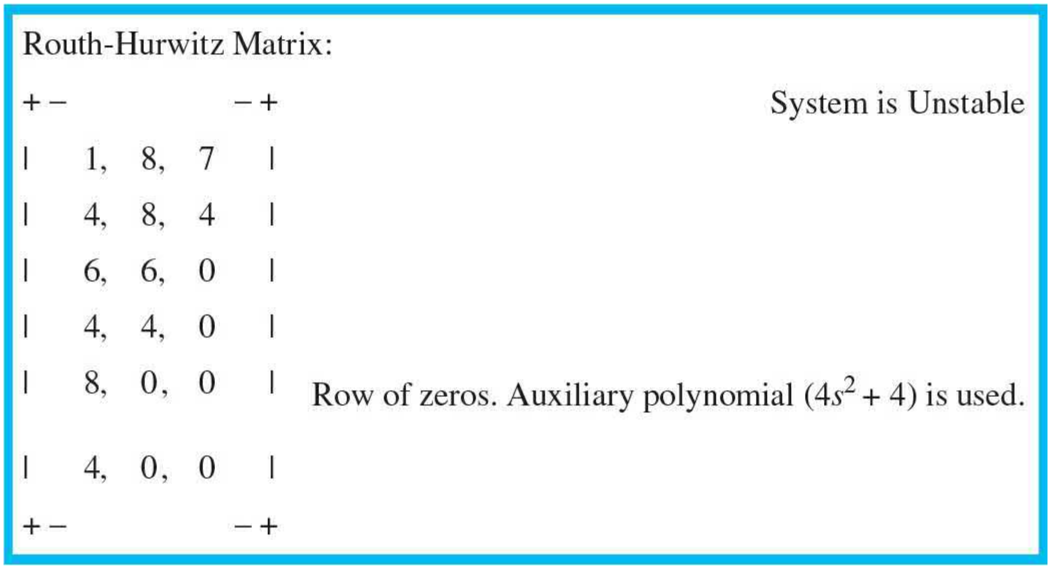
```
Routh-Hurwitz Matrix:


+ -                             - +                  System is Unstable
|       1,          2,    3       |
|       1,          2,    0       |
|      eps,         3,    0       |        First element is zero. Epsilon is used.
|   2 eps  - 3      0,    0       |                              ← Sign Change
|   ------------,                 |
|      eps                        |
|       3,          0,    0       |                              ← Sign Change
+ -                             - +
```

Hence, because of the final two sign changes, we expect to see two unstable poles.

EXAMPLE 5-4-3 Revisiting Example 5-3-3, use tfrouth to study the following characteristic equation:

$$s^5+4s^4+8s^3+8s^2+7s+4=0 \tag{5-33}$$

Enter the characteristic equation coefficients as [1 4 8 8 7 4]. The MATLAB output will be as follows:

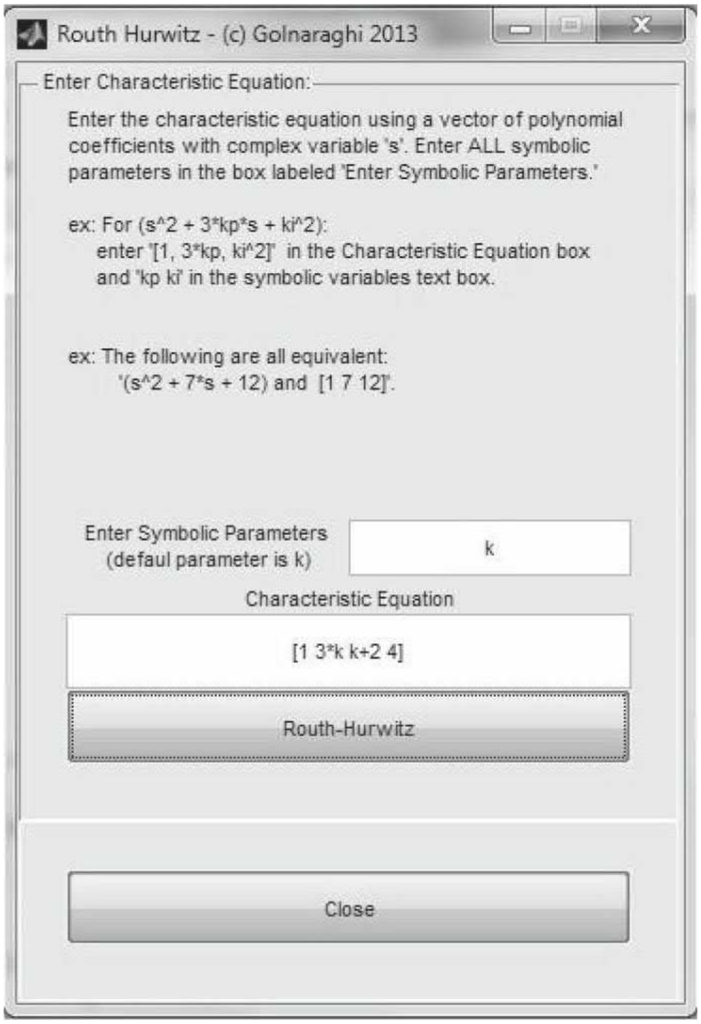


```
Routh-Hurwitz Matrix:
+−              −+                                   System is Unstable
|   1,   8,   7   |
|   4,   8,   4   |
|   6,   6,   0   |
|   4,   4,   0   |
|   8,   0,   0   |   Row of zeros. Auxiliary polynomial (4s^2 + 4) is used.

|   4,   0,   0   |
+−              −+
```

EXAMPLE 5-4-4 Considering the characteristic equation of a closed-loop control system

$$s^3 + 3Ks^2 + (K+2)s + 4 = 0 \qquad (5\text{-}34)$$

It is desired to find the range of *K* so that the system is stable. Using the default parameter “k,” enter the characteristic equation coefficients as [1 3 * *k k* + 2 4]—see Fig. 5-5.



Enter Characteristic Equation:

Enter the characteristic equation using a vector of polynomial coefficients with complex variable 's'. Enter ALL symbolic parameters in the box labeled 'Enter Symbolic Parameters.'

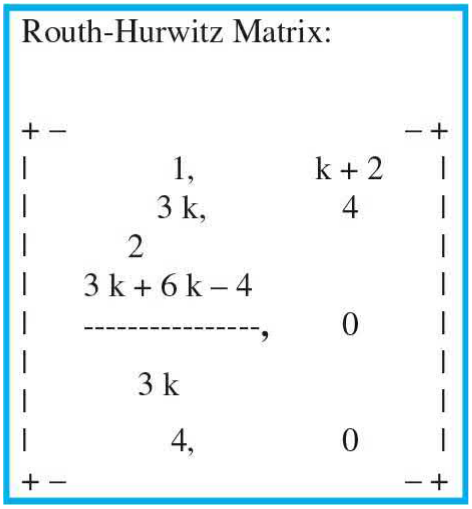
ex: For (s^2 + 3*kp*s + ki^2):
enter '[1, 3*kp, ki^2]' in the Characteristic Equation box
and 'kp ki' in the symbolic variables text box.

ex: The following are all equivalent:
'(s^2 + 7*s + 12) and [1 7 12]'.

Enter Symbolic Parameters (defaul parameter is k): k

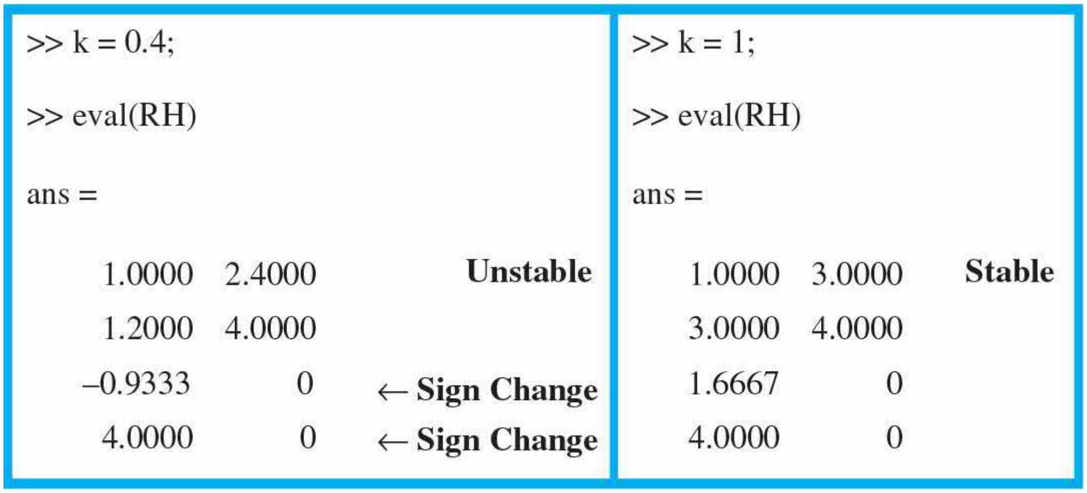
Characteristic Equation: [1 3*k k+2 4]

Figure 5-5 Entering characteristic polynomial for Example 5-4-4 using the tfrouth module.



```
Routh-Hurwitz Matrix:


+ -                              - +
|         1,           k + 2       |
|        3 k,            4         |
|      2                           |
|   3 k + 6 k - 4                  |
|   ----------------,    0         |
|                                  |
|       3 k                        |
|         4,             0         |
+ -                              - +
```

In tfrouth, the Routh array is store under variable name “RH.” In order to check the system stability for different values of *k*, you need to assign a value to *k* first. Then use the MATLAB command “eval(RH)” to obtain the numerical value of the Routh array as shown below:



```
>> k = 0.4;

>> eval(RH)

ans =

     1.0000   2.4000              Unstable
     1.2000   4.0000
    –0.9333        0        ← Sign Change
     4.0000        0        ← Sign Change
```

```
>> k = 1;

>> eval(RH)

ans =

     1.0000   3.0000      Stable
     3.0000   4.0000
     1.6667        0
     4.0000        0
```

In this case, the system is unstable for *k* = 0.4, and stable for *k* = 1. ▲

EXAMPLE 5-4-5 Considering the following state space system:

$$\begin{aligned}\frac{dx_1(t)}{dt} &= x_1(t) - 4x_2(t)\\ \frac{dx_2(t)}{dt} &= 5x_1(t) + u(t)\end{aligned} \tag{5-35}$$

State equation of the system is

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\,\mathbf{x}(t) + \mathbf{B}\,u(t)\\ \mathbf{y}(t) &= \mathbf{C}\,\mathbf{x}(t)\end{aligned} \tag{5-36}$$

where

$$\mathbf{A} = \begin{bmatrix} 1 & -4 \\ 5 & 0 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{5-37}$$

Taking the Laplace transform of Eq. (5-36), assuming zero initial conditions, and solving for **Y**(*s*), we have

$$\mathbf{Y}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{U}(s) \tag{5-38}$$

From the discussions in Secs. 3-7-1 and 3-7-2, the system transfer function in this case is

$$\begin{aligned}\mathbf{G}(s) &= \mathrm{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} = \mathbf{C}\frac{\mathbf{adj}(s\mathbf{I} - \mathbf{A})}{\mathbf{det}(s\mathbf{I} - \mathbf{A})}\mathbf{B} \\ &= \mathbf{C}\frac{[\mathbf{adj}(s\mathbf{I} - \mathbf{A})]\mathbf{B}}{|s\mathbf{I} - \mathbf{A}|} = \frac{\begin{bmatrix} -4 \\ s-1 \end{bmatrix}}{s^2 - s + 20}\end{aligned} \tag{5-39}$$

More specifically, noting **Y**(*s*) = **X**(*s*), we have

$$\frac{X_1(s)}{U(s)} = \frac{-4}{s^2 - s + 20} \tag{5-40}$$

$$\frac{X_2(s)}{U(s)} = \frac{s-1}{s^2 - s + 20} \tag{5-41}$$

Setting the denominator of the transfer-function matrix **G**(*s*) to zero, we get the **characteristic equation:**

$$|s\mathbf{I} - \mathbf{A}| = 0 \tag{5-42}$$

This implies that the roots of the characteristic equation are the **eigenvalues** of the matrix **A**. Hence,

$$|s\mathbf{I} - \mathbf{A}| = \begin{vmatrix} s-1 & 4 \\ -5 & s \end{vmatrix} = s^2 - s + 20 = 0 \tag{5-43}$$

From the first Routh-Hurwitz criterion, this system is **unstable—**see Fig. 5-6 for the time response.

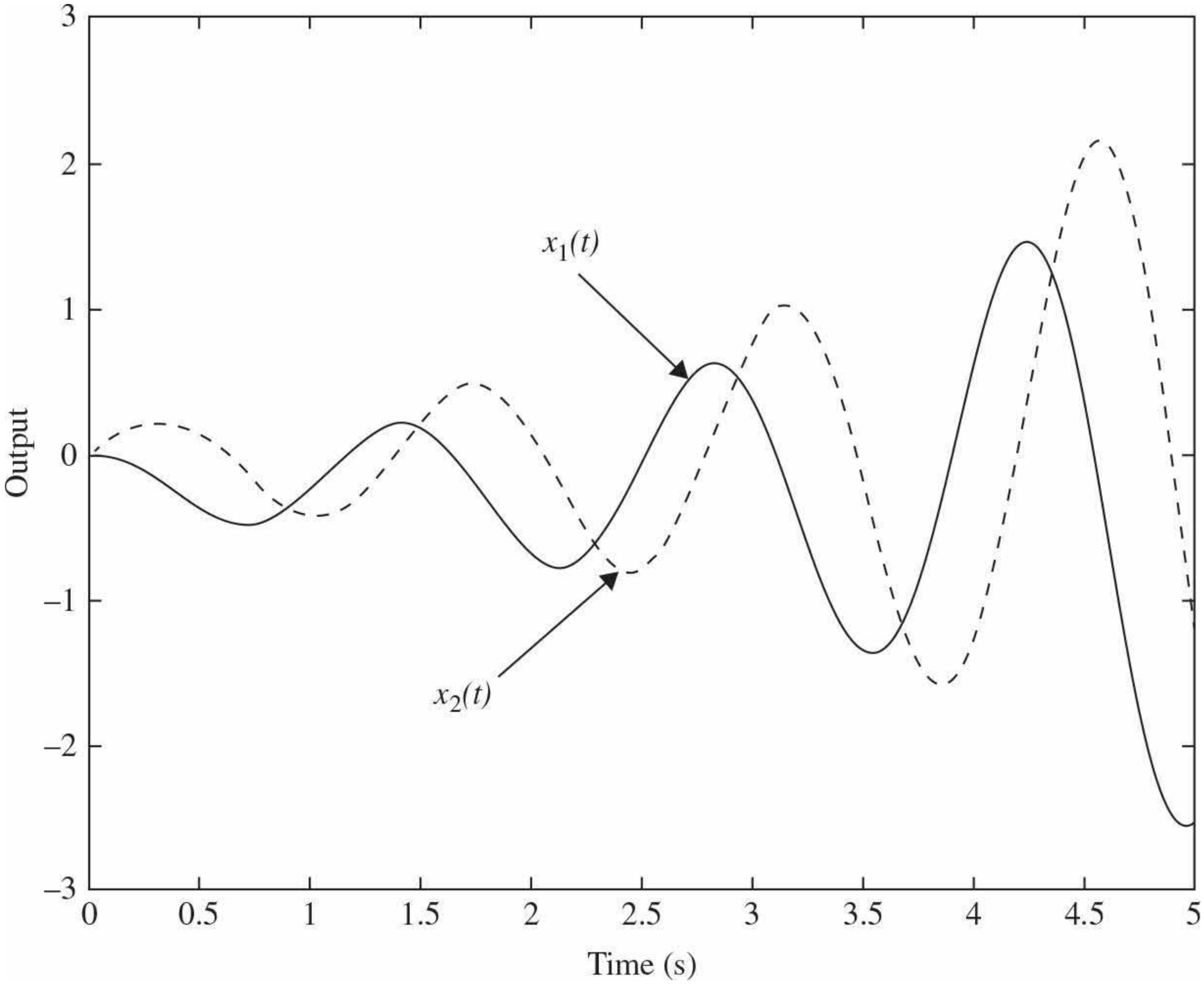


Figure 5-6 Time response of Eq. (5-34) for a unit step input. Note: $y(t) = x(t)$.

Using the following state feedback controller, we can address stability issues and other control goals (see Chap. 8 for more discussion on the subject):

$$u(t) = -k_1 x_1(t) - k_2 x_2(t) + r(t) \tag{5-44}$$

where k_1 and k_2 are real constants, and $r(t)$ is a unit step input.

The next step is to determine the system stability for parameters k_1 and k_2. Upon substitution of Eq. (5-44) into Eq. (5-36), we have the closed-loop system equation:

$$\dot{\mathbf{x}}(t) = (\mathbf{A} - \mathbf{BK})\mathbf{x}(t) + \mathbf{B}r(t) = \mathbf{A}^*\mathbf{x}(t) + \mathbf{B}r(t)$$
$$\mathbf{y}(t) = \mathbf{C}\,\mathbf{x} \tag{5-45}$$

where

$$\mathbf{K} = [k_1 \qquad k_2] \tag{5-46}$$

and

$$\mathbf{A}^* = \mathbf{A} - \mathbf{BK} = \begin{bmatrix} 1 & -4 \\ 5 - k_1 & -k_2 \end{bmatrix} \tag{5-47}$$

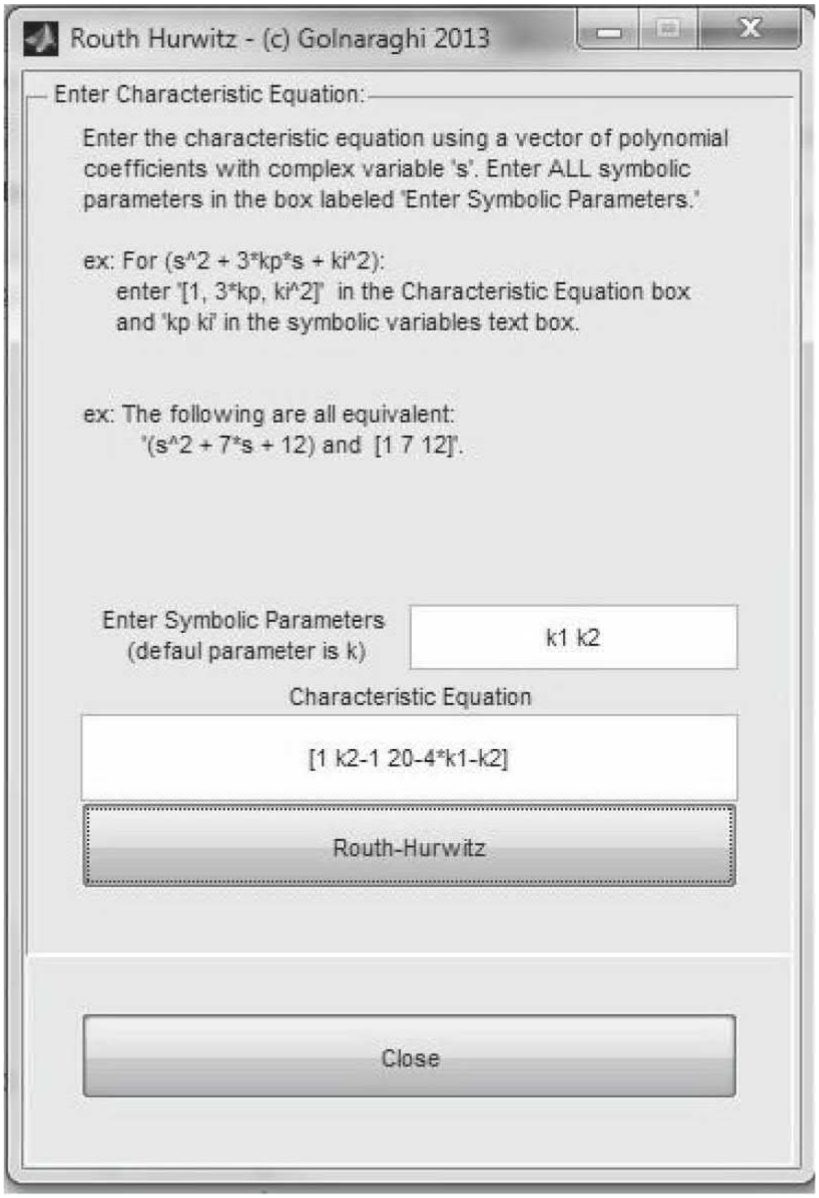
Taking the Laplace transform of Eq. (5-45), assuming zero initial conditions, and solving for **Y**(*s*), we have

$$\mathbf{Y}(s) = \mathbf{X}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A} + \mathbf{BK})^{-1}\,\mathbf{B}R(s) = \frac{\begin{bmatrix} -4 \\ s-1 \end{bmatrix}}{s^2 + (k_2 - 1)s + 20 - 4k_1 - k_2} \tag{5-48}$$

Following the same procedure as in Eq. (5-39), the characteristic equation of the closed-loop system is now:

$$|s\mathbf{I} - \mathbf{A} + \mathbf{BK}| = \begin{vmatrix} s-1 & 4 \\ -5 + k_1 & s + k_2 \end{vmatrix} = s^2 + (k_2 - 1)s + 20 - 4k_1 - k_2 = 0 \tag{5-49}$$

For stability requirements we us **tfrouth** as shown in Fig. 5-7. As a result, to ensure stability, all elements in the first column of the Routh array must be positive. This requires that both of the following conditions are met simultaneously.



Enter Characteristic Equation:

Enter the characteristic equation using a vector of polynomial coefficients with complex variable 's'. Enter ALL symbolic parameters in the box labeled 'Enter Symbolic Parameters.'

ex: For (s^2 + 3*kp*s + ki^2):
enter '[1, 3*kp, ki^2]' in the Characteristic Equation box
and 'kp ki' in the symbolic variables text box.

ex: The following are all equivalent:
'(s^2 + 7*s + 12) and [1 7 12]'.

Enter Symbolic Parameters (defaul parameter is k): k1 k2

Characteristic Equation: [1 k2-1 20-4*k1-k2]

Figure 5-7 Entering characteristic polynomial for Example 5-4-5 using the tfrouth module.

$$k_2 > 1 \tag{5-50}$$

$$k_2 < 20 - 4k_1 \tag{5-51}$$

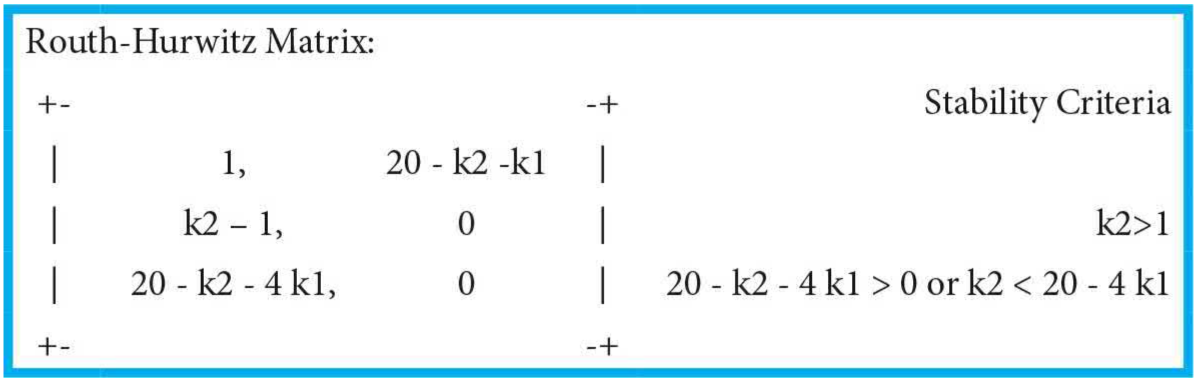
Hence, for stability we must ensure

$$1 < k_2 < 20 - 4k_1 \tag{5-52}$$

Using k_2 = 2 and k_1 = 2, both conditions are met for the system to be stable. In this case, we have

$$\mathbf{Y}(s)=\mathbf{C}(s\mathrm{I}-\mathbf{A}+\mathbf{BK})^{-1}\mathbf{B}R(s)=\frac{\begin{bmatrix}-4\\ s-1\end{bmatrix}}{s^2+s+10} \tag{5-53}$$

Using Toolbox 5-4-1, upon revising the denominator term to [1 1 10], we obtain the time response of the system for a unit step *r*(*t*), as shown in Fig. 5-8.



```
Routh-Hurwitz Matrix:

+-                                      -+                   Stability Criteria

|        1,            20 - k2 -k1       |

|      k2 – 1,              0            |                                 k2>1

|   20 - k2 - 4 k1,         0            |  20 - k2 - 4 k1 > 0 or k2 < 20 - 4 k1

+-                                      -+
```

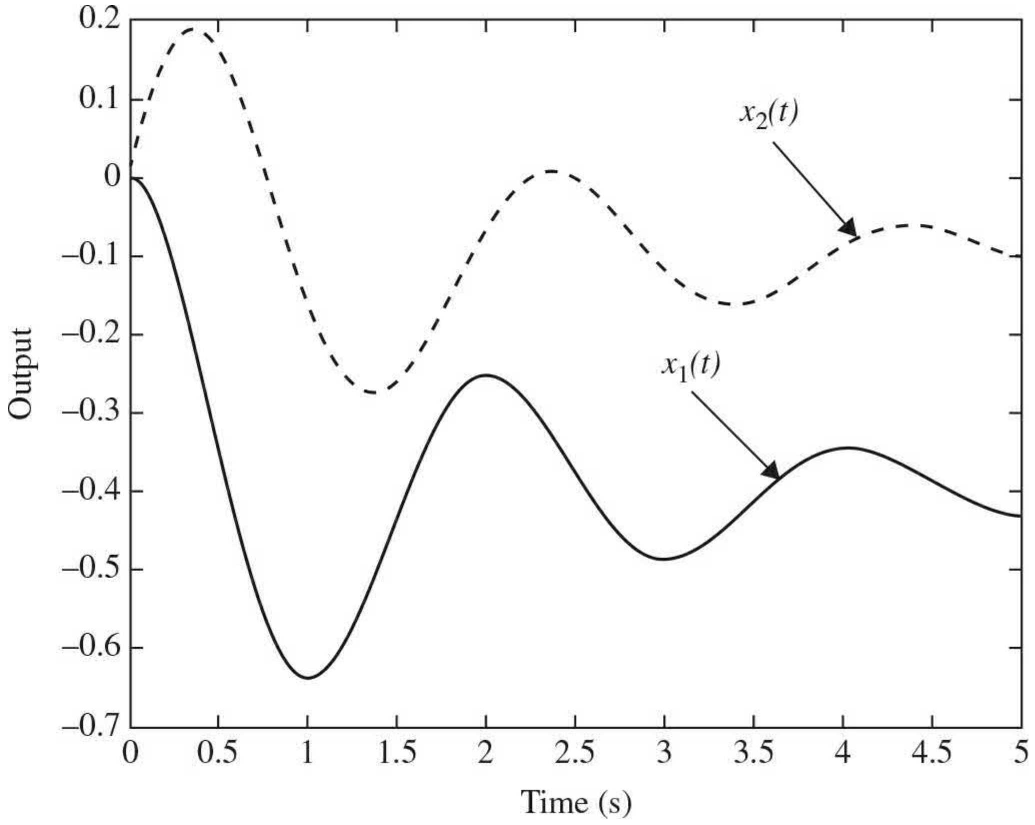
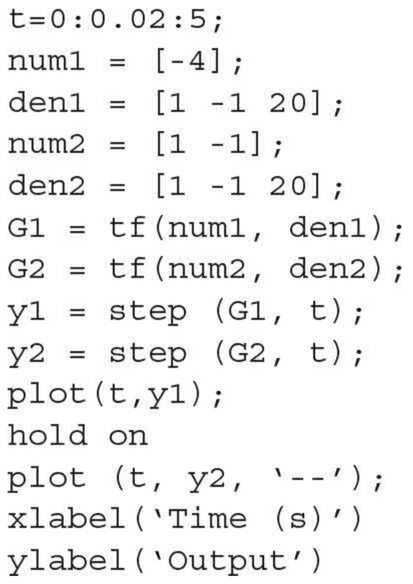


Figure 5-8 Time response of Eq. (5-53) for a unit step input. Note: $y(t) = x(t)$.

Toolbox 5-4-1

Time domain step response of Eq. (5-39) for a unit step input is shown in Fig. 5-6.



```
t=0:0.02:5;
num1 = [-4];
den1 = [1 -1 20];
num2 = [1 -1];
den2 = [1 -1 20];
G1 = tf(num1, den1);
G2 = tf(num2, den2);
y1 = step (G1, t);
y2 = step (G2, t);
plot(t,y1);
hold on
plot (t, y2, ‘--’);
xlabel(‘Time (s)’)
ylabel(‘Output’)
```

5-5 SUMMARY

In this chapter, we utilized the Routh-Hurwitz stability criterion for linear time-invariant continuous systems. We showed that the condition of these types of stability is related directly to the roots of the characteristic equation. For a continuous-data system to be stable, the roots of the characteristic equation must all be located in the left half of the *s*-plane. While the Routh-Hurwitz stability applies to the characteristic equations of the systems, we highlighted that this approach can also apply to examine the stability of the system in state space representation—because the eigenvalues of the **A** matrix are the same as the roots of the characteristic equation.

We solved various examples using tfrouth, the stability MATLAB tool developed for this book.

REFERENCES

1. F. Golnaraghi and B. C. Kuo, *Automatic Control Systems*, 9th Ed., John Wiley & Sons, New York, 2010.

2. F. R. Gantmacher, *Matrix Theory*, Vol. II, Chelsea Publishing Company, New York, 1964.

3. K. J. Khatwani, “On Routh-Hurwitz Criterion,” *IEEE Trans. Automatic Control*, Vol. AC-26, p. 583, April 1981.

4. S. K. Pillai, “The ε Method of the Routh-Hurwitz Criterion,” *IEEE Trans. Automatic Control*, Vol. AC-26, 584, April 1981.

5. B. C. Kuo and F. Golnaraghi, *Automatic Control Systems*, 8th Ed., John Wiley & Sons, New York, 2003.

PROBLEMS

5-1. Without using the Routh-Hurwitz criterion, determine if the following systems are asymptotically stable, marginally stable, or unstable. In each case, the closed-loop system transfer function is given.

(a) $M(s)=\dfrac{10(s+2)}{s^3+3s^2+5s}$

(b) $M(s)=\dfrac{s-1}{(s+5)(s^2+2)}$

(c) $M(s)=\dfrac{K}{s^3+5s+5}$

(d) $M(s)=\dfrac{100(s-1)}{(s+5)(s^2+2s+2)}$

(e) $M(s)=\dfrac{100}{s^3-2s^2+3s+10}$

(f) $M(s)=\dfrac{10(s+12.5)}{s^4+3s^3+50s^2+s+10^6}$

5-2. Use the ROOTS command in MATLAB to solve Prob. 5-33.

5-3. Using the Routh-Hurwitz criterion, determine the stability of the closed-loop system that has the following characteristic equations. Determine the number of roots of each equation that are in the right-half *s*-plane and on the $M(s)=\dfrac{10(s+2)}{s^3+3s^2+5s}$.

(a) $s^3 + 25s^2 + 10s + 450 = 0$

(b) $s^3 + 25s^2 + 10s + 50 = 0$

(c) $s^3 + 25s^2 + 250s + 10 = 0$

(d) $2s^4 + 10s^3 + 5.5s^2 + 10 = 0$

(e) $s^6 + 2s^5 + 8s^4 + 20s^2 + 16s + 16 = 0$

(f) $s^4 + 2s^3 + 20s + 5 = 0$

(g) $s^8 + 2s^7 + 8s^6 + 12s^5 + 20s^4 + 16s^3 + 16s^2 = 0$

5-4. Use MATLAB to solve Prob. 5-35.

5-5. Use MATLAB Toolbox 5-3-1 to find the roots of the following characteristic equations of linear continuous-data systems and determine the stability condition of the systems.

(a) $s^3 + 10s^2 + 10s + 130 = 0$

(b) $s^4 + 12s^3 + s^2 + 2s + 10 = 0$

(c) $s^4 + 12s^3 + 10s^2 + 10s + 10 = 0$

(d) $s^4 + 12s^3 + s^2 + 10s + 1 = 0$

(e) $s^6 + 6s^5 + 125s^4 + 100s^3 + 100s^2 + 20s + 10 = 0$

(f) $s^5 + 125s^4 + 100s^3 + 100s^2 + 20s + 10 = 0$

5-6. For each of the characteristic equations of feedback control systems given, use MATLAB to determine the range of *K* so that the system is asymptotically stable. Determine the value of *K* so that the system is marginally stable and determine the frequency of sustained oscillation, if applicable.

(a) $s^4 + 25s^3 + 15s^2 + 20s + K = 0$

(b) $s^4 + Ks^3 + 2s^2 + (K + 1)s + 10 = 0$

(c) $s^3 + (K + 2)^2 + 2Ks + 10 = 0$

(d) $s^3 + 20s^2 + 10K = 0$

(e) $s^4 + Ks^3 + 5s^2 + 10s + 10K = 0$

(f) $s^4 + 12.5s^3 + s^2 + 5s + K = 0$

5-7. The loop transfer function of a single-loop feedback control system is given as

$$G(s)H(s)=\frac{K(s+5)}{s(s+2)(1+Ts)}$$

The parameters K and T may be represented in a plane with K as the horizontal axis and T as the vertical axis. Determine the regions in the T-versus-K parameter plane where the closed-loop system is asymptotically stable and where it is unstable. Indicate the boundary on which the system is marginally stable.

5-8. Given the forward-path transfer function of unity-feedback control systems, apply the Routh-Hurwitz criterion to determine the stability of the closed-loop system as a function of K. Determine the value of K that will cause sustained constant-amplitude oscillations in the system. Determine the frequency of oscillation.

(a) $$G(s)=\frac{K(s+4)(s+20)}{s^3(s+100)(s+500)}$$

(b) $$G(s)=\frac{K(s+10)(s+20)}{s^2(s+2)}$$

(c) $$G(s)=\frac{K}{s(s+10)(s+20)}$$

(d) $$G(s)=\frac{K(s+1)}{s^3+2s^2+3s+1}$$

5-9. Use MATLAB to solve Prob. 5-40.

5-10. A controlled process is modeled by the following state equations.

$$\frac{dx_1(t)}{dt}=x_1(t)-2x_2(t) \qquad \frac{dx_2(t)}{dt}=10x_1(t)+u(t)$$

The control $u(t)$ is obtained from state feedback such that

$$u(t) = -k_1 x_1(t) - k_2 x_2(t)$$

where k_1 and k_2 are real constants. Determine the region in the k_1-versus-k_2 parameter plane in which the closed-loop system is asymptotically stable.

5-11. A linear time-invariant system is described by the following state equations.

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{B}u(\mathbf{t})$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -4 & -3 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

The closed-loop system is implemented by state feedback, so that $u(t) = -\mathbf{Kx}(t)$, where $\mathbf{K} = [k_1\ k_2\ k_3]$ and k_1, k_2, and k_3 are real constants. Determine the constraints on the elements of **K** so that the closed-loop system is asymptotically stable.

5-12. Given the system in state equation form,

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{B}u(t)$$

(a) $$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & -2 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

(b) $$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

Can the system be stabilized by state feedback $u(t=-\mathbf{K}\mathrm{x}(t)$, where $\mathbf{K}= [k_1\ k_2\ k_3]$?

5-13. Consider the open-loop system in Fig. 5P-13a.

where $\frac{d^2y}{dt^2}-\frac{g}{l}y=z$ and $f(t)=\tau\frac{dz}{dt}+z$

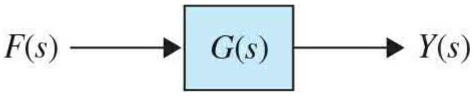


Figure 5P-13a

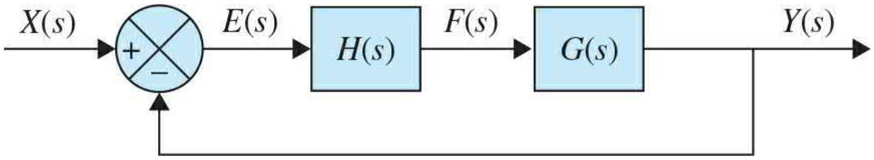


Figure 5P-13b

Our goal is to stabilize this system so the closed-loop feedback control will be defined as shown in the block diagram in Fig. 5P-13*b*.

Assuming $f(t)=k_pe+k_d\frac{de}{dt}$.

(a) Find the open-loop transfer function.

(b) Find the closed-loop transfer function.

(c) Find the range of *kp* and *kd* in which the system is stable.

(d) Suppose $\frac{g}{l}=10$ and τ = 0.1. If *y*(0) = 10 and $\frac{dy}{dt}=0,$ then plot the step response of the system with three different values for k_p and k_d. Then show that some values are better than others; however, all values must satisfy the Routh-Hurwitz criterion.

5-14. The block diagram of a motor-control system with tachometer feedback is shown in Fig. 5P-14. Find the range of the tachometer constant *Kt* so that the system is asymptotically stable.

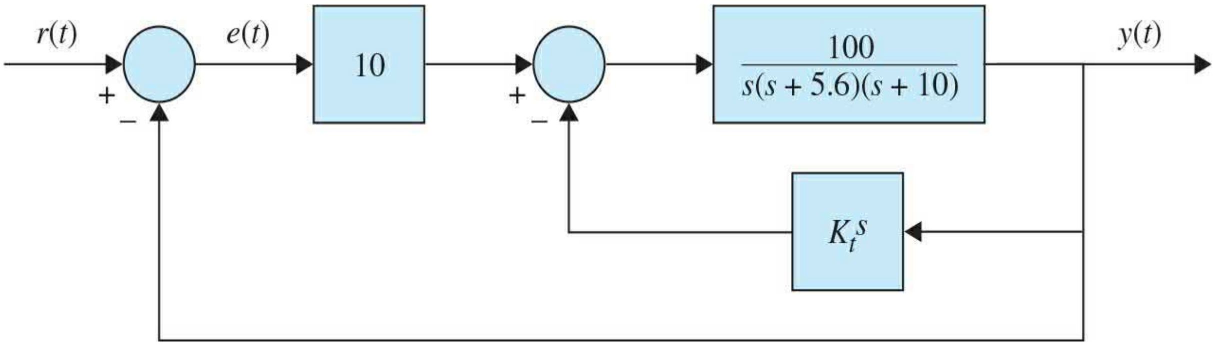


Figure 5P-14

5-15. The block diagram of a control system is shown in Fig. 5P-15. Find the region in the K-versus-α plane for the system to be asymptotically stable. (Use K as the vertical and $g\alpha$ as the horizontal axis.)

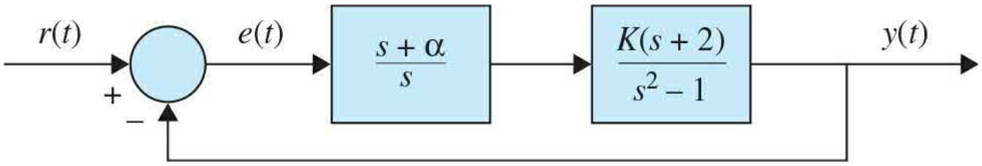


Figure 5P-15

5-16. The conventional Routh-Hurwitz criterion gives information only on the location of the zeros of a polynomial $F(s)$ with respect to the left half and right half of the s-plane. Devise a linear transformation $s = f(p, \alpha)$, where p is a complex variable, so that the Routh-Hurwitz criterion can be applied to determine whether $F(s)$ has zeros to the right of the line $s=-\alpha$, where α is a positive real number. Apply the transformation to the following characteristic equations to determine how many roots are to the right of the line $s = -1$ in the s-plane.

(a) $F(s)=s^2 + 5s +3 =0$

(b) $s^3 + 3s^2 + 3s + 1 =0$

(c) $F(s)= s^3 + 4s^2 + 3s + 10 =0$

(d) $s + 4s^2 + 4s + 4 =0$

5-17. The payload of a space-shuttle-pointing control system is modeled as a pure mass M. The payload is suspended by magnetic bearings so that no friction is encountered in the control. The attitude of the payload in the y

direction is controlled by magnetic actuators located at the base. The total force produced by the magnetic actuators is *f* (*t*). The controls of the other degrees of motion are independent and are not considered here. Because there are experiments located on the payload, electric power must be brought to the payload through cables. The linear spring with spring constant *Ks* is used to model the cable attachment. The dynamic system model for the control of the *y*-axis motion is shown in Fig. 5P-17. The force equation of motion in the *y*-direction is

$$f(t)=K_s y(t)+M\frac{d^2y(t)}{dt^2}$$

where K_s = 0.5 N m/m and *M* = 500kg. The magnetic actuators are controlled through state feedback, so that

$$f(t)=-K_P y(t)-K_D\frac{dy(t)}{dt}$$

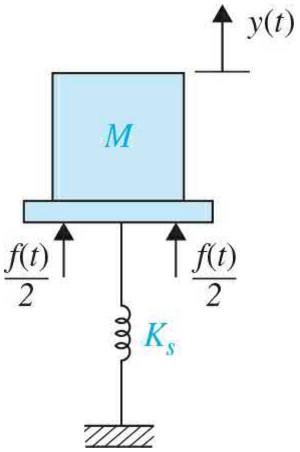


Figure 5P-17

(a) Draw a functional block diagram for the system.

(b) Find the characteristic equation of the closed-loop system.

(c) Find the region in the *KD*-versus-*KP* plane in which the system is asymptotically stable.

5-18. An inventory-control system is modeled by the following differential equations:

$$\frac{dx_1(t)}{dt} = -x_2(t) + u(t)$$
$$\frac{dx_2(t)}{dt} = -Ku(t)$$

where $x_1(t)$ is the level of inventory; $x_2(t)$, the rate of sales of product; $u(t)$, the production rate; and K, a real constant. Let the output of the system by $y(t) = x_1$ and $r(t)$ be the reference set point for the desired inventory level. Let $u(t) = r(t - y(t))$. Determine the constraint on K so that the closed-loop system is asymptotically stable.

5-19. Use MATLAB to solve Prob. 5-50.

5-20. Use MATLAB to

(a) Generate symbolically the time function of $f(t)$

$$f(t) = 5 + 2e^{-2t}\sin\left(2t + \frac{\pi}{4}\right) - 4e^{-2t}\cos\left(2t - \frac{\pi}{2}\right) + 3e^{-4t}$$

(b) Generate symbolically $G(s) = \dfrac{(s+1)}{s(s+2)(s^2+2s+2)}$

(c) Find the Laplace transform of $f(t)$ and name it $F(s)$.

(d) Find the inverse Laplace transform of $G(s)$ and name it $g(t)$.

(e) If $G(s)$ is the forward-path transfer function of unity-feedback control systems, find the transfer function of the closed-loop system and apply the Routh-Hurwitz criterion to determine its stability.

(f) If $F(s)$ is the forward-path transfer function of unity-feedback control systems, find the transfer function of the closed-loop system and apply the Routh-Hurwitz criterion to determine its stability.

[1] The formal and mathematical discussions of this topic are included in App. B.

[2] For Mac or Unix users refer to MATLAB help for information.

CHAPTER 6

Important Components of Feedback Control Systems

As mentioned in Chap. 1, the design process of a control system starts with development of a mathematical model of the system, which is normally composed of various subcomponents such as mechanical, electrical, chemical, sensors, actuators (motors), etc., otherwise known as **dynamic systems**. In Chap. 2, we looked at modeling of simple dynamic systems such as mechanical, electrical, fluid, and heat transfer systems, which may be considered as the subcomponents of most control systems encountered. Using the basic modeling principles such as Newton's second law of motion, Kirchoff's law, or the law of conservation of mass, the models of these dynamic systems are represented by differential equations. In Chap. 3, we utilized the transfer function and state space approaches to solve these differential equations to find the behavior of the dynamic systems. Later in Chap. 4, we learned how to graphically represent the system models using block, signal flow, and state diagrams and learned about the concept of feedback in control systems. We further learned that a typical feedback control system, as shown in Fig. 6-1, is composed of various components including

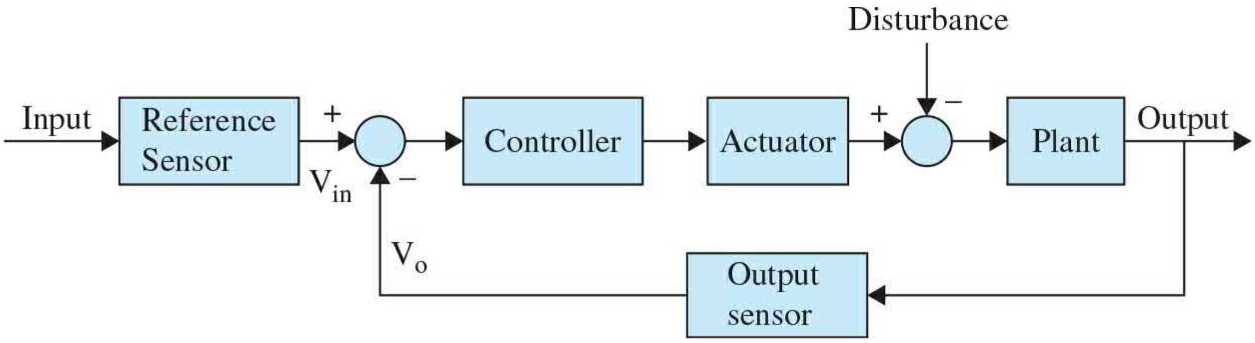


Figure 6-1 Block diagram representation of a general feedback control system.

Learning Outcomes

After successful completion of this chapter, you will be able to

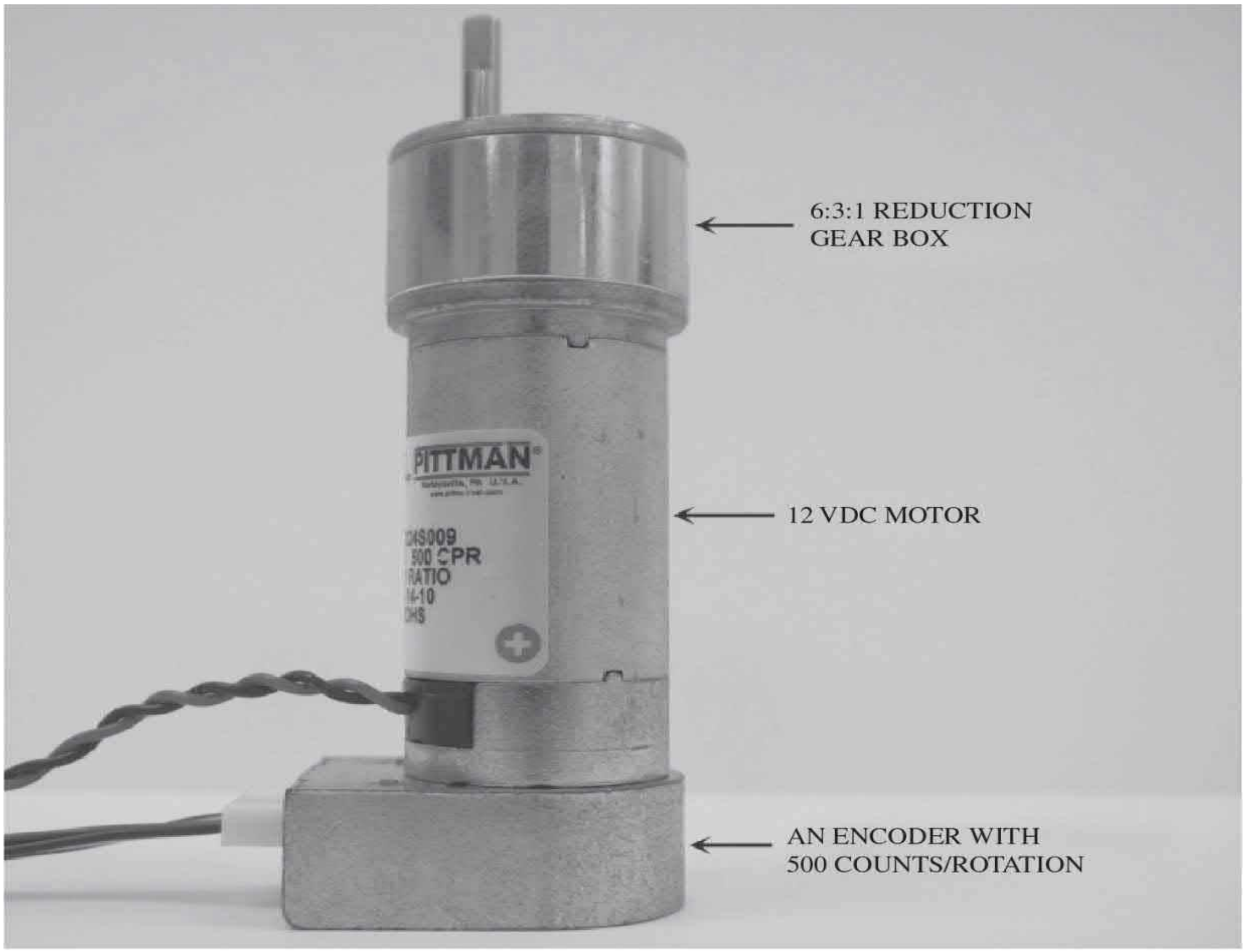
1. Understand the necessary components that make a feedback control system work, including sensors, actuators, and the controller.
2. Develop mathematical model for these components, including potenti-ometers, tachometers, encoders, op amps, and dc motors.
3. Establish the transfer functions for speed and position time response of dc motors.
4. Characterize a dc motor through series of experimental measurements.

- Reference sensor (or input sensor)
- Output sensor
- Actuator
- Controller
- Plant (the component whose variables are to be controlled—normally a dynamic system as described in Chap. 2)

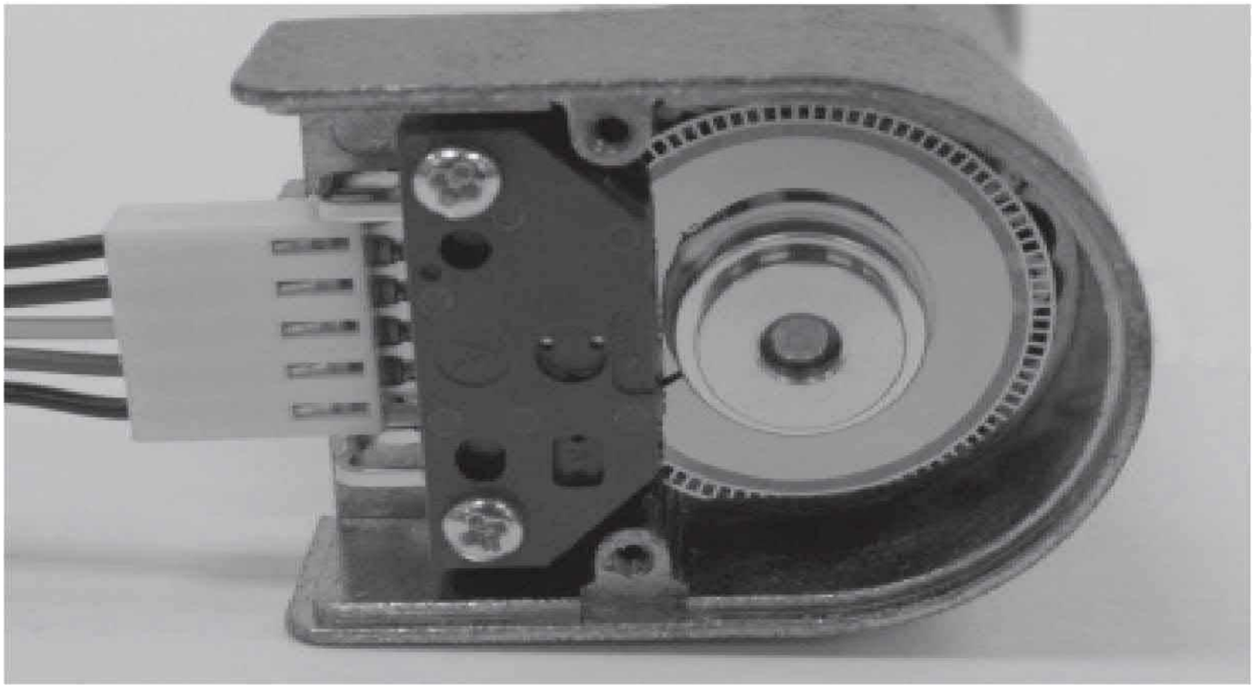
In a **feedback control system**, sensors are very important to sense various properties of the system, particularly, the output of the plant. A controller can then compare the output signal with the desired objective, or the input and adjust the performance of the overall system, using an actuator, to meet the desired goal.

Sensors, actuators, and the controller are the most important components of any control system.

In this chapter, we focus on the necessary components that make a feedback control system work. These include **sensors**, **actuators**, and the actual brain of the control system, that is, the **controller**. Of particular interest are components that possess **linear models—**or at least as close to linear as possible. In this textbook we will, for the sake of simplicity and to meet our linearity objective, use **dc motors** as actuators. We also look at sensors that may be used to quantify the motion of dc motors, namely, **encoders**, **tachometers**, and **potentiometers**. As an example, Fig. 6-2*a* shows a typical dc motor-gearbox system equipped with an encoder that senses the motor shaft output. A close up of the attached encoder is shown in Fig. 6-2*b*.


6:3:1 REDUCTION GEAR BOX
12 VDC MOTOR
AN ENCODER WITH 500 COUNTS/ROTATION
PITTMAN
500 CPR
RATIO

(a)



(b)

Figure 6-2 (*a*) A typical dc motor-gearbox system with an onboard encoder to measure motor shaft rotation—picture taken by the author from a GM8224S009 12VDC 500 CPR Ametec Pittman model. (*b*) A view of the encoder in the bottom of the motor.

In this chapter, we will also learn about **op-amps** and their role as building blocks of any control system. Finally, through case studies, we bring all the material learned in Chaps. 1 to 5 together.

By the end of this chapter, you are able to understand how to model a complete control system and its individual components and further understand how these components are related and interact with one another.

Finally, at the end of this chapter, we also provide a **Control Lab**, which includes a practical means of characterizing a dc motor through series of experimental measurements.

6-1 MODELING OF ACTIVE ELECTRICAL ELEMENTS: OPERATIONAL AMPLIFIERS

Operational amplifiers, or simply **op-amps**, offer a convenient way to build, implement, or realize continuous-data or *s*-domain transfer functions. In control systems, op-amps are often used to implement the controllers or compensators that evolve from the control-system design process, so in this section we illustrate common op-amp configurations. An in-depth presentation of op-amps is beyond the scope of this text. For those interested, many texts are available that are devoted to all aspects of op-amp circuit design and applications [8, 9].

Our primary goal here is to show how to implement first-order transfer functions with op-amps while keeping in mind that higher-order transfer functions are also important. In fact, simple high-order transfer functions can be implemented by connecting first-order op-amp configurations together. Only a representative sample of the multitude of op-amp configurations will be discussed. Some of the practical issues associated with op-amps are demonstrated in Chaps. 7 and 11.

6-1-1 The Ideal Op-Amp

An op-amp circuit is usually analyzed by considering the op-amp to be ideal. The ideal op-amp circuit is shown in Fig. 6-3, and it has the following properties:

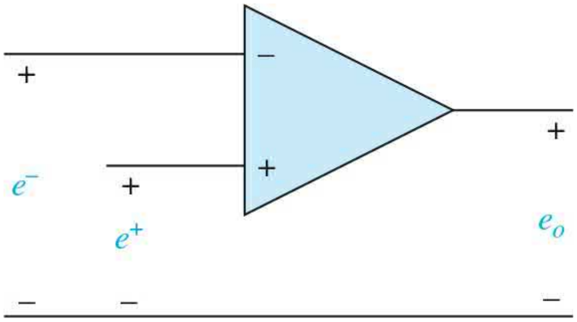


Figure 6-3 Schematic diagram of an op-amp.

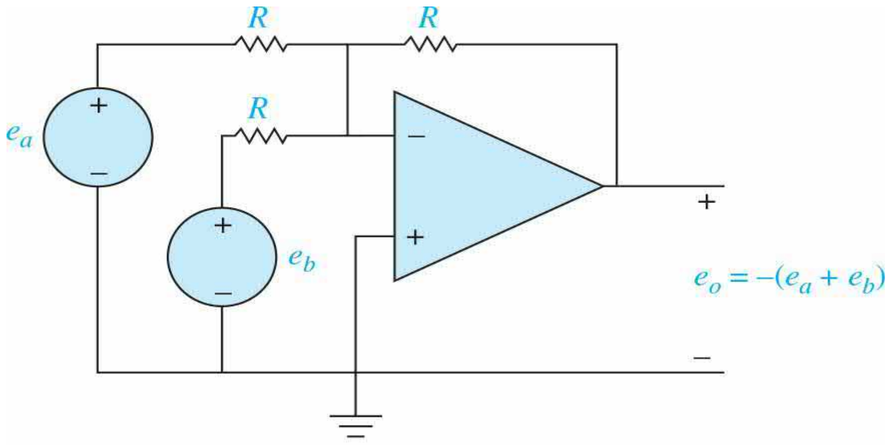
1. The voltage between the + and − terminals is zero, that is, $e^+ = e^-$. This property is commonly called the *virtual ground or virtual short*.
2. The currents into the + and − input terminals are zero. Thus, the input impedance is infinite.
3. The impedance seen looking into the output terminal is zero. Thus, the output is an ideal voltage source.
4. The input-output relationship is $e_o = A(e^+ - e^-)$, where the gain A approaches infinity.

The input-output relationship for many op-amp configurations can be determined by using these principles. An op-amp cannot be used as shown in Fig. 6-3. Rather, linear operation requires the addition of feedback of the output signal to the "–" input terminal.

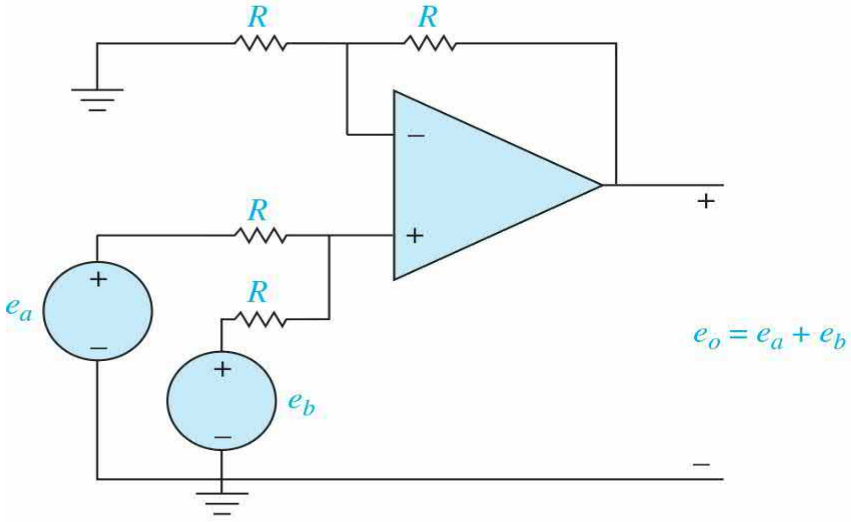
6-1-2 Sums and Differences

As illustrated in Chap. 4, one of the most fundamental elements in a block diagram or an SFG is the addition or subtraction of signals. When these signals are voltages, op-amps provide a simple way to add or subtract signals, as shown in Fig. 6-4, where all the resistors have the same value. Using superposition and the ideal properties given in the preceding section, the input-output relationship in Fig. 6-4*a* is $e_o = -(e_a - e_b)$. Thus, the output is the negative sum of the input voltages. When a positive sum is desired, the circuit shown in Fig. 6-4*b* can be used. Here, the output is given by $e_o = e_a +$

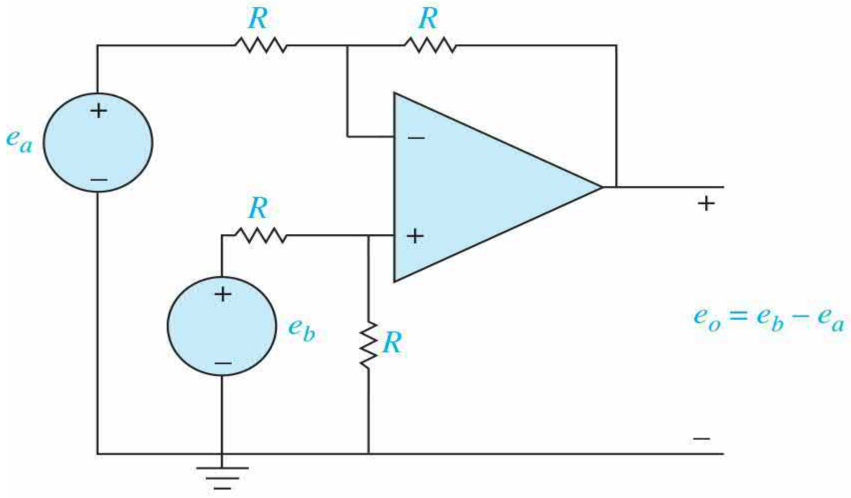
e_b. Modifying Fig. 6-4*b* slightly gives the differencing circuit shown in Fig. 6-34*c*, which has an input-output relationship of $e_o = i_b - e_a$.



(a)



(b)



(c)

Figure 6-4 Op-amps used to add and subtract signals.

6-1-3 First-Order Op-Amp Configurations

In addition to adding and subtracting signals, op-amps can be used to implement transfer functions of continuous-data systems. While many alternatives are available, we will explore only those that use the inverting op-amp configuration shown in Fig. 6-5. In the figure, $Z_1(s)$ and $Z_2(s)$ are impedances commonly composed of resistors and capacitors. Inductors are not commonly used because they tend to be bulkier and more expensive. Using ideal op-amp properties, the input-output relationship, or transfer function, of the circuit shown in Fig. 6-5 can be written in a number of ways, such as

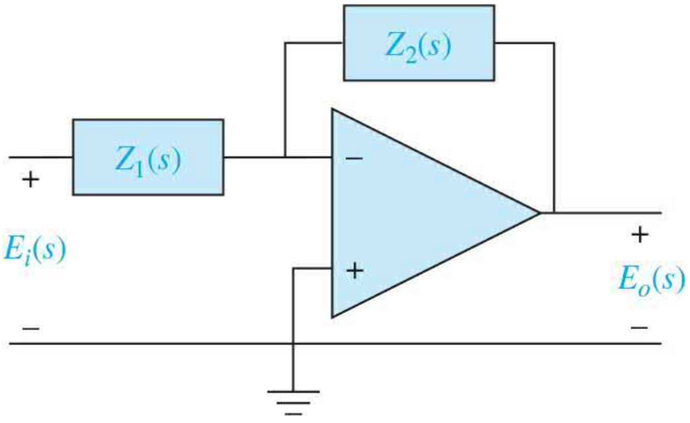


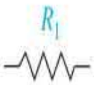
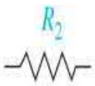
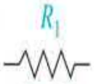
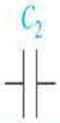
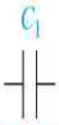
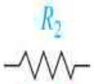
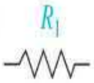
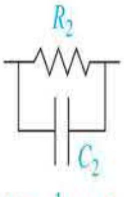
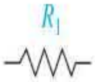
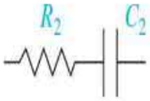
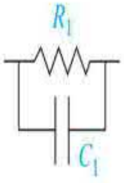
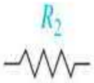
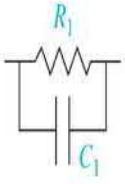
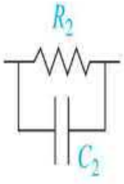
Figure 6-5 Inverting op-amp configuration.

$$
\begin{aligned}
G(s) &= \frac{E_o(s)}{E_i(s)} = -\frac{Z_2(s)}{Z_1(s)} = \frac{-1}{Z_1(s)Y_2(s)} \\
&= -Z_2(s)Y_1(s) = -\frac{Y_1(s)}{Y_2(s)}
\end{aligned}
\qquad (6\text{-}1)
$$

where $Y_1(s) = 1/Z_1(s)$ and $Y_2(s) = 1/Z_2(s)$ are the admittances associated with the circuit impedances. The different transfer function forms given in Eq. (6-1) apply conveniently to the different compositions of the circuit impedances.

Using the inverting op-amp configuration shown in Fig. 6-5 and using resistors and capacitors as elements to compose $Z_1(s)$ and $Z_2(s)$, it is possible to implement poles and zeros along the negative real axis as well as at the origin in the *s*-plane, as shown in Table 6-1. Because the inverting op-amp configuration has been used, all the transfer functions have negative gains. The negative gain is usually not an issue because it is simple to add a gain of −1 to the input and output signal to make the net gain positive.

TABLE 6-1 Inverting Op-Amp Transfer Functions

Input Element	Feedback Element	Transfer Function	Comments
(a)  R_1 $Z_1 = R_1$	 R_2 $Z_2 = R_2$	$-\dfrac{R_2}{R_1}$	Inverting gain; e.g., if $R_1 = R_2$, $e_o = -e_1$
(b)  R_1 $Z_1 = R_1$	 C_2 $Y_2 = sC_2$	$\left(\dfrac{-1}{R_1C_2}\right)\dfrac{1}{s}$	Pole at the origin; i.e., an integration
(c)  C_1 $Y_1 = sC_1$	 R_2 $Z_2 = R_2$	$(-R_2C_1)s$	Zero at the origin; i.e., a differentiator
(d)  R_1 $Z_1 = R_1$	 R_2 C_2 $Y_2 = \dfrac{1}{R_2} + sC_2$	$\dfrac{\dfrac{1}{R_1C_2}}{s + \dfrac{1}{R_2C_2}}$	Pole at $\dfrac{-1}{R_2C_2}$ with a dc gain of $-R_2/R_1$
(e)  R_1 $Z_1 = R_1$	 R_2 C_2 $Z_2 = R_2 + \dfrac{1}{sC_2}$	$\dfrac{-R_2}{R_1}\left(\dfrac{s + 1/R_2C_2}{s}\right)$	Pole at the origin and a zero at $-1/R_2C_2$; i.e., a PI controller
(f)  R_1 C_1 $Y_1 = \dfrac{1}{R_1} + sC_1$	 R_2 $Z_2 = R_2$	$-R_2C_1\left(s + \dfrac{1}{R_1C_1}\right)$	Zero at $s = \dfrac{-1}{R_1C_1}$; i.e., a PD controller
(g)  R_1 C_1 $Y_1 = \dfrac{1}{R_1} + sC_1$	 R_2 C_2 $Y_2 = \dfrac{1}{R_2} + sC_2$	$\dfrac{\dfrac{-C_1}{C_2}\left(s + \dfrac{1}{R_1C_1}\right)}{s + \dfrac{1}{R_2C_2}}$	Poles at $s = \dfrac{-1}{R_2C_2}$ and a zero at $s = \dfrac{-1}{R_1C_1}$; i.e., a lead or lag controller

EXAMPLE 6-1-1 As an example of op-amp realization of transfer functions, consider the transfer function

$$G(s) = K_p + \frac{K_I}{s} + K_D s \tag{6-2}$$

where K_P, K_D, and K_I are real constants. In Chaps. 7 and 11, this transfer function will be called the **PID controller**, since the first term is a **proportional** gain, the second is an **integral** term, and the third is a **derivative** term. Using Table 6-1, the proportional gain can be implemented using line (a), the integral term can be implemented using line (b), and the derivative term can be implemented using line (c). By superposition, the output of $G(s)$ is the sum of the responses due to each term in $G(s)$. This sum can be implemented by adding an additional input resistance to the circuit shown in Fig. 6-4*a*. By making the sum negative, the negative gains of the proportional, integral, and derivative term implementations are canceled, giving the desired result shown in Fig. 6-5. The transfer functions of the components of the op-amp circuit in Fig. 6-5 are

$$\textbf{Proportional}: \quad \frac{E_P(s)}{E(s)} = -\frac{R_2}{R_1} \tag{6-3}$$

$$\textbf{Integral}: \quad \frac{E_I(s)}{E(s)} = -\frac{1}{R_i C_i s} \tag{6-4}$$

$$\textbf{Derivative}: \quad \frac{E_D(s)}{E(s)} = -R_d C_d s \tag{6-5}$$

The output voltage is

$$E_o(s) = -[E_P(s) + E_I(s) + E_D(s)] \tag{6-6}$$

Thus, the transfer function of the PID op-amp circuit is

$$G(s) = \frac{E_o(s)}{E(s)} = \frac{R_2}{R_1} + \frac{1}{R_i C_i s} + R_d C_d s \tag{6-7}$$

By equating Eqs. (6-2) and (6-7), the design is completed by choosing the values of the resistors and the capacitors of the op-amp circuit so that the desired values of K_P, K_I, and K_D are matched. The design of the controller should be guided by the availability of standard capacitors and resistors.

It is important to note that Fig. 6-6 is just one of many possible implementations of Eq. (6-2). For example, it is possible to implement the PID controller with just three op-amps. Also, it is common to add components to limit the high-frequency gain of the differentiator and to limit the integrator output magnitude, which is often referred to as *antiwindup* protection. One advantage of the implementation shown in Fig. 6-6 is that each of the three constants K_P, K_I, and K_D can be adjusted or tuned individually by varying resistor values in its op-amp circuits. Op-amps are also used in control systems for A/D and D/A converters, sampling devices, and realization of nonlinear elements for system compensation.

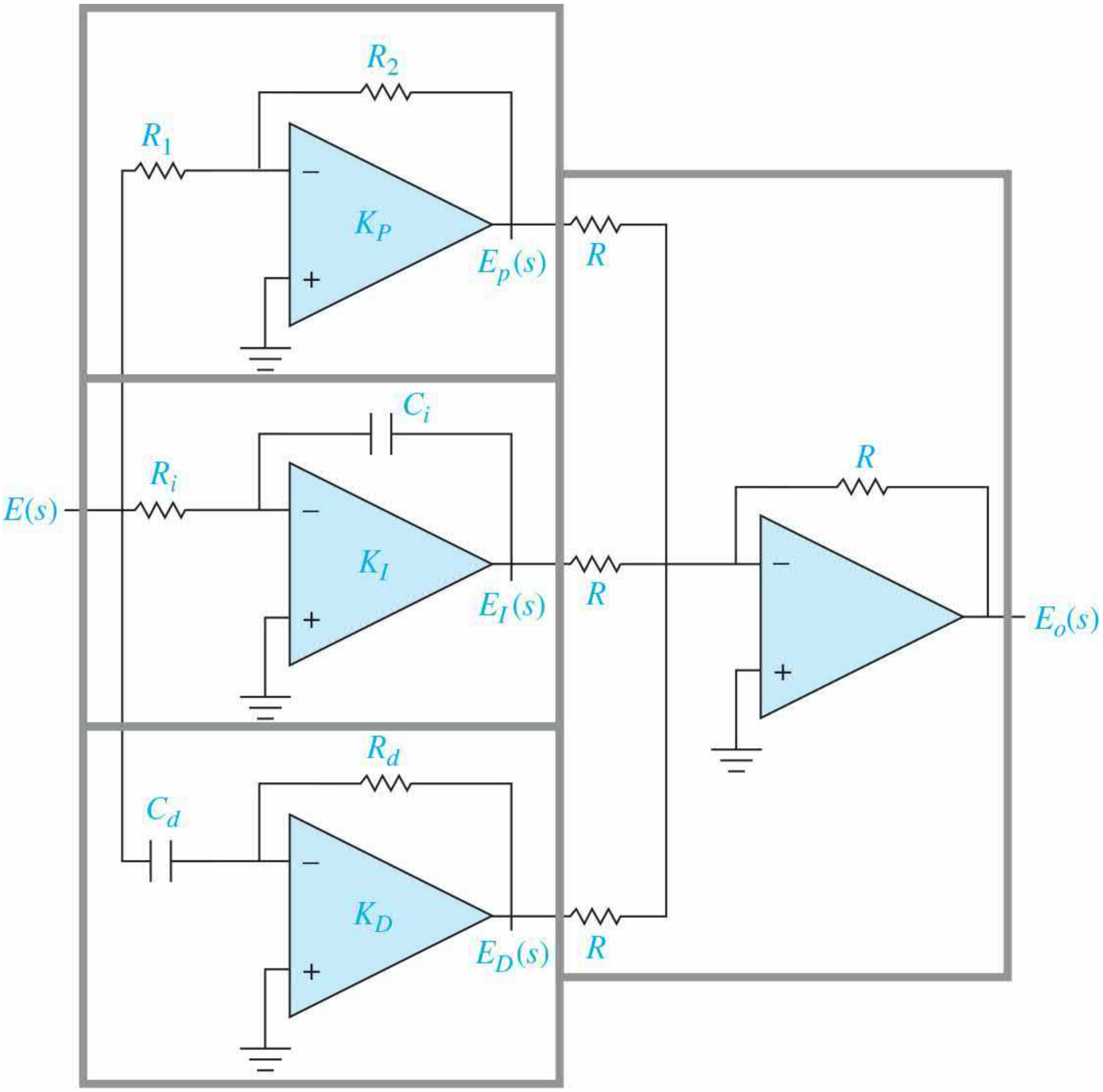


Figure 6-6 Implementation of a PID controller.

6-2 SENSORS AND ENCODERS IN CONTROL SYSTEMS

Sensors and encoders are important components used to monitor the performance and for feedback in control systems. In this section, the principle of operation and applications of some of the sensors and encoders that are commonly used in control systems are described.

6-2-1 Potentiometer

A potentiometer is an electromechanical transducer that converts mechanical energy into electrical energy. The input to the device is in the form of a mechanical displacement, either linear or rotational. When a voltage is applied across the fixed terminals of the potentiometer, the output voltage, which is measured across the variable terminal and ground, is proportional to the input displacement, either linearly or according to some nonlinear relation.

Rotary potentiometers are available commercially in single-revolution or multirevolution form, with limited or unlimited rotational motion. The potentiometers are commonly made with wirewound or conductive plastic resistance material. Figure 6-7 shows a cutaway view of a rotary potentiometer, and Fig. 6-8 shows a linear potentiometer that also contains a built-in operational amplifier. For precision control, the conductive plastic potentiometer is preferable because it has infinite resolution, long rotational life, good output smoothness, and low static noise.

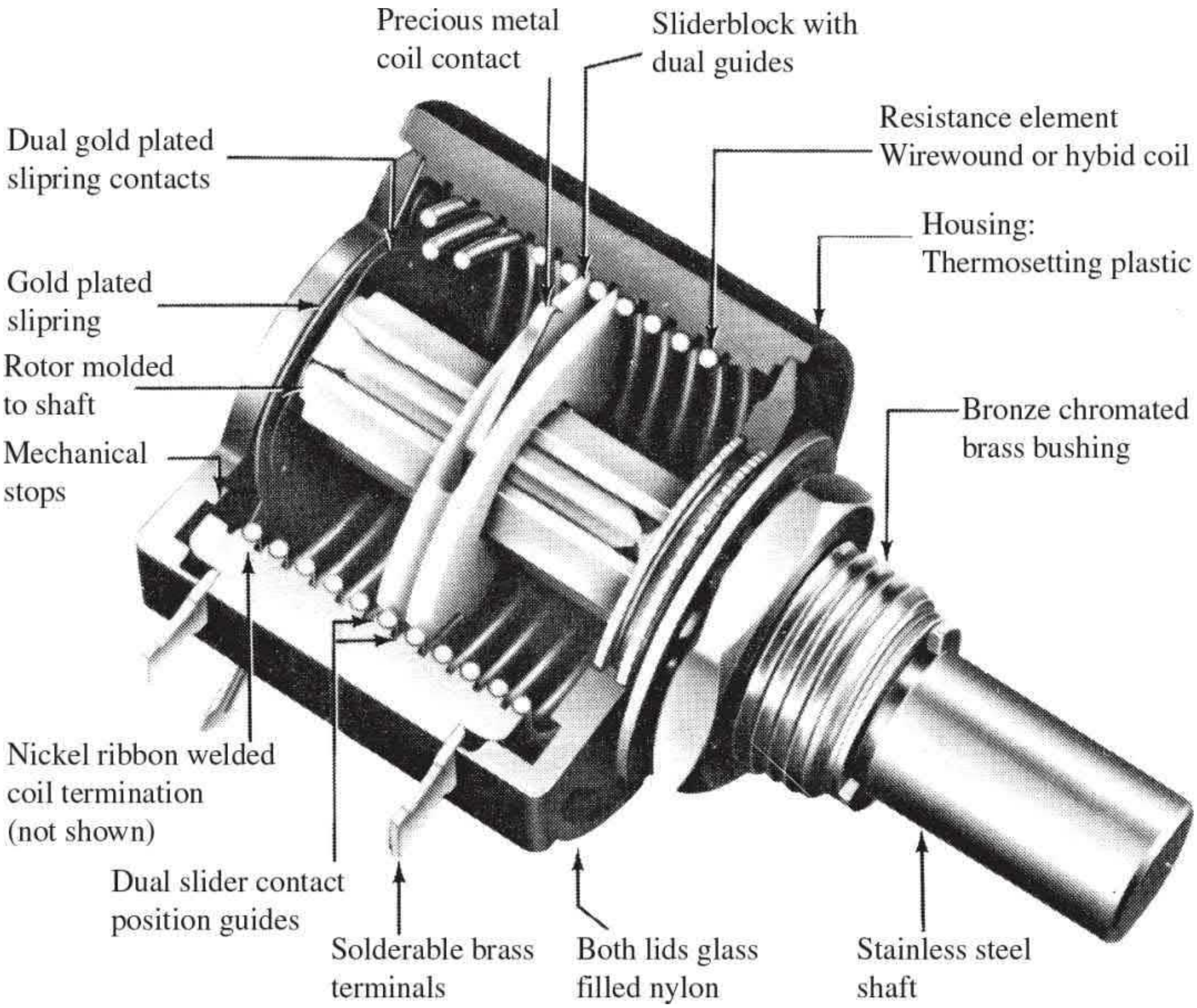

Precious metal coil contact
Sliderblock with dual guides
Dual gold plated slipring contacts
Resistance element Wirewound or hybid coil
Housing: Thermosetting plastic
Gold plated slipring
Rotor molded to shaft
Mechanical stops
Bronze chromated brass bushing
Nickel ribbon welded coil termination (not shown)
Dual slider contact position guides
Solderable brass terminals
Both lids glass filled nylon
Stainless steel shaft

Figure 6-7 Ten-turn rotary potentiometer. (Courtesy of Helipot Division of Beckman Instruments, Inc.)

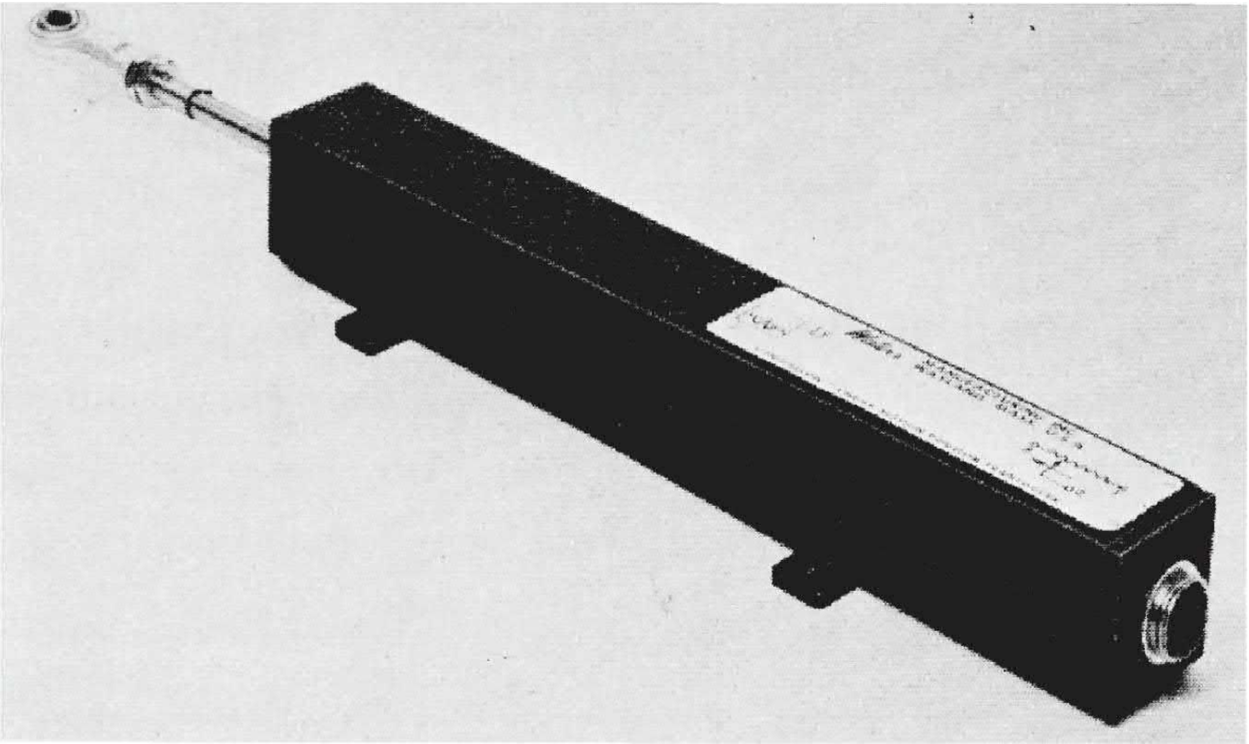


Figure 6-8 Linear motion potentiomenter with built-in operational amplifier. (Courtesy of Waters Manufacturing, Inc.)

Figure 6-9 shows the equivalent circuit representation of a potentiometer, linear or rotary. Because the voltage across the variable terminal and reference is proportional to the shaft displacement of the potentiometer, when a voltage is applied across the fixed terminals, the device can be used to indicate the absolute position of a system or the relative position of two mechanical outputs. Figure 6-10*a* shows the arrangement when the housing of the potentiometer is fixed at reference; the output voltage *e(t)* will be proportional to the shaft position $\theta_c(t)$ in the case of a rotary motion. Then

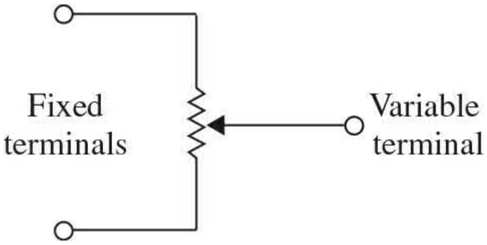


Figure 6-9 Electric circuit representation of a potentiometer.

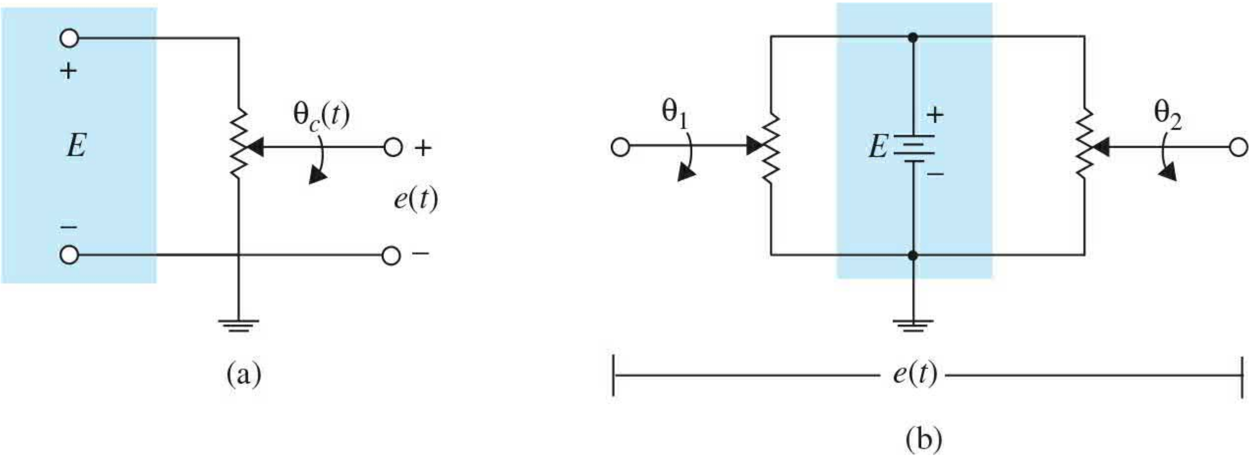


Figure 6-10 (*a*) Potentiometer used as a position indicator. (*b*) Two potentiometers used to sense the positions of two shafts.

$$e(t) = K_s \theta_c(t) \tag{6-8}$$

where K_s is the proportional constant. For an N-turn potentiometer, the total displacement of the variable arm is $2\pi N$ radians. The proportional constant K_s is given by

$$K_s = \frac{E}{2\pi N} \text{ V/rad} \tag{6-9}$$

where E is the magnitude of the reference voltage applied to the fixed terminals. A more flexible arrangement is obtained by using two potentiometers connected in parallel, as shown in Fig. 6-10*b*. This arrangement allows the comparison of two remotely located shaft positions. The output voltage is taken across the variable terminals of the two potentiometers and is given by

$$e(t) = K_s[\theta_1(t) - \theta_2(t)] \tag{6-10}$$

Figure 6-11 illustrates the block diagram representation of the setups in Fig. 6-10. In dc-motor control systems, potentiometers are often used for position feedback. Figure 6-12*a* shows the schematic diagram of a typical dc-motor, position-control system. The potentiometers are used in the feedback path to compare the actual load position with the desired reference position. If there is a discrepancy between the load position and the reference input, an error signal is generated by the potentiometers that will drive the motor in such a way that this error is minimized quickly. As shown in Fig. 6-12*a*, the error signal is amplified by a dc amplifier whose output drives the armature of a permanent-magnet dc motor. Typical waveforms of the signals in the system when the input $\theta_r(t)$ is a step function are shown in Fig. 6-13*b*. Note that the electric signals are all unmodulated. In control-systems terminology, a dc signal usually refers to an unmodulated signal. On the other hand, an ac signal refers to signals that are modulated by a modulation process. These definitions are different from those commonly used in electrical engineering, where dc simply refers to unidirectional signals and ac indicates alternating signals.

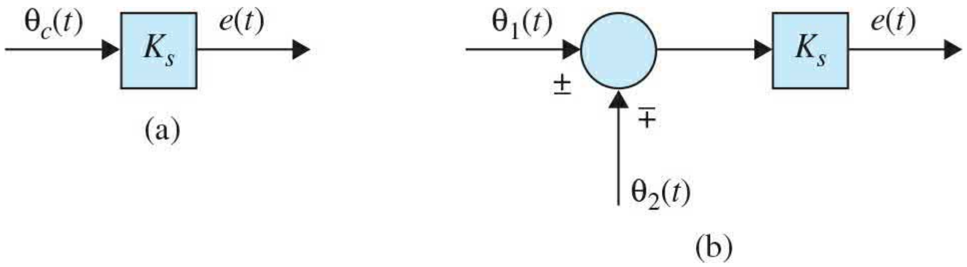
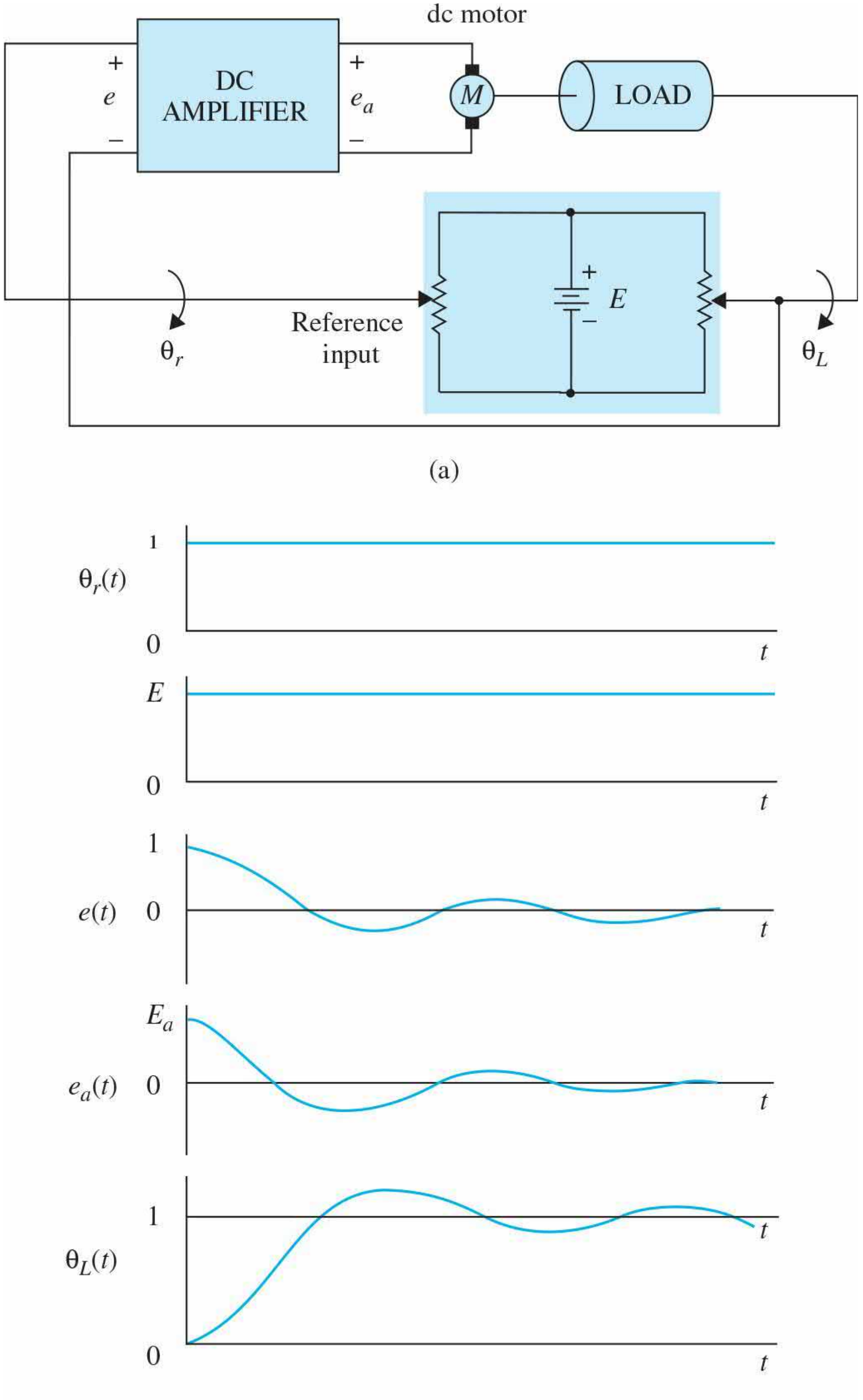


Figure 6-11 Block diagram representation of potentiometer arrangements in Fig. 6-10.



dc motor
DC
AMPLIFIER
+
e
−
+
e_a
−
M
LOAD
+
E
−
θ_r
Reference
input
θ_L
1
$\theta_r(t)$
0
t
E
0
t
1
e(t)
0
t
E_a
$e_a(t)$
0
t
1
$\theta_L(t)$
0
t
t

(a)

(b)

Figure 6-12 (*a*) A dc-motor, position-control system with potentiometers as error sensors. (*b*) Typical waveforms of signals in the control system of part (a).

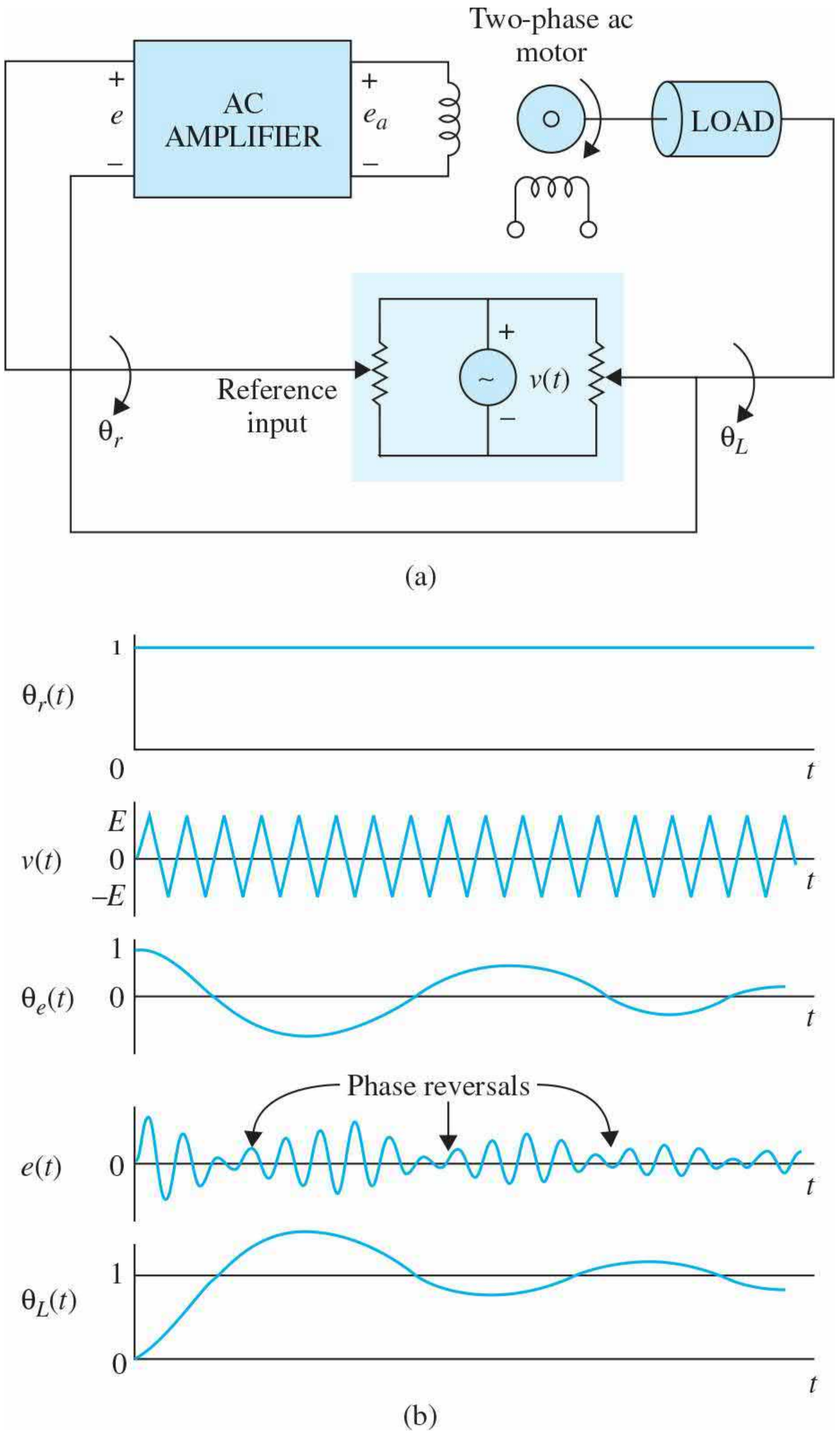


Figure 6-13 (*a*) An ac control system with potentiometers as error detectors. (*b*) Typical waveforms of signals in the control system of part (a).

Figure 6-13*a* illustrates a control system that serves essentially the same purpose as that of the system in Fig. 6-12*a*, except that ac signals prevail. In this case, the voltage applied to the error detector is sinusoidal. The frequency of this signal is usually much higher than that of the signal that is being transmitted through the system. Control systems with ac signals are usually found in aerospace systems that are more susceptible to noise.

Typical signals of an ac control system are shown in Fig. 6-13*b*. The signal *v(t)* is referred to as the carrier whose frequency is ω_c, or

$$v(t)=E\sin\omega_c t \tag{6-11}$$

Analytically, the output of the error signal is given by

$$e(t)=K_s\theta_e(t)v(t) \tag{6-12}$$

where $\theta_c(t)$ is the difference between the input displacement and the load displacement, or

$$\theta_e(t)=\theta_r(t)-\theta_L(t) \tag{6-13}$$

For the $\theta_c(t)$ shown in Fig. 6-13*b*, *e(t)* becomes a **suppressed-carrier-modulated** signal. A reversal in phase of *e(t)* occurs whenever the signal crosses the zero-magnitude axis. This reversal in phase causes the ac motor to reverse in direction according to the desired sense of correction of the error signal $\theta_e(t)$. The term *suppressed-carrier modulation* stems from the fact that when a signal $\theta_e(t)$ is modulated by a carrier signal *v(t)* according to Eq. (6-12), the resultant signal *e(t)* no longer contains the original carrier frequency ω_c. To illustrate this, let us assume that $\theta_e(t)$ is also a sinusoid given by

$$\theta_e(t)=\sin\omega_s t \tag{6-14}$$

where, normally, *e(t)*. Using familiar trigonometric relations and substituting Eqs. (6-11) and (6-14) into Eq. (6-12), we get

$$e(t)=\frac{1}{2}K_s E[\cos(\omega_c-\omega_s)t-\cos(\omega_c+\omega_s)t] \tag{6-15}$$

Therefore, *e(t)* no longer contains the carrier frequency E_c or the signal frequency ω_s but has only the two sidebands $\omega_c + \omega_s$ and $\omega_c - \omega_s$.

When the modulated signal is transmitted through the system, the motor acts as a demodulator, so that the displacement of the load will be of the same form as the dc signal before modulation. This is clearly seen from the waveforms of Fig. 6-13*b*. It should be pointed out that a control system need not contain all dc or all ac components. It is quite common to couple a dc component to an ac component through a modulator, or an ac device to a dc device through a demodulator. For instance, the dc amplifier of the system in Fig. 6-13*a* may be replaced by an ac amplifier that is preceded by a modulator and followed by a demodulator.

6-2-2 Tachometers

Tachometers are electromechanical devices that convert mechanical energy into electrical energy. The device works essentially as a voltage generator, with the output voltage proportional to the magnitude of the angular velocity of the input shaft. In control systems, most of the tachometers used are of the dc variety; that is, the output voltage is a dc signal. DC tachometers are used in control systems in many ways; they can be used as velocity indicators to provide shaft-speed readout, velocity feedback, speed control, or stabilization. Figure 6-14 is a block diagram of a typical velocity-control system in which the tachometer output is compared with the reference voltage, which represents the desired velocity to be achieved. The difference between the two signals, or the error, is amplified and used to drive the motor so that the velocity will eventually reach the desired value. In this type of application, the accuracy of the tachometer is highly critical, as the accuracy of the speed control depends on it.

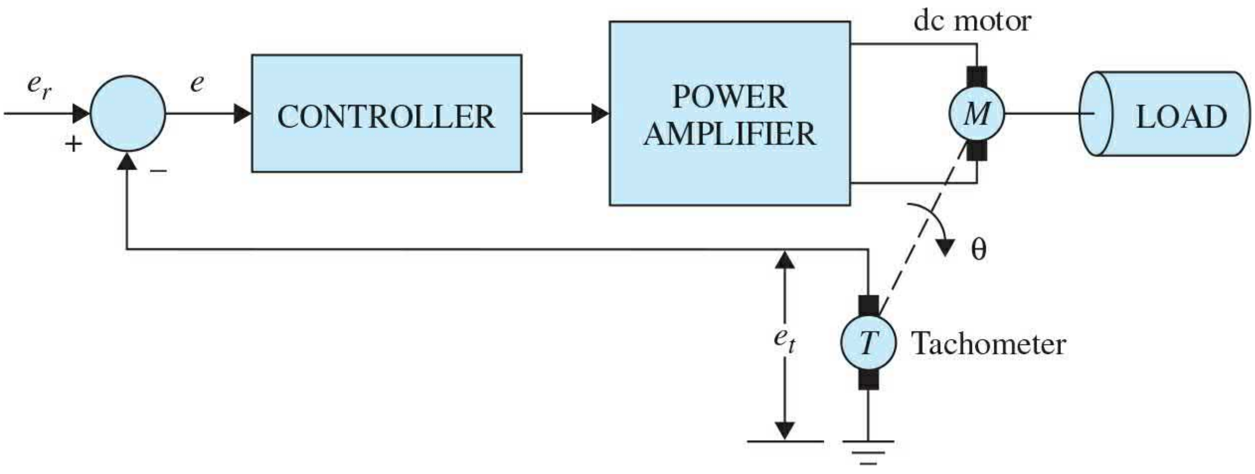


Figure 6-14 Velocity-control system with tachometer feedback.

In a position-control system, velocity feedback is often used to improve the stability or the damping of the closed-loop system. Figure 6-15 shows the block diagram of such an application. In this case, the tachometer feedback forms an inner loop to improve the damping characteristics of the system, and the accuracy of the tachometer is not so critical.

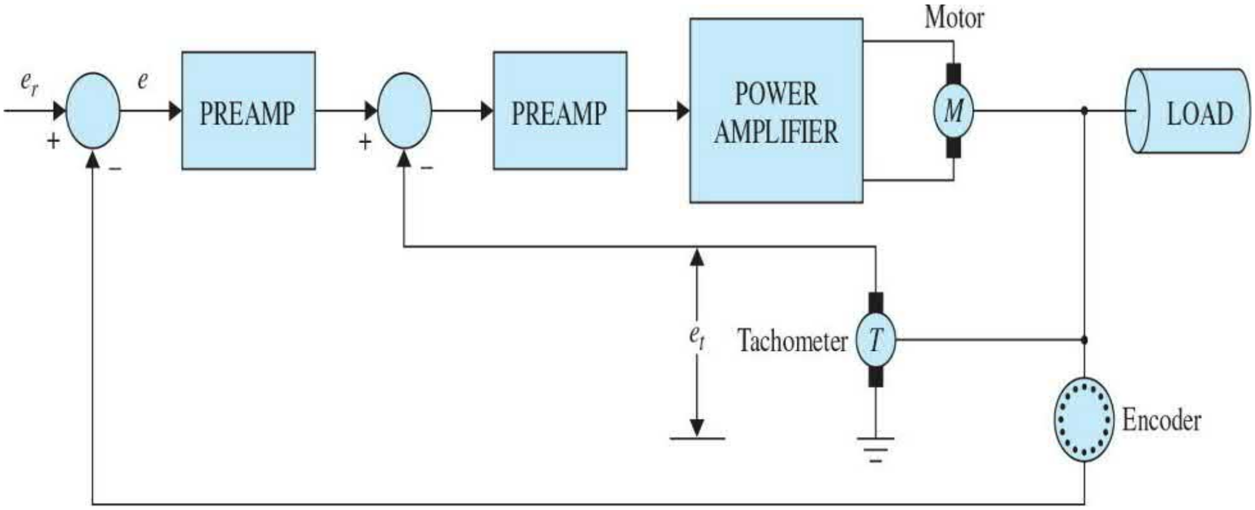


Figure 6-15 Position-control system with tachometer feedback.

The third and most traditional use of a dc tachometer is in providing the visual speed readout of a rotating shaft. Tachometers used in this capacity are generally connected directly to a voltmeter calibrated in revolutions per minute (rpm).

Mathematical Modeling of Tachometers

The dynamics of the tachometer can be represented by the equation

$$e_t(t) = K_t \frac{d\theta(t)}{dt} = K_t \omega(t) \tag{6-16}$$

where $e_t(t)$ is the output voltage; $\theta(t)$, the rotor displacement in radians; $\omega(t)$, the rotor velocity in rad/s; and K_t, the **tachometer constant** in V/rad/s. The value of K_t is usually given as a catalog parameter in **volts per 1000 rpm** (V/krpm).

The transfer function of a tachometer is obtained by taking the Laplace transform on both sides of Eq. (6-16). The result is

$$\frac{E_t(s)}{\Theta(s)} = K_t s \tag{6-17}$$

where $E_t(s)$ and $\Theta(s)$ are the Laplace transforms of $e_t(t)$ and $\theta(t)$, respectively.

6-2-3 Incremental Encoder

Incremental encoders are frequently found in modern control systems for converting linear or rotary displacement into digitally coded or pulse signals. The encoders that output a digital signal are known as absolute encoders. In the simplest terms, absolute encoders provide as output a distinct digital code indicative of each particular least significant increment of resolution. Incremental encoders, on the other hand, provide a pulse for each increment of resolution but do not make distinctions between the increments. In practice, the choice of which type of encoder to use depends on economics and control objectives. For the most part, the need for absolute encoders has much to do with the concern for data loss during power failure or the applications involving periods of mechanical motion without the readout under power. However, the incremental encoder's simplicity in construction, low cost, ease of application, and versatility have made it by far one of the most popular encoders in control systems. Incremental encoders are available in rotary and linear forms. Figures 6-16 and 6-17 show typical rotary and linear incremental encoders.

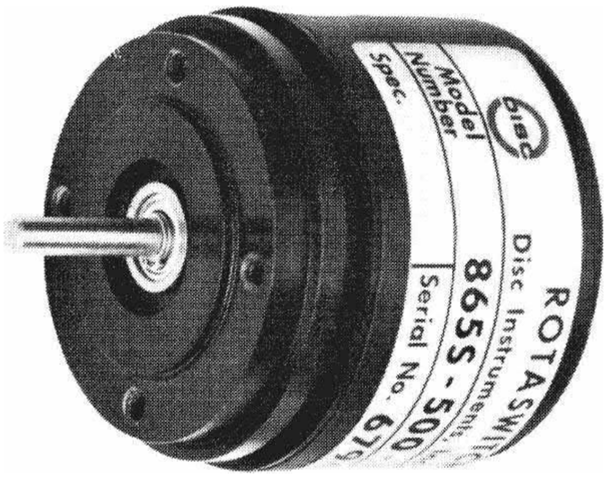


Figure 6-16 Rotary incremental encoder. (Courtesy of DISC Instruments, Inc.)

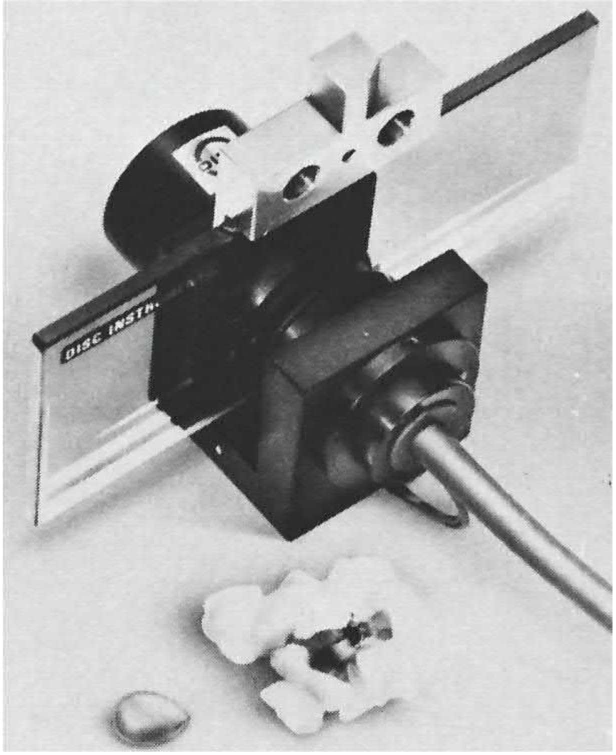


Figure 6-17 Linear incremental encoder. (Courtesy of DISC Instruments, Inc.)

A typical rotary incremental encoder has four basic parts: a light source, a rotary disk, a stationary mask, and a sensor, as shown in Fig. 6-18. The disk has alternate opaque and transparent sectors. Any pair of these sectors represents an incremental period. The mask is used to pass or block a beam of light between the light source and the photosensor located behind the mask. For encoders with relatively low resolution, the mask is not necessary. For fine-resolution encoders (up to thousands of increments per evolution), a multiple-slit mask is often used to maximize reception of the shutter light. The waveforms of the sensor outputs are generally triangular or sinusoidal, depending on the resolution required. Square-wave signals compatible with digital logic are derived by using a linear amplifier followed by a comparator. Figure 6-19*a* shows a typical rectangular output waveform of a single-channel incremental encoder. In this case, pulses are produced for both directions of shaft rotation. A dual-channel encoder with two sets of output pulses is necessary for direction sensing and other control functions. When the phase of the two-output pulse train is 90° apart electrically, the two signals are said to be in quadrature, as shown in Fig. 6-19*b*. The signals uniquely define 0-to-1 and 1-to-0 logic transitions with respect to the direction of rotation of the encoder disk so that a direction-sending logic circuit can be constructed to decode the signals. Figure 6-20 shows the single-channel output and the quadrature outputs with sinusoidal waveforms. The sinusoidal signals from the incremental encoder can be used for fine position control in feedback control systems. The following example illustrates some applications of the incremental encoder in control systems.

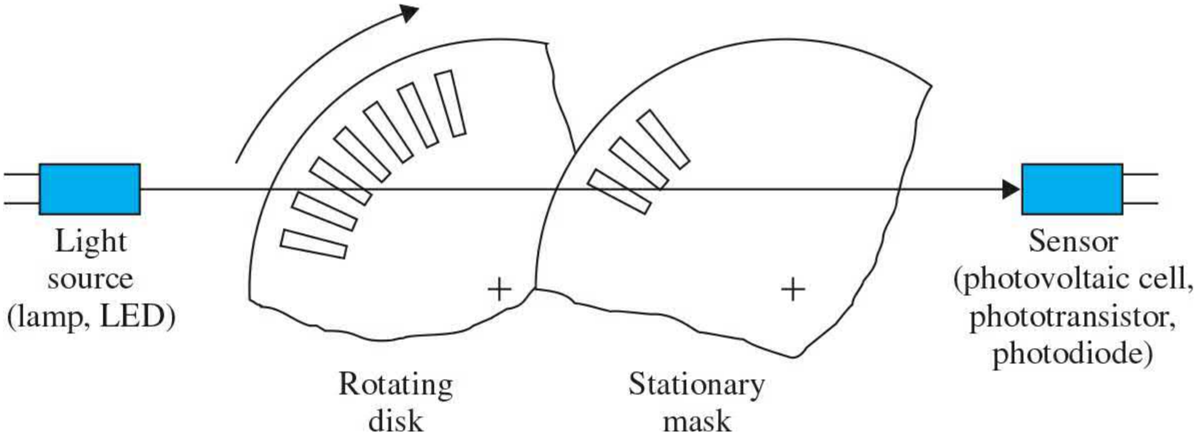


Figure 6-18 Typical incremental optomechanics.

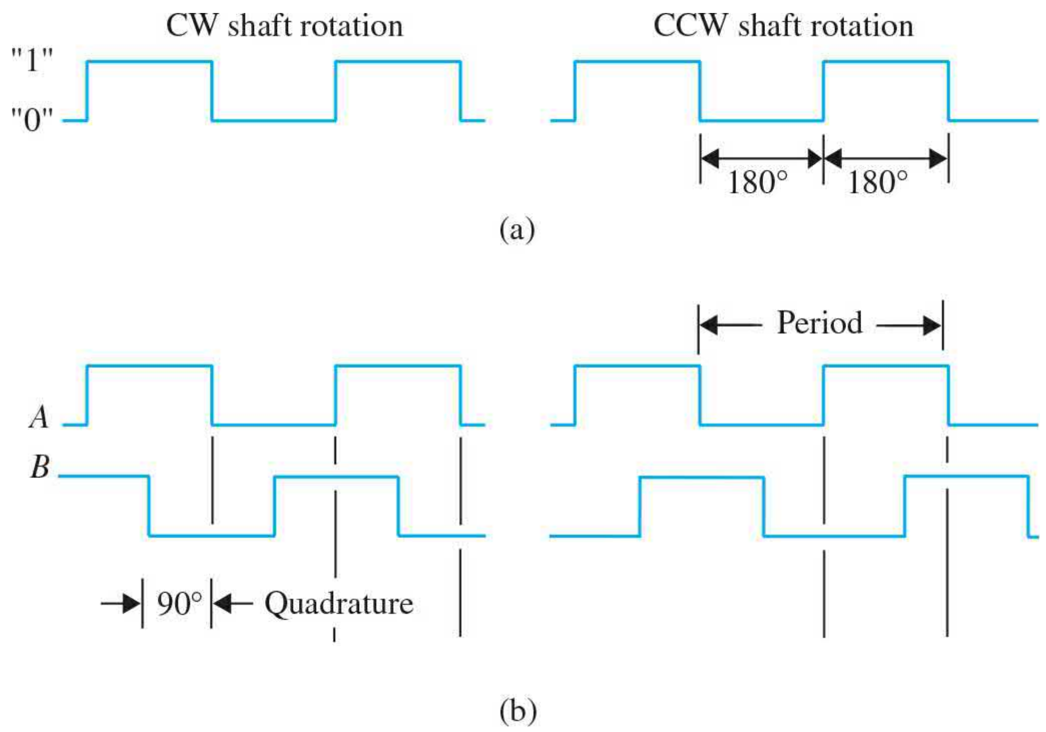


Figure 6-19 (*a*) Typical rectangular output waveform of a single-channel encoder device (bidirectional). (*b*) Typical dual-channel encoder signals in quadrature (bidirectional).

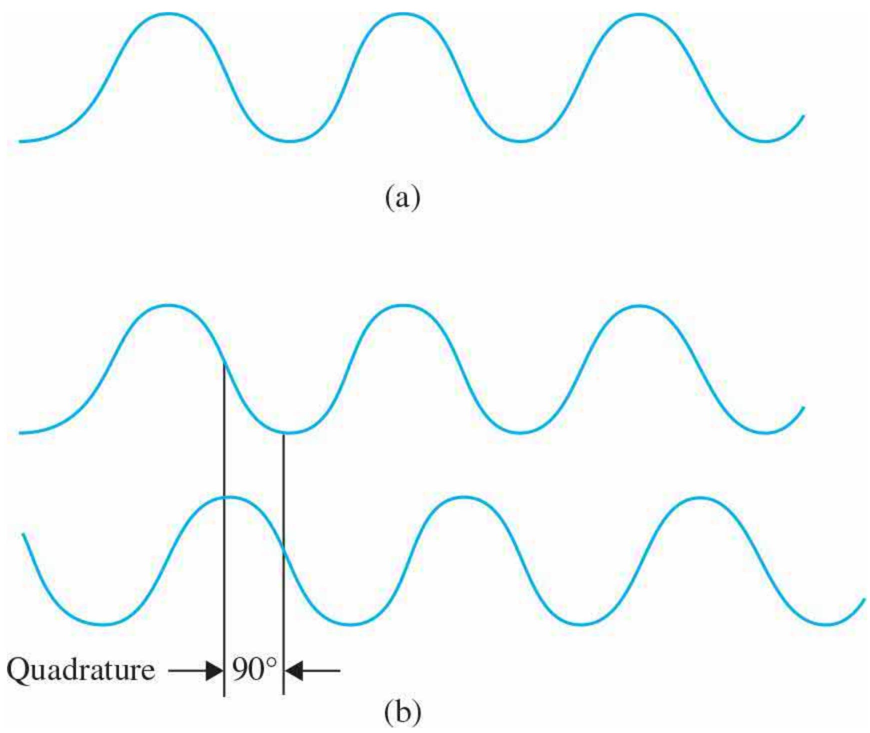


Figure 6-20 (*a*) Typical sinusoidal output waveform of a single-channel encoder device. (*b*) Typical dual-channel encoder signals in quadrature.

EXAMPLE 6-2-1 Consider an incremental encoder that generates two sinusoidal signals in quadrature as the encoder disk rotates. The output signals of the two channels are shown in Fig. 6-21 over one cycle. Note that the two encoder signals generate 4 zero crossings per cycle. These zero crossings can be used for position indication, position control, or speed measurements in control systems. Let us assume that the encoder shaft is coupled directly to the rotor shaft of a motor that directly drives the printwheel of an electronic typewriter or word processor. The printwheel has 96 character positions on its periphery, and the encoder has 480 cycles. Thus, there are 480 × 4 = 1920 zero crossings per revolution. For the 96-character printwheel, this corresponds to 1920/96 = 20 zero crossings per character; that is, there are 20 zero crossings between two adjacent characters.

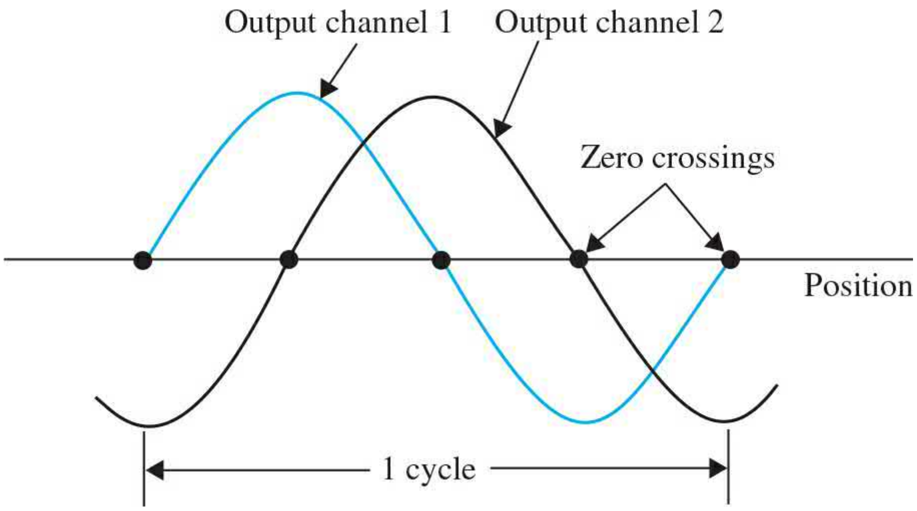


Figure 6-21 One cycle of the output signals of a dual-channel incremental encoder.

One way of measuring the velocity of the printwheel is to count the number of pulses generated by an electronic clock that occurs between consecutive zero crossings of the encoder outputs. Let us assume that a 500-kHz clock is used; that is, the clock generates 500,000 pulses/s. If the counter records, say, 500 clock pulses while the encoder rotates from the zero crossing to the next, the shaft speed is

$$\begin{aligned}\frac{500{,}000 \text{ pulses/s}}{500 \text{ pulses/zero crossing}} &= 1000 \text{ zero crossings/s}\\ &= \frac{1000 \text{ zero crossings/s}}{1920 \text{ zero crossings/rev}} = 0.52083 \text{ rev/s}\\ &= 31.25 \text{ rpm} \qquad (6\text{-}18)\end{aligned}$$

The encoder arrangement described can be used for fine position control of the printwheel. Let the zero crossing A of the waveforms in Fig. 6-21 correspond to a character position on the printwheel (the next character position is 20 zero crossings away), and the point corresponds to a stable equilibrium point. The coarse position control of the system must first drive the printwheel position to within 1 zero crossing on either side of position A; then, by using the slope of the sine wave at position A, the control system should null the error quickly.

6-3 DC MOTORS IN CONTROL SYSTEMS

Direct-current (dc) motors are one of the most widely used prime movers in the industry today. Years ago, the majority of the small servomotors used for control purposes were ac. In reality, ac motors are more difficult to control, especially for position control, and their characteristics are quite nonlinear, which makes the analytical task more difficult. DC motors, on the other hand, are more expensive because of their brushes and commutators, and variable-flux dc motors are suitable only for certain types of control applications. Before permanent-magnet technology was fully developed, the torque-per-unit volume or weight of a dc motor with a permanent-magnet (PM) field was far from desirable. Today, with the development of the rare-earth magnet, it is possible to achieve very high torque-to-volume PM dc motors at reasonable cost. Furthermore, the advances made in brush-and-commutator technology have made these wearable parts practically maintenance-free. The advancements made in power electronics have made brushless dc motors quite popular in high-performance control systems. Advanced manufacturing techniques have also produced dc motors with ironless rotors that have very low inertia, thus achieving a very high torque-to-inertia ratio. Low-time-constant properties have opened new applications for dc motors in computer peripheral equipment such as printers and disk drives, as well as in slew of other applications including automation and machine-tool industries.

6-3-1 Basic Operational Principles of DC Motors

The dc motor is basically a torque transducer that converts electric energy into mechanical energy. The torque developed on the motor shaft is directly proportional to the field flux and the armature current. As shown in Fig. 6-22, a current-carrying conductor is established in a magnetic field with flux ϕ, and the conductor is located at a distance r from the center of rotation. The relationship among the developed torque, flux ϕ, and current i_a is

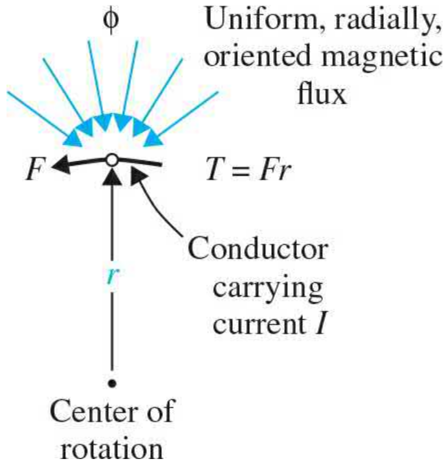

ϕ
Uniform, radially, oriented magnetic flux
F
$T = Fr$
Conductor carrying current I
r
Center of rotation

Figure 6-22 Torque production in a dc motor.

$$T_m = K_m \phi i_a \tag{6-19}$$

where T_m is the motor torque (in N · m, lb · ft, or oz · in.); ϕ, the magnetic flux (in webers); i_a, the armature current (in amperes); and K_m, a proportional constant.

In addition to the torque developed by the arrangement shown in Fig. 6-22, when the conductor moves in the magnetic field, a voltage is generated across its terminals. This voltage, the back emf, which is proportional to the shaft velocity, tends to oppose the current flow. The relationship between the back emf and the shaft velocity is

$$e_b = K_m \phi \omega_m \tag{6-20}$$

where e_b denotes the back emf (volts) and ω_m is the shaft velocity (rad/s) of the motor. Equations (6-19) and (6-20) form the basis of the dc-motor operation.

6-3-2 Basic Classifications of PM DC Motors

In general, the magnetic field of a dc motor can be produced by field windings or permanent magnets. Due to the popularity of PM dc motors in control system applications, we shall concentrate on this type of motor.

PM dc motors can be classified according to commutation scheme and armature design. Conventional dc motors have mechanical brushes and commutators. However, an important type of dc motors in which the commutation is done electronically is called **brushless dc.**

According to the armature construction, the PM dc motor can be broken down into three types of armature design: **iron-core**, **surface-wound,** and **moving-coil** motors.

Iron-Core PM DC Motors

The rotor and stator configuration of an iron-core PM dc motor is shown in Fig. 6-23. The permanent-magnet material can be barium ferrite, Alnico, or a rare-earth compound. The magnetic flux produced by the magnet passes through a laminated rotor structure that contains slots. The armature conductors are placed in the rotor slots. This type of dc motor is characterized by relatively high rotor inertia (since the rotating part consists of the armature windings), high inductance, low cost, and high reliability.

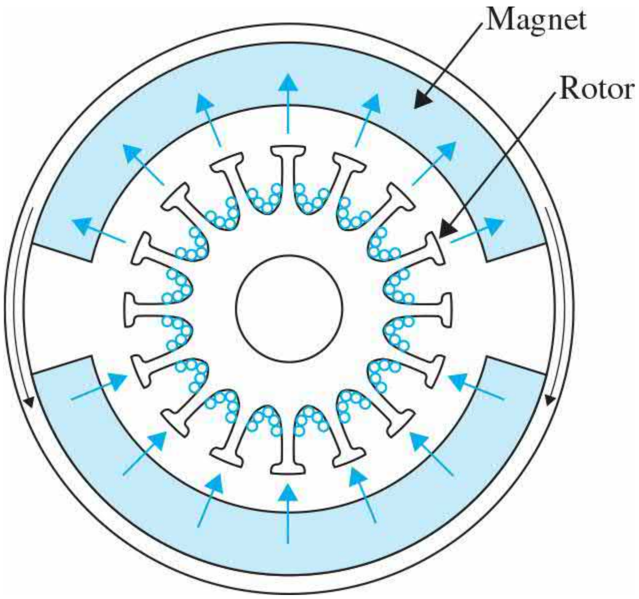


Figure 6-23 Cross-section view of a permanent-magnet (PM) iron-core dc motor.

6-3-3 Surface-Wound DC Motors

Figure 6-24 shows the rotor construction of a surface-wound PM dc motor. The armature conductors are bonded to the surface of a cylindrical rotor structure, which is made of laminated disks fastened to the motor shaft. Because no slots are used on the rotor in this design, the armature has no "cogging" effect. The conductors are laid out in the air gap between the rotor and the PM field, so this type of motor has lower inductance than that of the iron-core structure.

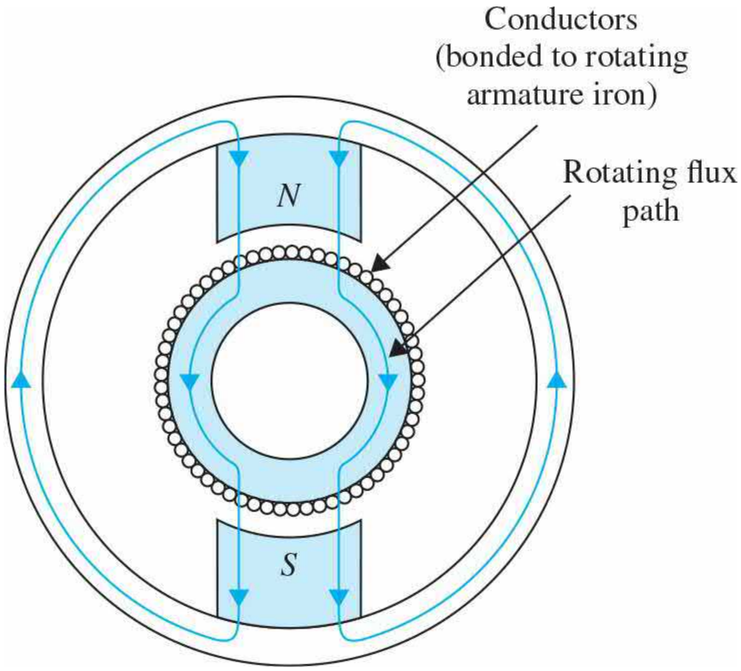


Figure 6-24 Cross-section view of a surface-wound permanent-magnet (PM) dc motor.

6-3-4 Moving-Coil DC Motors

Moving-coil motors are designed to have very low moments of inertia and very low armature inductance. This is achieved by placing the armature conductors in the air gap between a stationary flux return path and the PM structure, as shown in Fig. 6-25. In this case, the conductor structure is supported by nonmagnetic material—usually epoxy resins or fibreglass—to form a hollow cylinder. One end of the cylinder forms a hub, which is attached to the motor shaft. A cross-section view of such a motor is shown in Fig. 6-26. Because all unnecessary elements have been removed from the armature of the moving-coil motor, its moment of inertia is very low. Because the conductors in the moving-coil armature are not in direct contact with iron, the motor inductance is very low, and values of less than 100 µH are common in this type of motor. Its low-inertia and low-inductance properties make the moving-coil motor one of the best actuator choices for high-performance control systems.

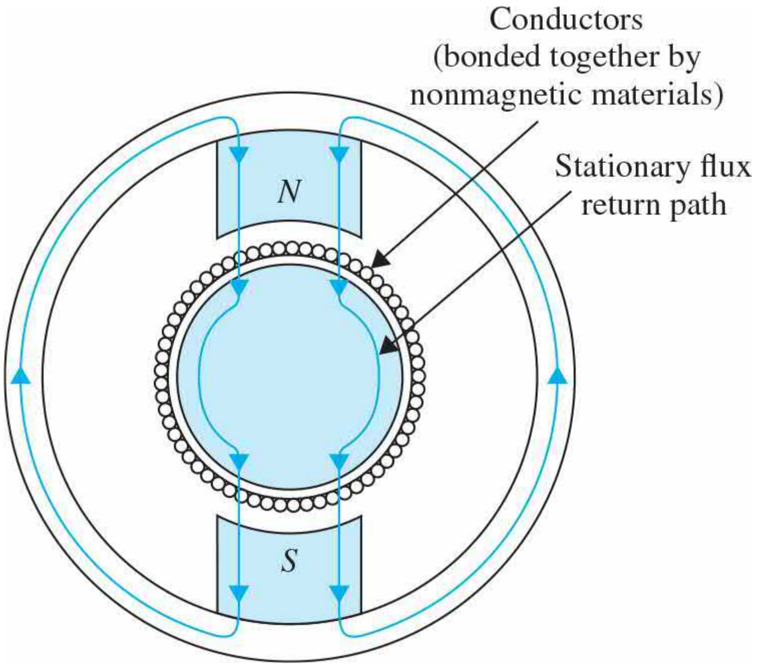


Figure 6-25 Cross-section view of a surface-wound permanent-magnet (PM) dc motor.

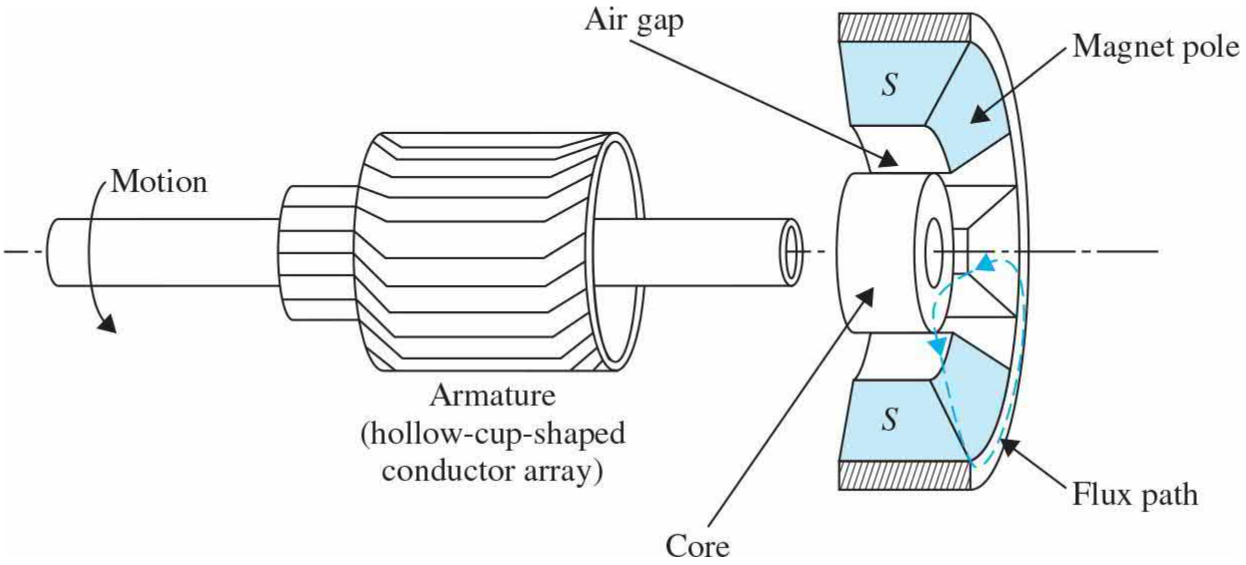


Figure 6-26 Cross-section side view of a moving-coil dc motor.

6-3-5 Brushless DC Motors

Brushless dc motors differ from the previously mentioned dc motors in that they employ electrical (rather than mechanical) commutation of the armature current. The most common configuration of brushless dc motors—especially for incremental-motion applications—is one in which the rotor consists of magnets and “back-iron” support and whose commutated windings are located external to the rotating parts, as shown in Fig. 6-27. Compared to the conventional dc motors, such as the one shown in Fig. 6-26, it is an inside-out configuration.

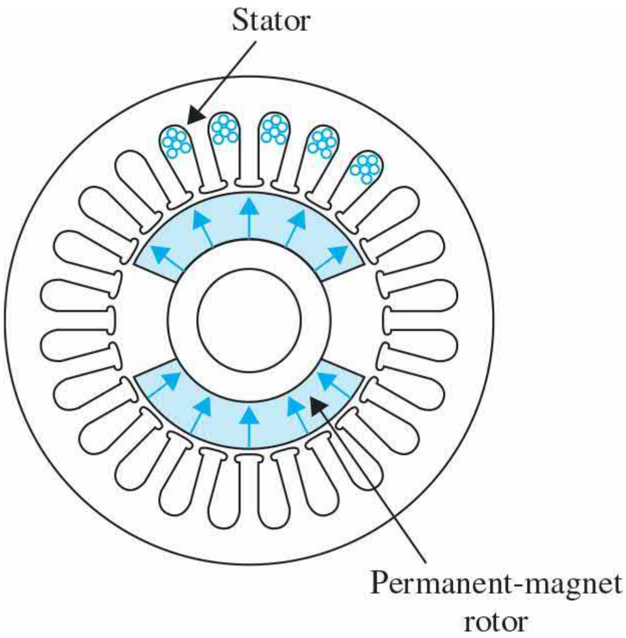


Figure 6-27 Cross-section view of a brushless, permanent-magnet (PM), iron-core dc motor.

Depending on the specific application, brushless dc motors can be used when a low moment of inertia is needed, such as the spindle drive in high-performance disk drives used in computers.

6-3-6 Mathematical Modeling of PM DC Motors

Dc motors are extensively used in control systems. In this section, we establish the mathematical model for dc motor. As it will be demonstrated here, the mathematical model of a dc motor is linear. We use the equivalent circuit diagram in Fig. 6-28 to represent a PM dc motor. The armature is modeled as a circuit with resistance R_a connected in series with an inductance L_a, and a voltage source e_b representing the back emf (electromotive force) in the armature when the rotor rotates. The motor variables and parameters are defined as follows:

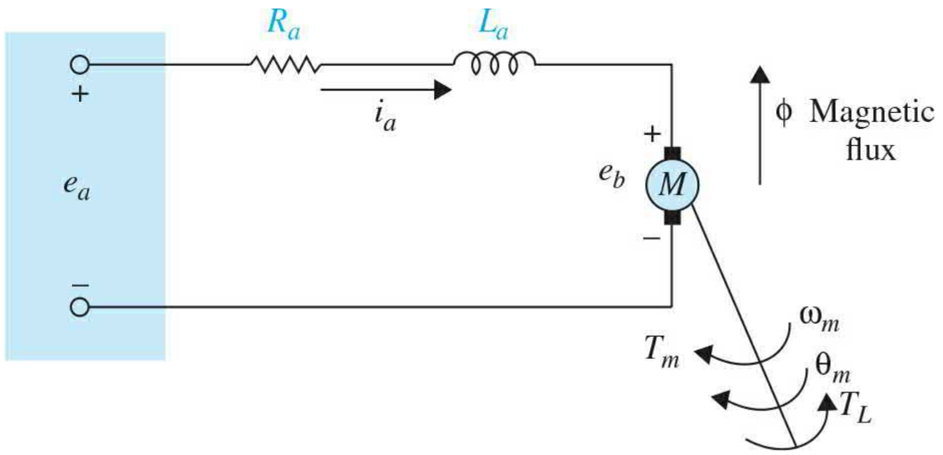


Figure 6-28 Model of a separately excited dc motor.

$i_a(t)$ = armature current

R_a = armature resistance

$e_b(t)$ = back emf

$T_L(t)$ = load torque

$T_m(t)$ = motor torque

$\theta_m(t)$ = rotor displacement

K_i = torque constant

L_a = armature inductance

$e_a(t)$ = applied voltage

K_b = back-emf constant

ϕ = magnetic flux in the air gap

$\omega_m(t)$ = rotor angular speed

J_m = rotor inertia

B_m = viscous-damping coefficient

With reference to the circuit diagram of Fig. 6-28, **the control of the dc motor is applied at the armature terminals in the form of the applied voltage *$e_a(t)$***. For linear analysis, we assume that the torque developed by the motor is proportional to the air-gap flux and the armature current. Thus,

$$T_m(t) = K_m(t)\phi i_a(t) \tag{6-21}$$

Because ϕ is constant, Eq. (6-21) is rewritten as

$$T_m(t) = K_i i_a(t) \tag{6-22}$$

where K_i is the **torque constant** in N · m/A, lb · ft/A, or oz · in/A.

Starting with the control input voltage $e_a(t)$, the cause-and-effect equations for the motor circuit in Fig. 6-28 are

$$\frac{di_a(t)}{dt} = \frac{1}{L_a}e_a(t) - \frac{R_a}{L_a}i_a(t) - \frac{1}{L_a}e_b(t) \tag{6-23}$$

$$T_m(t) = K_i i_a(t) \tag{6-24}$$

$$e_b(t) = K_b\frac{d\theta_m(t)}{dt} = K_b\omega_m(t) \tag{6-25}$$

$$\frac{d^2\theta_m(t)}{dt^2} = \frac{1}{J_m}T_m(t) - \frac{1}{J_m}T_L(t) - \frac{B_m}{J_m}\frac{d\theta_m(t)}{dt} \tag{6-26}$$

where $T_L(t)$ represents a load frictional torque such as Coulomb friction, which works as a disturbance reducing the speed of the motor. As an example, consider how the speed of a motor is reduced in case of an electric juicer once a fruit is pressed in.

Equations (6-23) through (6-26) consider that the applied voltage $e_a(t)$ is the cause; Eq. (6-23) considers that $di_a(t)/dt$ is the immediate effect due to $e_a(t)$; in Eq. (6-24), $i_a(t)$ causes the torque $T_m(t)$; Eq. (6-25) defines the back emf; and, finally, in Eq. (6-26), the torque $T_m(t)$ causes the angular speed $\omega_m(t)$ and displacement $\theta_m(t)$.

Upon taking the Laplace transform of Eqs. (6-23) through (6-26), and assuming zero initial conditions, we get

$$I_a(s) = \frac{1}{L_a}E_a(s) - \frac{R_a}{L_a}I_a(s) - \frac{1}{L_a}E_b(s) \tag{6-27}$$

$$T_m(s) = K_i I_a(s) \tag{6-28}$$

$$E_b(s) = K_b s\Theta_m(s) = K_b\Omega_m(s) \tag{6-29}$$

$$s^2\Theta_m(s) = \frac{1}{J_m}T_m(s) - \frac{1}{J_m}T_L(s) - \frac{B_m}{J_m}s\Theta_m(s) \tag{6-30}$$

After rearranging Eq. (6-27) and breaking down Eq. (6-30) to represent angular speed and position separately, Eqs. (6-27) through (6-30) take the following form

$$I_a(s)+\frac{R_a}{L_a}I_a(s)=\frac{1}{L_a}E_a(s)-\frac{1}{L_a}E_b(s) \tag{6-31}$$

$$T_m(s)=K_iI_a(s) \tag{6-32}$$

$$E_b(s)=K_b\Omega_m(s) \tag{6-33}$$

$$(J_ms+B_m)\,\Omega_m(s)=T_m(s)-T_L(s) \tag{6-34}$$

$$\Theta_m(s)=\frac{1}{s}\Omega_m(s) \tag{6-35}$$

the resulting equations may be individually represented by block diagrams, as shown in Fig. 6-29.

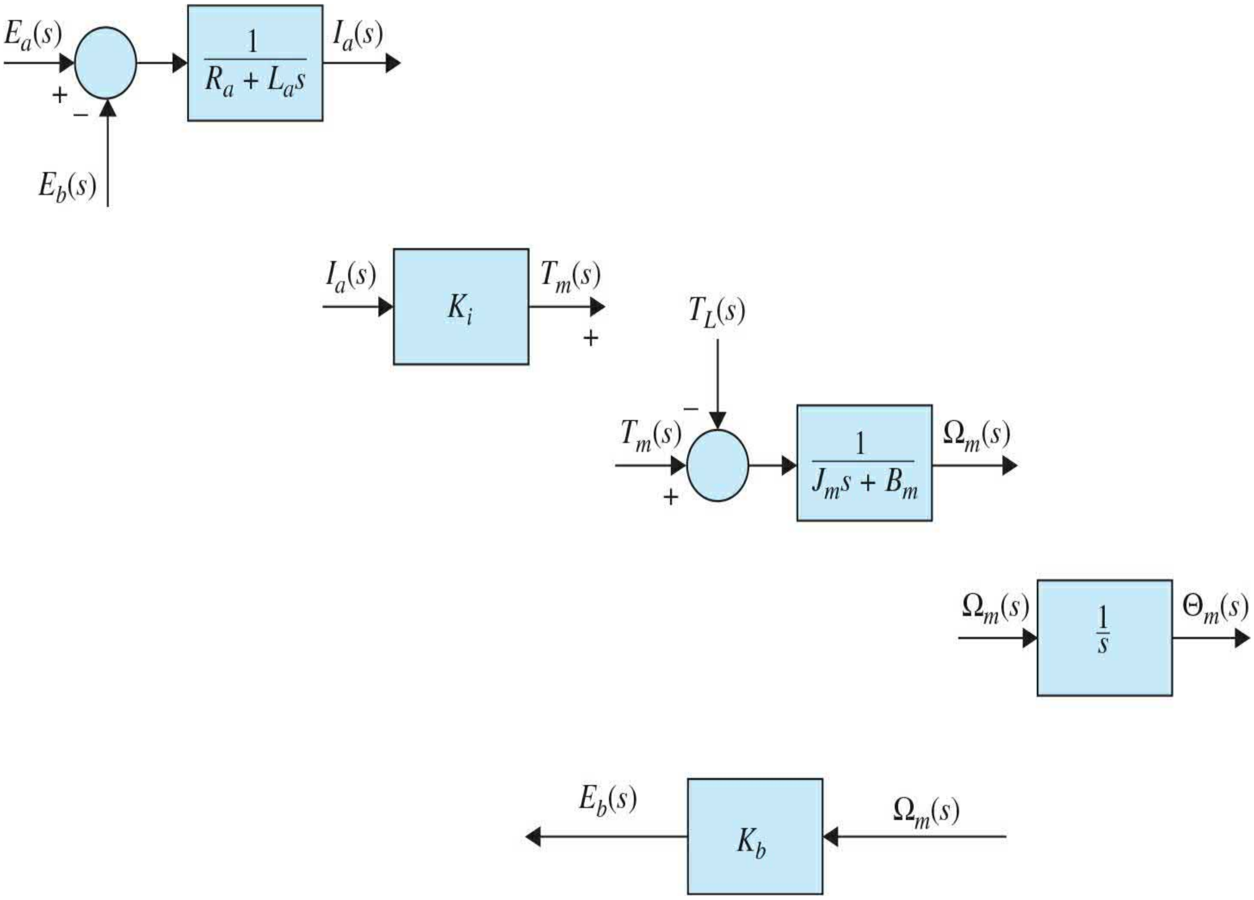


Figure 6-29 Individual block diagram representation of Eqs. (6-31) through (6-35).

Figure 6-30 shows a block-diagram representation of the dc-motor system. The advantage of using the block diagram is that it gives a clear picture of the transfer function relation between each block of the system. The transfer function between the motor displacement and the input voltage is obtained from the overall system block diagram in Fig. 6-30 as

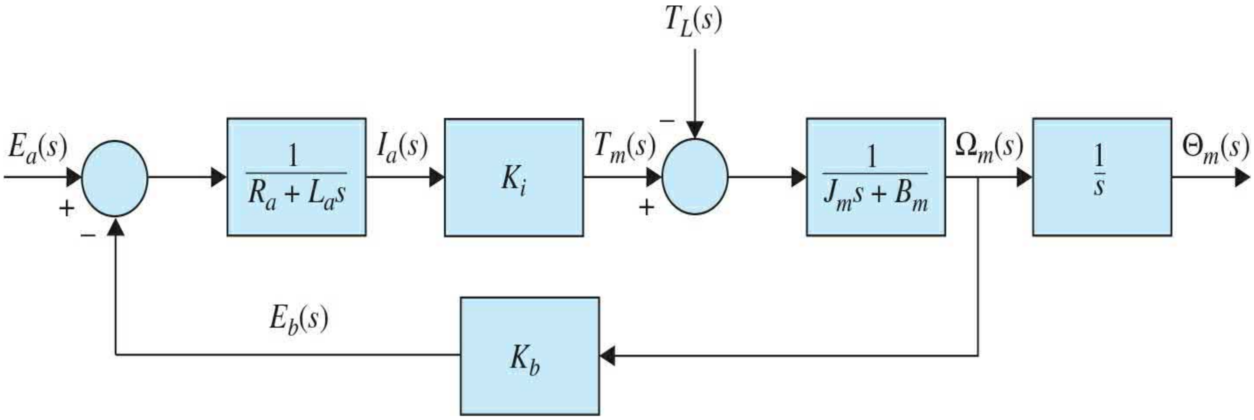


Figure 6-30 Block diagram of a dc-motor system.

$$\frac{\Theta_m(s)}{E_a(s)} = \frac{K_i}{L_a J_m s^3 + (R_a J_m + B_m L_a)s^2 + (K_b K_i + R_a B_m)s} \tag{6-36}$$

where $T_L(t)$ has been set to zero—that is, no load is applied to the motor.

Because an s can be factored out of the denominator of Eq. (6-36), *the significance of the transfer function $\Theta_m(s)/E_a(s)$ is that the dc motor is essentially an integrating device between these two variables*. This is expected because, if $e_a(t)$ is a constant input, the output motor displacement will behave as the output of an integrator; that is, it will increase linearly with time.

Although a dc motor by itself is basically an open-loop system, the block diagram of Fig. 6-30 shows that the motor has a "built-in" feedback loop caused by the back emf. Physically, the back emf represents the feedback of a signal that is proportional to the negative of the speed of the motor. As seen from Eq. (6-36), the back-emf constant K_b represents an added term to the resistance R_a and the viscous-friction coefficient B_m. Therefore, *the back-emf effect is equivalent to an "electric damping," which tends to improve the stability of the motor and, in general, the stability of the system*.

The **state variables** of the system can be defined as $i_a(t)$, $\omega_m(t)$, and $\theta_m(t)$.

By direct substitution and eliminating all the nonstate variables from Eqs. (6-23) through (6-26), the state equations of the dc-motor system are written in vector-matrix form:

$$\begin{bmatrix} \dfrac{di_a(t)}{dt} \\ \dfrac{d\omega_m(t)}{dt} \\ \dfrac{d\theta_m(t)}{dt} \end{bmatrix} = \begin{bmatrix} -\dfrac{R_a}{L_a} & -\dfrac{K_b}{L_a} & 0 \\ \dfrac{K_i}{J_m} & -\dfrac{B_m}{J_m} & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} i_a(t) \\ \omega_m(t) \\ \theta_m(t) \end{bmatrix} + \begin{bmatrix} \dfrac{1}{L_a} \\ 0 \\ 0 \end{bmatrix} e_a(t) + \begin{bmatrix} 0 \\ -\dfrac{1}{J_m} \\ 0 \end{bmatrix} T_L(t) \qquad (6\text{-}37)$$

The dc motor is essentially an integrating device between these two variables.

Notice that, in this case, $T_L(t)$ is treated as a second input in the state equations.

The SFG diagram of the system is drawn as shown in Fig. 6-31, following the procedure discussed in Sec. 4-3.

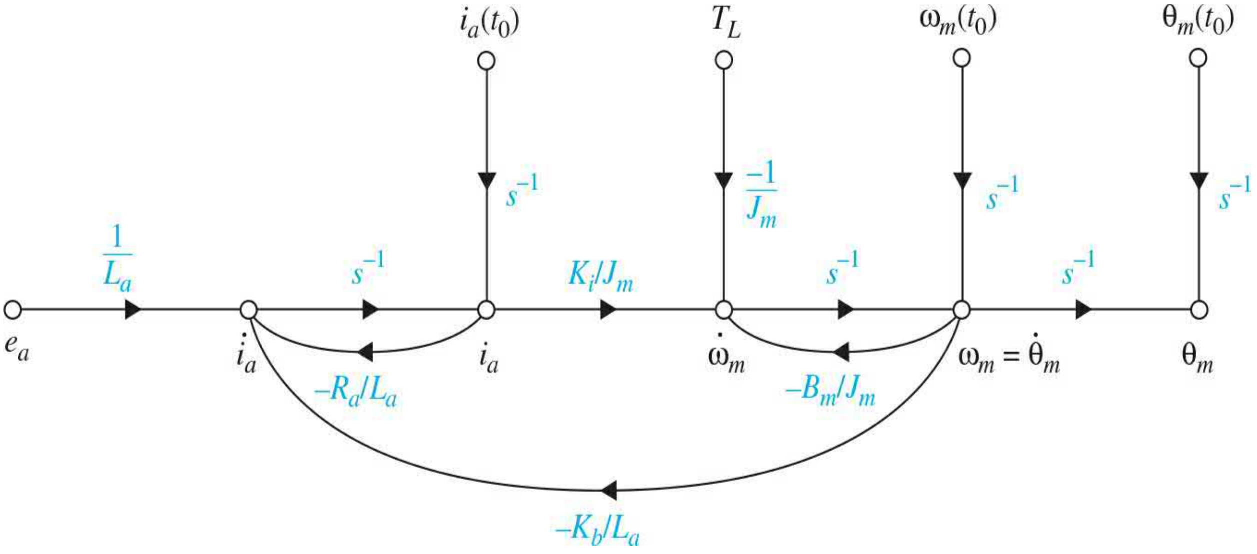


Figure 6-31 Signal-flow graph diagram of a dc-motor system with nonzero initial conditions.

6-3-7 Relation between K_i and K_b

Although functionally the torque constant K_i and back-emf constant K_b are two separate parameters, for a given motor their values are closely related. To show the relationship, we write the mechanical power developed in the armature as

$$P = e_b(t)i_a(t) \tag{6-38}$$

The mechanical power is also expressed as

$$P = T_m(t)\omega_m(t) \tag{6-39}$$

where, in SI units, $T_m(t)$ is in N · m and $w_m(t)$ is in rad/s. Now, substituting Eqs. (6-24) and (6-25) in Eqs. (6-38) and (6-39), we get

$$P = T_m(t)\omega_m(t) = K_b\omega_m(t)\frac{T_m(t)}{K_i} \tag{6-40}$$

from which we get, **in SI units**

$$K_b(\text{V/rad/s}) = K_i(\text{N}\cdot\text{m/A}) \tag{6-41}$$

In SI units, the values of K_b and K_i are identical.

Thus, we see that, in SI units, the values of K_b and K_i are identical if K_b is represented in V/rad/s and K_i is in N · m/A.

In the British unit system, we convert Eq. (6-38) into horsepower (hp); that is,

$$P = \frac{e_b(t)i_a(t)}{746}\ \text{hp} \tag{6-42}$$

In terms of torque and angular speed, P in Eq. (6-39) is rewritten in terms of hp as

$$P = \frac{T_m(t)\omega_m(t)}{550} \text{ hp} \tag{6-43}$$

where $T_m(t)$ is in ft-lb and $w_m(t)$ is in rad/s. Using Eqs. (6-24) and (6-25), and equating Eq. (6-42) to Eq. (6-43), we get

$$\frac{K_b\omega_m(t)T_m(t)}{746K_i} = \frac{T_m(t)\omega_m(t)}{550} \tag{6-44}$$

Thus,

$$K_b = \frac{746}{550}K_i = 1.356K_i \tag{6-45}$$

where K_b is in V/rad/s and K_i is in ft · lb/A.

6-4 SPEED AND POSITION CONTROL OF A DC MOTOR

Servomechanisms are probably the most frequently encountered electromechanical control systems. Applications include robots (each joint in a robot requires a position servo), numerical control (NC) machines, and laser printers, to name but a few. The common characteristic of all such systems is that the variable to be controlled (usually position or velocity) is fed back to modify the command signal. The servomechanism that will be used in the experiments in this chapter comprises a dc motor and amplifier that are fed back the motor speed and position values.

One of the key challenges in the design and implementation of a successful controller is obtaining an accurate model of the system components, particularly the actuator. In the previous section, we discussed various issues associated with modeling of dc motors. In this section, we study the speed and position control of dc motors.

6-4-1 Speed Response and the Effects of Inductance and Disturbance: Open-Loop Response

Consider the schematic diagram of an armature-controlled dc motor shown in Fig. 6-32, where the field current is held constant in this system. In this case, the sensor attached to the motor is a tachometer, which is used to sense the motor shaft speed. Depending on the application, for example, position control, a potentiometer, or an encoder may be used as a sensor instead—see Fig. 6-2. System parameters and variables include

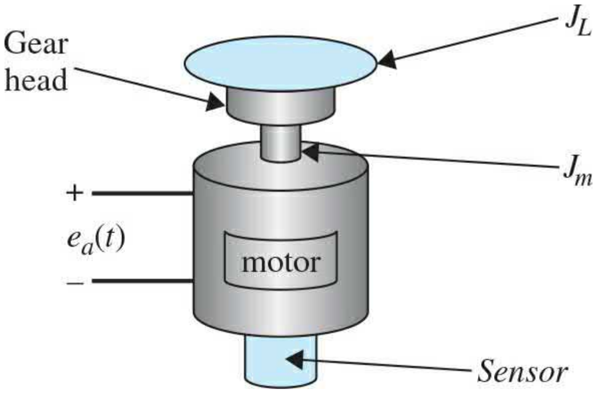


Figure 6-32 An armature-controlled dc motor with a gear head and a load inertia J_L.

R_a = armature resistance, Ω

L_a = armature inductance, H

e_a = applied armature voltage, V

e_b = back emf, V

θ_m = angular displacement of the motor shaft, rad

ω_m = angular speed of the motor shaft, rad/s

T_m = torque developed by the motor, N · m

J_L = moment of inertia of the load, kg · m^2

T_L = external load torque considered as a disturbance, N · m

J_m = moment of inertia of the motor (motor shaft), kg · m^2

J = equivalent moment of inertia of the motor and load connected to the motor-shaft, $J = J_L/n^2 + J_m$, $kg - m^2$ (refer to Chap. 2 for more details)

n = gear ratio

B_m = viscous-friction coefficient of the motor, N · m/rad/s

B_L = viscous-friction coefficient of the load, N · m/rad/s

B = equivalent viscous-friction coefficient of the motor and load referred to the motor shaft, N · m/rad/s (in the presence of gear ratio, B must be scaled by n; refer to Chap. 2 for more details)

K_i = torque constant, N · m/A

K_b = back-emf constant, V/rad/s

K_t = speed sensor (in this case a tachometer) gain, V/rad/s

As shown in Fig. 6-33, the armature-controlled dc motor is itself a feedback system, where back-emf voltage is proportional to the speed of the motor. In Fig. 6-33, we have included the effect of any possible external load (e.g., the load applied to a juice machine by the operator pushing in the fruit) as a disturbance torque T_L. The system may be arranged in input-output form, in Laplace domain, such that $E_a(s)$ is the input and $\Omega_m(s)$ is the output:

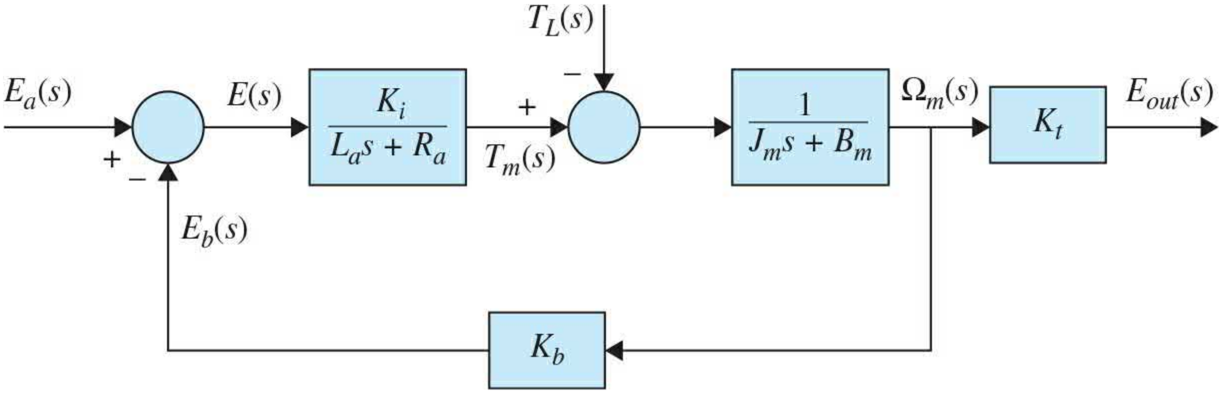


Figure 6-33 Block diagram of an armature-controlled dc motor.

$$\Omega_m(s) = \frac{\frac{K_i}{R_a J_m}}{\left(\frac{L_a}{R_a}\right)s^2 + \left(1 + \frac{B_m L_a}{R_a J_m}\right)s + \frac{K_i K_b + R_a B_m}{R_a J_m}} E_a(s) - \frac{\left\{1 + s\left(\frac{L_a}{R_a}\right)\right\} / J_m}{\left(\frac{L_a}{R_a}\right)s^2 + \left(1 + \frac{B_m L_a}{R_a J_m}\right)s + \frac{K_i K_b + R_a B_m}{R_a J_m}} T_L(s) \tag{6-46}$$

The ratio L_a/R_a is called the *motor electric-time constant*, which makes the system speed-response transfer function second order and is denoted by τ_e. Also, it introduces a zero to the disturbance-output transfer function. However, because L_a in the armature circuit is very small, τ_e may be neglected, resulting in the simplified transfer functions and the block diagram of the system. Thus, the speed of the motor shaft may be simplified to

$$\Omega_m(s)=\frac{\dfrac{K_i}{R_aJ_m}}{s+\dfrac{K_iK_b+R_aB_m}{R_aJ_m}}E_a(s)-\frac{\dfrac{1}{J_m}}{s+\dfrac{K_iK_b+R_aB_m}{R_aJ_m}}T_L(s) \qquad (6\text{-}47)$$

or

$$\Omega_m(s)=\frac{K_{\text{eff}}}{\tau_m s+1}E_a(s)-\frac{\dfrac{\tau_m}{J_m}}{\tau_m s+1}T_L(s) \qquad (6\text{-}48)$$

where $K_{eff} = K_i/(R_aB_m + K_iK_b)$ is the motor gain constant, and $\tau_m = R_aJ_m/(R_aB_m + K_iK_b)$ is the motor mechanical time constant.

Using superposition, we get

$$\Omega_m(s)=\Omega_m(s)\big|_{T_L(s)=0}+\Omega_m(s)\big|_{E_a(s)=0} \qquad (6\text{-}49)$$

To find the response $\omega_m(t)$, we use superposition and find the response due to the individual inputs. For $T_L = 0$ (no disturbance and $B = 0$) and an applied voltage $e_a(t) = A$, such that $E_a(s) = A/s$,

$$\omega_m(t)=\frac{AK_i}{K_iK_b+R_aB_m}(1-e^{-t/\tau_m}) \qquad (6\text{-}50)$$

In this case, note that the motor mechanical time constant τ_m is reflective of how fast the motor is capable of overcoming its own inertia J_m to reach a steady-state or constant speed dictated by voltage E_a. From Eq. (6-50), the speed final value is $\omega_{fv}=\dfrac{AK_i}{K_iK_b+R_aB_m}$. As it will be seen later in Chap. 7, this value is also known as the **reference input**, which reflects the desired output value for a give input voltage. As τ_m increases, the approach to steady state takes longer. See Fig. 6-34 for the typical time response associated with Eq. (6-50).

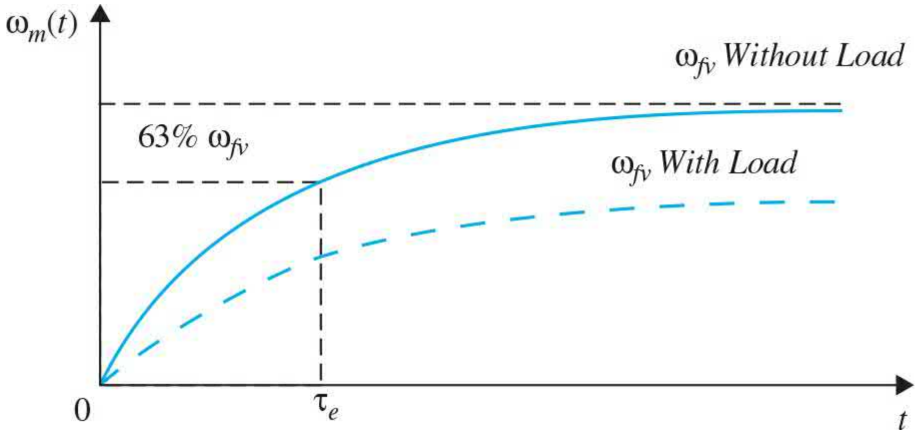


Figure 6-34 Typical speed response of a dc motor. Solid line represents a no- load response. Dashed line represents the effect of a constant load on the speed response.

If we apply a constant load torque of magnitude D to the system (i.e., T_L = D/s), the speed response from Eq. (6-48) will change to

$$\omega_m(t) = \frac{K_i}{K_i K_b + R_a B_m}\left(A - \frac{R_a D}{K_i}\right)(1 - e^{-t/\tau_m}) \tag{6-51}$$

which clearly indicates that the disturbance T_L affects the final speed of the motor. From Eq. (6-51), at steady state, the speed of the motor is $\omega_{fv} = \frac{K_i}{K_i K_b + R_a B_m}\left(A - \frac{R_a D}{K_i}\right).$ Here the final value of $\omega_m(t)$ is reduced by $R_a D/K_m K_b$, as shown in Fig. 6-34. A practical note is that **the value of** $T_L = D$ **may never exceed the motor stall torque**, and hence for the motor to turn, from Eq. (6-51), $AK_i/R_a > D$, which sets a limit on the magnitude of the torque T_L. For a given motor, the value of the stall torque can be found in the manufacturer's catalog.

In a realistic scenario, you must measure motor speed using a sensor. How would the sensor affect the equations of the system (see Fig. 6-33)?

6-4-2 Speed Control of DC Motors: Closed-Loop Response

As seen previously, the output speed of the motor is highly dependent on the value of torque T_L. We can improve the speed performance of the motor by using a proportional feedback controller. The controller is composed of a sensor (usually a tachometer for speed applications) to sense the speed and an amplifier with gain K (proportional control—refer to row (a) in Table 6-1) in the configuration shown in Fig. 6-35. The block diagram of the system is also shown in Fig. 6-36.

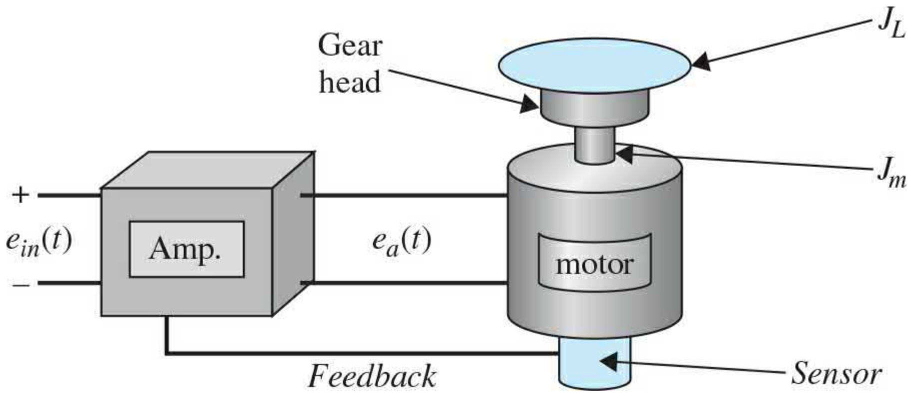


Figure 6-35 Feedback control of an armature-controlled dc motor with a load inertia.

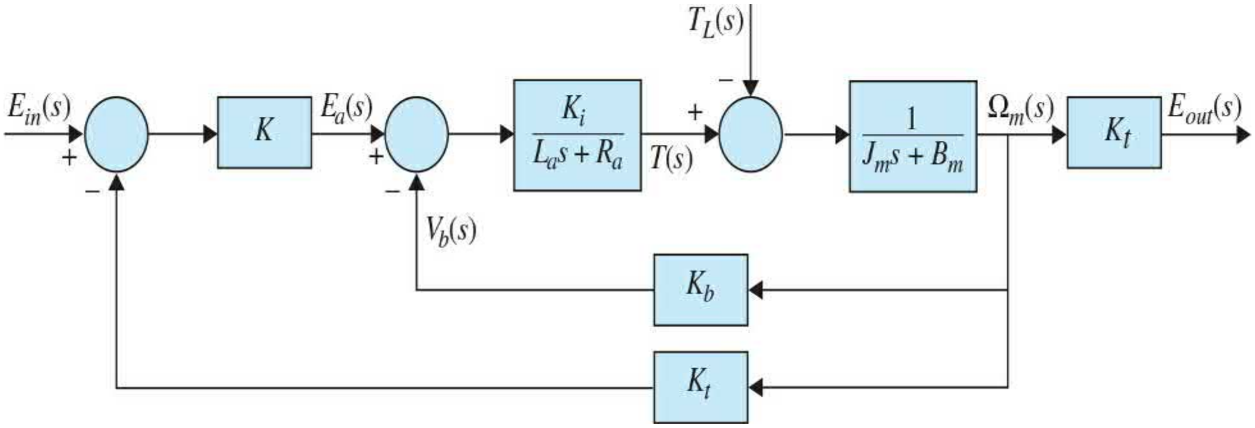


Figure 6-36 Block diagram of a speed-control, armature-controlled dc motor.

Note that the speed at the motor shaft is sensed by the tachometer with a gain K_t. For ease in comparison of input and output, the input to the control system is converted from voltage E_{in} to speed Ω_{in} using the tachometer gain K_t. Hence, for $L_a \simeq 0$, we have

$$\Omega_m(s) = \frac{\dfrac{K_tK_iK}{R_aJ_m}}{s+\left(\dfrac{K_iK_b+R_aB_m+K_tK_iK}{R_aJ_m}\right)}\Omega_{\text{in}}(s) - \frac{\dfrac{1}{J_m}}{s+\left(\dfrac{K_iK_b+R_aB_m+K_tK_iK}{R_aJ_m}\right)}T_L(s) \tag{6-52}$$

For a step input $\Omega_{\text{in}} = A/s$ and disturbance torque value $T_L = D/s$, the output becomes

$$\omega_m(t) = \frac{AKK_iK_t}{R_aJ_m}\tau_c(1-e^{-t/\tau_c}) - \frac{\tau_cD}{J_m}(1-e^{-t/\tau_c}) \tag{6-53}$$

where $\tau_c = \dfrac{R_aJ_m}{K_iK_b+R_aB+K_tK_iK}$ is the system time constant. The steady-state response in this case is

$$\omega_{fv} = \left(\frac{AKK_iK_t}{K_iK_b+R_aB_m+K_tK_iK} - \frac{R_aD}{K_iK_b+R_aB_m+K_tK_iK}\right) \tag{6-54}$$

where $\omega_{fv} \rightarrow A$ as $K \rightarrow \infty$. While the time response of Eq. (6-53) has a similar graphical representation as in Fig. 6-34, the control gain can reduce the effect of disturbance, because for high **K** values, the disturbance effect is reduced. Of course, there are limitations. For example, practically speaking, we are limited by amplifier saturation and the motor's input voltage limits. The system in this case will also exhibit steady-state error, which is later discussed in Chaps. 7 and 8. As in Sec. 6-4-1, the reader should investigate what happens if the inertia J_L is included in this model. If the load inertia J_L is too large, will the motor be able to turn? Again, as in Sec. 6-4-1, you will have to read the speed-sensor voltage to measure speed. How will that affect your equations?

6-4-3 Position Control

The position response in the open-loop case may be obtained by integrating the speed response. Then, considering Fig. 6-33, we have $\Theta_m()s)$ = $\Omega_m(s)/s$. The open-loop transfer function is therefore

$$\frac{\Theta_m(s)}{E_a(s)} = \frac{K_i}{s(L_aJs^2 + (L_aB_m + R_aJ)s + R_aB_m + K_iK_b)} \tag{6-55}$$

where in this case we have used the total inertia $J = J_L/n^2 + J_m$. For small L_a, the time response in this case is

$$\theta_m(t) = \frac{A}{K_b}(t + \tau_m e^{-t/\tau_m} - \tau_m) \tag{6-56}$$

which implies that the motor shaft is turning without stop at a constant steady-state speed A/K_b. To control the position of the motor shaft, the simplest strategy is to use an amplifier with gain K, as shown in row (a) in Table 6-1. The block diagram of the closed-loop system is shown in Fig. 6-37. The system is composed of an angular position sensor (usually an encoder or a potentiometer for position applications). Note that, for simplicity, the input voltage can be scaled to a position input $\Theta_{\text{in}}(s)$ so that the input and output have the same units and scale. Alternatively, the output can be converted into voltage using the sensor gain value. The closed-loop transfer function in this case becomes

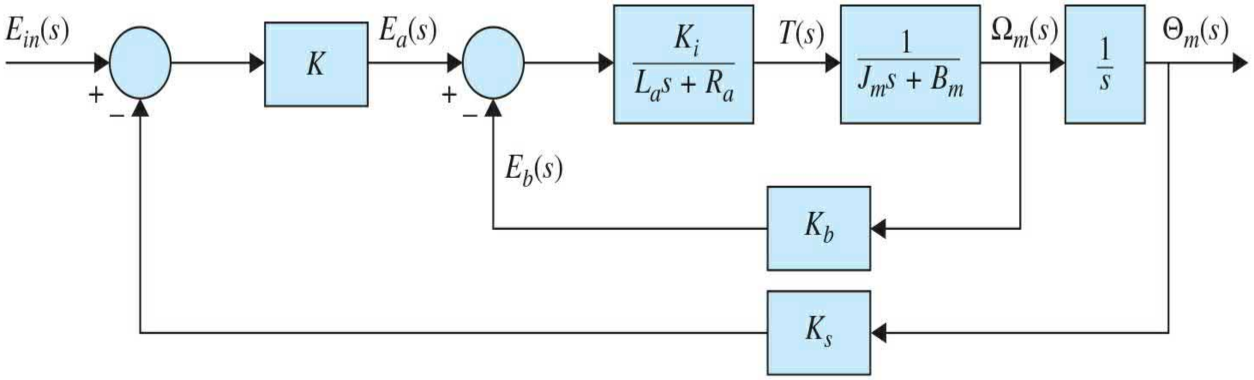


Figure 6-37 Block diagram of a position-control, armature-controlled dc motor.

$$\frac{\Theta_m(s)}{\Theta_{\text{in}}(s)}=\frac{\dfrac{KK_iK_s}{R_a}}{(\tau_e s+1)\left\{Js^2+\left(B_m+\dfrac{K_bK_i}{R_a}\right)s+\dfrac{KK_iK_s}{R_a}\right\}} \tag{6-57}$$

where K_s is the sensor gain (for the sake of argument let's use a potentiometer in this case), and, as before, the motor electrical time constant $\tau_e = (L_a/R_a)$ may be neglected for small L_a. As a result the position transfer function is simplified to

$$\frac{\Theta_m(s)}{\Theta_{\text{in}}(s)}=\frac{\dfrac{KK_iK_s}{R_aJ}}{s^2+\left(\dfrac{R_aB_m+K_iK_b}{R_aJ}\right)s+\dfrac{KK_iK_s}{R_aJ}}=\frac{\omega_n^2}{\left(s^2+2\zeta\omega_n s+\omega_n^2\right)} \tag{6-58}$$

where Eq. (6-58) is a second-order system, and

$$2\zeta\omega_n=\frac{R_aB_m+K_iK_b}{R_aJ} \tag{6-59}$$

$$\omega_n^2=\frac{KK_iK_s}{R_aJ} \tag{6-60}$$

As a result, for a step input $\Theta_{\text{in}}(s) = 1/s$, the position response of the system will be the same as that of a prototype second-order system, as shown in Chap. 3 (Fig. 3-11). For a given motor, all parameters are known, and the only varying term is the amplifier gain K—the controller gain. Upon varying K, we can directly change ω_n and indirectly change ζ to achieve a desired response. For a positive K, regardless of the type of response (e.g., critically damped or underdamped), the final value of the system is $\theta_{fv} = 1$, which implies that the output will follow the input (recall we used a unit step input). Hence, the position will not increase as in the uncontrolled system that is represented by Eq. (6-56).

Later, in Chaps. 7 and 8, we set up numerical and experimental case studies to test and verify the preceding concepts and learn more about other practical issues.

6-5 CASE STUDIES: PRACTICAL EXAMPLES

EXAMPLE 6-5-1 Modeling of a Sun-Seeker Control System

In this case study, we model a sun-seeker control system whose purpose is to control the attitude of a space vehicle so that it will track the sun with high accuracy. In the system described here, tracking the sun in only one plane is accomplished. A schematic diagram of the system is shown in Fig. 6-38. The principal elements of the error discriminator are two small rectangular silicon photovoltaic cells mounted behind a rectangular slit in an enclosure. The cells are mounted in such a way that when the sensor is pointed at the sun, a beam of light from the slit overlaps both cells. Silicon cells are used as current sources and connected in opposite polarity to the input of an op-amp. Any difference in the short-circuit current of the two cells is sensed and amplified by the op-amp. Because the current of each cell is proportional to the illumination on the cell, an error signal will be present at the output of the amplifier when the light from the slit is not precisely centered on the cells. This error voltage, when fed to the servo-amplifier, will cause the motor to drive the system back into alignment. The description of each part of the system is given in following sections.

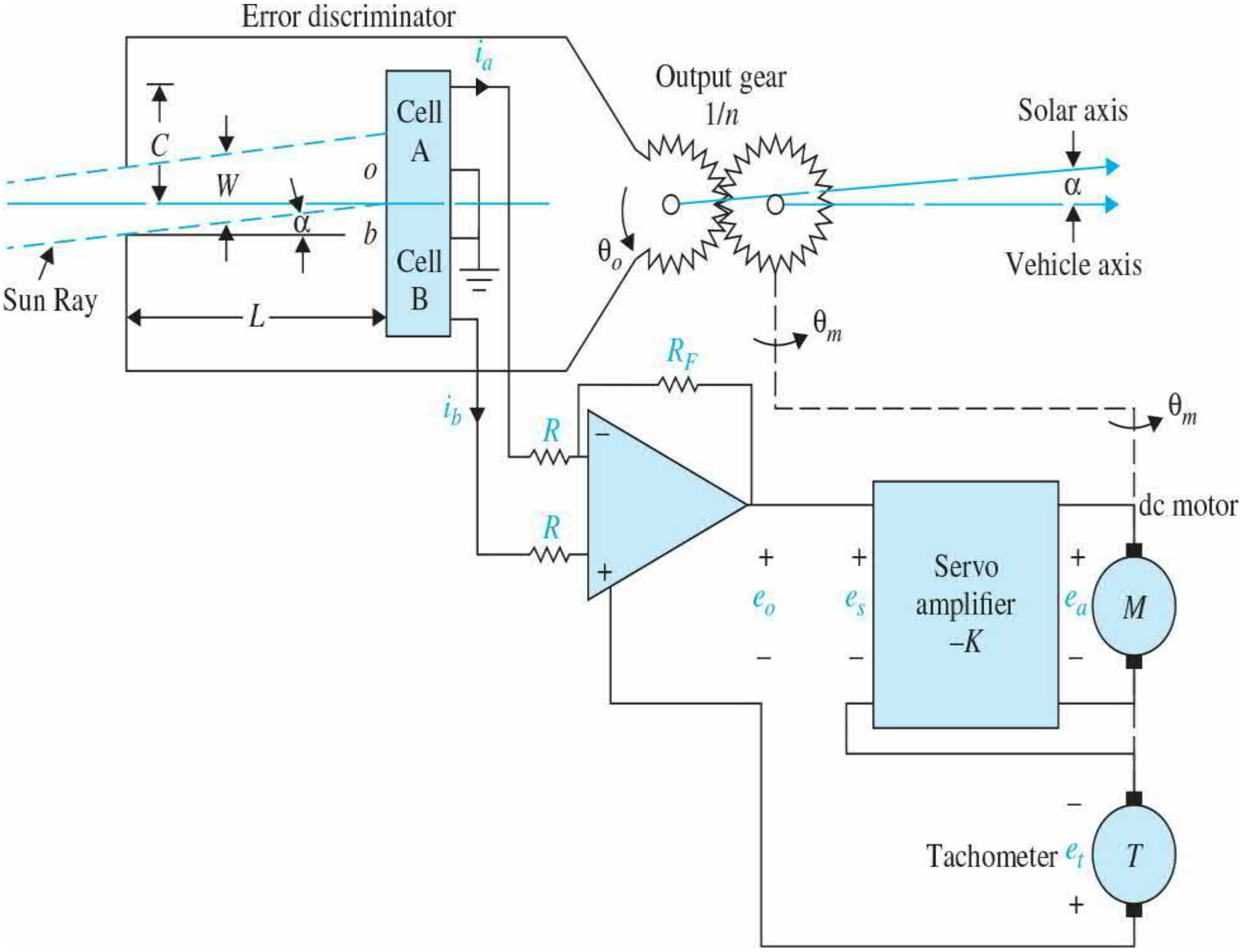


Figure 6-38 Schematic diagram of a sun-seeker system.

Coordinate System The center of the coordinate system is considered to be at the output gear of the system. The reference axis is taken to be the fixed frame of the dc motor, and all rotations are measured with respect to this axis. The solar axis, or the line from the output gear to the sun, makes an angle $\theta_r(t)$ with respect to the reference axis, and $\theta_o(t)$ denotes the vehicle axis with respect to the reference axis. The objective of the control system is to maintain the error between $\theta_r(t)$ and $\theta_o(t)$, $\alpha(t)$, near zero

$$\alpha(t) = \theta_r(t) - \theta_o(t) \tag{6-61}$$

The coordinate system described is illustrated in Fig. 6-39.

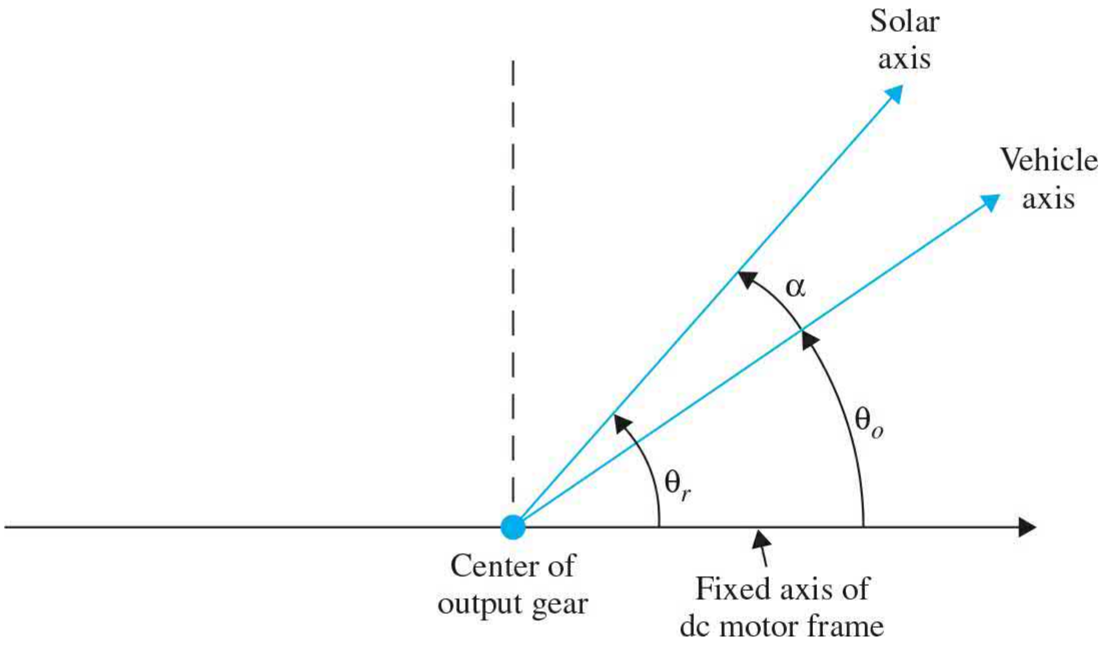


Figure 6-39 Coordinate system of the sun-seeker system.

Error Discriminator When the vehicle is aligned perfectly with the sun, *α(t)* = 0, and $i_o(t) = i_b(t) = I$, or $i_a(t) = i_b(t) = 0$. From the geometry of the sun ray and the photovoltaic cells shown Fig. 6-39, we have

$$oa = \frac{W}{2} + L\tan\alpha(t) \tag{6-62}$$

$$ob = \frac{W}{2} - L\tan\alpha(t) \tag{6-63}$$

where *oa* denotes the width of the sun ray that shines on cell *A* and *ob* is the same on cell *B*, for a given *α(t)*. Because the current $i_a(t)$ is proportional to *oa* and $i_b(t)$ is proportional to *ob*, we have

$$i_a(t) = I + \frac{2LI}{W}\tan\alpha(t) \tag{6-64}$$

$$i_b(t) = I - \frac{2LI}{W}\tan\alpha(t) \tag{6-65}$$

for $0 \le \tan\alpha(t) \le W/2L$. For $W/2L \le \tan\alpha(t) \le (C - W/2)/L$, the sun ray is completely on cell *A*, and $i_a(t) = 2I$, $i_b(t) = 0$. For $(C - W/2)L \le \tan\alpha(t) \le (C +$

W/2)*L*,$i_a(t)$ decreases linearly from 2*I* to zero. $i_a(t) = i_b(t) = 0$ for tan$\alpha(t) \geq (C + W/2)/L$. Therefore, the error discriminator may be represented by the nonlinear characteristic of Fig. 6-40, where for small angle $\alpha(t)$, tan$\alpha(t)$ has been approximated by $\alpha(t)$ on the abscissa.

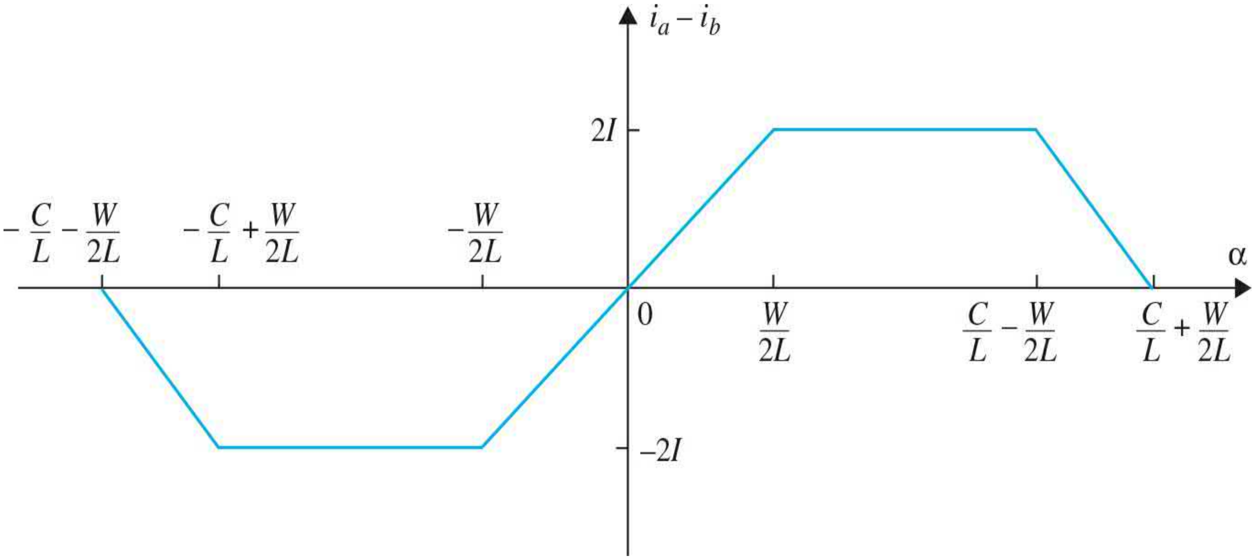


Figure 6-40 Nonlinear characteristic of the error discriminator. The abscissa is tan *α*, but it is approximated by *α* for small values of *α*.

The relationship between the output of the op-amp and the currents $i_a(t)$ and $i_b(t)$ is

$$e_o(t) = -R_F[i_a(t) - i_b(t)] \tag{6-66}$$

Servo-Amplifier The gain of the servo-amplifier is −*K*. With reference to Fig. 6-41, the output of the servo-amplifier is expressed as

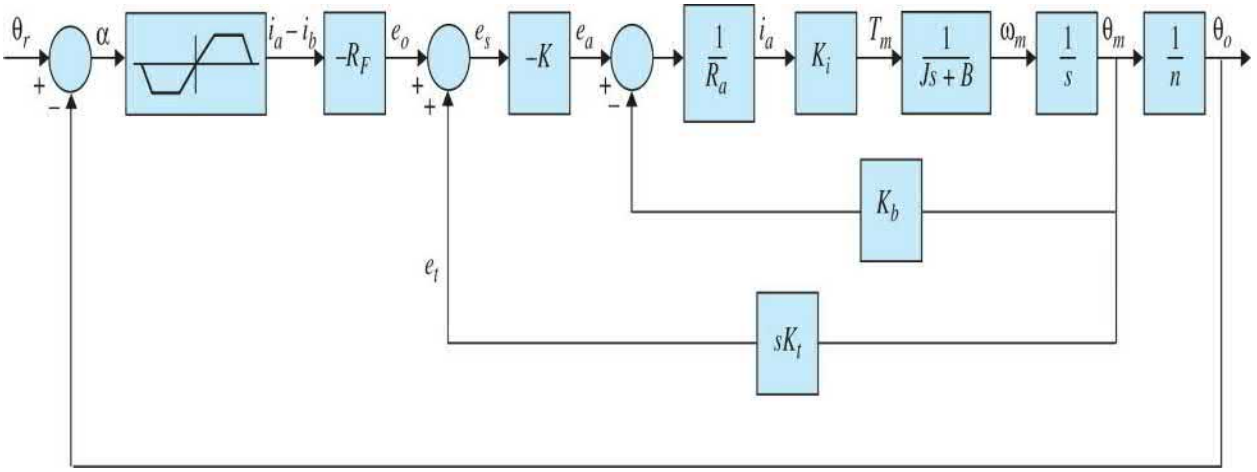


Figure 6-41 Block diagram of the sun-seeker system.

$$e_a(t) = -K[e_o(t) + e_t(t)] = -Ke_s(t) \tag{6-67}$$

Tachometer The output voltage of the tachometer e_t is related to the angular velocity of the motor through the tachometer constant K_t:

$$e_t(t) = K_t\omega_m(t) \tag{6-68}$$

The angular position of the output gear is related to the motor position through the gear ratio 1/*n*. Thus,

$$\theta_o = \frac{1}{n}\theta_m \tag{6-69}$$

DC Motor The dc motor has been modeled in Sec. 6-3. The equations are

$$e_a(t) = R_a i_a(t) + e_b(t) \tag{6-70}$$

$$e_b(t) = K_b\omega_m(t) \tag{6-71}$$

$$T_m(t) = K_i i_a(t) \tag{6-72}$$

$$T_m(t) = J\frac{d\omega_m(t)}{dt} + B\omega_m(t) \tag{6-73}$$

where *J* and *B* are the inertia and viscous-friction coefficient seen at the motor shaft. The inductance of the motor is neglected in Eq. (6-70) because it is assumed to be small—recall discussions on small motor electric time constant in Sec. 6-4-1. A block diagram that characterizes all the functional relations of the system is shown in Fig. 6-41.

EXAMPLE 6-5-2 Modeling of a Quarter-Car Active Suspension System

For the test vehicle shown in Fig. 6-42, a four-post shaker is used to test the performance of the vehicle suspension system by applying various excitations to the vehicle.

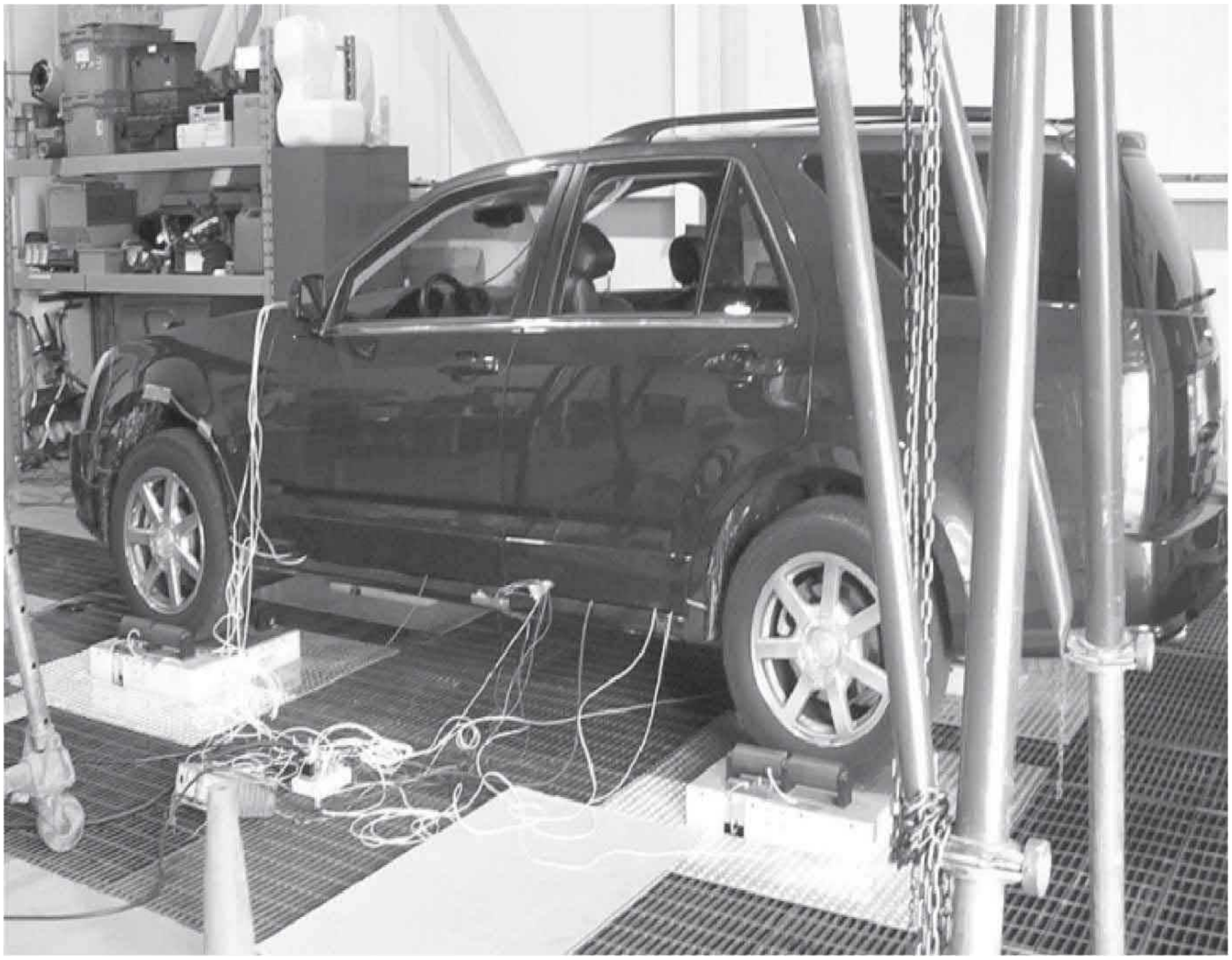


Figure 6-42 A Cadillac SRX 2005 model on a four-post shaker test facility (author’s research on active suspension system).

Classically, the quarter-car model, shown in Fig. 6-43, is used in the study of the vehicle suspension systems and the resulting dynamic response due to various road inputs. Typically, the inertia, stiffness, and damping characteristics of the system as illustrated in Fig. 6-43*a* are modeled as a two-degree of freedom (2-DOF) system, as shown in Fig. 6-43*b*. Although a 2-DOF system is a more accurate model, it is sufficient for the following analysis to assume a 1-DOF model, as shown in Fig. 6-43*c*.

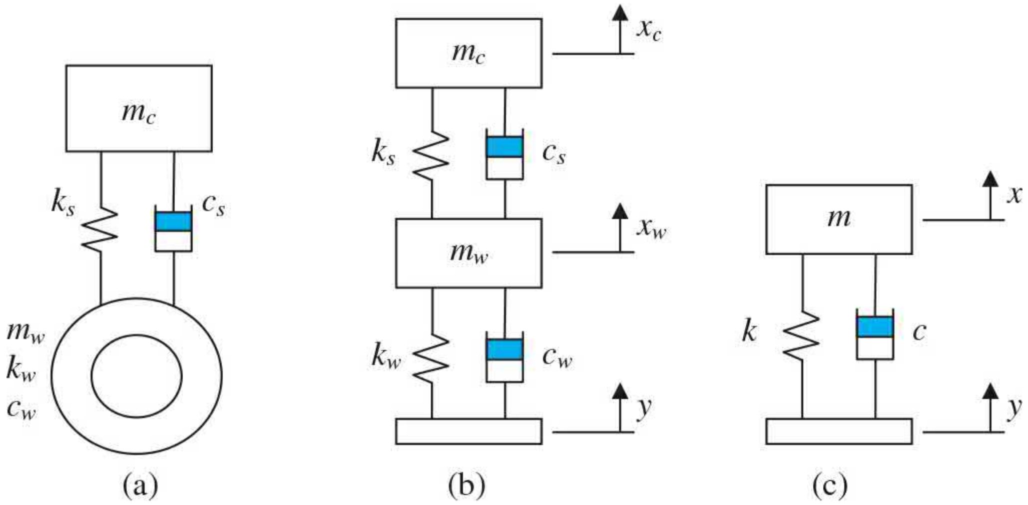
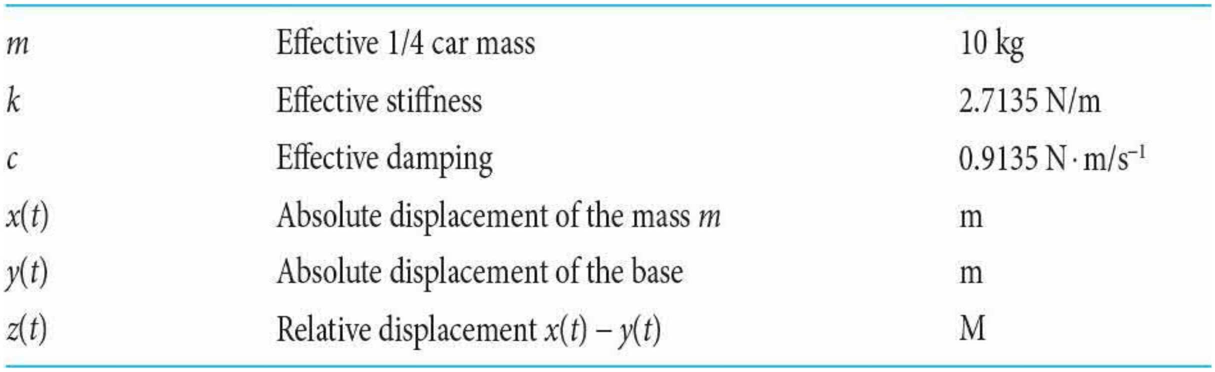


Figure 6-43 Quarter-car model realization. (*a*) Quarter car. (*b*) Two degrees of freedom. (*c*) One degree of freedom.

Open-Loop Base Excitation Given the simplified system representing a miniaturized vehicle model suspension, illustrated in Fig. 6-43*c*, where



m	Effective 1/4 car mass	10 kg
k	Effective stiffness	2.7135 N/m
c	Effective damping	$0.9135\ \mathrm{N\cdot m/s^{-1}}$
$x(t)$	Absolute displacement of the mass m	m
$y(t)$	Absolute displacement of the base	m
$z(t)$	Relative displacement $x(t)-y(t)$	M

the equation of motion of the system is defined as follows:

$$m\ddot{x}(t)+c\dot{x}(t)+kx(t)=c\dot{y}(t)+ky(t) \tag{6-74}$$

which can be simplified by substituting the relation $z(t) = x(t) - y(t)$ and non-dimensionalizing the coefficients to the form

$$\ddot{z}(t)+2\zeta\omega_n\dot{z}(t)+\omega_n^2 z(t)=-\ddot{y}(t)=-a(t) \tag{6-75}$$

The Laplace transform of Eq. (6-75) yields the input–output relationship

$$\frac{Z(s)}{A(s)}=\frac{-1}{s^2+2\zeta\omega_n s+\omega_n^2} \tag{6-76}$$

where the base acceleration $A(s)$ is the Laplace transform of $a(t)$ and is the input, and relative displacement $Z(s)$ is the output. In this case, initial conditions are assumed to be zero.

Closed-Loop Position Control Active control of the suspension system is to be achieved using the same dc motor described in Sec. 6-4-3 used in conjunction with a rack as shown in Fig. 6-43.

In Fig. 6-44, *T(t)* is the torque produced by the motor with shaft rotation θ, and r is the radius of the motor drive gear. Thus, Eq. (6-74) is rewritten to include the active component, $F(t)$,

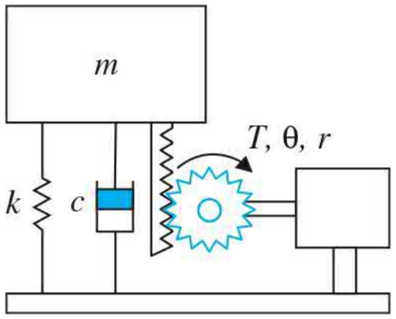


Figure 6-44 Active control of the 1-DOF model via a dc motor and rack.

$$m\ddot{x}+c\dot{x}+kx=c\dot{y}+ky+F(t) \tag{6-77}$$

where

$$m\ddot{z}+c\dot{z}+kz=F(t)-m\ddot{y}=F(t)-ma(t) \tag{6-78a}$$

$$F(t)=\frac{T(t)-(J_m\ddot{\theta}+B_m\dot{\theta})}{r} \tag{6-78b}$$

Because $z = \theta r$, we can substitute Eq. (6-78) into Eq. (6-77), rearrange, and take the Laplace transform to get

$$Z(s) = \frac{r}{(mr^2 + J_m)s^2 + (cr^2 + B_m)s + kr^2}[T(s) - mrA(s)] \tag{6-79}$$

Noting that the term *mrA(s)* is interpreted as a disturbance torque.

The motor equations from Sec. 6-3 are

$$I_a(s) = \frac{1}{L_a}E_a(s) - \frac{R_a}{L_a}I_a(s) - \frac{1}{L_a}E_b(s) \tag{6-80}$$

$$T(s) = K_i I_a(s) \tag{6-81}$$

$$E_b(t) = K_b \frac{Z(s)}{r} \tag{6-82}$$

Using $J = mr^2 + J_m$, $B = cr^2 + B_m$, and $K = kr^2$ in Eq. (6-79), and combining the resulting equation with Eqs. (6-80) through (82), the transfer function of the overall system may be obtained for the applied motor voltage $E_a(s)$ and the disturbance torque. The new equation of the system is in the following form:

$$Z(s) = \frac{\frac{K_i r}{R_a}}{\left(\frac{L_a}{R_a}s + 1\right)(Js^2 + Bs + K) + \frac{K_i K_b}{R_a}s}E_a(s) - \frac{\left(\frac{L_a}{R_a}s + 1\right)r}{\left(\frac{L_a}{R_a}s + 1\right)(Js^2 + Bs + K) + \frac{K_i K_b}{R_a}s}mrA(s) \tag{6-83}$$

In order to control the bounce of the car, we need to sense *z(t)* using a position sensor, for example, string potentiometer with a gain K_s. In that case, addition of the sensor and a controller—in this case a proportional controller with a gain *K* as in Sec. 6-4-3—will create a feedback control system as shown in Fig. 6-45. In this case, the input $Z_{in}(s)$ reflects the desired bounce level—normally $z_{in}(t) = 0$ for the control objective of having no bounce in response to disturbances from the road.

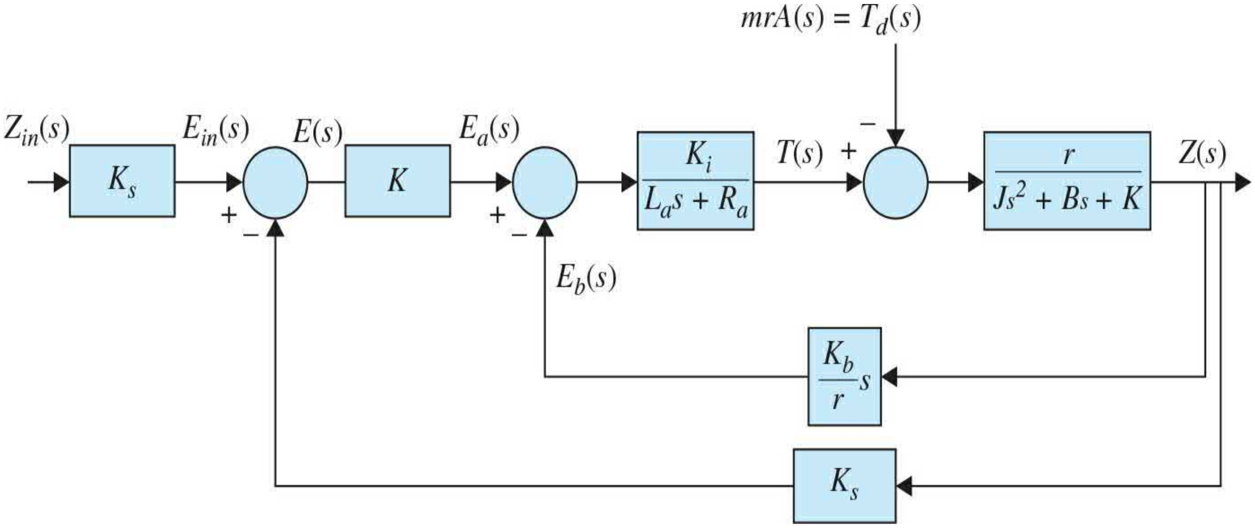


Figure 6-45 Block diagram of the active feedback control quarter-car system, for bounce control.

6-6 THE CONTROL LAB: INTRODUCTION TO LEGO MINDSTORMS NXT MOTOR—MODELING AND CHARACTERIZATION

This section offers a practical approach to modeling and characterization of a dc motor, without any reliance on manufacturer datasheets. It intends to provide an understanding and appreciation of where each term, discussed earlier in this chapter, comes from, and how to experimentally obtain or verify their value.

In this section, we continue our work with the project described earlier in Sec. 2-6 (and in detail in App. D), and experimentally characterize the motor parameters to arrive at an accurate model of the overall electrometrical system.

6-6-1 NXT Motor

The NXT motor used in this project is a 9-V dc motor specific to the LEGO MINDSTORMS NXT set. As shown in Fig. 6-46, the motor contains a **gear-train** from the motor to the output shaft to provide greater torque.

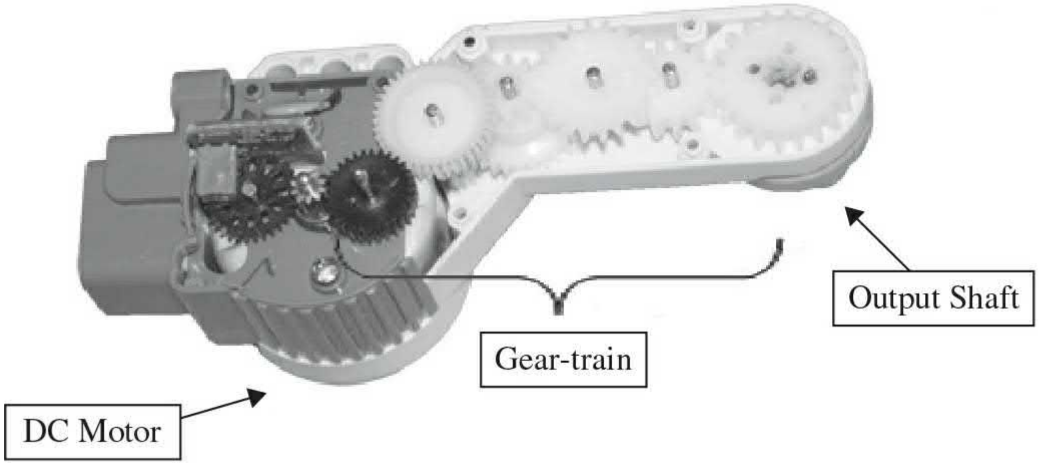
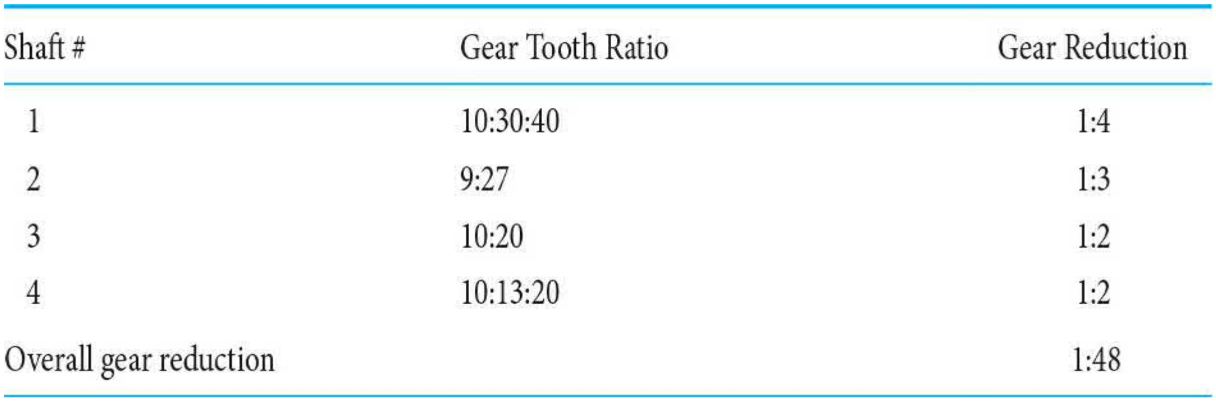


Figure 6-46 NXT motor internals.

The overall gear ratio calculation from motor to output shaft is shown in Table 6-2. In this case, as discussed in App. D, because *the encoder measures the rotational position of the output shaft, and not that of the motor shaft, the gear ratio and its model may be absorbed into the motor model*—see earlier discussions in this chapter for details. As a result, from this point forward, any reference to the motor implicitly refers to the motor-gear-train system. Further, all parameters obtained will be for the motor-gear-train combination.

TABLE 6-2 NXT Motor to Output Shaft Gear Reductions



Shaft #	Gear Tooth Ratio	Gear Reduction
1	10:30:40	1:4
2	9:27	1:3
3	10:20	1:2
4	10:13:20	1:2
Overall gear reduction		1:48

The gear ratio and its model may be absorbed into the motor model.

Note: All parameters obtained will be for the motor-gear-train combination.

In order to fully model the motor, both the electrical and mechanical characteristics of the motor are obtained experimentally.

6-6-2 Electrical Characteristics[1]

The electrical characteristics required to model the motor are its armature resistance and armature inductance. The following sections will provide you with tests that can be done on the motor to measure its electrical characteristics (see App. D for more details).

Armature Resistance

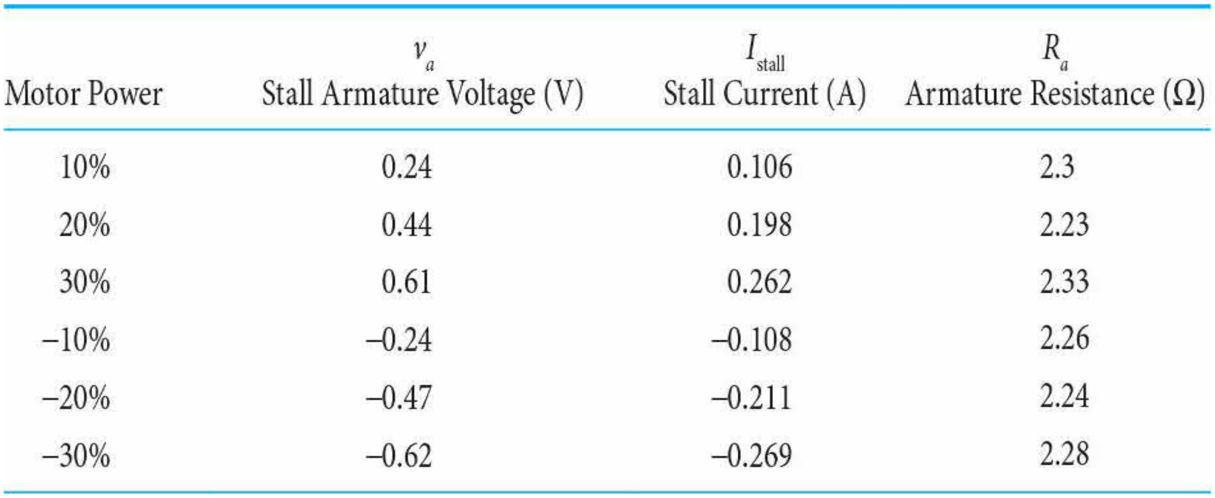
To begin, we must measure the armature current using a multimeter. Using the procedure as described in App. D. You will notice the current will drastically increase when the motor is stalled. Record the **stall current**, I_{stall}, that is reported by the multimeter and repeat for various trials. After measuring the stall current, the armature voltage must also be measured with the multimeter. Next calculate the experimental armature resistance, as suggested in App. D, using

$$R_a = \frac{v_a}{I_{\text{stall}}} \qquad (6\text{-}84)$$

where v_a is the voltage measured when the motor is stalled.

Experimental data for the NXT motor for various input power values (as a percentage of maximum power) are shown in Table 6-3. Applying Eq. (6-84), the average experimental armature resistance of the NXT motor was found to be $R_a = 2.27\ \Omega$.

TABLE 6-3 Armature Resistance Measurement Experimental Data



Motor Power	v_a Stall Armature Voltage (V)	I_{stall} Stall Current (A)	R_a Armature Resistance (Ω)
10%	0.24	0.106	2.3
20%	0.44	0.198	2.23
30%	0.61	0.262	2.33
−10%	−0.24	−0.108	2.26
−20%	−0.47	−0.211	2.24
−30%	−0.62	−0.269	2.28

Armature Inductance

There are a number of ways to measure the motor inductance. One way commonly practiced in most undergraduate control labs is to connect a known resistor R (select a value close to R_a) in series with the motor, stall the motor as in the previous section, provide the system with a constant input voltage, turn off the input and measure the electric time constant $L_a/(R + R_a)$. Knowing the time constant and resistance values, you can calculate L_a. We took the easy way out by using a multimeter that is able to measure inductance. Simply connect the multimeter to the motor terminals, and set the multimeter to measure inductance, as shown in Fig. 6-47. The experimentally measured armature inductance was found to be $L_a = 4.7$ mH.

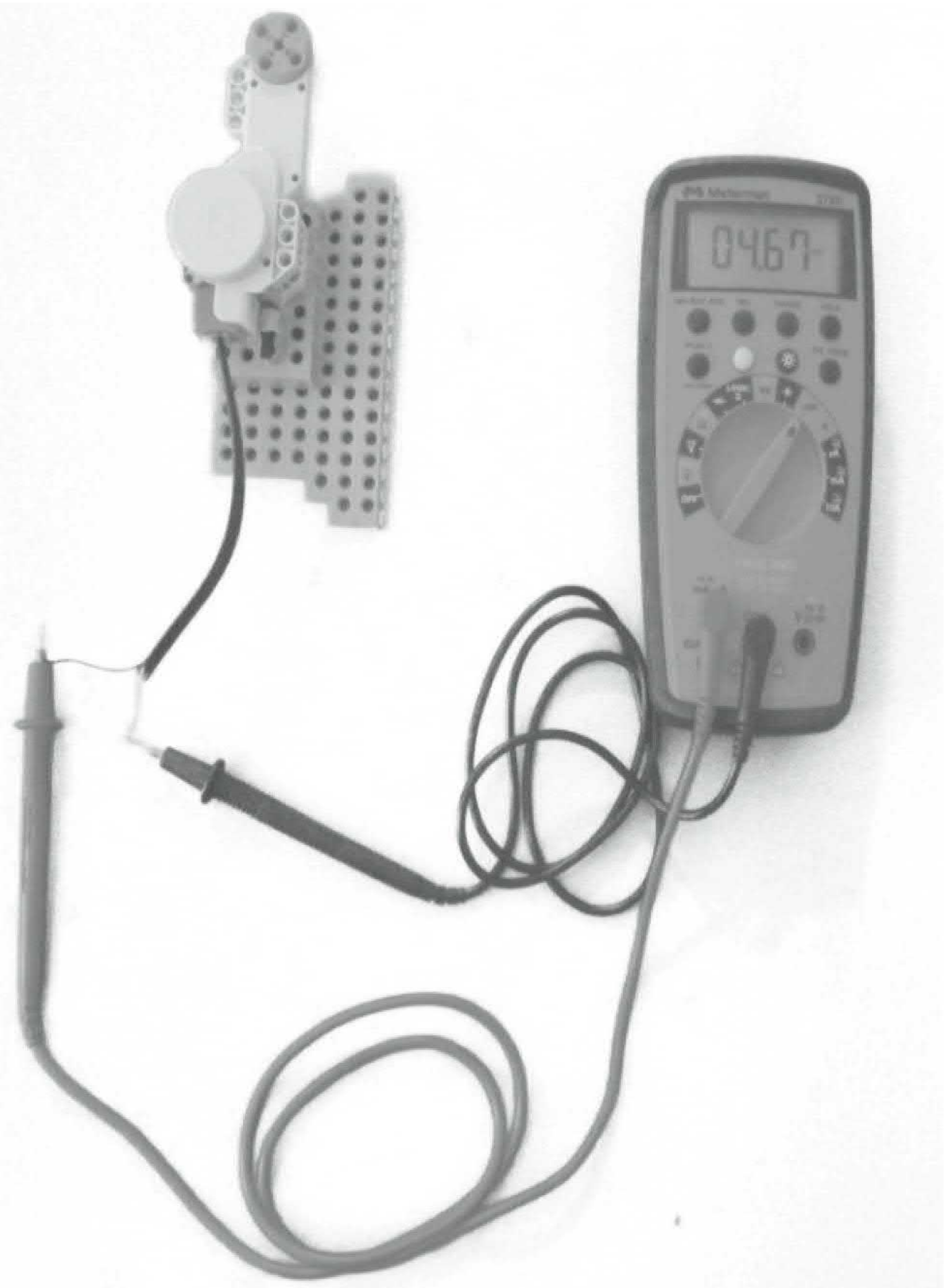

Meterman
04.67

Figure 6-47 Motor direct inductance measurement with a multimeter.

6-6-3 Mechanical Characteristics[2]

The mechanical characteristics required to model the motor are the torque constant, the back-emf constant, viscous-friction coefficient (recall from Chap. 2 that for simplicity we assume all friction is modeled as viscous damping), armature and load moments of inertia, and system mechanical time constant. The following sections will provide you with tests that you can use to measure the NXT motor mechanical characteristics.

Motor Torque Constant

As discussed in Sec. 6-3, the motor torque constant K_i is obtained from

$$T_m = K_i i_a \tag{6-85}$$

where T_m is the motor torque and i_a is the armature current. Determining the torque constant experimentally requires you to measure both the current supplied to the motor and the torque that it provides.

Start by attaching a shaft and spool to the end of the motor as illustrated in Fig. 6-48 as a pulley. Finally, attach a weight with known mass to the end of the thread; this will act as the external torque that the motor has to overcome in order to rotate. The torque can be calculated using

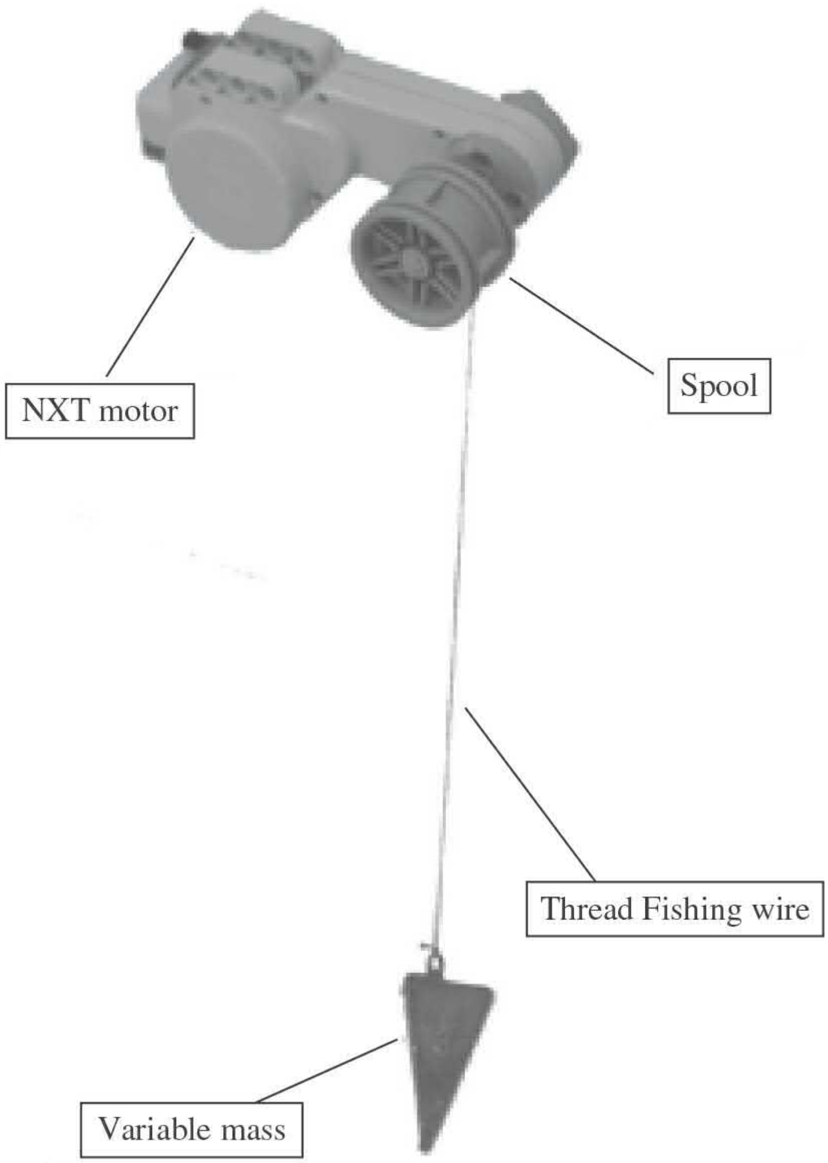


Figure 6-48 Torque constant test setup.

$$T = T_m - T_W = K_i i_a - r_{\text{spool}} W \tag{6-86}$$

where r_{spool} is the radius of the spool and W is the weight corresponding to mass M. The motor will stall when $T = 0$. The corresponding T_m and i_a are motor stall torque and current, respectively.

In this experiment, the motor rotation results in the mass **M** to move up or down. You will be required to measure the current supplied to the motor for a variable mass—recall the earlier instructions on how to measure the motor current form Sec. 6-6-2. Start by applying an input to the motor and let the mass rise to the top. While the motor is pulling the mass upward, measure the current supplied to the motor using a multimeter. Repeat this process for various masses and plot the experimental torque T_m from Eq. (6-85) against the measured current. It should be noted that K_i is independent of the input voltage value. Table 6-4 shows some of the measurements for our experiment. Note that for $T_w = 0$ N · m, $i_a = 0.041$ A, which is the current in the motor to overcome the internal motor friction. This value can later be used to calculate the motor damping parameter. Also note that for mass M = 0.874 kg the motor **stalls and the corresponding stall torque is $T_{stall} = T_w =$ 0.116 N · m**. The experimental motor torque curve for our NXT motor is shown in Fig. 6-49, where the experimental motor torque curve is extrapolated from the data points using the linear regression tool in MATLAB. The NXT experimentally measured motor constant is the inverse of the slope (3.95 A/N · m) or $K_i = 0.252$ N · m/A.

TABLE 6-4 Motor Torque Constant Measurement Experiment (r_{spool} = 0.013575 m)

Trial	M Mass Attached to String (kg)	W Weight (N)	$T_W = r_{\text{spool}} W$ Torque of Motor to Overcome Mass (N · m)	i_a Motor Current (A)
1	0	0	0	0.041
2	0.067	0.6566	0.008913	0.085
3	0.119	1.1662	0.015831	0.113
4	0.173	1.6954	0.023015	0.137
5	0.234	2.2932	0.03113	0.161
6	0.293	2.8714	0.038979	0.187
7	0.353	3.4594	0.046961	0.224
8	0.413	4.0474	0.054943	0.245
9	0.471	4.6158	0.062659	0.283
10	0.532	5.2136	0.070775	0.322
11	0.643	6.3014	0.085542	0.387
12	0.702	6.8796	0.093391	0.435
13	0.874	8.5654	0.116275	0.465 (stall)

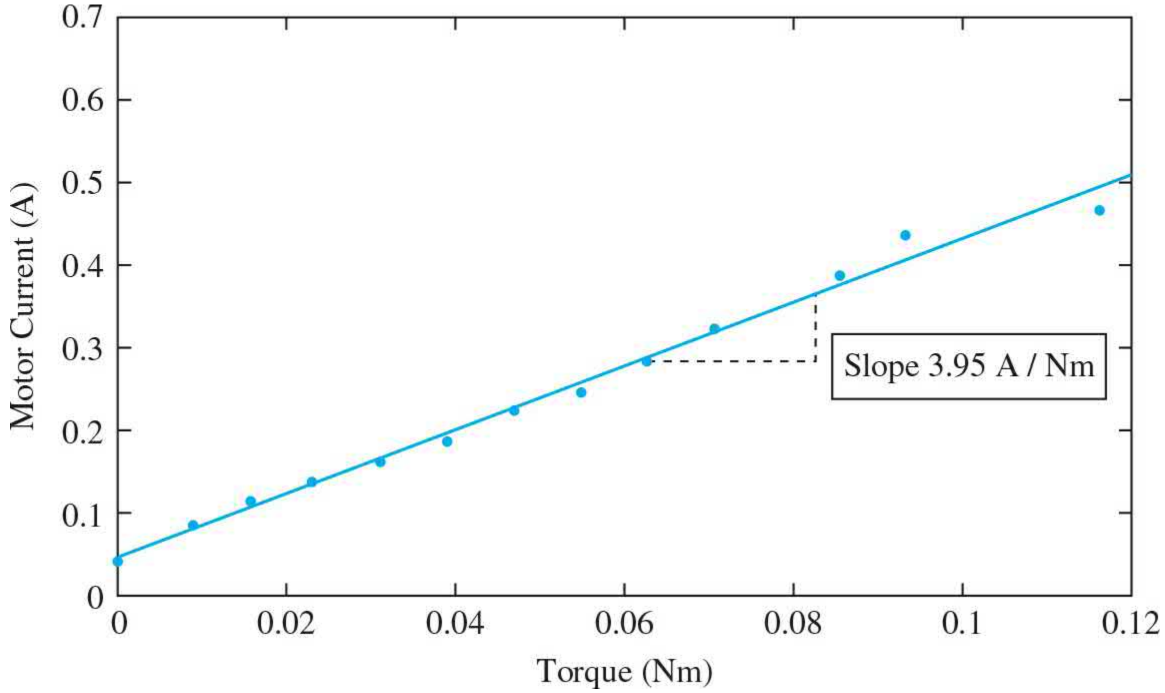


Figure 6-49 Experiment motor torque T_w versus current curve for NXT motor to calculate K_i.

Back-emf Constant

As discussed earlier in this chapter, the back-emf constant is obtained from

$$e_b = K_b \omega_m \tag{6-87}$$

where e_b is the back-emf or motor voltage and ω_m is the motor angular speed. To measure the motor back-emf constant, you will be required to test the motor's open-loop speed response using Simulink as well as measuring the supplied voltage using a multimeter (see App. D).

A sample open-loop step response for a 2.0-V step input is shown in Fig. 6-50. You will observe that there is noise visible in the output. The noise is a result of differentiation of the low resolution position signal from the encoder to find speed, and it is also attributed to the gear backlash. As a result, you need to record the average steady-state speed for various step inputs and record the steady-state speed and armature voltage for each trial. Finally, plot the experimental armature voltage against the measured steady-state average speeds. A sample plot is shown in Fig. 6-51, where the experimental data points are shown as well as an extrapolated trend line found using the linear regression tool in MATLAB. The slope of this line is the back-emf constant of the motor. The back-emf constant of the NXT motor was experimentally measured to be K_b = 0.249 V/rad/s.

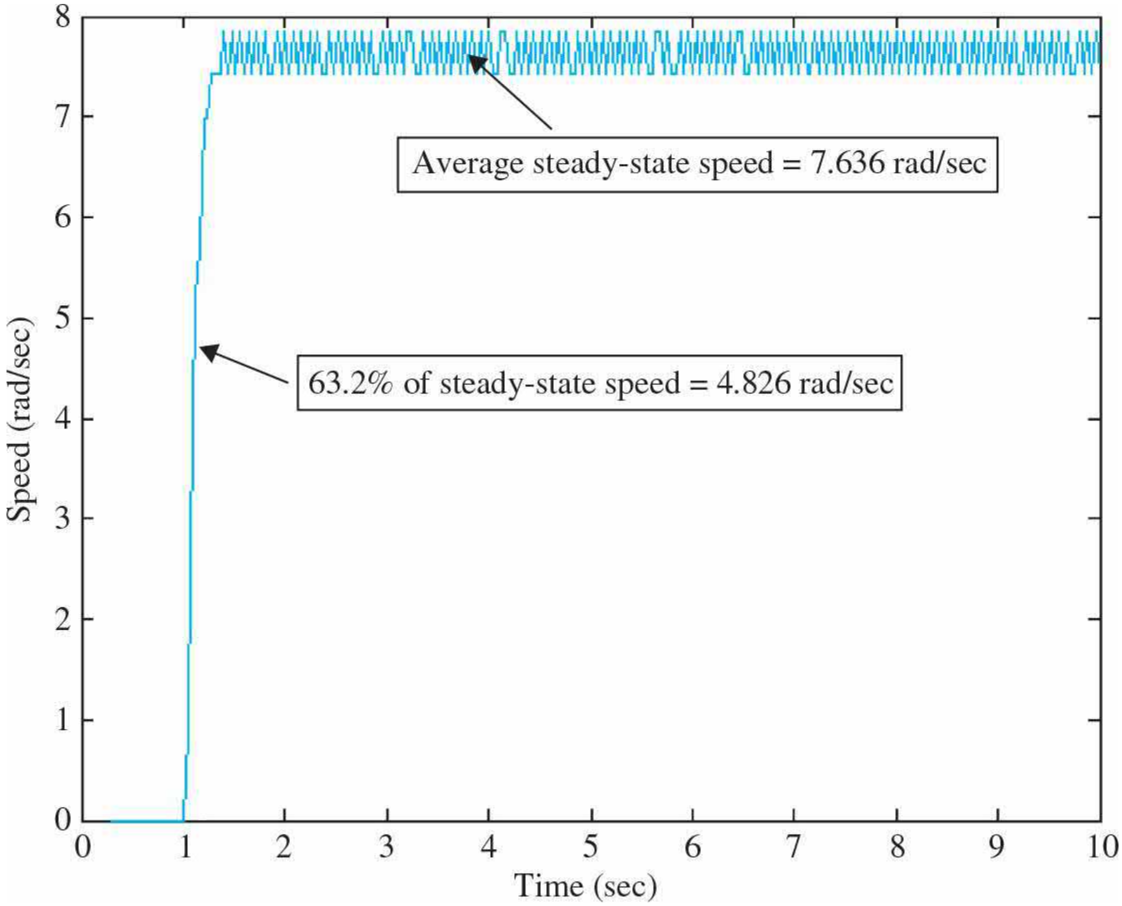


Figure 6-50 Open-loop speed response for 2.0-V input (50% power).

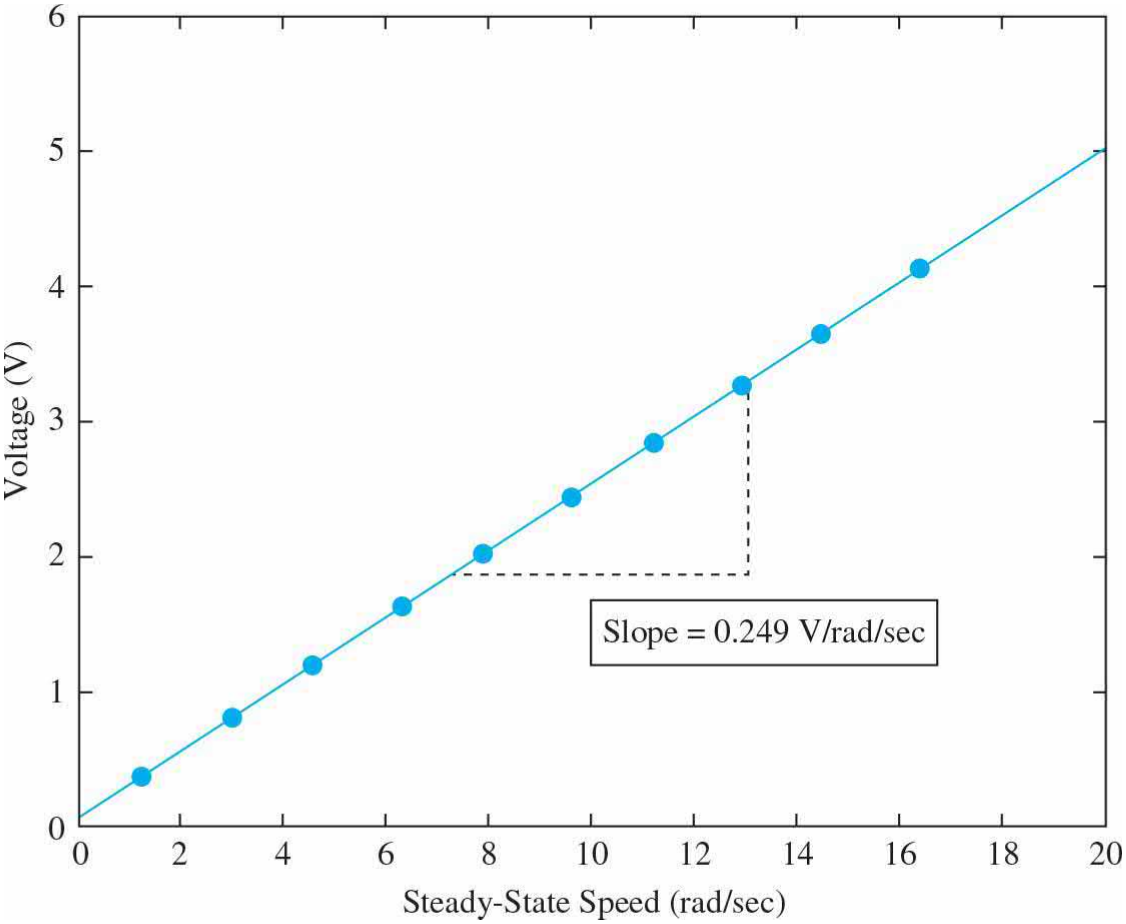


Figure 6-51 Voltage versus steady-state speed.

It should be noted that, *in an ideal case, the back-emf constant and motor torque constant are numerically equal in the SI units*. However, because these values have been experimentally measured, their corresponding experimentally measured values are close, but not equal. To equate the two constants, we can find the average value that exists between them so that both K_b and K_i are numerically equal. The average value between $K_b = 0.249$ V/rad/s and $K_i = 0.252$ N · m/A is 0.25; therefore, this average value will be used for both K_b and K_i.

Note: All parameters obtained will be for the motor-gear-train combination.

Viscous-Friction Coefficient

The viscous-friction coefficient describes the amount of friction that exists in the system. In reality, the friction may not be viscous. However, as discussed in Chap. 2, this is an assumption that we make to arrive at an approximate linear model for the motor-gear-train combination—again any reference to motor implicitly refers to the motor-gear-train combination. **Important note:** Because of various nonlinear affects such as friction and gear backlash, it is not expected that this parameter can be estimated accurately.

Using small inductance assumption, that is, $L_a \simeq 0$, we can arrive at the effective damping due the electrical and mechanical components. From Sec. 6-3-6, the speed response of the motor in Fig. 6-48 (assuming negligible spool inertia) is

$$J_m \frac{d\omega(t)}{dt} + \left(B_m + \frac{K_i K_b}{R_a} \right) \omega(t) = \frac{e_a(t) K_i}{R_a} - T_w \tag{6-88}$$

where B_m is the viscous-friction coefficient, R_a is the motor armature resistance, K_i is the motor torque constant and K_b is the back-emf constant.

Next, using the formula for steady-state speed,

$$\omega_{fv} = \lim_{t \to \infty} \omega(t) = \frac{2K_i}{K_i K_b + R_a B_m} \tag{6-89}$$

We can calculate B_m experimentally, using

$$\begin{aligned} B_m &= \left(\frac{2K_i}{\omega_{fv}} - K_i K_b \right) \left(\frac{1}{R_a} \right) \\ &= \left(\frac{2(0.25)}{7.636} - (0.25)^2 \right) \left(\frac{1}{2.27} \right) = 1.31 \times 10^{-3} \ \ \text{N} \cdot \text{m/s} \end{aligned} \tag{6-90}$$

To measure the viscous-friction coefficient, apply a step input—in this case 2 V—to the motor, using the procedure outlined earlier in back-emf constant measurement section, and observe the open-loop speed response, as shown in Fig. 6-50. Record the steady-state speed—in this case 7.636 rad/s—

and substitute the values in Eq. (6-90), as shown.

Alternatively, using the mechanical equation of the motor, and replacing the motor torque with armature current using Eq. (6-87), we have

$$J_m \frac{d\omega(t)}{dt} + B_m \omega(t) = K_i i_a - T_W \tag{6-91}$$

Hence, the viscous-friction coefficient at steady-state angular speed can also be measured using

$$B_m = \frac{K_i}{\omega_{fv}} i_a - \frac{T_W}{\omega_{fv}} \tag{6-92}$$

Using the no-load case ($T_w = 0$), and from Table 6-4 the no-load armature current $i_a = 0.041$ A, we can calculate B_m experimentally to be

$$B_m = \frac{K_i}{\omega_{fv}} i_a = \frac{(0.41)(.25)}{7.636} = 1.34 \times 10^{-3} \text{ N} \cdot \text{m/s} \tag{6-93}$$

The no-load value of current is what the motor needs to overcome internal friction.

You can also obtain the viscous-friction coefficient for different load torque values using Eq. (6-93) and Table 6-4, provided that you have the corresponding steady-state angular speed for each T_w. Constructing the motor speed-torque curve, as in Fig. 6-52, will help you with this task. You can experimentally obtain this curve using the same procedure as in K_i calculation that was discussed earlier in this section. In Fig. 6-48, the relation between angular speed and torque is

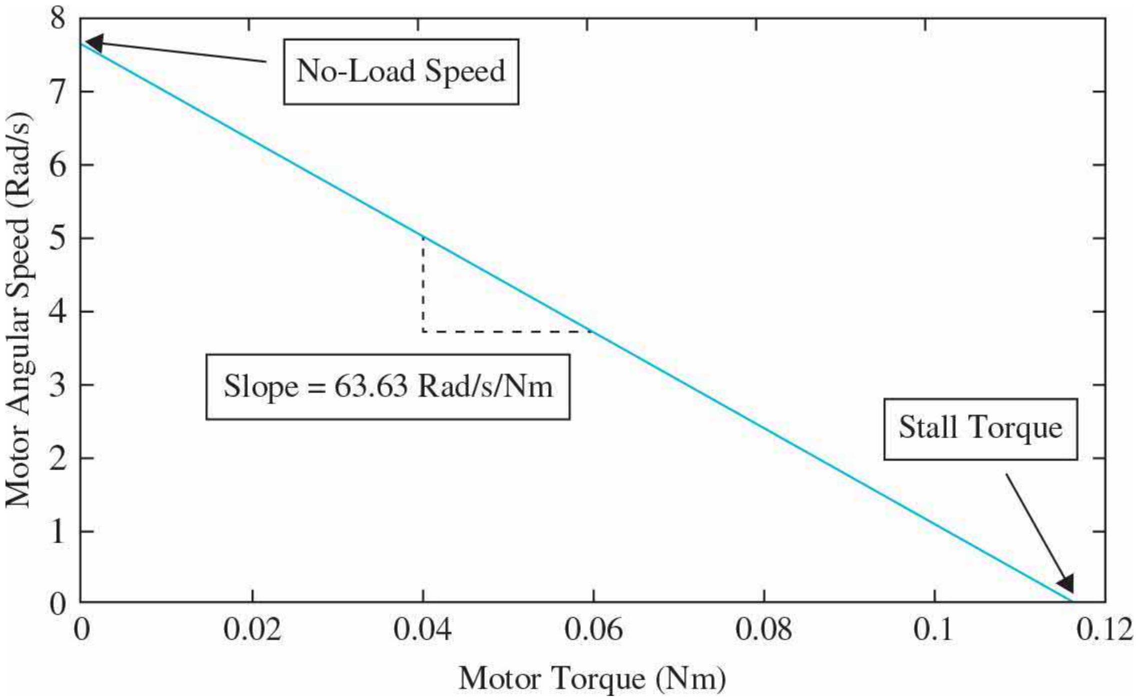


Figure 6-52 Motor speed-torque curve.

$$\begin{aligned}\omega_{fv} &= -63.63T_m + \omega_{fv(\text{noload})} \\ &= -63.63K_i i_a + 7.636 \qquad (6\text{-}94)\end{aligned}$$

From Table 6-4 and Eqs. (6-90) and (6-92), the average value of viscous-friction coefficient $B_m = 1.36 \times 10^{-3}$ N · m/s.

Caution: The value of B_m varies with the power provided to the motor. In our case, we have calculated the viscous damping coefficient at power equal to 50%. Upon application of different percentage of power to the motor, we can find a relationship between viscous-friction coefficient and percentage of motor power. As shown in Fig. 6-53, the viscous-friction coefficient value decreases as the motor power increases. In this project, therefore, the lower value of viscous-friction coefficient of the NXT motor with no load was adopted—that is, $B_m = 1.31 \times 10^{-3}$ N · m/s.

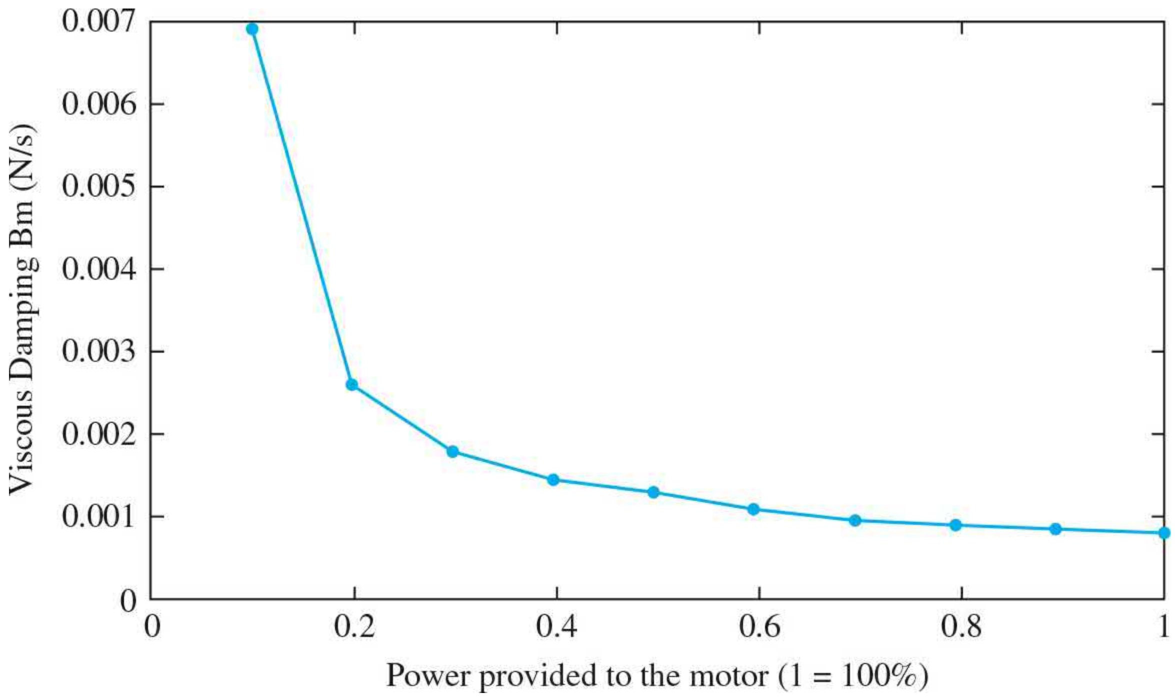


Figure 6-53 Measured damping coefficient as the power supplied to the motor varies from 10 to 100 percent.

The experiment was repeated **with the arm attached to the motor** and the equivalent motor-payload viscous-friction coefficient was experimentally measured using Eqs. (6-90) and (6-93) to be $B = B_m + B_{arm/payload} = 2.7 \times 10^{-3}$ N · m/s. The higher value of B in this case amounts to the higher initial torque required to overcome the higher internal friction associated with moving the arm/payload system. Please note that this measurement required the robot arm to rotate with the payload attached for about ten seconds. In most practical applications, this approach, while simple, may not be feasible (or safe!). An alternative approach may be to use the position control response (see Chap. 7 for position response) to obtain or fine tune the value of B.

Mechanical Time Constant

As discussed earlier in this chapter, the mechanical time constant τ_m is defined as the time required for the motor speed to reach 63.2% of its final value for a step input. To measure the time constant, you will be required to use the open-loop speed response experiment described earlier in back-emf calculations. First, ensure there is no external load connected to the motor. Next, apply a step input to the motor using the model shown in Fig. 6-50 and plot the resulting response. Find the average steady-state speed and calculate 63.2% of the steady-state speed as shown in Fig. 6-50.

In this trial, the average steady-state speed is measured to be 7.636 rad/s while 63.2% of the steady-state speed (to measure the time constant) is calculated to be 4.826 rad/s. The mechanical time constant for the NXT motor with no load was experimentally measured to be $\tau_m = 0.081$ s. The experiment was repeated with the robotic arm and the time constant was experimentally measured to be $\tau_m = 0.10$ s—obviously the slower response is due to a higher inertia added through the payload.

Moment of Inertia

The combination armature-load moment of inertia J_m can be experimentally calculated using

$$J_m = \tau_m \left(B_m + \frac{K_i K_b}{R_a} \right) \tag{6-95}$$

where this equation relates the overall motor-gear-train moment of inertia to the other parameters found in the previous sections. Applying Eq. (6-95) and substituting the parameters found in previous sections, the moment of inertia of the motor-gear-train is experimentally calculated to be $J_m = J_{motor} + J_{gear} = 2.33 \times 10^{-3}$ kg · m^2.

As a final check, for the no-load motor with speed response shown in Fig. 6-50, after reaching the final speed value, turn off the input and record how the speed decays with time to reach zero, as shown in Fig. 6-54. The system equation with power off is

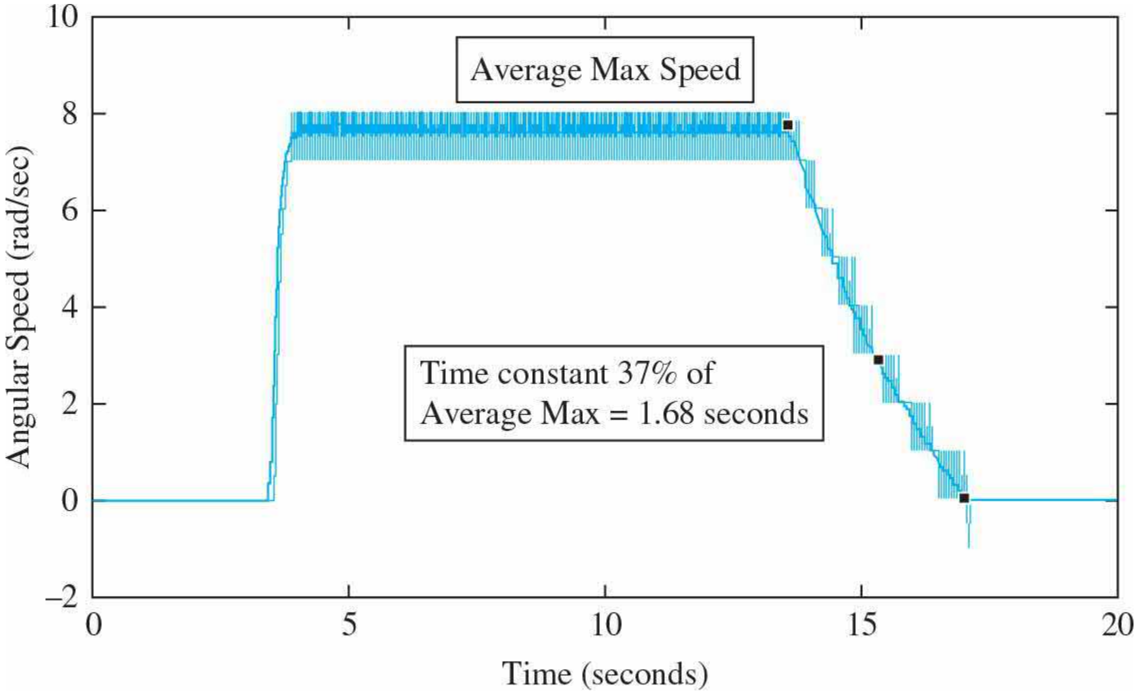


Figure 6-54 Time constant measurement with the motor input off after reaching steady-state response.

$$J_m \frac{d\omega(t)}{dt} + B_m \omega(t) = T_m \tag{6-96}$$

where, the time constant of the system in this case is $\tau = J_m/B_m$. Based on our estimated values of J_m and B_m the system time constant is $\tau = 1.78$ s, which is very close to the measure value of $\tau = 1.68$ s, shown in Fig. 6-54.

As a result, we have a very high confidence on the accuracy of our estimated parameters of the system.

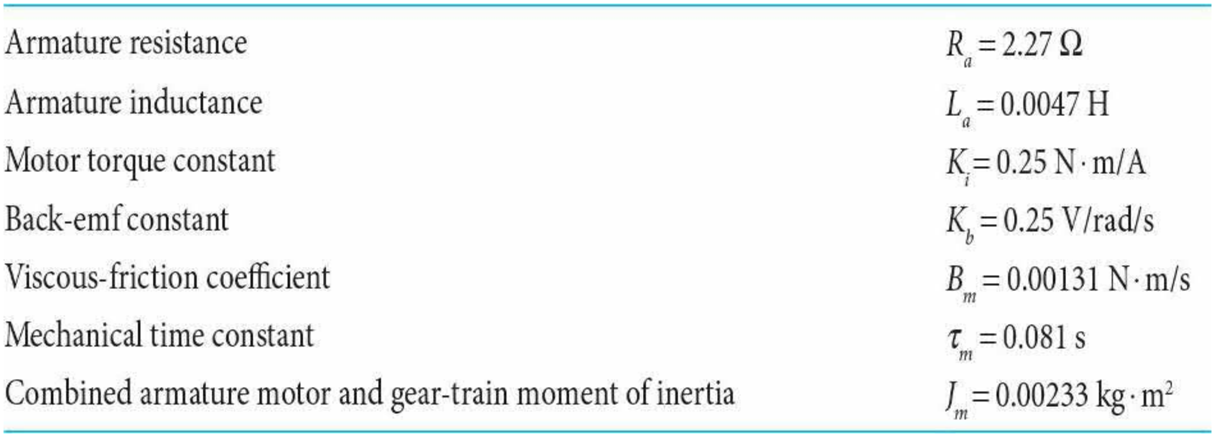
Similarly, the moment of inertia of the motor with the arm attached and the payload is experimentally calculated to be $J_{total} = J_m + J_{gear} + J_{arm/payload} = 3.02 \times 10^{-3}$ kg · m^2. Note the total inertia was obtained using Eq. (8-13) with $\tau_m = 0.10$ and $B = B_m + B_{arm/payload} = 2.7 \times 10^{-3}$.

Alternatively, you can calculate the arm/payload inertia by first identifying the combined mass center using techniques such as parallel axis theorem that you have learned in your second year dynamics course (or through using a CAD software). Then, by measuring the mass of the arm/payload, you can estimate $J_{arm/payload} = M_{arm/payload}\, r^2_{cm}$. This approach assumes the arm/payload system as a point mass M at a distance r_{cm} away from the axis of rotation. You may, however, find this a bit time consuming. It all depends on how much time you plan to spend on finding an approximate model. In practice, *good enough* is good enough!

6-6-4 Speed Response and Model Verification

Now that the motor parameters have been measured, the mathematical model of the speed response system can be developed and fine-tuned by comparing the simulated response to the response of the actual motor (see App. D for the simulation details). Using the parameter values in Table 6-5 and a step input of amplitude 2.0 V starting at time equal to 1 s, the speed response, shown in Fig. 6-55, **closely matches** that of the actual system that was earlier shown in Fig. 6-50.

TABLE 6-5 NXT No-Load Motor Experimental Parameters



Armature resistance	$R_a = 2.27\ \Omega$
Armature inductance	$L_a = 0.0047\ \mathrm{H}$
Motor torque constant	$K_i = 0.25\ \mathrm{N \cdot m/A}$
Back-emf constant	$K_b = 0.25\ \mathrm{V/rad/s}$
Viscous-friction coefficient	$B_m = 0.00131\ \mathrm{N \cdot m/s}$
Mechanical time constant	$\tau_m = 0.081\ \mathrm{s}$
Combined armature motor and gear-train moment of inertia	$J_m = 0.00233\ \mathrm{kg \cdot m^2}$

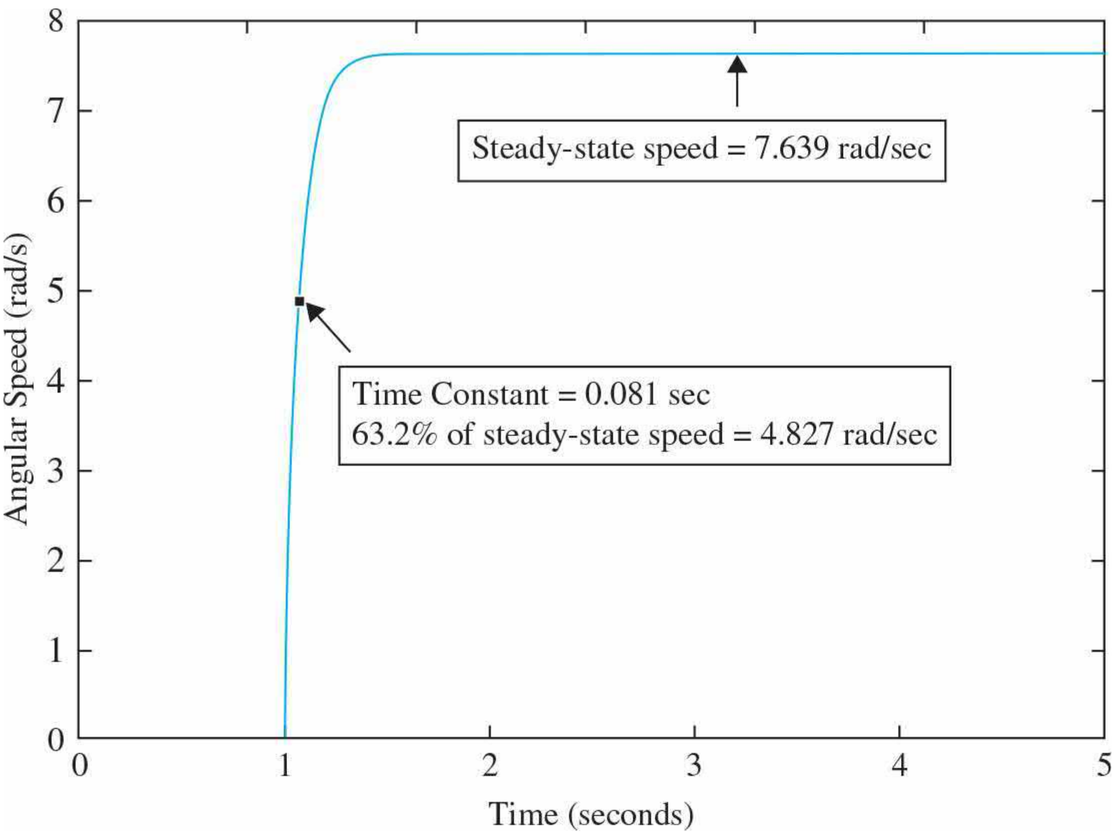


Figure 6-55 Speed response of modeled system for 2.0-V input.

In order to further verify the model with the arm attached, it is recommended that you look at the position control response, discussed in Chap. 7.

6-7 SUMMARY

In a **feedback control system**, sensors are very important to sense various properties of the system, particularly, the output of the plant. A controller can then compare the output signal with the desired objective, or the input, and adjust the performance of the overall system, using an actuator, to meet the desired goal. This chapter is devoted to the mathematical modeling of necessary components that make a feedback control system work. These include **sensors**, **actuators**, and the actual brain of the control system, that is, the **controller**. Of particular interest are components that possess **linear models—**or at least as close to linear as possible. For linear systems, differential equations, state equations, and transfer functions are the fundamental tools of modeling.

In this chapter, we used **dc motors** as actuators—because of their simplicity of models and high usage in the field. We also looked at sensors that may be used to quantify the motion of dc motors, namely, **encoders**, **tachometers,** and **potentiometers**. In this chapter, we also learned about **op-amps** and their role as building blocks of any control system.

We also discussed the concepts of speed and position response of dc motors and introduced speed and position control of dc motors. In the end, case study examples were presented that reflect mathematical modeling and motor parameter estimation in practical applications.

Upon successful completion of this chapter, you are now able to understand how to model a complete control system and its individual components, and further understand how these components are related and interact with one another.

REFERENCES

1. W. J. Palm III, *Modeling, Analysis, and Control of Dynamic Systems*, 2nd Ed., John Wiley & Sons, New York, 1999.
2. K. Ogata, *Modern Control Engineering*, 4th Ed., Prentice Hall, NJ, 2002.
3. I. Cochin and W. Cadwallender, *Analysis and Design of Dynamic Systems*, 3rd Ed., Addison-Wesley, 1997.
4. A. Esposito, *Fluid Power with Applications*, 5th Ed., Prentice Hall, NJ, 2000.
5. H. V. Vu and R. S. Esfandiari, *Dynamic Systems*, Irwin/McGraw-Hill, 1997.
6. J. L. Shearer, B. T. Kulakowski, and J. F. Gardner, *Dynamic Modeling and Control of Engineering Systems*, 2nd Ed., Prentice Hall, NJ, 1997.
7. R. L. Woods and K. L. Lawrence, *Modeling and Simulation of Dynamic Systems*, Prentice Hall, NJ, 1997.
8. E. J. Kennedy, *Operational Amplifier Circuits*, Holt, Rinehart and Winston, Fort Worth, TX, 1988.
9. J. V. Wait, L. P., Huelsman, and G. A. Korn, *Introduction to Operational Amplifier Theory and Applications*, 2nd Ed., McGraw-Hill, New

York, 1992.

10. B. C. Kuo, *Automatic Control Systems*, 7th Ed., Prentice Hall, NJ, 1995.

11. B. C. Kuo and F. Golnaraghi, *Automatic Control Systems*, 8th Ed., John Wiley & Sons, New York, 2003.

12. F. Golnaraghi and B. C. Kuo, *Automatic Control Systems*, 9th Ed., John Wiley & Sons, New York, 2010.

PROBLEMS

6-1. Write the force equations of the linear translational systems shown in Fig. 6P-1.

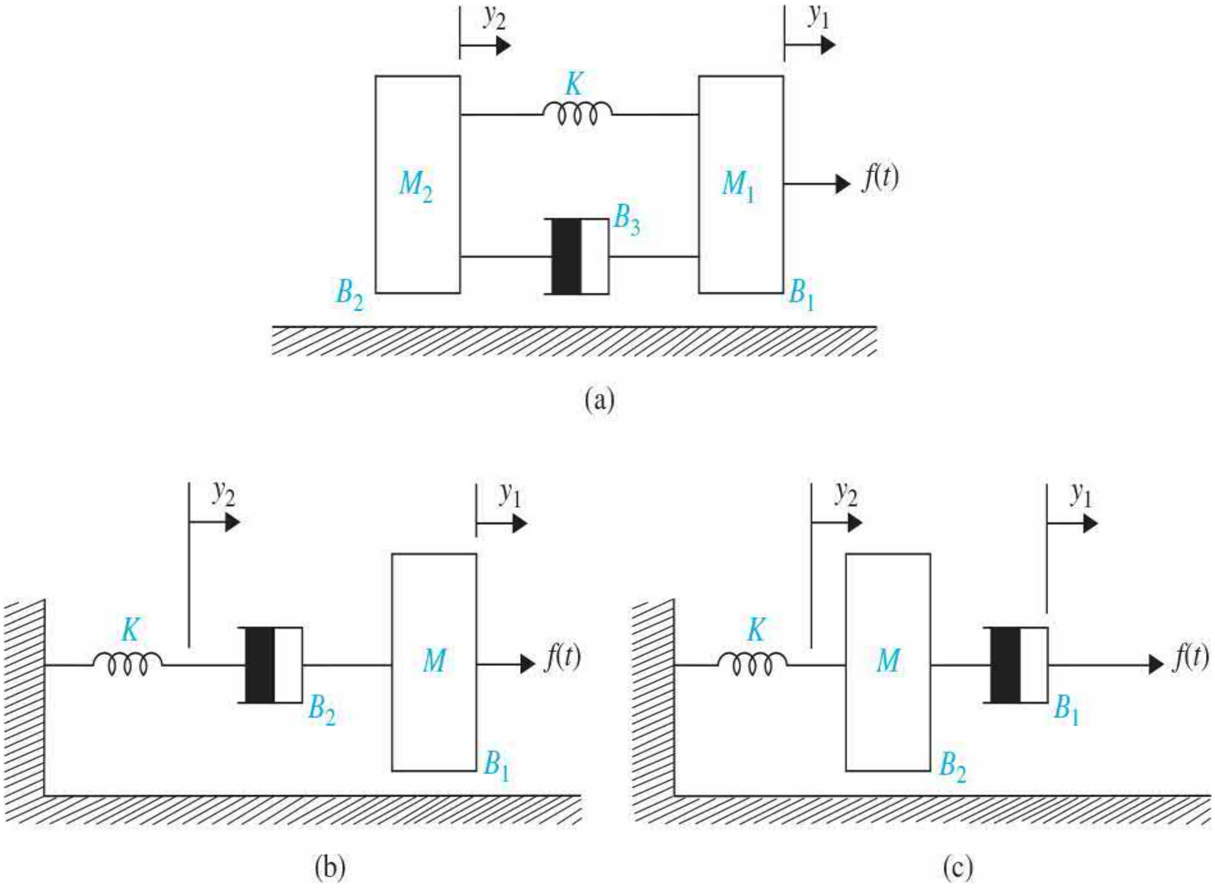


Figure 6P-1

(a) Draw state diagrams using a minimum number of integrators. Write the state equations from the state diagrams.

(b) Define the state variables as follows:

(i) $x_1 = y_2$, $x_2 = dy_2/dt$, $x_3 = y_1$, and $x_4 = dy_1/dt$

(ii) $x_1 = y_2$, $x_2 = y_1$, and $x_3 = dy_1/dt$

(iii) $x_1 = y_1$, $x_2 = y_2$, and $x_3 = dy_2/dt$

Write the state equations and draw the state diagram with these state variables. Find the transfer functions $Y_1(s)/F(s)$ and $Y_2(s)/F(s)$.

6-2. Write the force equations of the linear translational system shown in Fig. 6P-2. Draw the state diagram using a minimum number of integrators. Write the state equations from the state diagram. Find the transfer functions $Y_1(s)/F(s)$ and $Y_2(s)/F(s)$. Set $Mg = 0$ for the transfer functions.

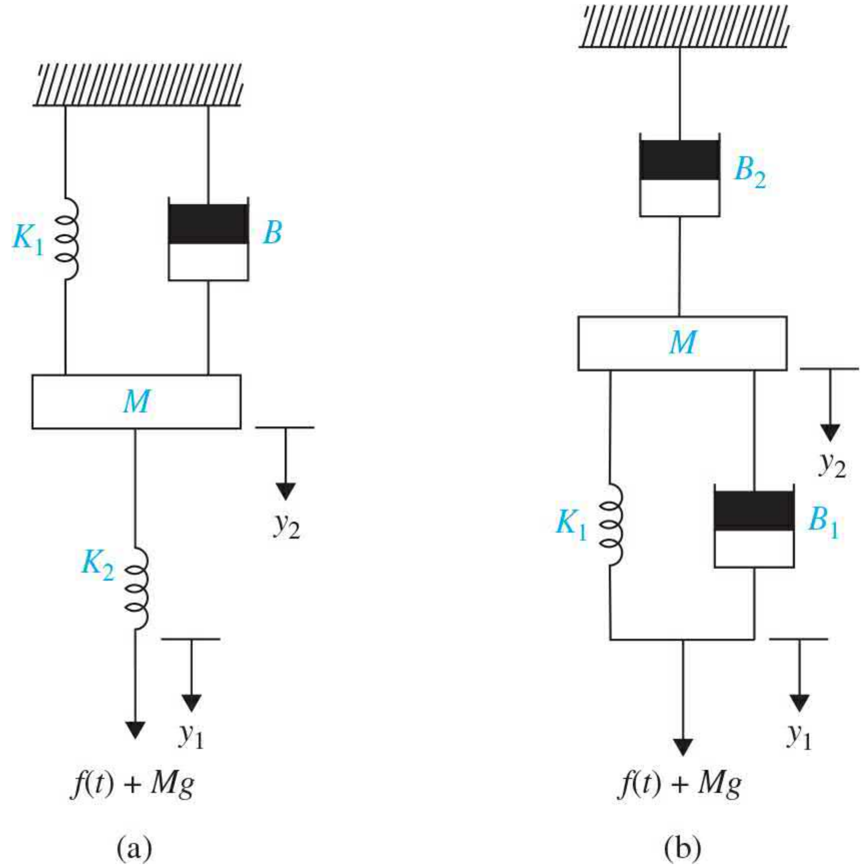


Figure 6P-2

6-3. Write the torque equations of the rotational systems shown in Fig. 6P-3. Draw state diagrams using a minimum number of integrators. Write the state equations from the state diagrams. Find the transfer function $\Theta(s)/T(s)$

for the system in (a). Find the transfer functions $\Theta_1(s)/T(s)$ and $\Theta_2(s)/T(s)$ for the systems in parts (b), (c), (d), and (e).

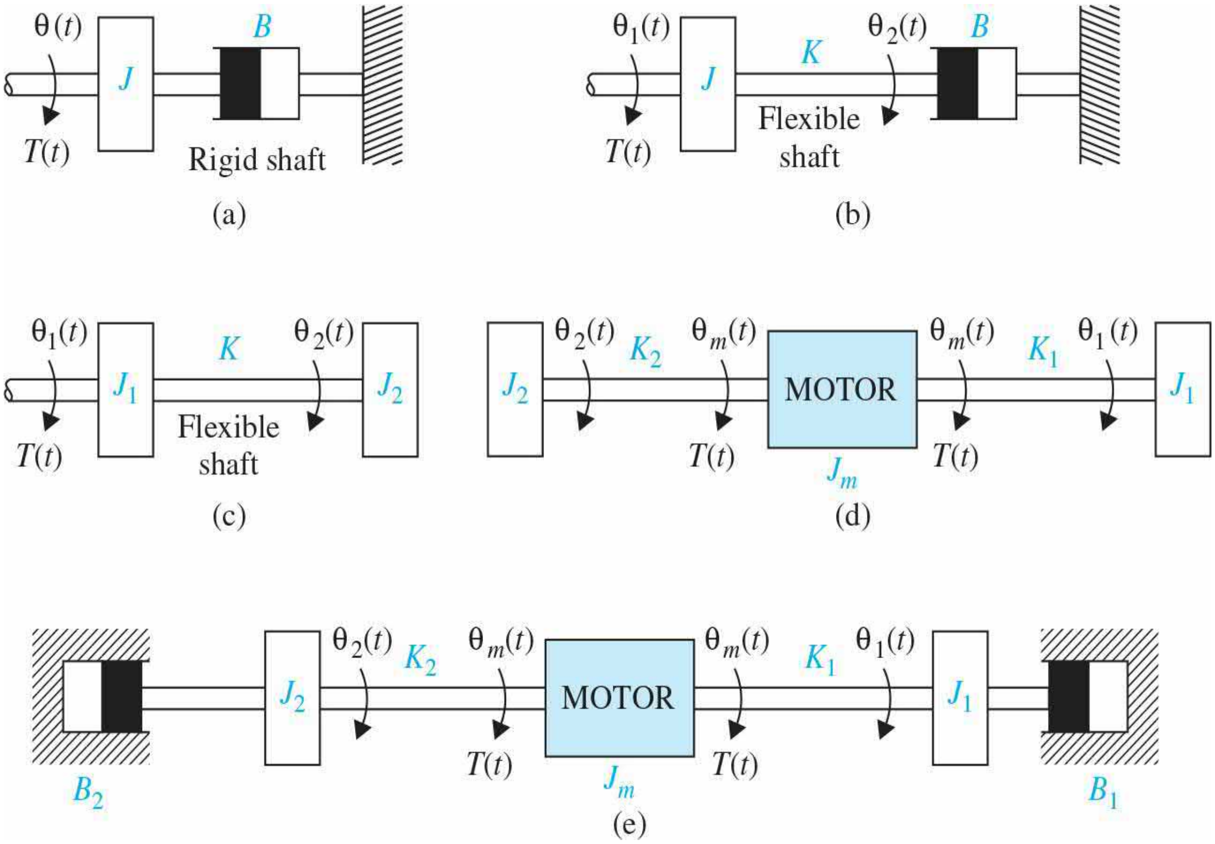


Figure 6P-3

6-4. An open-loop motor control system is shown in Fig. 6P-4. The potentiometer has a maximum range of 10 turns (20π red.). Find the transfer functions $E_o(s)/T_m(s)$. The following parameters and variables are defined: $\theta_m(t)$ is the motor displacement; $\theta_L(t)$, the load displacement; $T_m(t)$, the motor torque; J_m, the motor inertia; B_m, the motor viscous-friction coefficient; B_p, the potentiometer viscous-friction coefficient; $e_o(t)$, the output voltage, and K, the torsional spring constant.

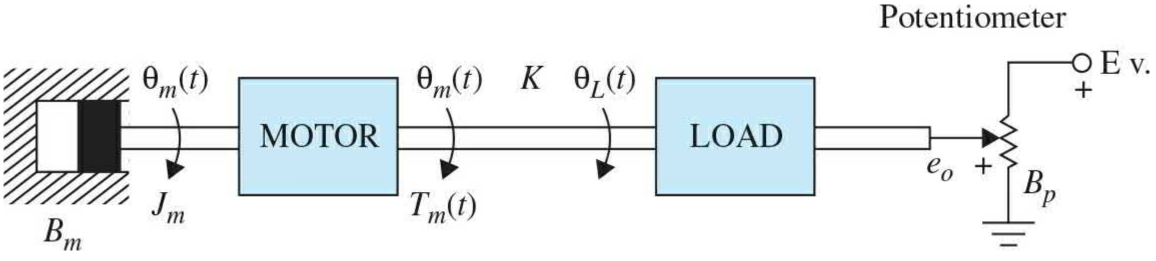


Figure 6P-4

6-5. Write the torque equations of the gear-train system shown in Fig. 6P-5. The moments of inertia of gears are lumped as J_1, J_2, and J_3. $T_m(t)$ is the applied torque; N_1, N_2, N_3, and N_4 are the number of gear teeth. Assume rigid shafts.

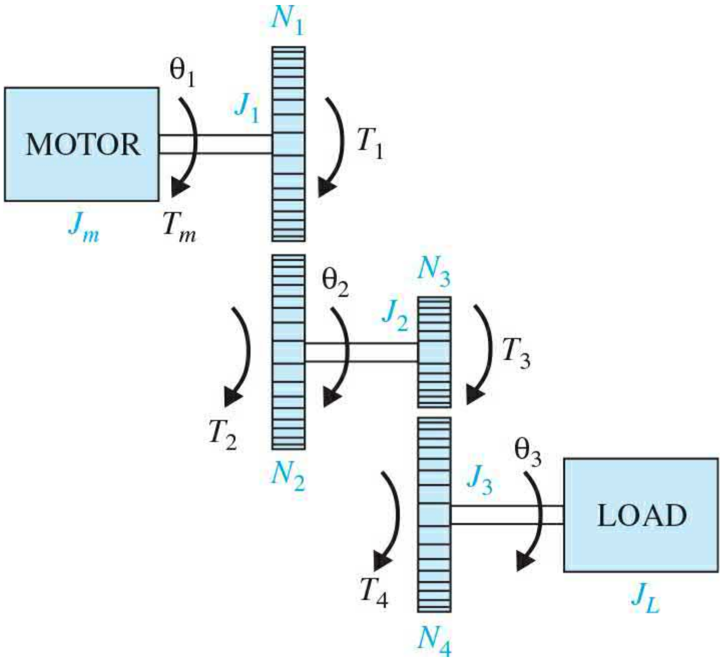


Figure 6P-5

(a) Assume that J_1, J_2, and J_3 are negligible. Write the torque equations of the system. Find the total inertia of the motor.

(b) Repeat part (a) with the moments of inertia J_1, J_2, and J_3.

6-6. A vehicle towing a trailer through a spring-damper coupling hitch is shown in Fig. 6P-6. The following parameters and variables are defined: M is the mass of the trailer; K_h, the spring constant of the hitch; B_h, the viscous damping coefficient of the hitch; B_t, the viscous-friction coefficient of the trailer; $y_1(t)$, the displacement of the towing vehicle; $y_2(t)$, the displacement of the trailer; and $f(t)$, the force of the towing vehicle.

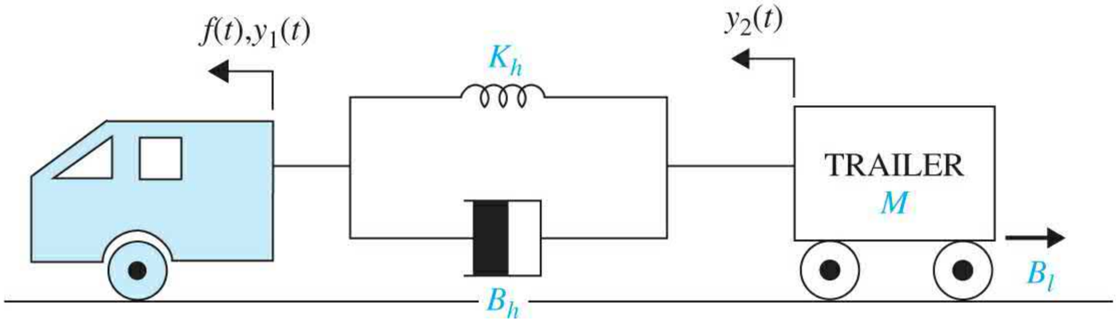


Figure 6P-6

(a) Write the differential equation of the system.

(b) Write the state equations by defining the following state variables: $x_1(t) = y_1(t) - y_2(t)$ and $x_2(t) = dy_2(t)dt$.

6-7. Figure 6P-7 shows a motor-load system coupled through a gear train with gear ratio $n = N_1/N_2$. The motor torque is $T_m(t)$, and $T_L(t)$ represents a load torque.

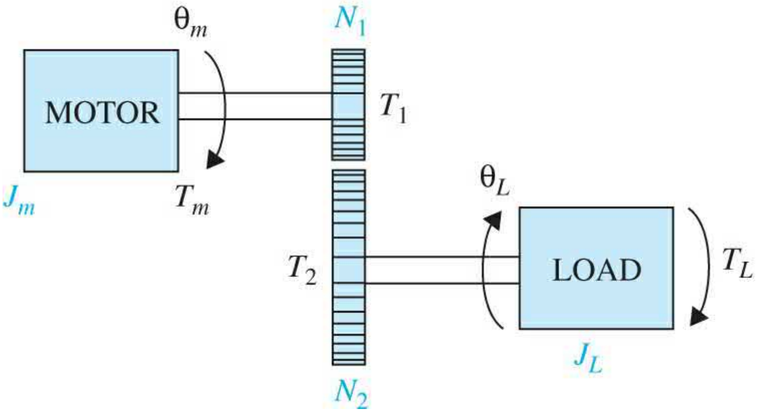


Figure 6P-7

(a) Find the optimum gear ratio n^* such that the load acceleration $\alpha_L = d^2\theta_L/dt^2$ is maximized.

(b) Repeat part (a) when the load torque is zero.

6-8. Figure 6P-8 shows the simplified diagram of the printwheel control system of a word processor. The printwheel is controlled by a dc motor through belts and pulleys. Assume that the belts are rigid. The following parameters and variables are defined: $T_m(t)$ is the motor torque; $\theta_m(t)$, the motor displacement; $y(t)$, the linear displacement of the printwheel; J_m, the motor inertia; B_m, the motor viscous-friction coefficient; r, the pulley radius; M, the mass of the printwheel.

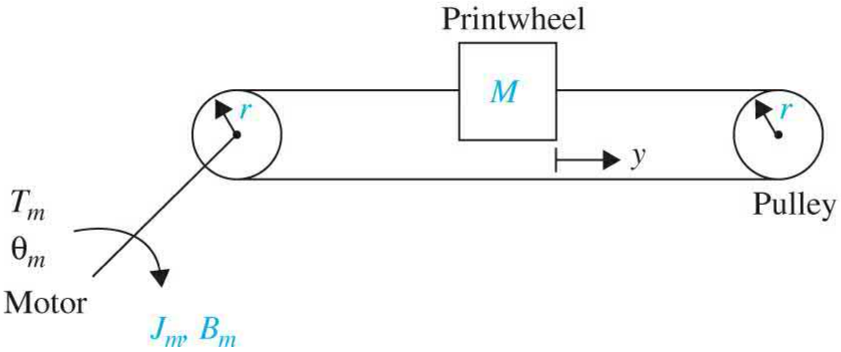


Figure 6P-8

(a) Write the differential equation of the system.

(b) Find the transfer function $Y(s)/T_m(s)$.

6-9. Figure 6P-9 shows the diagram of a printwheel system with belts and pulleys. The belts are modeled as linear springs with spring constants K_1 and K_2.

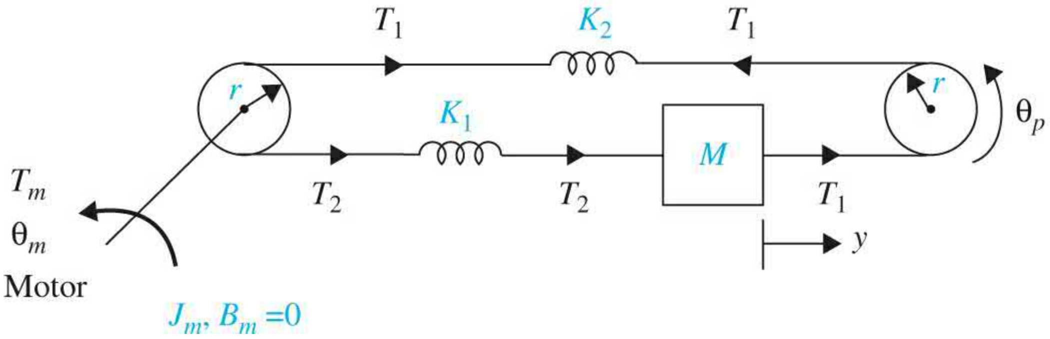


Figure 6P-9

(a) Write the differential equations of the system using θ_m and y as the dependent variables.

(b) Write the state equations using $x_1 = r\theta_m - y$, $x_2 = dy/dt$, and $x_3 = \omega_m = d\theta_m/dt$ as the state variables.

(c) Draw a state diagram for the system.

(d) Find the transfer function $Y(s)/T_m(s)$.

(e) Find the characteristic equation of the system.

6-10. The schematic diagram of a motor-load system is shown in Fig. 6P-10. The following parameters and variables are defined: $T_m(t)$ is the motor torque; $\omega_m(t)$, the motor velocity; $\theta_m(t)$, the motor displacement; $\omega_L(t)$, the load velocity; $\theta_L(t)$, the load displacement; K, the torsional spring constant; J_m, the motor inertia; B_m, the motor viscous-friction coefficient; and B_L, the load viscous-friction coefficient.

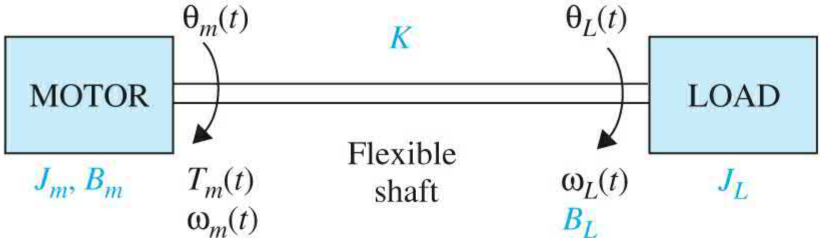


Figure 6P-10

(a) Write the torque equations of the system.

(b) Find the transfer functions $\Theta_L(s)/T_m(s)$ and $\Theta_m(s)/T_m(s)$.

(c) Find the characteristic equation of the system.

(d) Let $T_m(t) = T_m$ be a constant applied torque; show that $\omega_m = \omega_L =$ constant in the steady state. Find the steady-state speeds ω_m and ω_L.

(e) Repeat part (d) when the value of J_L is doubled, but J_m stays the same.

6-11. The schematic diagram of a control system containing a motor coupled to a tachometer and an inertial load is shown in Fig. 6P-11. The following parameters and variables are defined: T_m is the motor torque; J_m, the motor inertia; J_t, the tachometer inertia; J_L, the load inertia; K_1 and K_2, the spring constants of the shafts; θ_t, the tachometer displacement; θ_m, the motor velocity; θ_L, the load displacement; ω_t, the tachometer velocity; ω_L, the load velocity; and B_m, the motor viscous-friction coefficient.

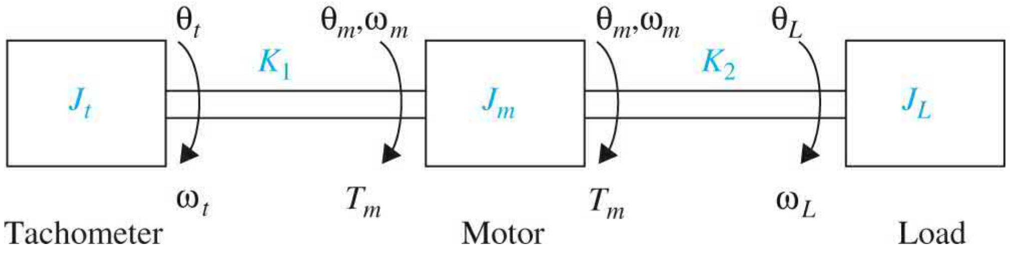


Figure 6P-11

(a) Write the state equations of the system using θ_L, ω_L, θ_t, ω_t, θ_m, and ω_m as the state variables (in the listed order). The motor torque T_m is the input.

(b) Draw a signal flow diagram with T_m at the left and ending with θ_L on the far right. The state diagram should have a total of 10 nodes. Leave out the initial states.

(c) Find the following transfer functions: $\dfrac{\Theta_L(s)}{T_m(s)} \quad \dfrac{\Theta_t(s)}{T_m(s)} \quad \dfrac{\Theta_m(s)}{T_m(s)}$.

(d) Find the characteristic equation of the system.

6-12. The voltage equation of a dc motor is written as

$$e_a(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + K_b \omega_m(t)$$

where $e_a(t)$ is the applied voltage; $i_a(t)$, the armature current; R_a, the armature resistance; L_a, the armature inductance; K_b, the back-emf constant; $\omega_m(t)$, the motor velocity; and $\omega_n(t)$, the reference input voltage. Taking the Laplace transform on both sides of the voltage equation, with zero initial conditions and solving for $\Omega_m(s)$, we get

$$\Omega_m(s) = \frac{E_a(s) - (R_a + L_a s) I_a(s)}{K_b}$$

which shows that the velocity information can be generated by feeding back the armature voltage and current. The block diagram in Fig. 6P-12 shows a dc-motor system, with voltage and current feedbacks, for speed control.

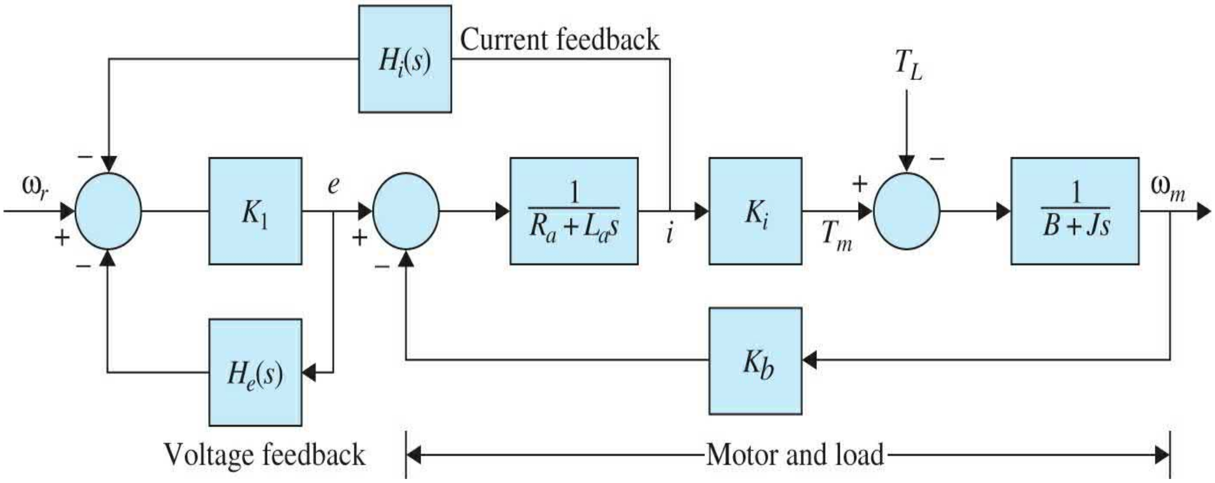


Figure 6P-12

(a) Let K_1 be a very high gain amplifier. Show that when $H_i(s)/H_e(s) = -(R_a + L_a s)$, the motor velocity $\omega_m(t)$ is totally independent of the load-disturbance torque T_L.

(b) Find the transfer function between $\Omega_m(s)$ and $\Omega_r(s)(T_L = 0)$ when $H_i(s)$ and $H_e(s)$ are selected as in part (a).

6-13. This problem deals with the attitude control of a guided missile. When traveling through the atmosphere, a missile encounters aerodynamic forces that tend to cause instability in the attitude of the missile. The basic concern from the flight-control standpoint is the lateral force of the air, which tends to rotate the missile about its center of gravity. If the missile centerline is not aligned with the direction in which the center of gravity C is traveling, as shown in Fig. 6P-13, with angle θ, which is also called the angle of attack, a side force is produced by the drag of the air through which the missile travels. The total force F_α may be considered to be applied at the center of pressure P. As shown in Fig. 6P-20, this side force has a tendency to cause the missile to tumble end over end, especially if the point P is in front of the center of gravity C. Let the angular acceleration of the missile about the point C, due to the side force, be denoted by α_F. Normally, α_F is directly proportional to the angle of attack θ and is given by

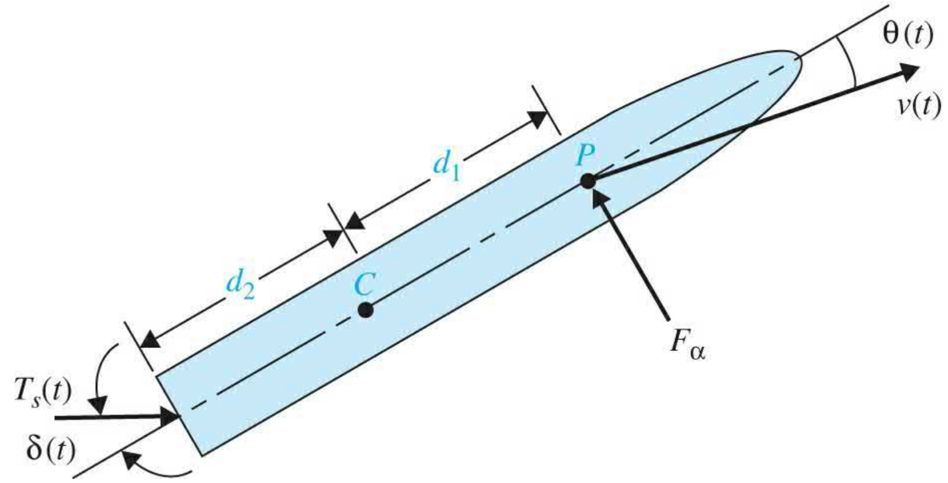


Figure 6P-13

$$\alpha_F = \frac{K_F d_1}{J}\theta$$

where K_F is a constant that depends on such parameters as dynamic pressure, velocity of the missile, air density, and so on, and

J = missile moment of inertia about C

d_1 = distance between C and P

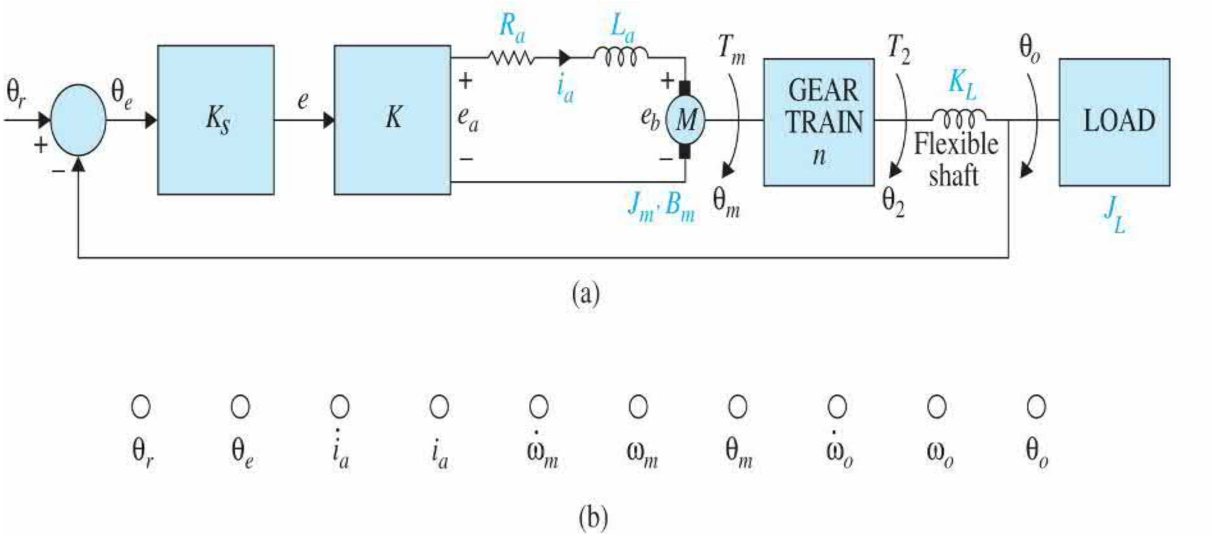
The main objective of the flight-control system is to provide the stabilizing action to counter the effect of the side force. One of the standard control means is to use gas injection at the tail of the missile to deflect the direction of the rocket engine thrust T_s, as shown in the figure.

(a) Write a torque differential equation to relate among T_s, δ, θ, and the system parameters given. Assume that δ is very small, so that sin $\delta(t)$ is approximated by $\delta(t)$.

(b) Assume that T_s is a constant torque. Find the transfer function $\Theta(s)/\Delta(s)$, where $\Theta(s)$ and $\Delta(s)$ are the Laplace transforms of $\theta(t)$ and $\delta(t)$, respectively. Assume that $\delta(t)$ is very small.

(c) Repeat parts (a) and (b) with points C and P interchanged. The d_1 in the expression of α_F should be changed to d_2.

6-14. Figure 6P-14*a* shows the schematic diagram of a dc-motor control system for the control of a printwheel of a word processor. The load in this case is the printwheel, which is directly coupled to the motor shaft. The following parameters and variables are defined: K_s is the error-detector gain (V/rad); K_i, the torque constant (oz · in/A); K, the amplifier gain (V/V); K_b, the back-emf constant (V/rad/s); n, the gear train ratio $\theta_2/\theta_m = T_m/T_2$; B_m, the motor viscous-friction coefficient (oz · in · s); J_m, the motor inertia (oz · in · s^2); K_L the torsional spring constant of the motor shaft (oz · in/rad); and J_L, the load inertia (oz · in · sec^2).



(a)

(b)

Figure 6P-14

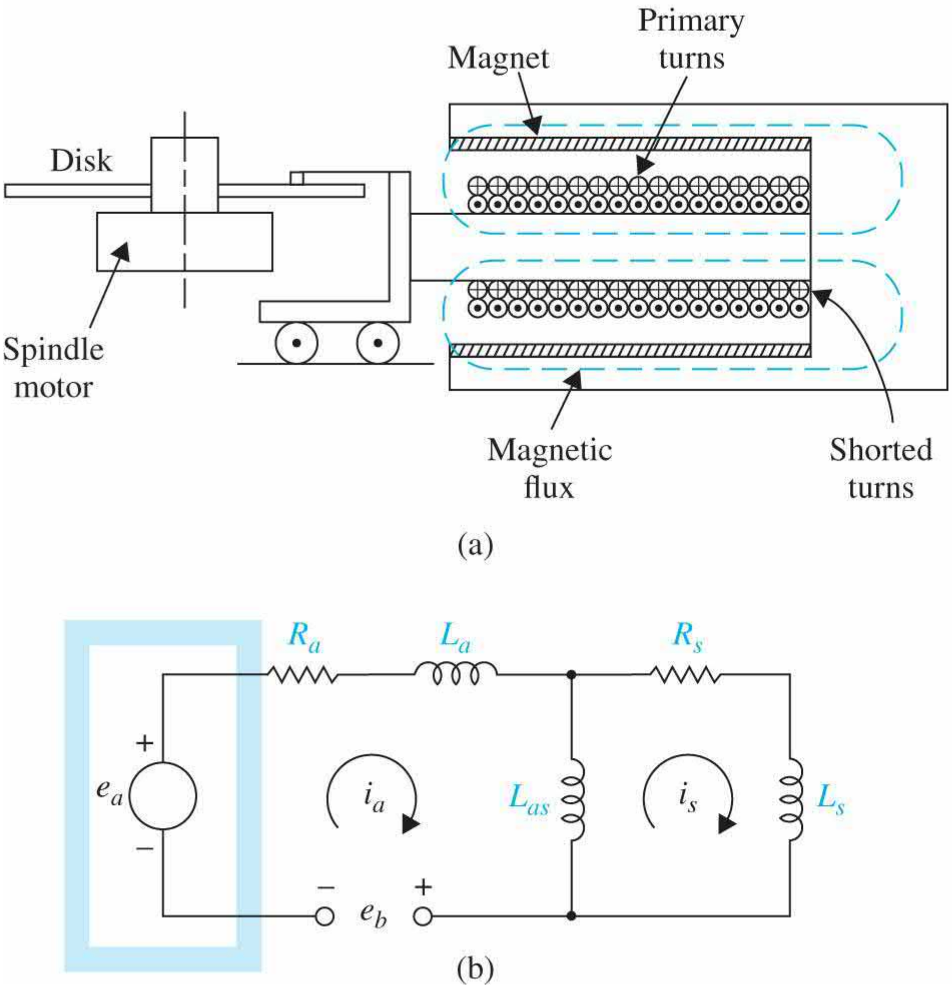
(a) Write the cause-and-effect equations of the system. Rearrange these equations into the form of state equations with $x_1 = \theta_o$, $x_2 = \theta_o$, $x_3 = \theta_m$, $x_4 = \omega_m$, and $x_5 = i_a$.

(b) Draw a state diagram using the nodes shown in Fig. 3P-38*b*.

(c) Derive the forward-path transfer function (with the outer feedback path open): $G(s) = \Theta_o(s)/\Theta_e(s)$. Find the closed-loop transfer function $M(s) = \Theta_o(s)/\Theta_r(s)$.

(d) Repeat part (c) when the motor shaft is rigid, that is, $K_L = \infty$ Show that you can obtain the solutions by taking the limit as K_L approaches infinity in the results in part (c).

6-15. The schematic diagram of a voice-coil motor (VCM), used as a linear actuator in a disk memory-storage system, is shown in Fig. 6P-15*a*. The VCM consists of a cylindrical permanent magnet (PM) and a voice coil. When current is sent through the coil, the magnetic field of the PM interacts with the current-carrying conductor, causing the coil to move linearly. The voice coil of the VCM in Fig. 6P-15*a* consists of a primary coil and a shorted-turn coil. The latter is installed for the purpose of effectively reducing the electric constant of the device. Figure 6P-15*b* shows the equivalent circuit of the coils. The following parameters and variables are defined: $e_a(t)$ is the applied coil voltage; $i_a(t)$, the primary-coil current; $i_s(t)$, the shorted-turn coil current; R_a, the primary-coil resistance; L_a, the primary-coil inductance; L_{as}, the mutual inductance between the primary and shorted-turn coils; $v(t)$, the velocity of the voice coil; $y(t)$, the displacement of the voice coil; $f(t) = K_i v(t)$, the force of the voice coil; K_i, the force constant; K_b, the back-emf constant; $e_b(t) = K_b v(t)$, the back emf; M_T, the total mass of the voice coil and load; and B_T, the total viscous-friction coefficient of the voice coil and load.



(a)

(b)

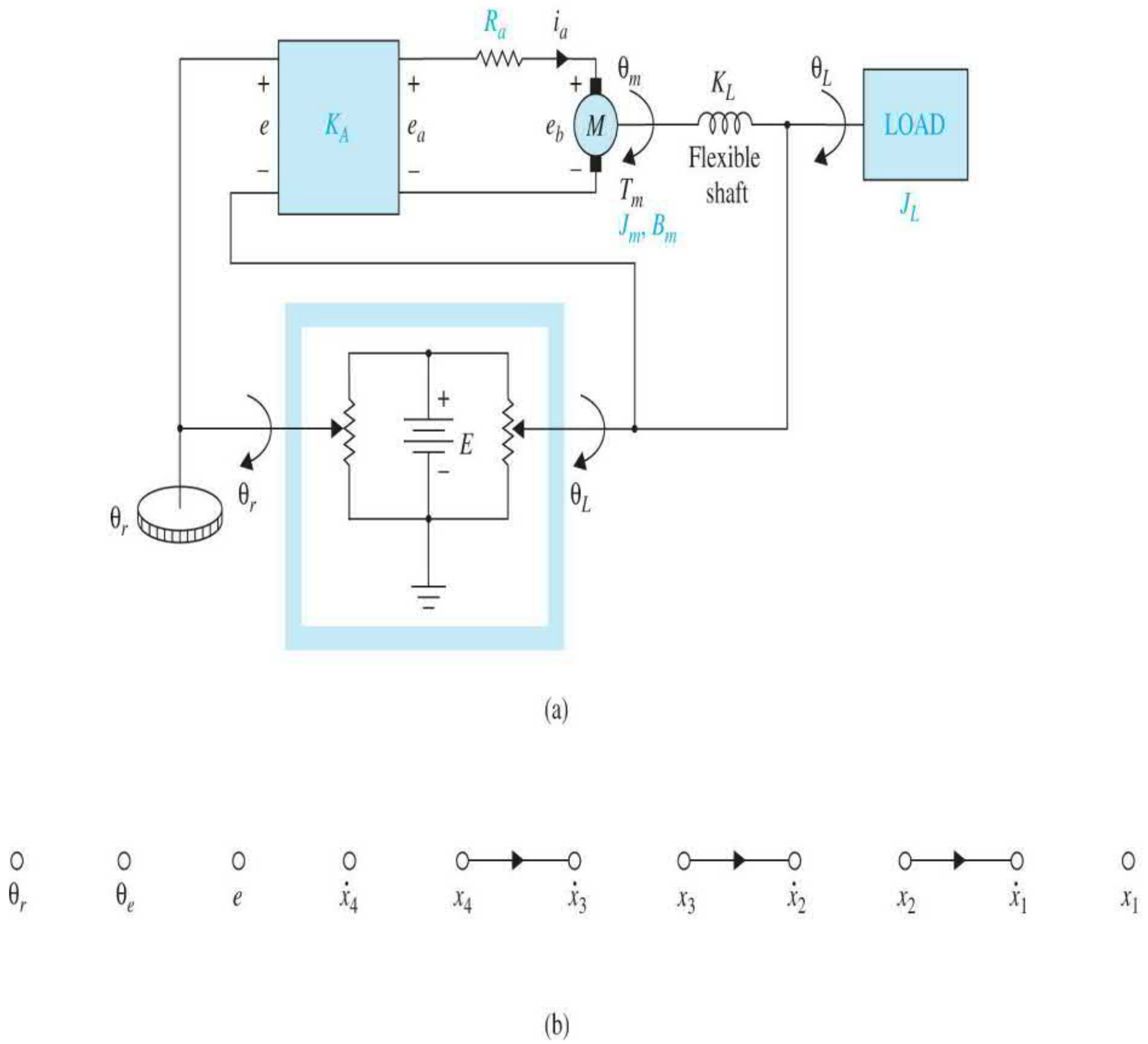
Figure 6P-15

(a) Write the differential equations of the system.

(b) Draw a block diagram of the system with $E_a(s)$, $I_a(s)$, $I_s(s)$, $V(s)$, and $Y(s)$ as variables.

(c) Derive the transfer function $Y(s)/E_a(s)$.

6-16. A dc-motor position-control system is shown in Fig. 6P-16*a*. The following parameters and variables are defined: e is the error voltage; e_r, the reference input; θ_L, the load position; K_A, the amplifier gain; e_a, the motor input voltage; e_b, the back emf; i_a, the motor current; T_m, the motor torque; J_m, the motor inertia = 0.03 oz · in · s^2; B_m, the motor viscous-friction coefficient = 10 oz · in · s^2; K_L, the torsional spring constant = 50,000 oz · in/rad; J_L, the load inertia = 0.05 oz · in · s^2; K_i, the motor torque constant = 21 oz · in/A; K_b, the back-emf constant = 15.5 V/1000 rpm; K_s, the error-detector gain = $E/2\pi$; E, the error-detector applied voltage = $2\pi V$; R_a, the motor resistance = 1.15 Ω; and $\theta_e = e_r - \theta_L$.



(a)

(b)

Figure 6P-16

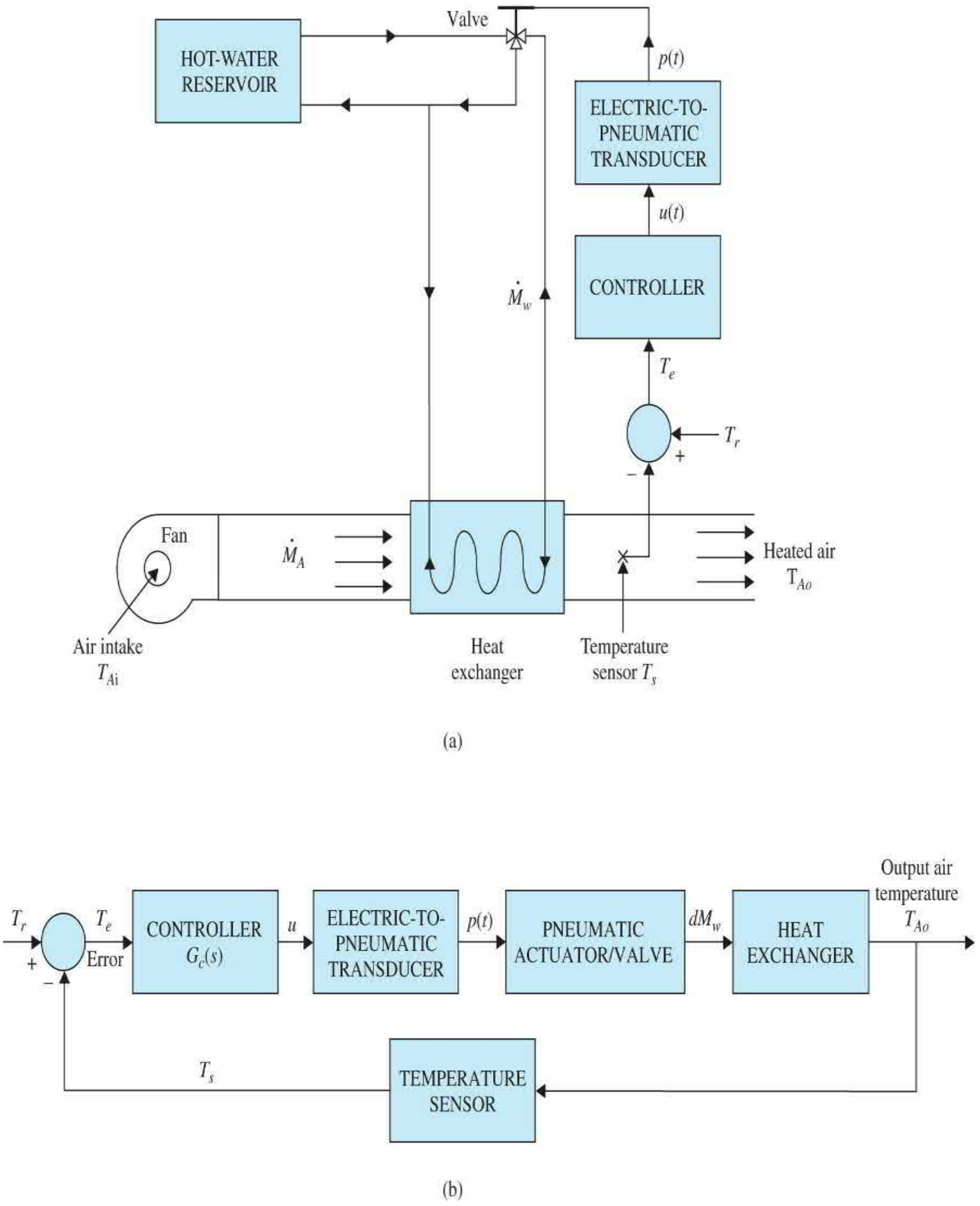
(a) Write the state equations of the system using the following state variables: $x_1 = \theta_L$, $x_2 = d\theta_L/dt = \omega_L$, $x_3 = \theta_3$, and $x_4 = d\theta_m/dt = \omega_m$.

(b) Draw a signal flow diagram using the nodes shown in Fig. 6P-16*b*.

(c) Derive the forward-path transfer function $G(s) = \Theta_L(s)/\Theta_e(s)$ when the outer feedback path from θ_L is opened. Find the poles of $G(s)$.

(d) Derive the closed-loop transfer function $M(s) = \Theta_L(s)/\Theta_e(s)$. Find the poles of $M(s)$ when $K_A = 1{,}2738$, and 5476. Locate these poles in the s-plane, and comment on the significance of these values of K_A.

6-17. Figure 6P-17*a* shows the setup of the temperature control of an air-flow system. The hot-water reservoir supplies the water that flows into the heat exchanger for heating the air. The temperature sensor senses the air temperature T_{AO} and sends it to be compared with the reference temperature T_r. The temperature error T_e is sent to the controller, which has the transfer function $G_c(s)$. The output of the controller, $u(t)$, which is an electric signal, is converted to a pneumatic signal by a transducer. The output of the actuator controls the water-flow rate through the three-way valve. Figure 6P-17*b* shows the block diagram of the system.



(a)

(b)

Figure 6P-17

The following parameters and variables are defined: dM_w is the flow rate of the heating fluid = $k_M u$, K_M = 0.054 kg/s/V; T_w, the water temperature = $K_R dM_w$; K_R = 65°C/kg/s; and T_{AO} the output air temperature.

Heat-transfer equation between water and air:

$$\tau_c \frac{dT_{AO}}{dt} = T_w - T_{AO} \qquad \tau_c = 10 \text{ s}$$

Temperature sensor equation:

$$\tau_s \frac{dT_s}{dt} = T_{AO} - T_s \qquad \tau_s = 2 \text{ s}$$

(a) Draw a functional block diagram that includes all the transfer functions of the system.

(b) Derive the transfer function $T_{AO}(s)/T_{r(s)}$ when $G_c(s)$ = 1.

6-18. The objective of this problem is to develop a linear analytical model of the automobile engine for idle-speed control system shown in Fig. 1-2. The input of the system is the throttle position that controls the rate of air flow into the manifold (see Fig. 6P-18). Engine torque is developed from the buildup of manifold pressure due to air intake and the intake of the air/gas mixture into the cylinder. The engine variations are as follows:

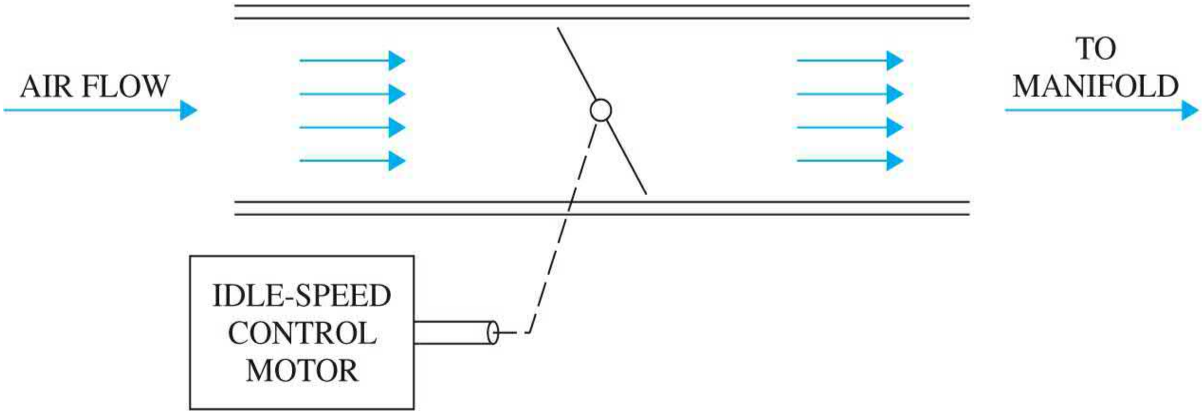


Figure 6P-18

$q_i(t)$ = amount of air flow across throttle into manifold

$dq_i(t)/dt$ = rate of air flow across throttle into manifold

$q_m(t)$ = average air mass in manifold

$q_o(t)$ = amount of air leaving intake manifold through intake valves

$dq_o(t)/dt$ = rate of air leaving intake manifold through intake valves

$T(t)$ = engine torque

T_d = disturbance torque due to application of auto accessories = constant

$\omega(t)$ = engine speed

$\alpha(t)$ = throttle position

τ_D = time delay in engine

J_e = inertia of engine

The following assumptions and mathematical relations between the engine variables are given:

1. The rate of air flow into the manifold is linearly dependent on the throttle position:

$$\frac{dq_i(t)}{dt} = K_1\alpha(t) \qquad K_1 = \text{proportional constant}$$

2. The rate of air flow leaving the manifold depends linearly on the air mass in the manifold and the engine speed:

$$\frac{dq_o(t)}{dt} = K_2\, q_m(t) + K_3\omega(t) \qquad K_2,\, K_3 = \text{constants}$$

3. A pure time delay of τ_D seconds exists between the change in the manifold air mass and the engine torque:

$$T(t) = K_4 q_m(t - \tau_D) \qquad K_4 = \text{constant}$$

4. The engine drag is modeled by a viscous-friction torque $B\omega(t)$, where B is the viscous-friction coefficient.

5. The average air mass $q_m(t)$ is determined from

$$q_m(t) = \int\left(\frac{dq_i(t)}{dt} - \frac{dq_o(t)}{dt}\right)dt$$

6. The equation describing the mechanical components is

$$T(t) = J\frac{d\omega(t)}{dt} + B\omega(t) + T_d$$

(a) Draw a functional block diagram of the system with *α(t)* as the input, *ω(t)* as the output, and T_d as the disturbance input. Show the transfer function of each block.

(b) Find the transfer function Ω(*s*)/*α(s)* of the system.

(c) Find the characteristic equation and show that it is not rational with constant coefficients.

(d) Approximate the engine time delay by

$$e^{-\tau_D s} \cong \frac{1 - \tau_D s/2}{1 + \tau_D s/2}$$

and repeat parts (b) and (c).

6-19. Phase-locked loops are control systems used for precision motor-speed control. The basic elements of a phase-locked loop system incorporating a dc motor is shown in Fig. 6P-19*a*. An input pulse train represents the reference frequency or desired output speed. The digital encoder produces digital pulses that represent motor speed. The phase detector compares the motor speed and the reference frequency and sends an error voltage to the filter (controller) that governs the dynamic response of the system. Phase detector gain = K_p, encoder gain = K_e, counter gain = 1/*N*, and dc-motor torque constant = K_i. Assume zero inductance and zero friction for the motor.

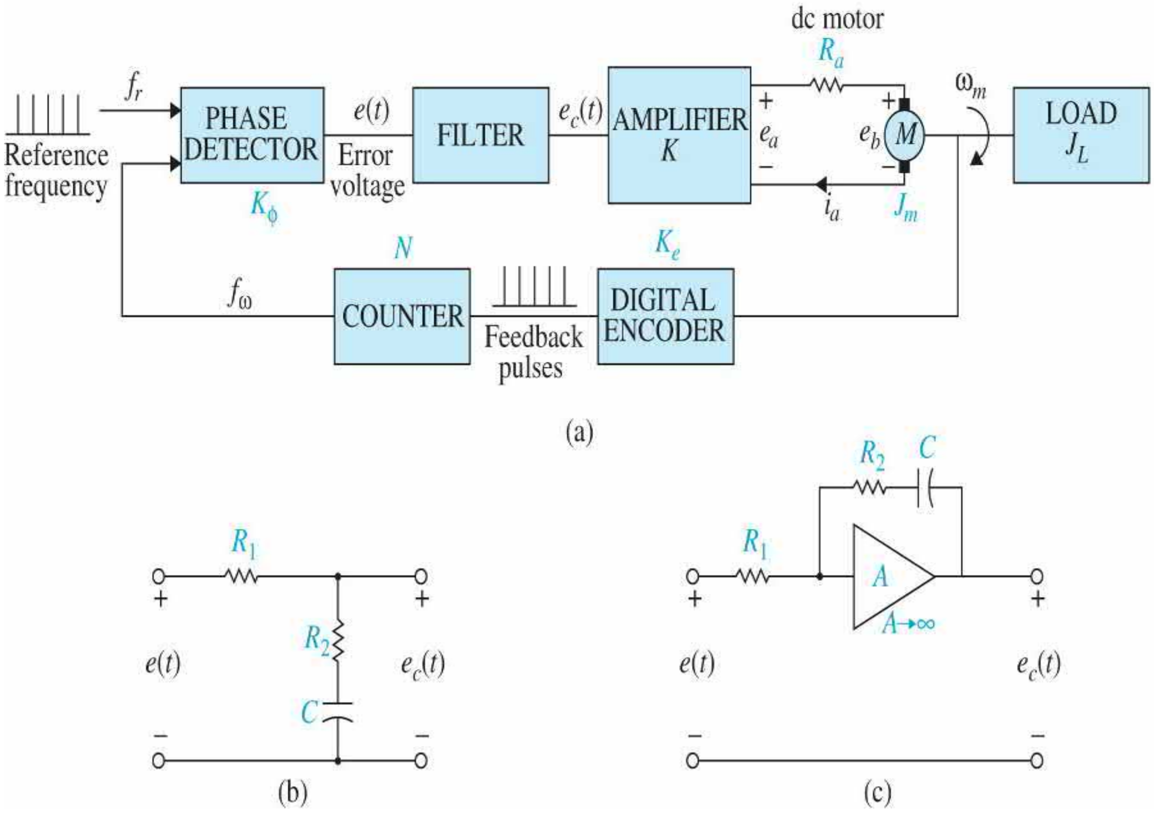


Figure 6P-19

(a) Derive the transfer function $E_c(s)/E(s)$ of the filter shown in Fig. 6P-19*b*. Assume that the filter sees infinite impedance at the output and zero impedance at the input.

(b) Draw a functional block diagram of the system with gains or transfer functions in the blocks.

(c) Derive the forward-path transfer function $\Omega_m(s)/E(s)$ when the feedback path is open.

(d) Find the closed-loop transfer function $\Omega_m(s)/F_r(s)$.

(e) Repeat parts (a), (c), and (d) for the filter shown in Fig. 6P-19(c).

(f) The digital encoder has an output of 36 pulses per revolution. The reference frequency f_r is fixed at 120 pulse/s. Find K_e in pulse/rad. The idea of using the counter N is that with f_r fixed, various desired output speeds can be attained by changing the value of N. Find N if the desired output speed is 200 rpm. Find N if the desired output speed is 1800 rpm.

6-20. The linearized model of a robot arm system driven by a dc motor is shown in Fig. 6P-20. The system parameters and variables are given as follows:

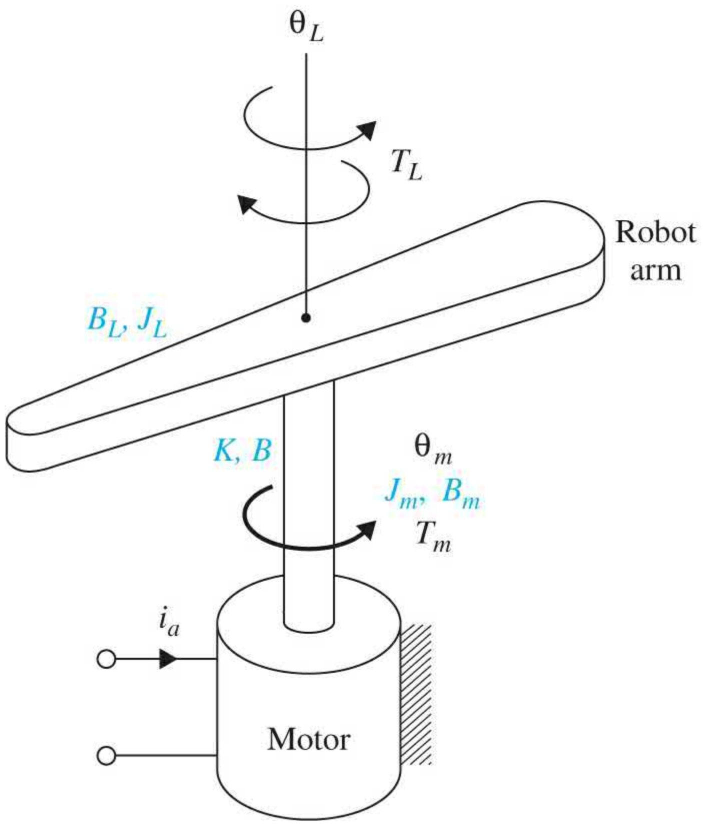


Figure 6P-20

DC Motor	Robot Arm
T_m = motor torque = $K_i i_a$	J_L = inertia of arm
K_i = torque constant	T_L = disturbance torque on arm
i_a = armature current of motor	θ_L = arm displacement
J_m = motor inertia	K = torsional spring constant
B_m = motor viscous-friction coefficient	θ_m = motor-shaft displacement
B = viscous-friction coefficient of shaft between the motor and arm	
B_L = viscous-friction coefficient of the robot arm shaft	

(a) Write the differential equations of the system with $i_a(t)$ and $T_L(t)$ as input and $\theta_m(t)$ and $\theta_L(t)$ as outputs.

(b) Draw an SFG using $I_a(s)$, $T_L(s)$, $\Theta_L(s)$ as node variables.

(c) Express the transfer-function relations as

$$\begin{bmatrix} \Theta_m(s) \\ \Theta_L(s) \end{bmatrix} = \mathbf{G}(s)\begin{bmatrix} I_a(s) \\ -T_L(s) \end{bmatrix}$$

Find **G**(s).

6-21. The following differential equations describe the motion of an electric train in a traction system:

$$\frac{dx(t)}{dt} = v(t)$$

$$\frac{dv(t)}{dt} = -k(v) - g(x) + f(t)$$

where

$x(t)$ = linear displacement of train

$v(t)$ = linear velocity of train

$k(v)$ = resistance force on train [odd function of v, with the properties: $k(0) = 0$ and $dk(v)/dv = 0$

$g(x)$ = gravitational force for a nonlevel track or due to curvature of track

$f(t)$ = tractive force

The electric motor that provides the tractive force is described by the following equations:

$$e(t) = K_b\phi(t)v(t) + R_a i_a(t)$$

$$f(t) = K_i\phi(t)i_a(t)$$

where $e(t)$ is the applied voltage; $i_a(t)$, the armature current; $i_f(t)$, the field current; R_a, the armature resistance; $\phi(t)$, the magnetic flux from a separately excited field = $K_f i_f(t)$; and K_i, the force constant.

(a) Consider that the motor is a dc series motor with the armature and field windings connected in series, so that $i_a(t) = i_f(t)$, $g(x) = 0$, $k(v) = Bv(t)$, and $R_a = 0$. Show that the system is described by the following nonlinear state equations:

$$\frac{dx(t)}{dt} = v(t)$$
$$\frac{dv(t)}{dt} = -Bv(t) + \frac{K_i}{K_b^2 K_f v^2(t)} e^2(t)$$

(b) Consider that for the conditions stated in part (a), $i_a(t)$ is the input of the system [instead of $e(t)$]. Derive the state equations of the system.

(c) Consider the same conditions as in part (a) but with $\phi(t)$ as the input. Derive the state equations.

6-22. Figure 6P-22*a* shows a well-known “broom-balancing” system. The objective of the control system is to maintain the broom in the upright position by means of the force $u(t)$ applied to the car as shown. In practical applications, the system is analogous to a one-dimensional control problem of balancing a unicycle or a missile immediately after launching. The free-body diagram of the system is shown in Fig. 6P-22*b*, where

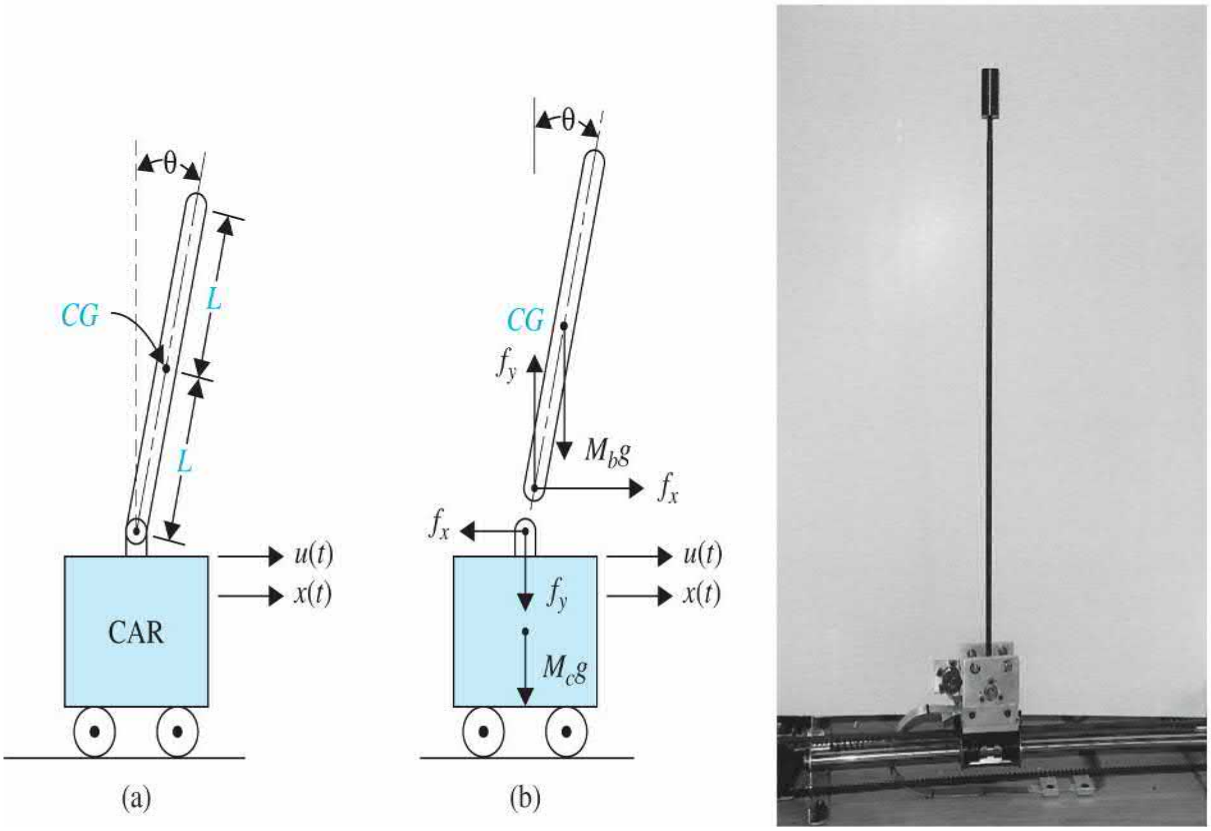


Figure 6P-22

f_x = force at broom base in horizontal direction

f_y = force at broom base in vertical direction

M_b = mass of broom

g = gravitational acceleration

M_c = mass of car

J_b = moment of inertia of broom about center of gravity $CG = M_b L^2/3$

(a) Write the force equations in the x and the y directions at the pivot point of the broom. Write the torque equation about the center of gravity CG of the broom. Write the force equation of the car in the horizontal direction.

(b) Express the equations obtained in part (a) as state equations by assigning the state variables as $x_1 = \theta$, $x_2 = d\theta/dt$, $x_3 = x$, and $x_4 = dx/dt$. Simplify these equations for small θ by making the approximations $\sin\theta \cong \theta$ and $\cos\theta \cong 1$.

(c) Obtain a small-signal linearized state-equation model for the system in the form of

$$\frac{d\Delta\mathbf{x}(t)}{dt} = \mathbf{A} * \Delta\mathbf{x}(t) + \mathbf{B} * \Delta\mathbf{r}(t)$$

at the equilibrium point $x_{01}(t) = 1$, $x_{02}(t) = 0$, $x_{03}(t) = 0$, and $x_{04}(t) = 0$.

6-23. Figure 6P-23 shows the schematic diagram of a ball-suspension control system. The steel ball is suspended in the air by the electromagnetic force generated by the electromagnet. The objective of the control is to keep the metal ball suspended at the nominal equilibrium position by controlling the current in the magnet with the voltage *e(t)*. The practical application of this system is the magnetic levitation of trains or magnetic bearings in high-precision control systems.

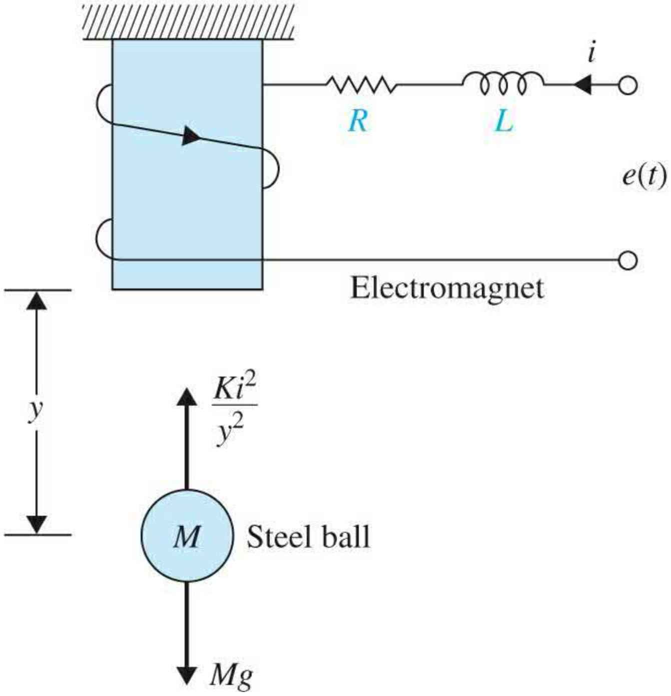


Figure 6P-23

The resistance of the coil is *R*, and the inductance is *L(y)* = *L/y(t)*, where *L* is a constant. The applied voltage *e(t)* is a constant with amplitude *E*.

(a) Let E_{eq} be a nominal value of *E*. Find the nominal values of *y(t)* and *dy(t)/dt* at equilibrium.

(b) Define the state variables at $x_1(t) = i(t)$, $x_2(t) = y(t)$, and $x_3(t) = dy(t)/dt$. Find the nonlinear state equations in the form of

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(\mathbf{x}, e)$$

(c) Linearize the state equations about the equilibrium point and express the linearized state equations as

$$\frac{d\Delta\mathbf{x}(t)}{dt} = \mathbf{A} * \Delta\mathbf{x}(t) + \mathbf{B} * \Delta e(t)$$

The force generated by the electromagnet is $Ki^2(t)/y(t)$, where K is a proportional constant, and the gravitational force on the steel ball is Mg.

6-24. Figure 6P-24*a* shows the schematic diagram of a ball-suspension system. The steel ball is suspended in the air by the electromagnetic force generated by the electromagnet. The objective of the control is to keep the metal ball suspended at the nominal position by controlling the current in the electromagnet. When the system is at the stable equilibrium point, any small perturbation of the ball position from its floating equilibrium position will cause the control to return the ball to the equilibrium position. The free-body diagram of the system is shown in Fig. 6P-24*b*, where

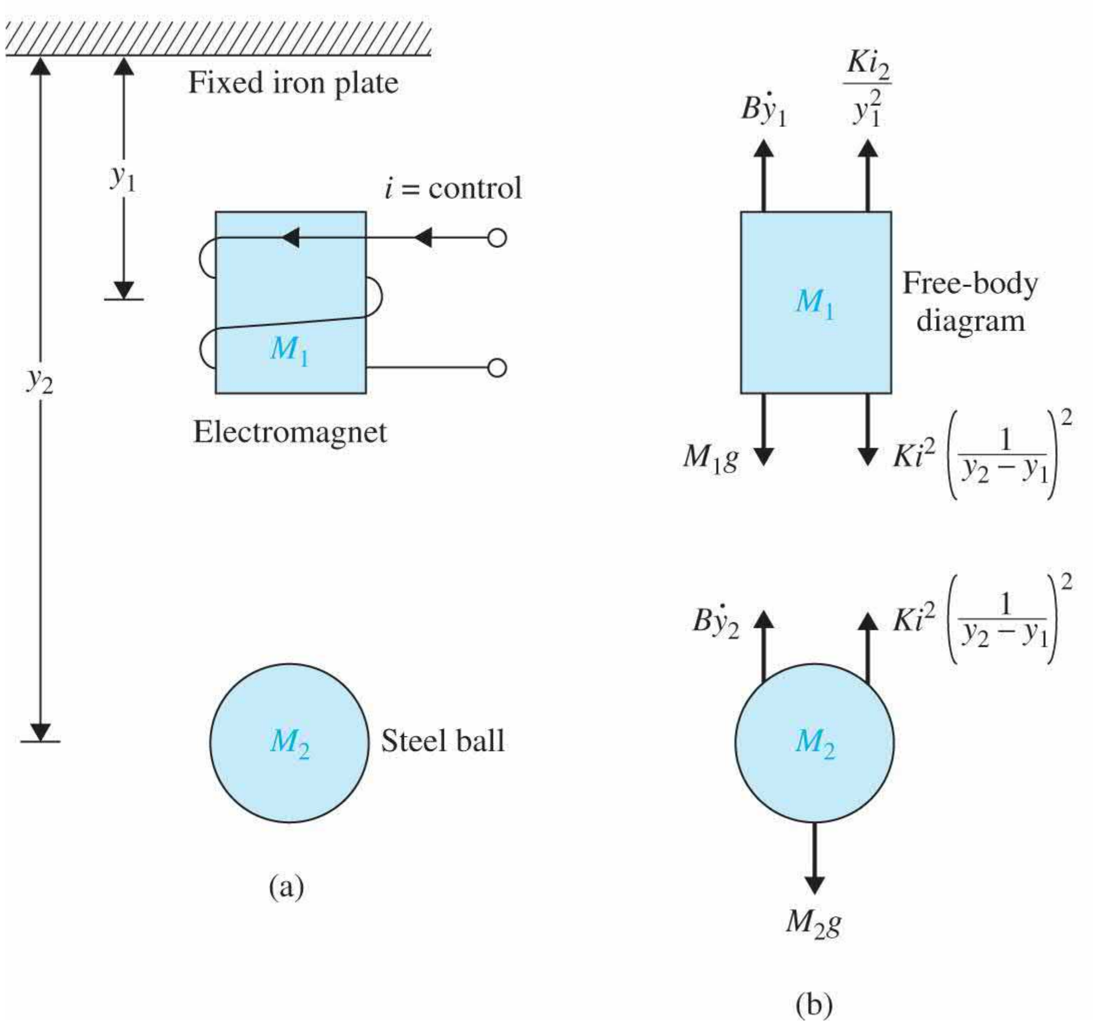


Figure 6P-24

M_1 = mass of electromagnet = 2.0

M_2 = mass of steel ball = 1.0

B = viscous-friction coefficient of air = 0.1

K = proportional constant of electromagnet = 1.0

g = gravitational acceleration = 32.2

Assume all units are consistent. Let the stable equilibrium values of the variable, $i(t)$, $y_1(t)$, and $y_2(t)$ be I, Y_1, and Y_2, respectively. The state variables are defined as $x_1(t) = y_1(t)$, $x_2(t) = dy_1(t)/dt$, $x_3(t) = y_2(t)$, and $x_4(t) = dy_2(t)/dt$.

(a) Given $Y_1 = 1$, find I and Y_2.

(b) Write the nonlinear state equations of the system in the form of $dx(t)/dt = f(x, i)$.

(c) Find the state equations of the linearized system about the equilibrium state I, Y_1, and Y_2 in the form:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A} * \Delta\mathbf{x}(t) + \mathbf{B} * \Delta i(t)$$

6-25. The schematic diagram of a steel-rolling process is shown in Fig. 6P-25. The steel plate is fed through the rollers at a constant speed of V ft/s. The distance between the rollers and the point where the thickness is measured is d ft. The rotary displacement of the motor, $\theta_m(t)$, is converted to the linear displacement $y(t)$ by the gear box and linear-actuator combination $y(t) = n\theta_m(t)$, where n is a positive constant in ft/rad. The equivalent inertia of the load that is reflected to the motor shaft is J_L.

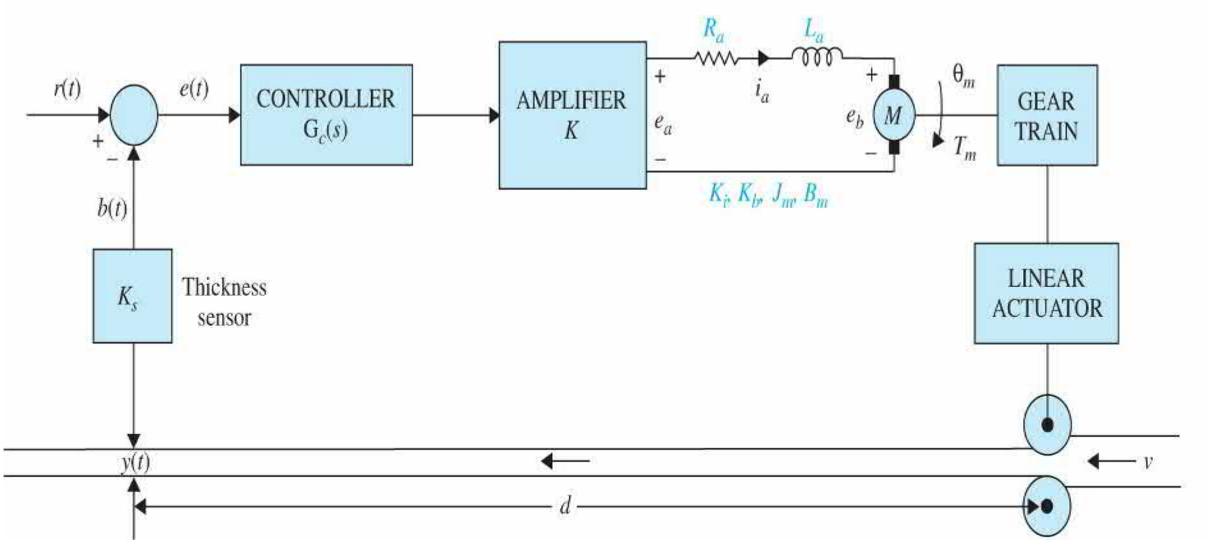


Figure 6P-25

(a) Draw a functional block diagram for the system.

(b) Derive the forward-path transfer function $Y(s)/E(s)$ and the closed-loop transfer function $Y(s)/R(s)$.

[1] It is important to note that the measured values provided here may vary from motor to motor, and you are best to conduct your own experiments to determine your system's parameter values.

[2] It is important to note that the measured values provided here may vary from motor to motor, and you are best to conduct your own experiments to determine your system's parameter values.

CHAPTER 7

Time-Domain Performance of Control Systems

In this chapter, building upon the background material discussed in Chaps. 1 to 3, we arrive at the time response of simple control systems. In order to find the time response of a control system, we first need to model the overall system dynamics and find a mathematical model that sufficiently represents the system. In many practical instances, the system is nonlinear and has to be linearized. The system could be composed of mechanical, electrical, or other subsystems. Each subsystem may have sensors and actuators to sense the environment and to interact with it. Next, using Laplace transforms, we can find the transfer function of all the subcomponents and use the block diagram approach or signal flow diagrams, to find the interactions among the system components. Finally, we can find the overall transfer function of the system and, using inverse Laplace transforms, obtain the time response of the system to a test input—normally a step input.

Learning Outcomes

After successful completion of this chapter, you will be able to

1. Analyze transient and steady-state time response of a simple control system.
2. Develop simple design criteria for manipulating the time response.
3. Determine speed and position time response of dc motors.
4. Apply basic control techniques and look at the effects of adding a simple gain or poles and zeros to the system transfer function.
5. Use MATLAB to study the time response of simple control system.

In this chapter, we look at more details of the time-response analysis, discuss **transient** and **steady-state** time response of simple control systems, and develop design criteria for manipulating the time response. In the end, we introduce basic control techniques and look at the effects of adding a simple gain or poles and zeros to the system transfer function. We also introduce simple examples of proportional, derivative, and integral controller design concepts in time domain. The controller design at this level is purely introductory and relies on observation of the time response.

Throughout the chapter various examples involving the speed and position control of a dc motor provide practical relevance of the subjects discussed here. Finally a case study is provided at the end of the chapter that looks at a practical example involving position control of a dc motor and the issues surrounding the simplification of dc motor model from a third-order system to a second order. MATLAB toolboxes have been used throughout the chapter to facilitate analysis and interpretation of various concepts discussed.

Finally, most undergraduate control courses have labs dealing with time response and control of dc motors—namely, speed response, speed control, position response, and position control. In many cases, because of high cost of control lab equipment, student exposure to test equipment is limited, and as a result, many students do not gain a practical insight into the subject. In this textbook, recognizing these limitations, we introduce the concept of **Control Lab**, which includes two classes of experiments: **SIMLab (model-based** simulation) and **LEGOLab (physical experiments).** These experiments are intended to supplement, or replace, the experimental exposure of the students in a traditional undergraduate control course.

After completing this chapter, the reader is encouraged to refer to App. A to experience practical aspects of designing a control system, using LEGOLab.

7-1 TIME RESPONSE OF CONTINUOUS-DATA SYSTEMS: INTRODUCTION

Because time is used as an independent variable in most control systems, it is usually of interest to evaluate the *state* and *output* responses with respect to time or, simply, the **time response**. In the analysis problem, a **reference (test) input signal** is applied to a system, and the **performance** of the system is evaluated by studying the system response in the time domain. For instance, if the objective of the control system is to have the output variable track the input signal, starting at some initial time and initial condition, it is necessary to compare the input and output responses as functions of time. Therefore, *in most control-system problems, the final evaluation of the performance of the system is based on the time responses*.

> In most control-system problems, the final evaluation of the performance of the system is based on the time responses.
>
> The time response of a control system is usually divided into two parts: the **transient response** and the **steady-state** response.
>
> In **stable** control systems, transient response is defined as the part of the time response that goes to zero as time becomes very large.
>
> The steady-state response is simply the part of the total response that remains after the transient has died out.

As discussed in earlier chapters, the time response of a control system is usually divided into two parts: the **transient response** and the **steady-state** response. Let $y(t)$ denote the time response of a continuous-data system; then, in general, it can be written as

$$y(t) = y_t(t) + y_{ss}(t) \tag{7-1}$$

where $y_t(t)$ denotes the transient response and $y_{ss}(t)$ denotes the steady-state response.

In **stable** control systems, *transient response* is defined as the part of the time response that goes to zero as time becomes very large. Thus, $y_t(t)$ has the property

$$\lim_{t\to\infty} y_t(t) = 0 \tag{7-2}$$

The steady-state response is simply the part of the total response that remains after the transient has died out. Thus, the steady-state response can still vary in a fixed pattern, such as a sine wave, or a ramp function that increases with time.

All real, stable control systems exhibit transient phenomena to some extent before the steady state is reached. Because inertia, mass, and inductance are unavoidable in physical systems, the response of a typical control system cannot follow sudden changes in the input instantaneously, and transients are usually observed. Therefore, *the control of the transient response is necessarily important* because it is a significant part of the dynamic behavior of the system, and the deviation between the output response and the input or the desired response, before the steady state is reached, must be closely controlled.

> The study of a stable control system in the time domain essentially involves the evaluation of the transient and the steady-state responses of the system.

The *steady-state response of a control system is also very important* because it indicates where the system output ends up when time becomes large. For a position-control system, the steady-state response when compared with the desired **reference** position gives an indication of the final accuracy of the system. In general, if the steady-state response of the output does not agree with the desired reference exactly, the system is said to have a **steady-state error**.

The study of a stable control system in the time domain essentially involves the evaluation of the transient and the steady-state responses of the system. In the design problem, specifications are usually given in terms of the transient and the steady-state performances, and controllers are designed so that the specifications are all met by the designed system.

7-2 TYPICAL TEST SIGNALS TO EVALUATE TIME-RESPONSE PERFORMANCE OF CONTROL SYSTEMS

Unlike electric networks and communication systems, the inputs to most practical control systems are not exactly known ahead of time. *In many cases, the actual inputs of a control system may vary in random fashion with respect* *to time*. For instance, in a radar-tracking system for antiaircraft missiles, the position and speed of the target to be tracked may vary in an unpredictable manner, so that they cannot be predetermined. This poses a problem for the designer because it is difficult to design a control system so that it will perform satisfactorily to all possible forms of input signals. For the purpose of analysis and design, it is necessary to assume some basic types of **test inputs** so that the **performance** of a system can be evaluated. By selecting these basic test signals properly, not only is the mathematical treatment of the problem systematized but also the response due to these inputs allows the prediction of the system's performance to other more complex inputs. In the **design** problem, **performance criteria** may be specified with respect to these **test signals** so that the system may be designed to meet the criteria. This approach is particularly useful for linear systems, since the response to complex signals can be determined by superposing those due to simple test signals.

As later discussed in Chap. 10, the response of a linear time-invariant system may also be analyzed in the frequency domain by using a test sinusoidal input. When the input frequency is swept from zero to beyond the significant range of the system characteristics, curves in terms of the amplitude ratio and phase between the input and the output are drawn as functions of frequency. In that case, it is possible to predict the time-domain behavior of the system from its frequency-domain characteristics.

To facilitate the time-domain analysis, the following deterministic test signals are used.

> The **step** function is very useful as a test signal because its initial instantaneous jump in amplitude reveals a great deal about a system's quickness in responding to inputs with abrupt changes.

Step-Function Input

The step-function is the most important and widely used input, and it represents an instantaneous change in the reference input. For example, if the input is an angular position of a motor shaft, a step input represents the sudden rotation of the shaft. The mathematical representation of a step function or magnitude *R* is

$$\begin{aligned} r(t) &= R \quad t \geq 0 \\ &= 0 \quad t < 0 \end{aligned} \tag{7-3}$$

where *R* is a real constant. Or

$$r(t) = Ru_s(t) \tag{7-4}$$

where *us*(*t*) is the unit-step function. The step function as a function of time is shown in Fig. 7-1*a*. The step function is very useful as a test signal because its initial instantaneous jump in amplitude reveals a great deal about a system’s quickness in responding to inputs with abrupt changes. Also, because the step function contains, in principle, a wide band of frequencies in its spectrum, as a result of the jump discontinuity, it is equivalent to the application of numerous sinusoidal signals with a wide range of frequencies.

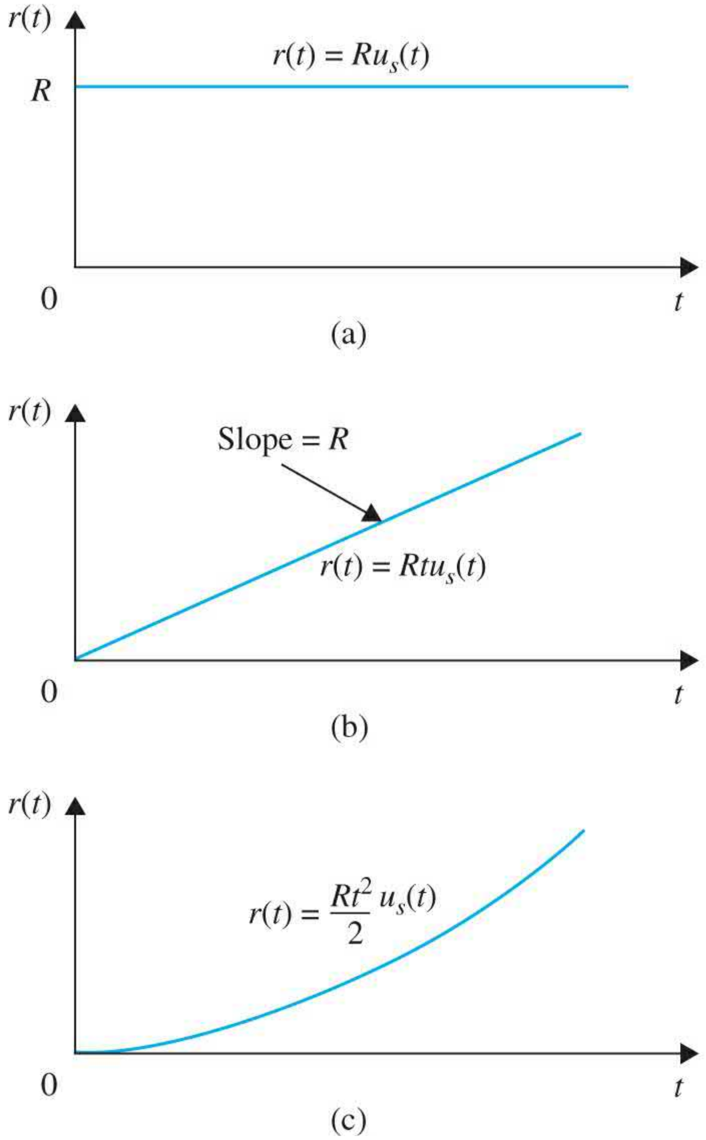


Figure 7-1 Basic time-domain test signals for control systems. (*a*) Step function. (*b*) Ramp function. (*c*) Parabolic function.

Ramp-Function Input

The ramp function is a signal that changes constantly with time. Mathematically, a ramp function is represented by

$$r(t) = Rtu_s(t) \tag{7-5}$$

where R is a real constant. The ramp function is shown in Fig. 7-1*b*. If the input variable represents the angular displacement of a motor shaft, the ramp input denotes the constant-speed rotation of the shaft. The ramp function has the ability to test how the system would respond to a signal that changes linearly with time.

Parabolic-Function Input

The parabolic function represents a signal that is one order faster than the ramp function. Mathematically, it is represented as

$$r(t) = \frac{Rt^2}{2} u_s(t) \tag{7-6}$$

where R is a real constant and the factor ½ is added for mathematical convenience because the Laplace transform of $r(t)$ is simply R/s^3. The graphical representation of the parabolic function is shown in Fig. 7-1*c*.

These signals all have the common feature that they are simple to describe mathematically. From the step function to the parabolic function, the signals become progressively faster with respect to time. In theory, we can define signals with still higher rates, such as t^3, which is called the *jerk function*, and so forth. However, in reality, we seldom find it necessary or feasible to use a test signal faster than a parabolic function.

7-3 THE UNIT-STEP RESPONSE AND TIME-DOMAIN SPECIFICATIONS

As defined earlier, the transient portion of the time response is the part that goes to zero as time becomes large. Nevertheless, the transient response of a control system is necessarily important because both the amplitude and the time duration of the transient response must be kept within tolerable or prescribed limits. For example, in the automobile idle-speed control system described in Chap. 1, in addition to striving for a desirable idle speed in the steady state, the transient drop in engine speed must not be excessive, and the recovery in speed should be made as quickly as possible. For linear control systems, the characterization of the transient response is often done by use of the **unit-step** function $us(t)$ as the input. *The response of a control system when the input is a unit-step function is called the unit-step response.* Figure

7-2 illustrates a typical unit-step response of a linear control system. With reference to the unit-step response, performance criteria commonly used for the characterization of linear control systems in the time domain are defined as follows:

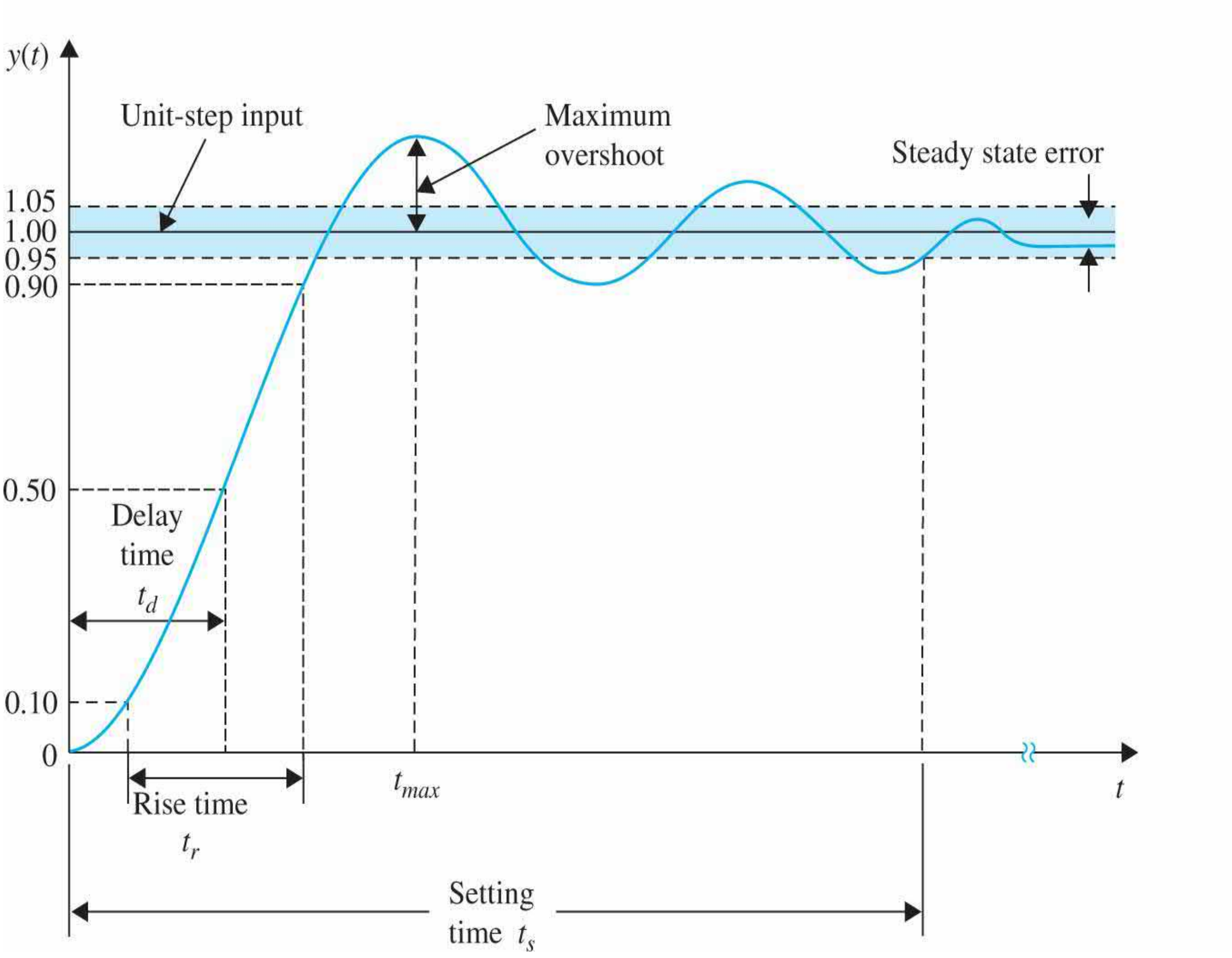


Figure 7-2 Typical unit-step response of a control system illustrating the time-domain specifications.

1. **Maximum overshoot**. Let $y(t)$ be the unit-step response. Let y_{max} denote the maximum value of $y(t)$; y_{ss}, the steady-state value of $y(t)$; and $y_{max} \geq y_{ss}$. The maximum overshoot of $y(t)$ is defined as

$$\text{Maximum overshoot} = y_{\max} - y_{ss} \tag{7-7}$$

For linear control systems, the characterization of the transient response is often done by use of the **unit-step** function $u_s(t)$ as the input.

The maximum overshoot is often represented as a percentage of the final value of the step response; that is,

$$\text{Percent maximum overshoot} = \frac{\text{Maximum overshoot}}{y_{ss}} \times 100\% \qquad (7\text{-}8)$$

The maximum overshoot is often used to measure the relative stability of a control system. A system with a large overshoot is usually undesirable. For design purposes, the maximum overshoot is often given as a time-domain specification. The unit-step illustrated in Fig. 7-2 shows that the maximum overshoot occurs at the first overshoot. For some systems, the maximum overshoot may occur at a later peak, and, if the system transfer function has an odd number of zeros in the right-half *s*-plane, a negative undershoot may even occur[4,5] —see Secs. 7-9-3 and 7-9-4.

2. **Delay time.** The delay time t_d is defined as the time required for the step response to reach 50 percent of its final value. This is shown in Fig. 7-2.
3. **Rise time.** The rise time t_r is defined as the time required for the step response to rise from 10 to 90 percent of its final value, as shown in Fig. 7-2. An alternative measure is to represent the rise time as the reciprocal of the slope of the step response at the instant that the response is equal to 50 percent of its final value.
4. **Settling time.** The settling time t_s is defined as the time required for the step response to decrease and stay within a specified percentage of its final value. A frequently used figure is 5 percent.
5. **Steady-state error**. The steady-state error of a system response is defined as the discrepancy between the output and the reference input when the steady state ($t \to \infty$) is reached.

The first four quantities, just defined, give a direct measure of the transient characteristics of a control system in terms of the unit-step response. These time-domain specifications are relatively easy to measure when the step response is well defined, as shown in Fig. 7-2. Analytically, these quantities are difficult to establish, except for simple systems lower than the third order.

It should be pointed out that the steady-state error may be defined for any test signal such as a step function, ramp function, parabolic function, or even a sinusoidal input, although Fig. 7-2 only shows the error for a step input.

7-4 TIME RESPONSE OF A PROTOTYPE FIRST-ORDER SYSTEM

Recalling the **prototype first-order system** that was discussed in Chap. 3:

$$\frac{dy(t)}{dt}+\frac{1}{\tau}y(t)=\frac{1}{\tau}u(t) \tag{7-9}$$

Time constant is a measure of how fast the system responds to an input.

where τ is known as the **time constant** of the system, which is a measure of how fast the system responds to initial conditions or external excitations. For a test unit-step input

$$u(t)=u_s(t)=\begin{cases} 0, & t<0, \\ 1, & t\geq 0 \end{cases} \tag{7-10}$$

If $y(0)=\dfrac{dy(0)}{dt}=0, \mathcal{L}(y(t))=U(s)=\dfrac{1}{s}$ and $\mathcal{L}(y(t))=Y(s)$,

then

$$\frac{Y(s)}{U(s)} = \frac{1/\tau}{s+1/\tau} \tag{7-11}$$

where $s = -1/\tau$ is the single pole of the transfer function. Using inverse Laplace transform, the time response of Eq. (7-9) is

$$y(t) = 1 - e^{-t/\tau} \tag{7-12}$$

where $t = \tau$ is the time for $y(t)$ to reach 63 percent of its final value of $\lim_{t\to\infty} y(t) = 1$.

See MATLAB Toolbox 3-4-1.

Using MATLAB Toolbox 3-4-1, we can obtain the time response shown in Eq. (7-12). Figure 7-3 shows typical unit-step responses of $y(t)$ for a two arbitrary values of τ. As the value of time constant τ decreases, the system response becomes faster. Notice from Fig. 7-3, that with the increase in the time constant, the pole $s = -1/\tau$ moves further to the left in the s-plane. Finally, for any positive τ values, the pole will always stay in the left-half s-plane, as shown in Fig. 7-4, and the system is always **stable**.

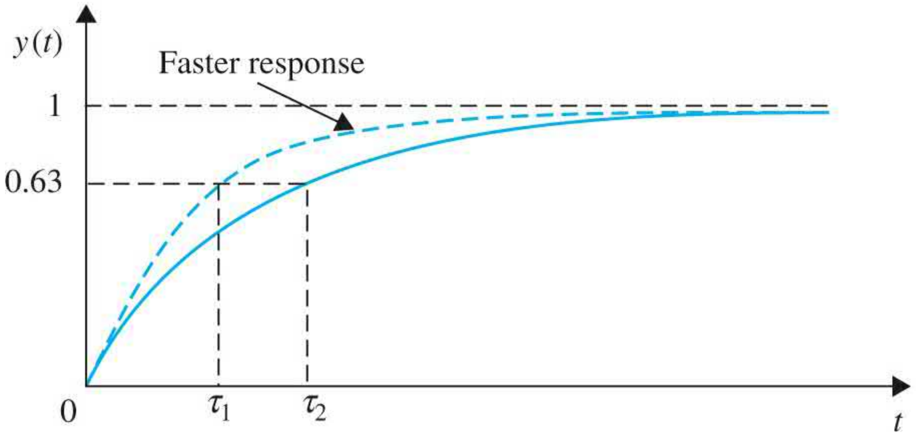


Figure 7-3 Unit-step response of a prototype first-order system.

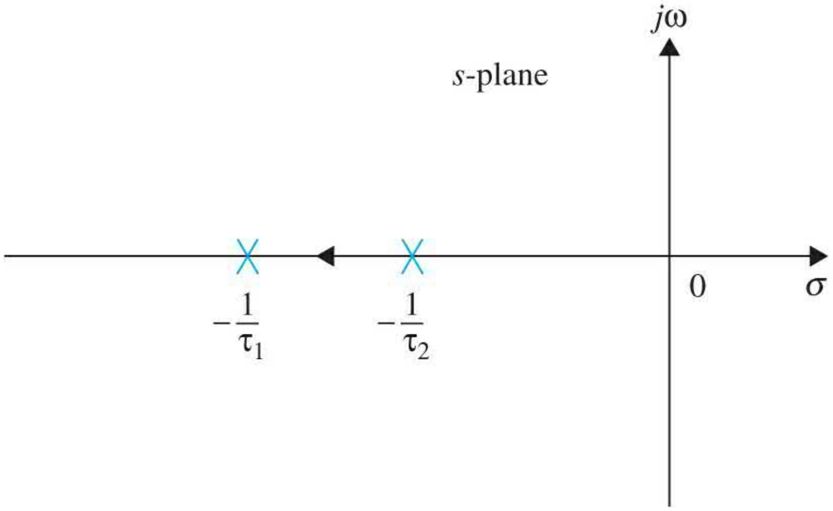


Figure 7-4 Pole position of the transfer function, in Eq. (7-12), of a prototype first-order system as the system time constant decreases.

EXAMPLE 7-4-1 For the dc motor in Sec. 7-4-1, as shown in Fig. 7-5, examine the effect of increased damping applied through a magnetic brake on the speed response. Note in this case, the magnetic brake may also be considered as an external load. But for simplicity, we consider it to strictly change the damping of the system.

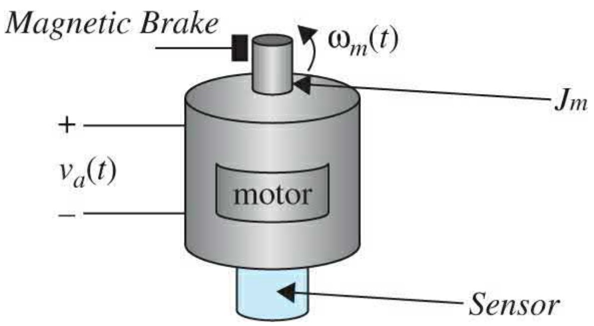


Figure 7-5 An armature-controlled dc motor.

Speed Response of a DC Motor

As discussed in Sec. 7-4-1, the ratio L_a/R_a, the *motor electric-time constant* denoted by τ_e, is very small and may be neglected, resulting in the simplified block diagram of the system, as shown in Fig. 7-6.

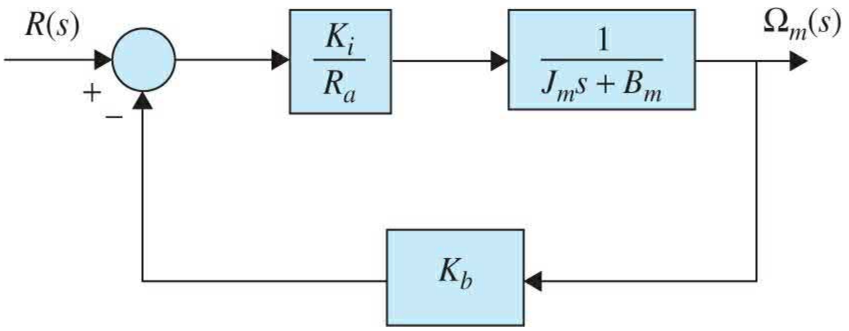


Figure 7-6 Simplified block diagram of a dc motor for speed response assuming a negligible electric time constant.

Thus, the speed of the motor shaft in Laplace domain is modeled as

$$\Omega_m(s) = \frac{\dfrac{K_i}{R_a J_m}}{s + \dfrac{K_i K_b + R_a B_m}{R_a J_m}} R(s) \qquad (7\text{-}13)$$

or

$$\Omega_m(s) = \frac{K_{\text{eff}}}{\tau_m s + 1} R(s) \tag{7-14}$$

where $K_{eff} = K_i/(R_a B_m + K_i K_b)$ is the motor gain constant, and $\tau_m = R_a J_m/(R_a B_m + K_i K_b)$ is the motor mechanical time constant.

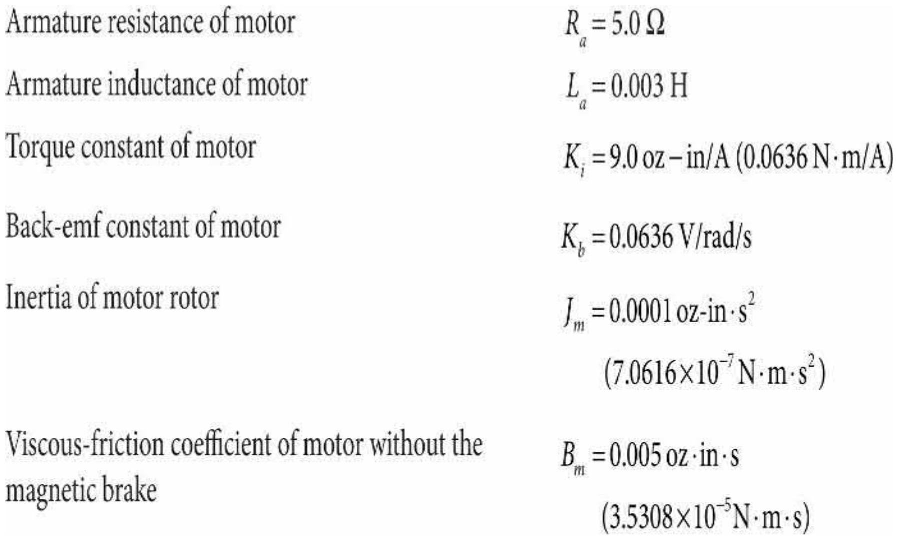
To find the response $\omega_m(t)$, for a unit step input *voltage* such that $r(t)$ or $R(s)=1/s$,

$$\omega_m(t) = \frac{AK_i}{K_i K_b + R_a B_m}(1 - e^{-t/\tau_m}) \tag{7-15}$$

In this case, the motor mechanical time constant τ_m is reflective of how fast the motor is capable of overcoming its own inertia J_m to reach a steady state or constant speed dictated by voltage $R(s)$. As τ_m increases, the approach to steady state takes longer. See Fig. 7-34 for the typical time response associated with Eq. (7-50). From Eq. (7-15), the speed final value is

$$\omega_{fv} = \lim_{t\to\infty} \omega(t) = \frac{K_i}{K_i K_b + R_a B_m} \tag{7-16}$$

Let’s select a dc motor with given parameters:



Armature resistance of motor	$R_a = 5.0\ \Omega$
Armature inductance of motor	$L_a = 0.003$ H
Torque constant of motor	$K_i = 9.0$ oz − in/A (0.0636 N·m/A)
Back-emf constant of motor	$K_b = 0.0636$ V/rad/s
Inertia of motor rotor	$J_m = 0.0001$ oz-in·s^2 (7.0616×10^{-7} N·m·s^2)
Viscous-friction coefficient of motor without the magnetic brake	$B_m = 0.005$ oz·in·s (3.5308×10^{-5} N·m·s)

Application of a magnetic brake increases the viscous damping in the system. Using Toolbox 7-4-1 (in Chap. 6), we now examine the damping effect on the speed response of the motor. Table 7-1 illustrates the motor mechanical time constant τ_m and final value ω_{fv} for three damping values in response to a unit step input. The three-speed responses are shown in Fig. 7-7. As shown with higher damping, the motor response is faster, while the final value decreases. This obviously meets the common sense notion that by using the brake the final speed of the motor drops.

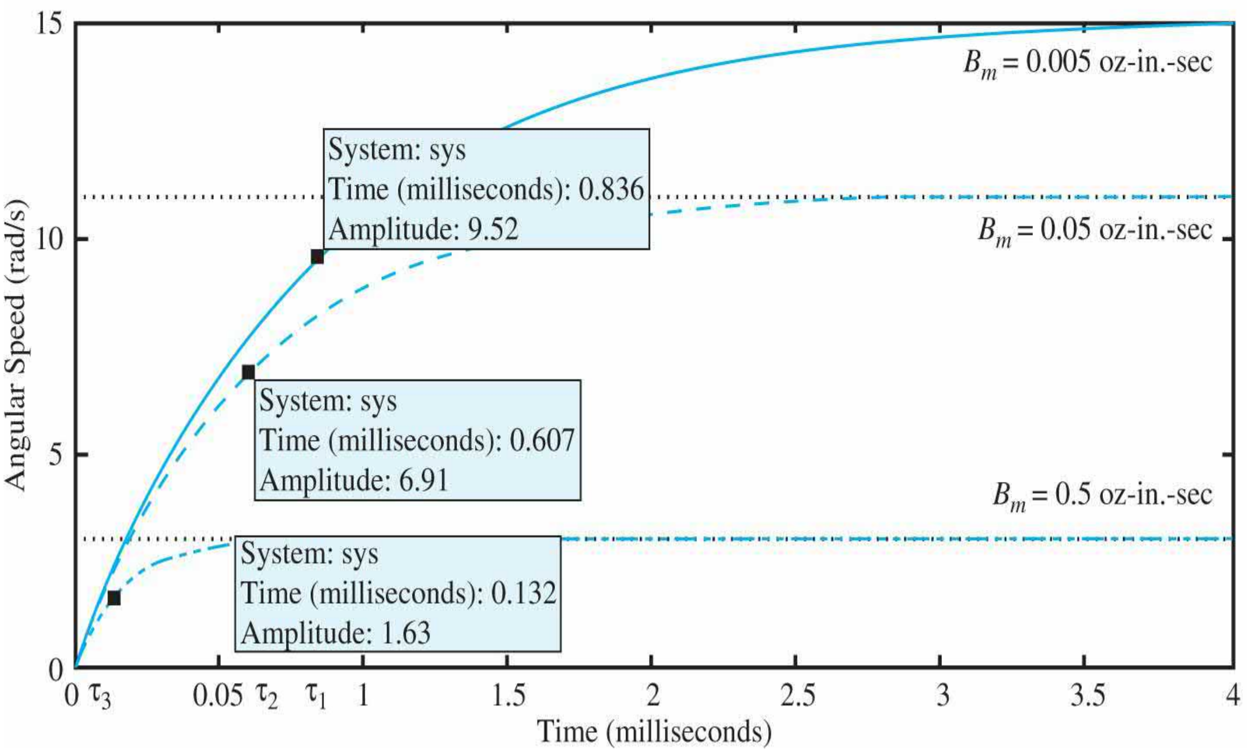
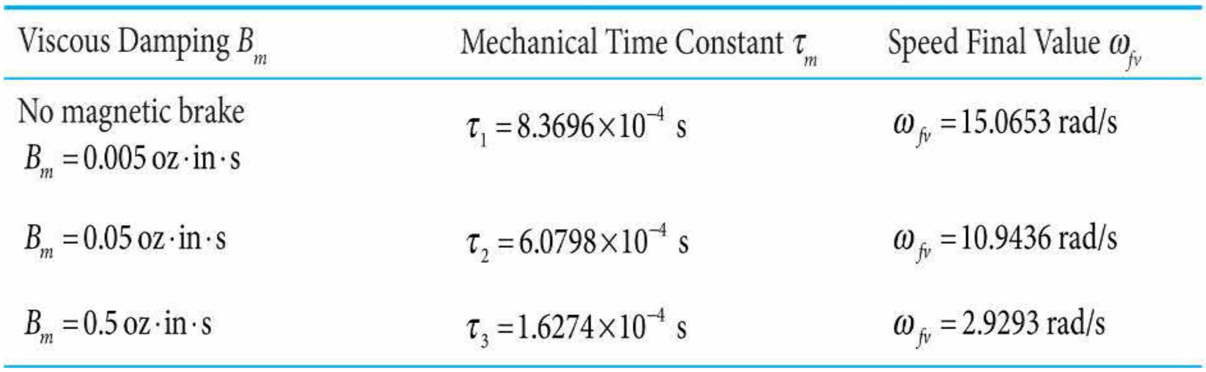


Figure 7-7 Speed response of dc motor in Example 7-4-1 to a unit step input. (*a*) Solid line represents no magnetic brake B_m = 0.005 oz·in·s. (*b*) Dashed line represents the effect of a magnetic brake when B_m = 0.05 oz·in·s. (*c*) Dot-dashed line represent the response to a more dominant damping B_m = 0.5 oz·in·s.

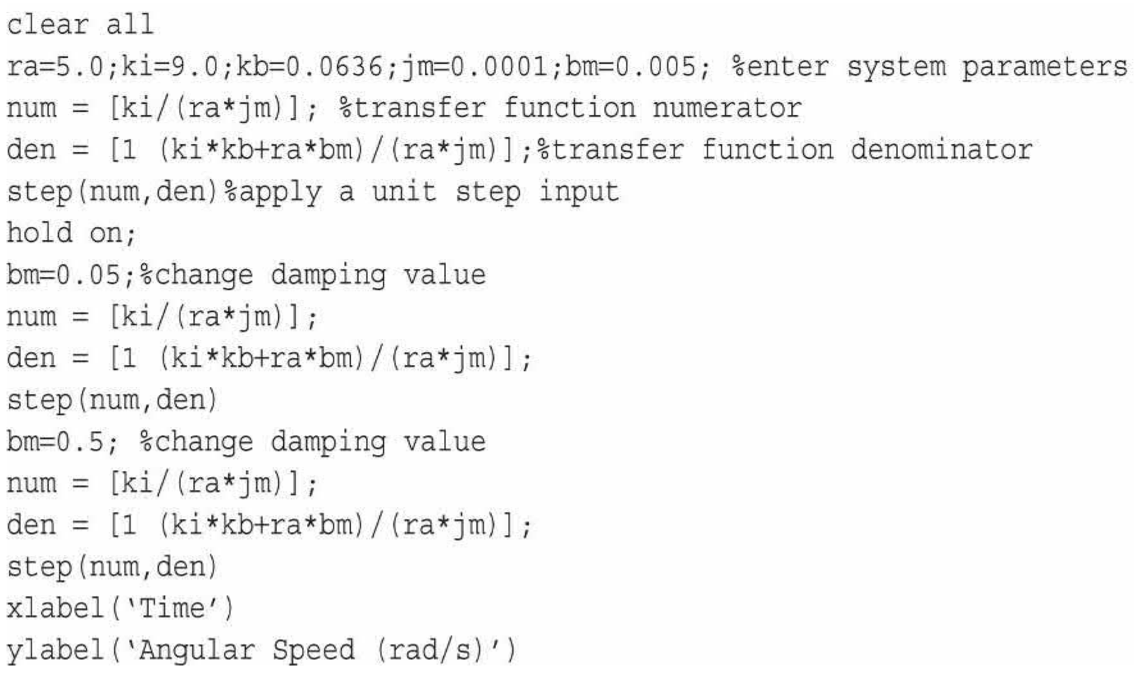
TABLE 7-1 Motor Mechanical Time Constant and Final Speed for Three Different Damping Values



Viscous Damping B_m	Mechanical Time Constant τ_m	Speed Final Value ω_{fv}
No magnetic brake $B_m = 0.005\ \text{oz} \cdot \text{in} \cdot \text{s}$	$\tau_1 = 8.3696 \times 10^{-4}$ s	$\omega_{fv} = 15.0653$ rad/s
$B_m = 0.05\ \text{oz} \cdot \text{in} \cdot \text{s}$	$\tau_2 = 6.0798 \times 10^{-4}$ s	$\omega_{fv} = 10.9436$ rad/s
$B_m = 0.5\ \text{oz} \cdot \text{in} \cdot \text{s}$	$\tau_3 = 1.6274 \times 10^{-4}$ s	$\omega_{fv} = 2.9293$ rad/s

Toolbox 7-4-1

Speed response of Example 7-4-1 using MATLAB.



```
clear all
ra=5.0;ki=9.0;kb=0.0636;jm=0.0001;bm=0.005; %enter system parameters
num = [ki/(ra*jm)]; %transfer function numerator
den = [1 (ki*kb+ra*bm)/(ra*jm)];%transfer function denominator
step(num,den)%apply a unit step input
hold on;
bm=0.05;%change damping value
num = [ki/(ra*jm)];
den = [1 (ki*kb+ra*bm)/(ra*jm)];
step(num,den)
bm=0.5; %change damping value
num = [ki/(ra*jm)];
den = [1 (ki*kb+ra*bm)/(ra*jm)];
step(num,den)
xlabel(‘Time’)
ylabel(‘Angular Speed (rad/s)’)
```

7-5 TRANSIENT RESPONSE OF A PROTOTYPE SECOND-ORDER SYSTEM

Although true second-order control systems are rare in practice, their analysis generally helps to form a basis for the understanding of analysis and design of higher-order systems, especially the ones that can be approximated by second-order systems.

Consider that a second-order control system with unity feedback is represented by the block diagram shown in Fig. 7-8. The open-loop transfer function of the system is

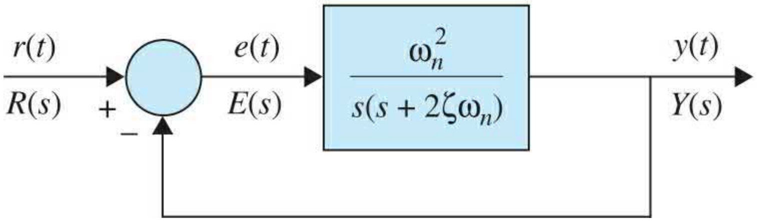


Figure 7-8 Prototype second-order control system.

$$G(s)=\frac{Y(s)}{E(s)}=\frac{\omega_n^2}{s(s+2\zeta\omega_n)} \tag{7-17}$$

where ζ and ω_n are real constants. The closed-loop transfer function of the system is

$$\frac{Y(s)}{R(s)}=\frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2} \tag{7-18}$$

As earlier discussed in Chaps. 2 and 3, the system in Fig. 7-8 with the closed-loop transfer function given by Eq. (7-18) is defined as the **prototype second-order system**.

7-5-1 Damping Ratio and Natural Frequency

The characteristic equation of the prototype second-order system is obtained by setting the denominator of Eq. (7-18) to zero:

$$\Delta(s)=s^2+2\zeta\omega_n s+\omega_n^2=0 \tag{7-19}$$

The two roots of Eq. (7-19) are the poles of the transfer function in Eq. (7-18), and can be expressed as

$$\begin{aligned} s_1,s_2 &= -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \\ &= -\sigma \pm j\omega \end{aligned} \tag{7-20}$$

where

$$\sigma=\zeta\omega_n \tag{7-21}$$

and

$$\omega=\omega_n\sqrt{1-\zeta^2} \tag{7-22}$$

Figure 7-9 illustrates the relationships among the location of the characteristic equation roots and σ, ζ, ω_n, and ω, when $0 < \zeta < 1$. For the complex-conjugate roots shown,

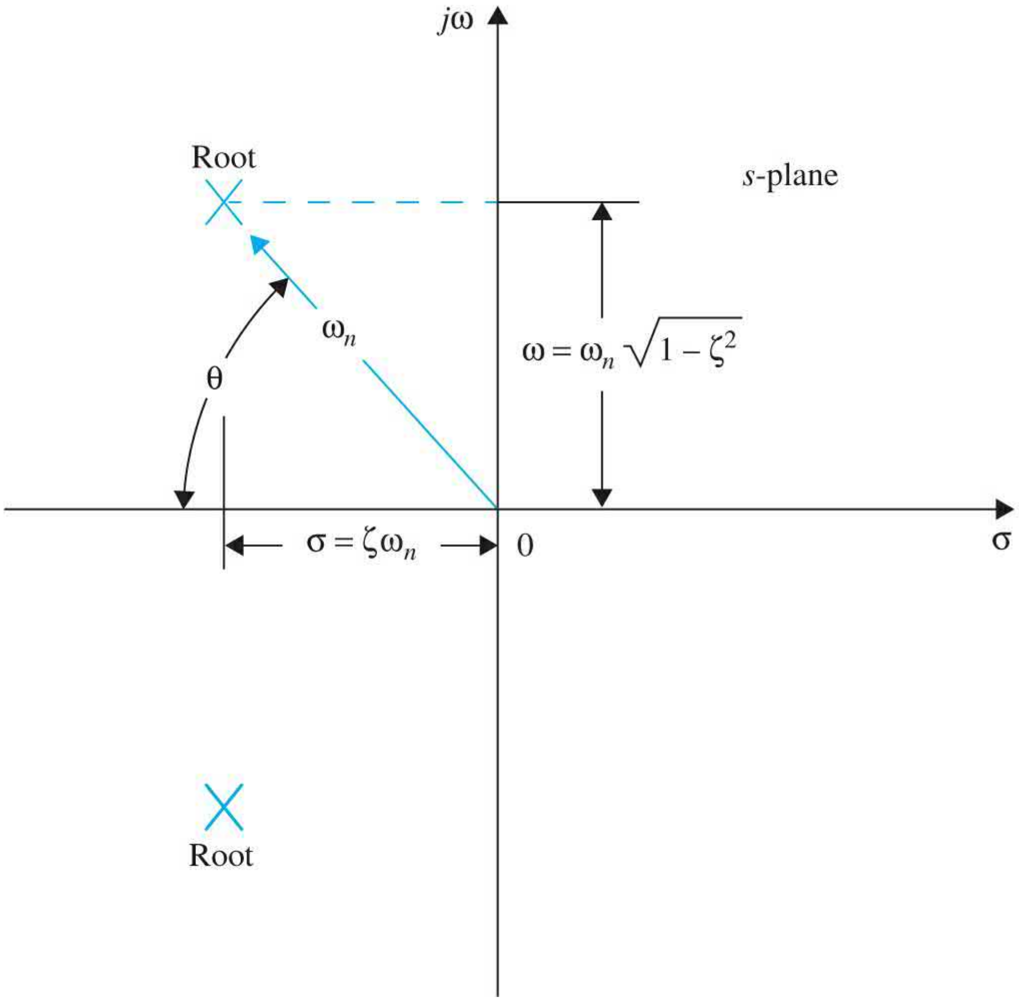


Figure 7-9 Relationship's among the characteristic-equation roots of the prototype second-order system and σ, ζ, ω_n, and ω.

- ω_n is the radial distance from the roots to the origin of the *s*-plane.
- σ is the real part of the roots.
- ω is the imaginary part of the roots.
- ζ is the cosine of the angle between the radial line to the roots and the negative axis when the roots are in the left-half *s*-plane, or $\zeta = \cos\theta$

The physical significance of ζ and σ, in Eq. (7-21), is now investigated. As seen from Chap. 3, the **damping constant** σ appears as the constant that is multiplied to t in the exponential term of $y(t)$. Therefore, σ controls the rate of rise or decay of the unit-step response $y(t)$. In other words, σ controls the "damping" of the system. The inverse of σ, $1/\sigma$, is proportional to **the time constant** of the system.

> Damping ratio ζ has a direct impact on the overshoot, while natural frequency ω_n has a direct effect on the rise time, delay time, and settling time.

When the two roots of the characteristic equation are real and equal, we called the system critically damped. From Eq. (7-20), we see that critical damping occurs when $\zeta = 1$. Under this condition, the damping factor is simply $\sigma = \omega_n$. Thus, we can regard ζ as the **damping ratio** (unitless); that is,

$$\zeta = \cos\theta = \text{Damping ratio} = \frac{\sigma}{\omega_n} = \frac{\text{Actual damping factor}}{\text{Damping factor at critical damping}} \qquad (7\text{-}23)$$

The parameter ω_n is defined as the **natural frequency**. As seen from Eq. (7-20), when $\zeta = 0$, the damping is zero, the roots of the characteristic equation are imaginary, and Eq. (7-27) in Table 7-2 shows that the unit-step response is purely sinusoidal. Therefore, ω_n corresponds to the frequency of the undamped sinusoidal response. Equation (7-20) shows that, when $0 < \zeta < 1$, the imaginary part of the roots has the magnitude of ω . When $\zeta \neq 0$, the response of $y(t)$ is not a periodic function, and ω defined in Eq. (7-22) is not a frequency. However, for the purpose of reference, ω in Eq. (7-22) is sometimes defined as the **conditional frequency**, or the **damped frequency**.

The unit step response classification, for the system represented by transfer function in Eq. (7-18), with respect to the damping ratio ζ is made in Table 7-2. Figure 7-10 illustrates typical unit-step responses that correspond to the various root locations.

TABLE 7-2 Classification of Prototype Second-Order System Based on Damping Ratio

$0<\zeta<1$	$s_1, s_2 = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2}$ $y(t) = 1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}\sin\left[\left(\omega_n\sqrt{1-\zeta^2}\right)t + \cos^{-1}\zeta\right]$ (7-24)	Underdamped
$\zeta=1$	$s_1, s_2 = -\omega_n$ $y(t) = 1 - e^{-\omega_n t} - te^{-\omega_n t}$ (7-25)	Critically damped
$\zeta>1$	$s_1, s_2 = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2-1}$ $1 - \frac{e^{-\omega_n\zeta t}}{\sqrt{\zeta^2-1}}\left(\cosh\left(\omega_n\sqrt{\zeta^2-1}\right)t + \zeta\sinh\left(\omega_n\sqrt{\zeta^2-1}\right)t\right)$ (7-26)	Overdamped
$\zeta=0$	$s_1, s_2 = \pm j\omega_n$ $y(t) = 1 - \cos\omega_n t$ (7-27)	Undamped
$\zeta<0$	$s_1, s_2 = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2}$ (7-28) *(unstable response)*	Negatively damped

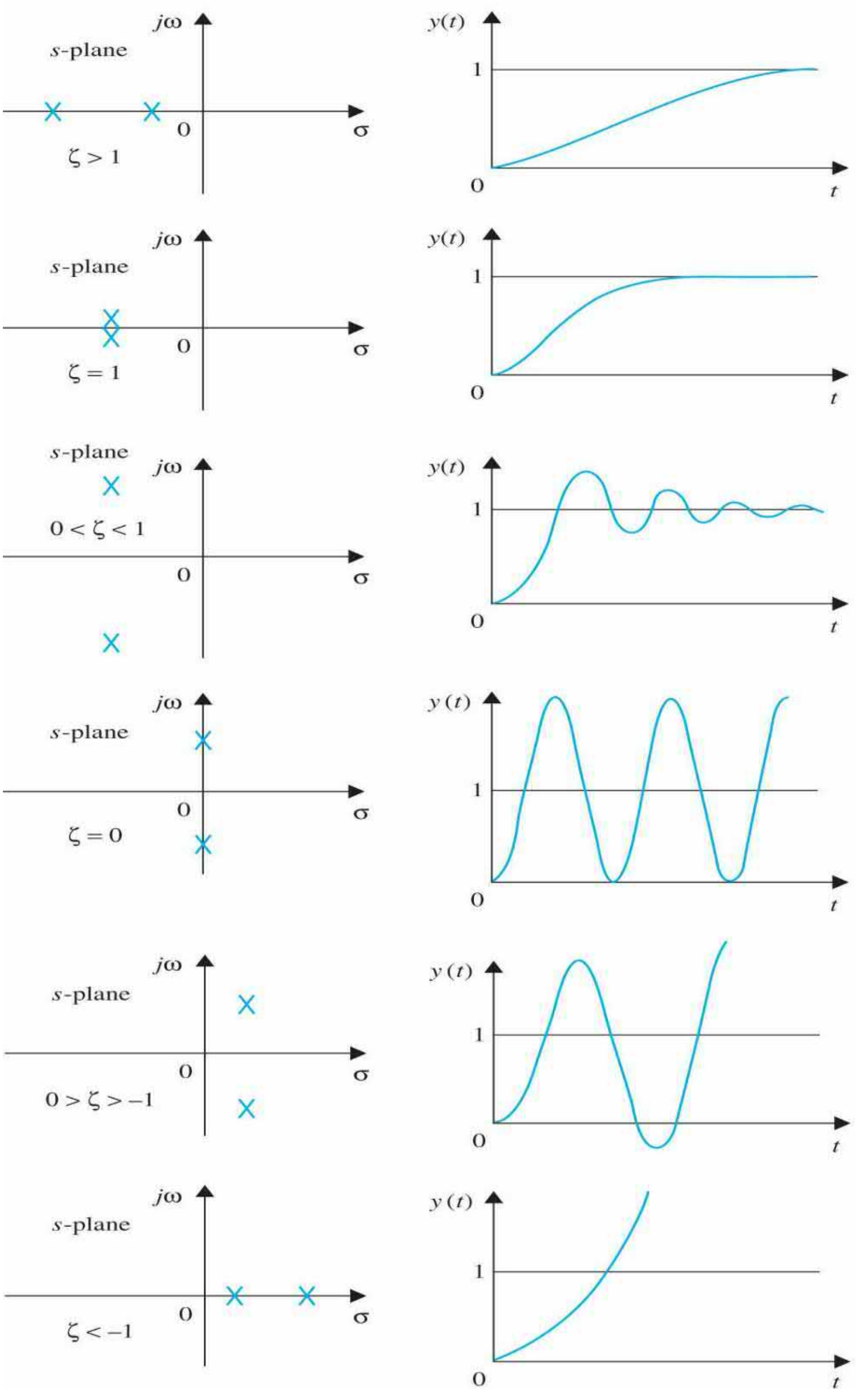
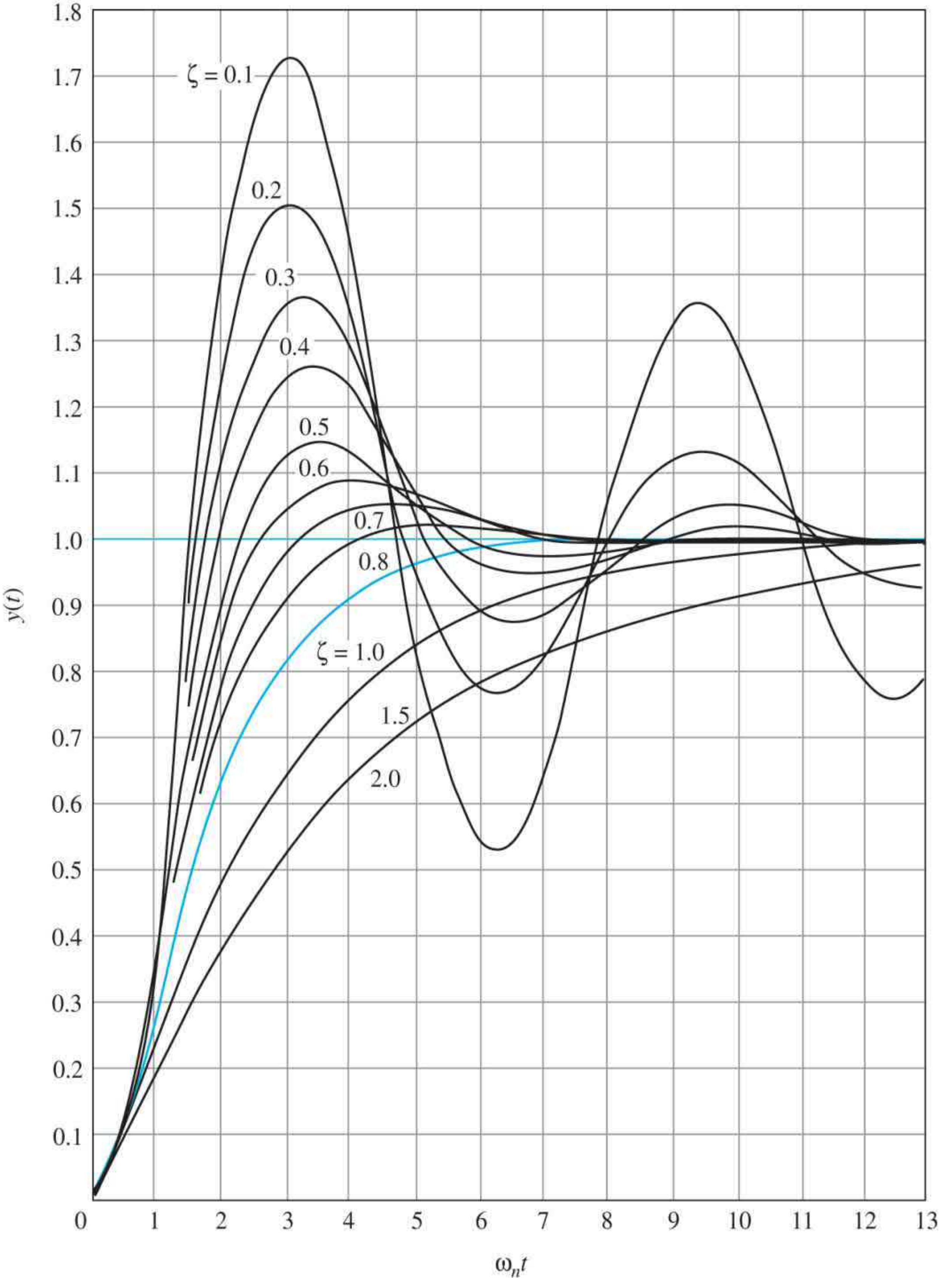

jω
s-plane
0
σ
ζ > 1
y(t)
1
0
t
jω
s-plane
0
σ
ζ = 1
y(t)
1
0
t
s-plane
jω
0 < ζ < 1
0
σ
y(t)
1
0
t
jω
s-plane
0
σ
ζ = 0
y(t)
1
0
t
jω
s-plane
0
σ
0 > ζ > −1
y(t)
1
0
t
jω
s-plane
0
σ
ζ < −1
y(t)
1
0
t

Figure 7-10 Step-response comparison for various characteristic equation root locations in the *s*-plane.

In most control applications, when the system is stable, the transient response corresponding to $\zeta > 0$ are of great interest. Figure 7-11 shows the unit-step responses of Eqs. (7-24) through (7-26) plotted as functions of the normalized time $\omega_n t$ for various values of ζ. As seen, the response becomes more oscillatory with larger overshoot as ζ decreases. When $\zeta \geq 1$, the step response does not exhibit any overshoot; that is, $y(t)$ never exceeds its final value during the transient. The responses also show that ω_n has a direct effect on the rise time, delay time, and settling time but does not affect the overshoot.



y(t)
1.8
1.7
1.6
1.5
1.4
1.3
1.2
1.1
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
$\omega_n t$
$\zeta = 0.1$
0.2
0.3
0.4
0.5
0.6
0.7
0.8
$\zeta = 1.0$
1.5
2.0

Figure 7-11 Unit-step responses of the prototype second-order system with various damping ratios.

As mentioned earlier, overshoot, rise time, delay time and settling time are quantities that define transient response **performance** of a control system, and will be studied in more detail in the following sections.

Finally, it is important to establish a relation between the system pole locations and the time response of the system. Figure 7-12 shows in the *s*-plane (a) the constant-ω_n loci, (b) the constant-ζ loci, (c) the constant-σ loci, and (d) the constant-ω loci. Regions in the *s*-plane are identified with the system damping as follows:

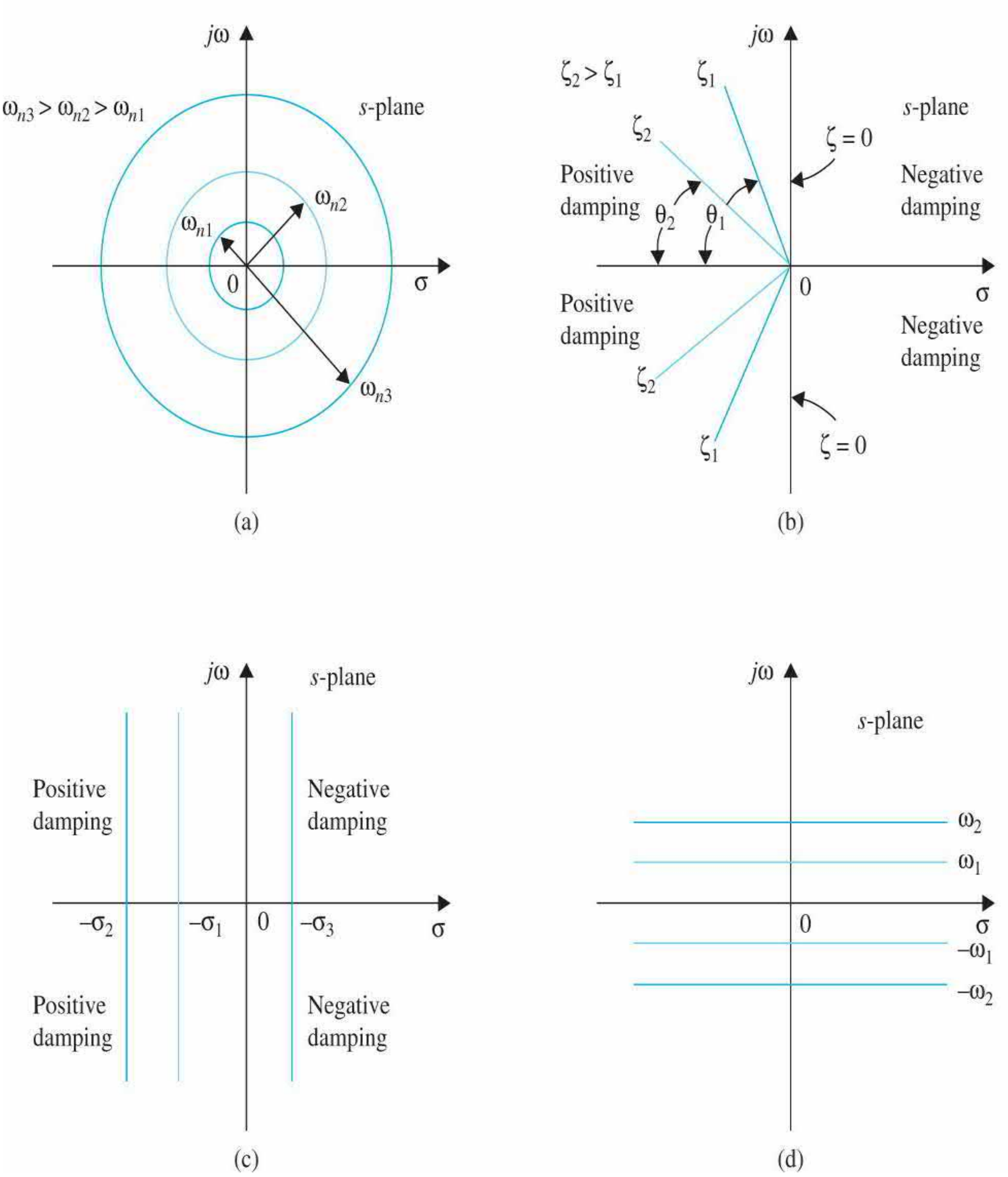


Figure 7-12 (*a*) Constant-natural frequency loci. (*b*) Constant-damping-ratio loci. (*c*) Constant-damping-factor loci. (*d*) Constant-conditional-frequency loci.

- The **left-half s-plane** corresponds to **positive damping**; that is, the damping factor or damping ratio is positive. Positive damping causes the unit-step response to settle to a constant final value in the steady state due to the negative exponent of exp($-\zeta\omega_n t$). The system is **stable**.

- The **right-half s-plane** corresponds to **negative damping**. Negative damping gives a response that grows in magnitude without bound, and the system is **unstable**.
- The imaginary axis corresponds to zero damping ($\sigma = 0$ or $\zeta = 0$). **Zero damping** results in a **sustained oscillation** response, and the system is marginally stable or marginally unstable.

Thus, we have demonstrated with the help of the simple prototype second-order system that the location of the characteristic equation roots plays an important role in the transient response of the system.

7-5-2 Maximum Overshoot ($0 < \zeta < 1$)

Maximum overshoot is one of the most important transient response performance criteria for a control system. For example, in a pick and place robot arm, overshoot represents how far the robot end effector moves away from the final drop-off destination. Generally, the transient response of a prototype second-order system with overshoot is oscillatory (i.e., $0 < \zeta < 1$)—in special circumstances the zeros of a transfer function may also cause overshoot without oscillations (see Sec. 7-10).

When $0 < \zeta < 1$ (underdamped response), for a unit-step function input, $R(s) = 1/s$, the output response of the system is obtained by taking the inverse Laplace transform of the output transform:

$$Y(s) = \frac{\omega_n^2}{s(s^2 + 2\zeta\omega_n s + \omega_n^2)} \tag{7-29}$$

In most cases, step input transient response of a prototype second-order system with overshoot is oscillatory (i.e., $0 < \zeta < 1$).

A critically damped or overdamped transfer function with a zero, in special circumstances, may exhibit overshoot without oscillations—see Sec. 7-7.

This can be done by referring to the Laplace transform table in App. D. The result is

$$y(t)=1-\frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}\sin\left(\omega_n\sqrt{1-\zeta^2}t-\cos^{-1}\zeta\right)\quad t\geq 0 \tag{7-30}$$

The exact relation between the damping ratio and the amount of overshoot can be obtained by taking the derivative of Eq. (7-30) with respect to *t* and setting the result to zero. Thus,

$$\frac{dy(t)}{dt}=\frac{\omega_n e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}\left[\zeta\sin(\omega t+\theta)-\sqrt{1-\zeta^2}\cos(\omega t+\theta)\right]\quad t\geq 0 \tag{7-31}$$

where *ω* and *θ* are defined in Eqs. (7-22) and (7-23), respectively. From Fig. 7-9 it is easy to see

$$\zeta=\cos\theta \tag{7-32}$$

$$\sqrt{1-\zeta^2}=\sin\theta \tag{7-33}$$

Hence, can show that the quantity inside the square bracket in Eq. (7-31) can be reduced to sin *ωt*. Thus, Eq. (7-31) is simplified to

$$\frac{dy(t)}{dt}=\frac{\omega_n}{\sqrt{1-\zeta^2}}e^{-\zeta\omega_n^t}\sin\omega_n\sqrt{1-\zeta^2}t\quad t\geq 0 \tag{7-34}$$

Setting *dy*(*t*)/*dt* to zero, we have the solutions: *t* = ∞ and

$$\omega_n\sqrt{1-\zeta^2}t=n\pi\quad n=0,1,2,\ldots \tag{7-35}$$

from which we get

$$t = \frac{n\pi}{\omega_n \sqrt{1-\zeta^2}} \qquad n = 0, 1, 2, \ldots \tag{7-36}$$

The solution at $t = \infty$ is the maximum of $y(t)$ only when $\zeta \geq 1$. For the unit-step responses shown in Fig. 7-13, the first overshoot is the maximum overshoot. This corresponds to $n = 1$ in Eq. (7-36). Thus, the time at which the maximum overshoot occurs is

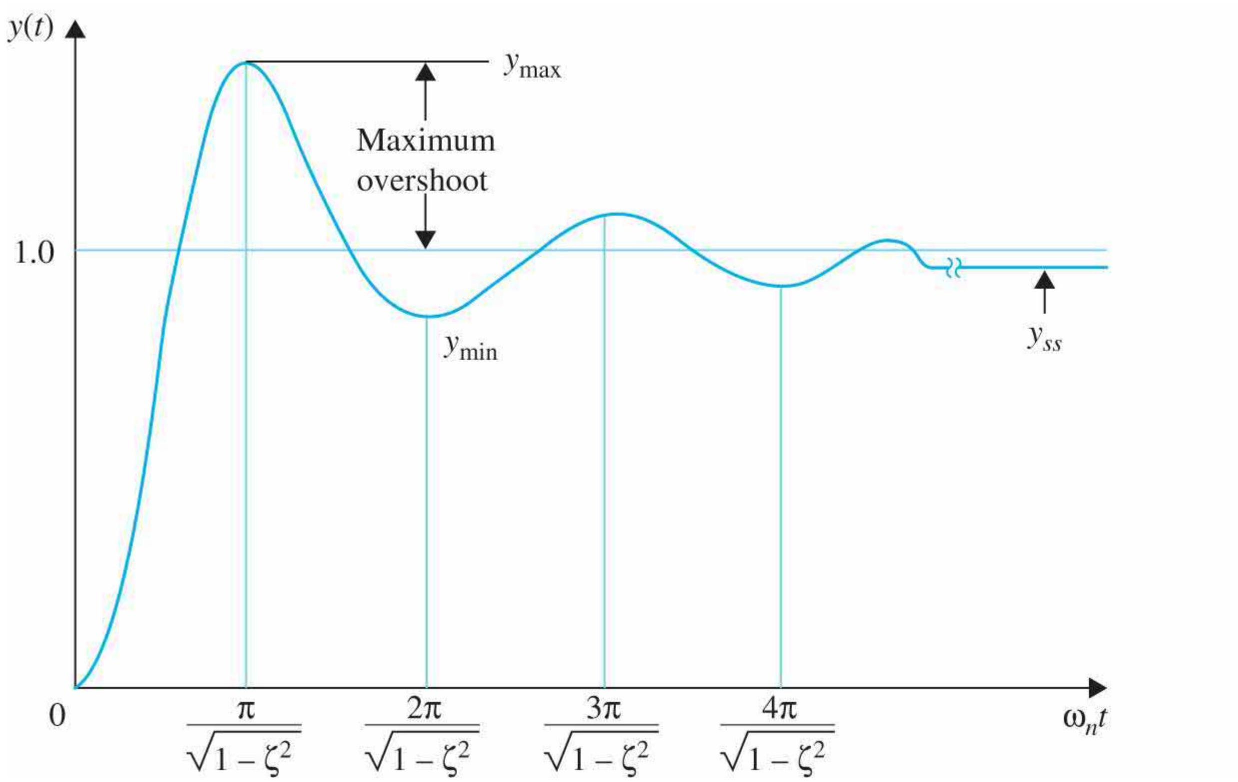


Figure 7-13 Unit-step response illustrating that the maxima and minima occur at periodic intervals.

$$t_{\max} = \frac{\pi}{\omega_n \sqrt{1-\zeta^2}} \tag{7-37}$$

With reference to Fig. 7-13, the overshoots occur at odd values of n, that is, $n = 1, 3, 5, \ldots$, and the undershoots occur at even values of n. Whether the extremum is an overshoot or an undershoot, the time at which it occurs is given by Eq. (7-36). It should be noted that, *although the unit-step response for $\zeta = 0$ is not periodic, the overshoots and the undershoots of the response* *do occur at periodic intervals, as shown in Fig. 7-13*.

The magnitudes of the overshoots and the undershoots can be determined by substituting Eq. (7-36) into Eq. (7-30). The result is

$$t_{\max} = \frac{\pi}{\omega_n\sqrt{1-\zeta^2}} \tag{7-37}$$

or

$$y(t)\big|_{\max\ or\ \min} = 1 - \frac{e^{-n\pi\zeta/\sqrt{1-\zeta^2}}}{\sqrt{1-\zeta^2}}\sin(n\pi+\theta) \quad n = 1,2,\ldots \tag{7-38}$$

From Eq. (7-7), the maximum overshoot is obtained by

$$y(t)\big|_{\max\ or\ \min} = 1 + (-1)^{n-1}e^{-n\pi\zeta/\sqrt{1-\zeta^2}} \quad n = 1,2,\ldots \tag{7-39}$$

where, $y_{ss} = \lim_{t\to\infty} y(t) = 1.$ Letting n = 1 in Eq. (7-39), we get

$$\text{Maximum overshoot} = y_{\max} - 1 = e^{-\pi\zeta/\sqrt{1-\zeta^2}} \tag{7-40}$$

Also from Eq. (7-8) the percent maximum overshoot is

$$\text{Percent maximum overshoot} = \frac{\text{Maximum overshoot}}{y_{ss}} \times 100\% \tag{7-41}$$

or

$$\text{Percent maximum overshoot} = 100e^{-\pi\zeta/\sqrt{1-\zeta^2}} \tag{7-42}$$

Equation (7-41) shows that the maximum overshoot of the step response of the prototype second-order system is a function of only the damping ratio ζ.

The relationship between the percent maximum overshoot and the damping ratio given in Eq. (7-104) is plotted in Fig. 7-14. The time t_{max} in Eq. (7-100) is a function of both ζ and ω_n.

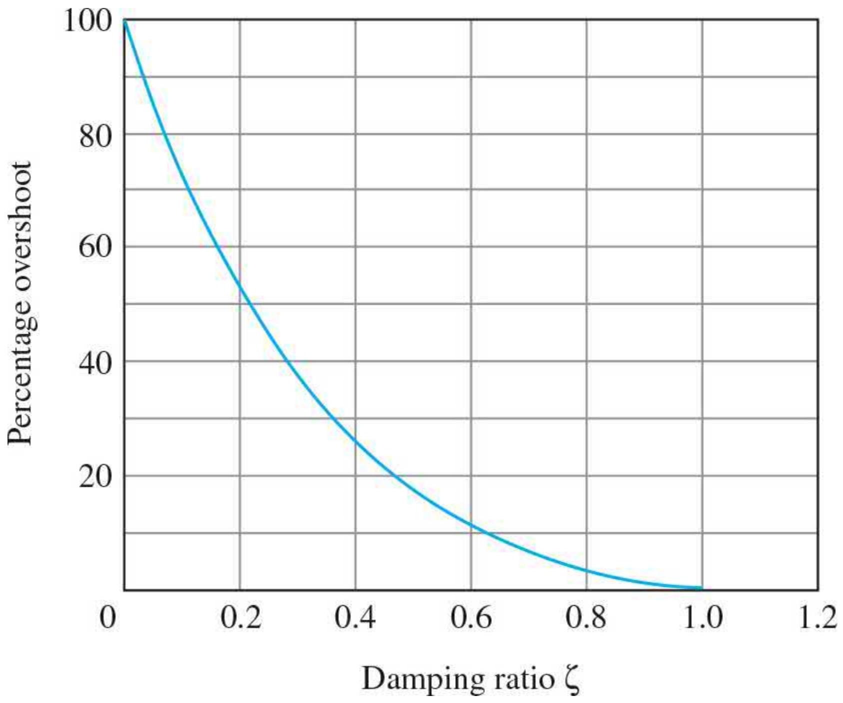


Figure 7-14 Percent overshoot as a function of damping ratio for the step response of the prototype second-order system.

Finally, because maximum overshoot is only a function of ζ, referring to Fig. 7-12*b* all points along the ζ_1 line, regardless of the values of ω_n, will have the same maximum overshoot value. Similarly all points along the ζ_2 line will experience the same maximum overshoot, and because $\zeta_2 > \zeta_1$, the maximum overshoot corresponding to ζ_2 is going to be smaller.

7-5-3 Delay Time and Rise Time ($0 < \zeta < 1$)

Delay and rise time are the measures of how fast a control system responds to an input or initial conditions.

Delay and rise time are the measures of how fast a control system responds to an input or initial conditions. It is more difficult to determine the exact analytical expressions of the delay time t_d, rise time t_r, and settling time t_s, even for just the simple prototype second-order system. For instance, for the delay time, we would have to set $y(t) = 0.5$ in Eq. (7-30) and solve for t. An easier way would be to plot $\omega_n t_d$ versus ζ, as shown in Fig. 7-15, and then approximate the curve by a straight line or a curve over the range of $0 < \zeta < 1$. From Fig. 7-15, the delay time for the prototype second-order system is approximated as

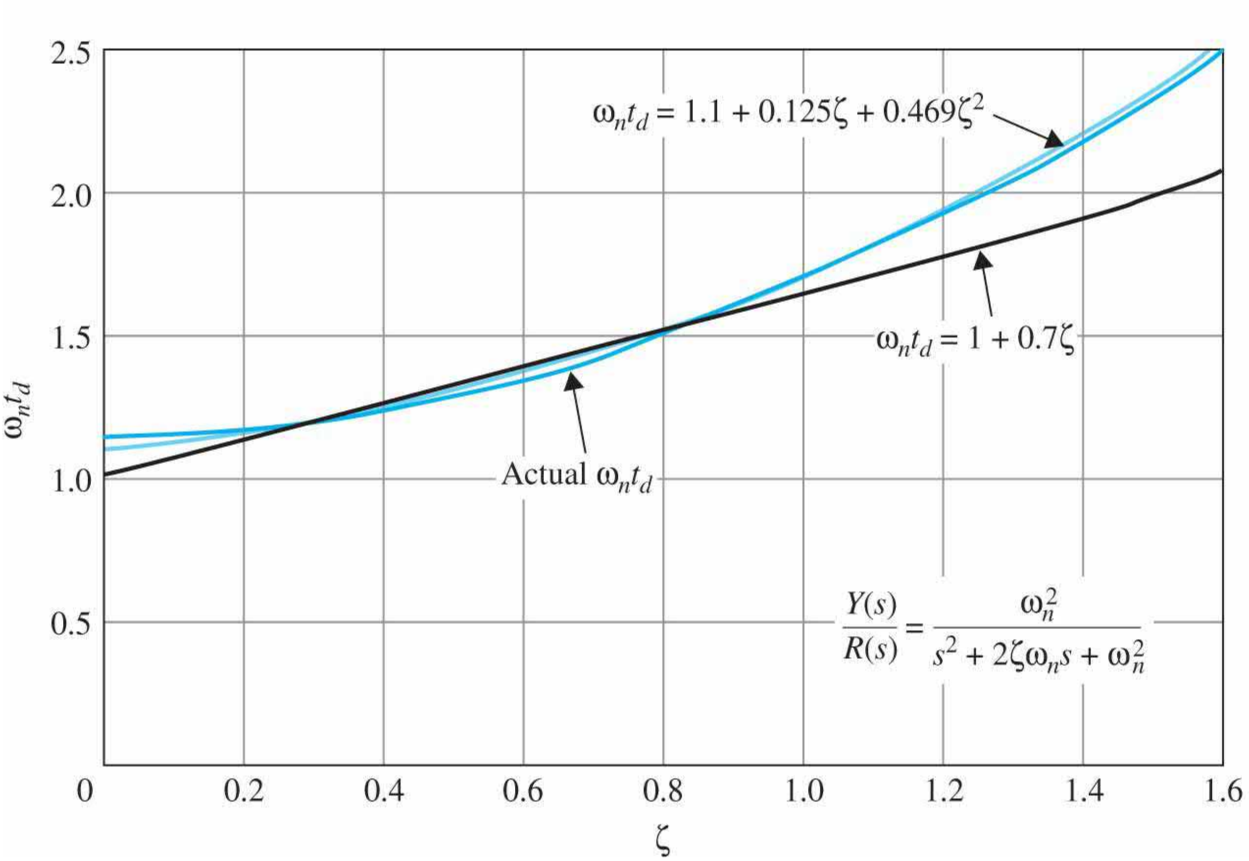


Figure 7-15 Normalized delay time versus ζ for the prototype second-order system.

$$t_d \cong \frac{1+0.7\zeta}{\omega_n} \qquad 0<\zeta<1.0 \tag{7-43}$$

We can obtain a better approximation by using a second-order equation for *td*:

$$t_d \cong \frac{1.1 + 0.125\zeta + 0.469\zeta^2}{\omega_n} \quad 0 < \zeta < 1.0 \tag{7-44}$$

For the **rise time** t_r, which is the time for the step response to reach from 10 to 90 percent of its final value, the exact value can be determined directly from the responses of Fig. 7-11. The plot of $\omega_n t_r$ versus ζ is shown in Fig. 7-16. In this case, the relation can again be approximated by a straight line over a limited range of ζ:

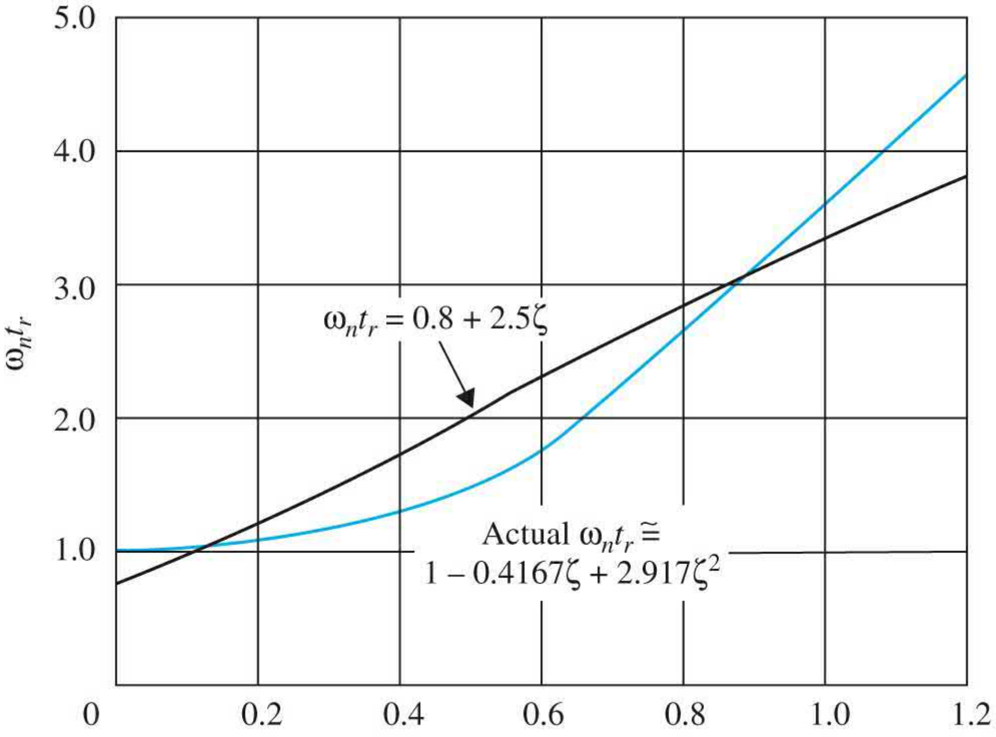


Figure 7-16 Normalized rise time versus ζ for the prototype second-order system.

$$t_r = \frac{0.8 + 2.5\zeta}{\omega_n} \quad 0 < \zeta < 1 \tag{7-45}$$

A better approximation can be obtained by using a second-order equation:

$$t_r = \frac{1 - 0.4167\zeta + 2.917\zeta^2}{\omega_n} \quad 0 < \zeta < 1 \tag{7-46}$$

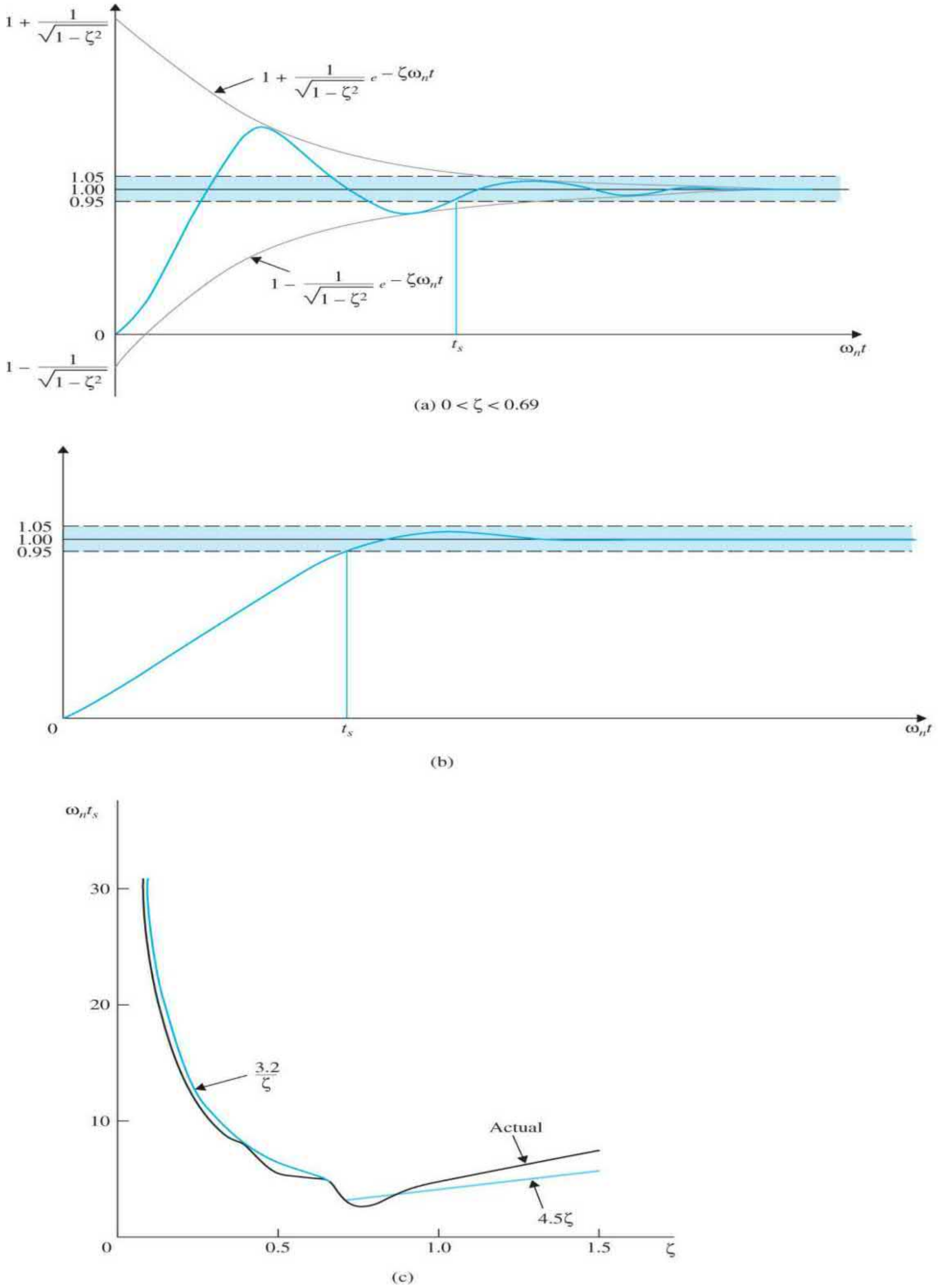
From this discussion, the following conclusions can be made on the rise time and delay time of the prototype second-order system:

- t_r and t_d are proportional to ζ and inversely proportional to ω_n.
- Increasing (decreasing) the natural frequency ω_n will reduce (increase) t_r and t_d.

7-5-4 Settling Time (5 and 2 Percent)

Settling time is a measure of how fast the step response settles to its final value.

As the name implies, *settling time should be used to measure how fast the step response settles to its final value*. From Fig. 7-11, we see that when $0 < \zeta < 0.69$, the unit-step response has a maximum overshoot greater than 5 percent, and the response can enter the band between 0.95 and 1.05 for the last time from either the top or the bottom. When ζ is greater than 0.69, the overshoot is less than 5 percent, and the response can enter the band between 0.95 and 1.05 only from the bottom. Figure 7-17*a* and *b* show the two different situations. Thus, the settling time has a discontinuity at $\zeta = 0.69$. The exact analytical description of the settling time t_s is difficult to obtain. We can obtain an approximation for t_s for $0 < \zeta < 0.69$ by using the envelope of the damped sinusoid of $y(t)$, as shown in Fig. 7-17*a* for a 5 percent requirement. In general, when the settling time corresponds to an intersection with the upper envelope of $y(t)$, the following relation is obtained:



(a) $0 < \zeta < 0.69$

(b)

(c)

Figure 7-17 Five-percent settling time of the unit-step response.

$$1+\frac{1}{\sqrt{1-\zeta^2}}e^{-\zeta\omega_n t_s} = \text{Upper bound of unit-step response} \tag{7-47}$$

When the settling time corresponds to an intersection with the bottom envelope of $y(t)$, t_s must satisfy the following condition:

$$1-\frac{1}{\sqrt{1-\zeta^2}}e^{-\zeta\omega_n t_s} = \text{Lower bound of unit-step response} \tag{7-48}$$

For the 5 percent requirement on settling time, the right-hand side of Eq. (7-47) would be 1.05, and that of Eq. (7-48) would be 0.95. It is easily verified that the same result for t_s is obtained using either equation.

Solving Eq. (7-47) for $\omega_n t_s$, we have

$$\omega_n t_s = -\frac{1}{\zeta}\ln\left(c_{ts}\sqrt{1-\zeta^2}\right) \tag{7-49}$$

where c_{ts} is the percentage set for the settling time. For example, if the threshold is **5 percent**, the c_{ts} = 0.05. Thus, for a 5-percent settling time, the right-hand side of Eq. (7-49) varies between 3.0 and 3.32 as ζ varies from 0 to 0.69. We can approximate the settling time for the prototype second-order system as

$$\textbf{5\% settling time: } t_s \cong \frac{3.2}{\zeta\omega_n} \quad 0<\zeta<0.69 \tag{7-50}$$

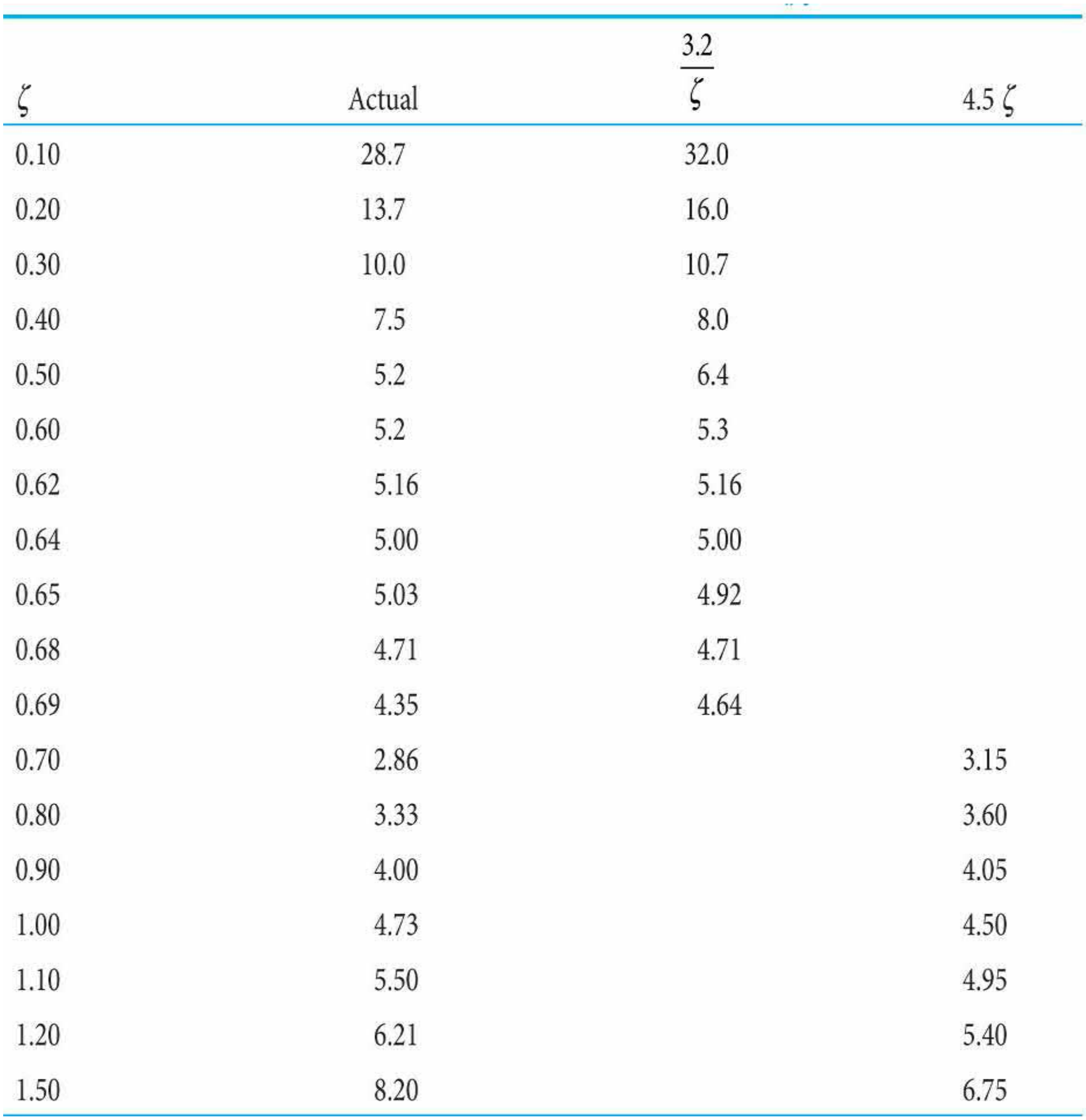
In general, settling time is inversely proportional to ζ and ω_n. A practical way of reducing the settling time is to increase ω_n while holding ζ constant. Although the response will be more oscillatory, the maximum overshoot depends only on ζ and can be controlled independently.

When the damping ratio ζ is greater than 0.69, from Fig. 7-17*b*, the unit-step response will always enter the band between 0.95 and 1.05 from below. We can show by observing the responses in Fig. 7-11 that the value of $\omega_n t_s$ is almost directly proportional to ζ. The following approximation is used for t_s for $\zeta > 0.69$.

$$\textbf{5\% settling time: } t_s = \frac{4.5\zeta}{\omega_n} \quad \zeta > 0.69 \tag{7-51}$$

Figure 7-17*c* shows the actual values of $\omega_n t_s$ versus ζ for the prototype second-order system described by Eq. (7-18), along with the approximations using Eqs. (7-50) and (7-51) for their respective effective ranges. The numerical values are shown in Table 7-3.

TABLE 7-3 Comparison of 5% Settling Times of Prototype Second-Order System, $\omega_n t_s$



ζ	Actual	$\frac{3.2}{\zeta}$	$4.5\,\zeta$
0.10	28.7	32.0	
0.20	13.7	16.0	
0.30	10.0	10.7	
0.40	7.5	8.0	
0.50	5.2	6.4	
0.60	5.2	5.3	
0.62	5.16	5.16	
0.64	5.00	5.00	
0.65	5.03	4.92	
0.68	4.71	4.71	
0.69	4.35	4.64	
0.70	2.86		3.15
0.80	3.33		3.60
0.90	4.00		4.05
1.00	4.73		4.50
1.10	5.50		4.95
1.20	6.21		5.40
1.50	8.20		6.75

It should also be pointed out that the 5 percent threshold is by no means a number cast in stone. More stringent design problems may require the system response to settle in any number less than 5 percent. For the 2-percent settling time, following the same procedure, we get

$$\textbf{2\% settling time: } t_s \cong \frac{4.0}{\zeta\omega_n} \quad 0 < \zeta < 0.9 \tag{7-52}$$

7-5-5 Transient Response Performance Criteria—Final

Remarks

The performance transient response criteria discussed in the previous sections, are also considered as transient response design criteria, as they are used in design of control systems. In order to effectively design a control system, it is important to fully appreciate how the pole movements affect PO, rise time, and settling time just from the *s*-plane. As shown in Fig. 7-18, for the prototype second-order transfer function in Eq. (7-18):

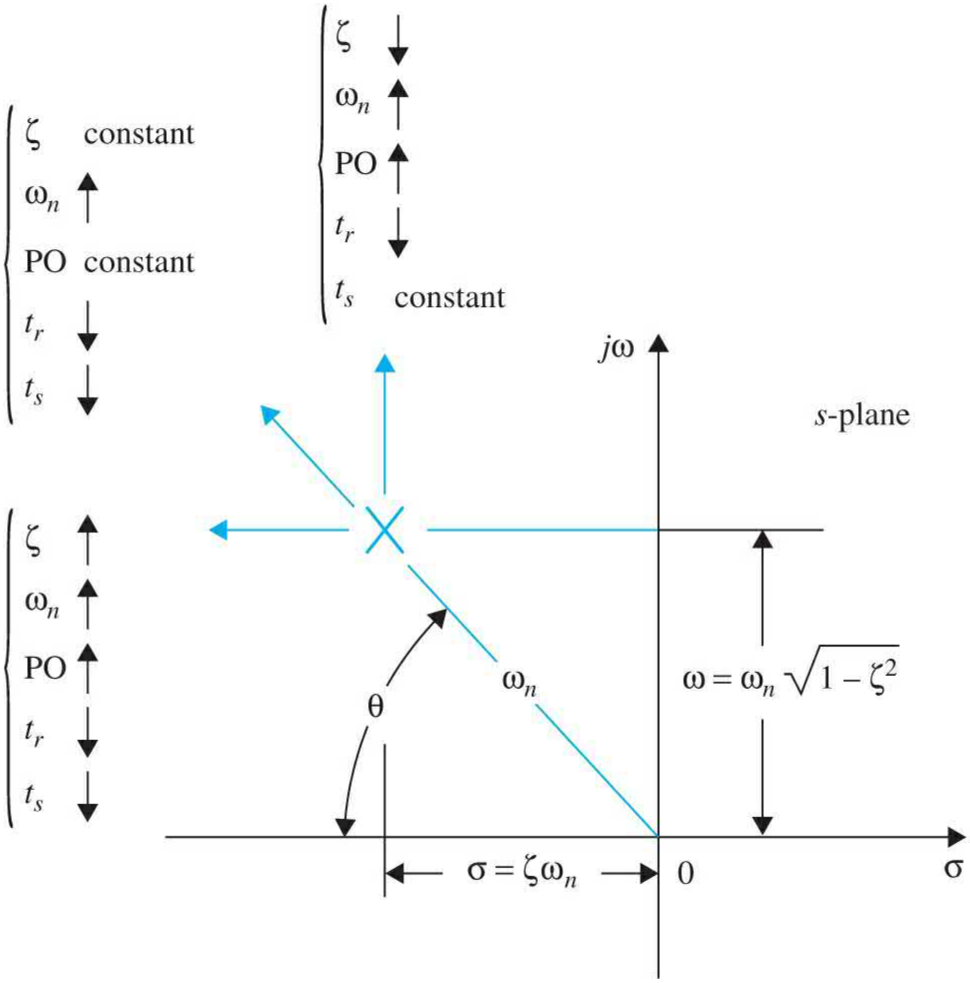


Figure 7-18 Relationship's among the pole location and ζ, ω_n, PO, t_r , and t_s.

- As the poles move diagonally away from the origin, because θ remains constant, the damping ratio ζ remains constant while natural frequency ω_n increases. Considering the definitions of the performance criteria from previous sections, PO remains constant while t_{max}, t_r , and t_s decrease.

- As the poles move away vertically, the natural frequency of the system increases while the damping ratio decreases. In this case PO increases, both t_{max} and t_r decrease, while t_s remains constant.
- As the poles move horizontally to the left, because q remains constant, the natural frequency of the system increases. In this case, PO increases while both t_r and t_s decrease. Note t_{max} in this case remains constant.

Since the equations reflecting t_s are based on approximations, which are different depending on the threshold percentage (e.g., 2 or 5 percent), the above *s*-plane observations may not always be accurate—refer to Example 7-5-1.

Finally, please keep in mind that, while the definitions on y_{max}, t_{max}, t_d, t_r , and t_s apply to a system of any order, the damping ratio ζ and the natural undamped frequency ω_n strictly apply only to a second-order system whose closed-loop transfer function is given in Eq. (7-18). Naturally, the relationships among t_d, t_r , and t_s and ζ and ω_n are valid only for the same second-order system model. However, these relationships can be used to measure the performance of higher-order systems that can be approximated by second-order ones, under the stipulation that some of the higher-order poles can be neglected.

EXAMPLE 7-5-1 For the dc motor position control problem discussed in Sec. 7-4-3, and shown in Fig. 7-19, use the motor with parameter values of Example 7-4-1 to study the effect of controller gain *K* on overshoot, rise time, and settling time. This example is a very important introduction to position control of dc motors.

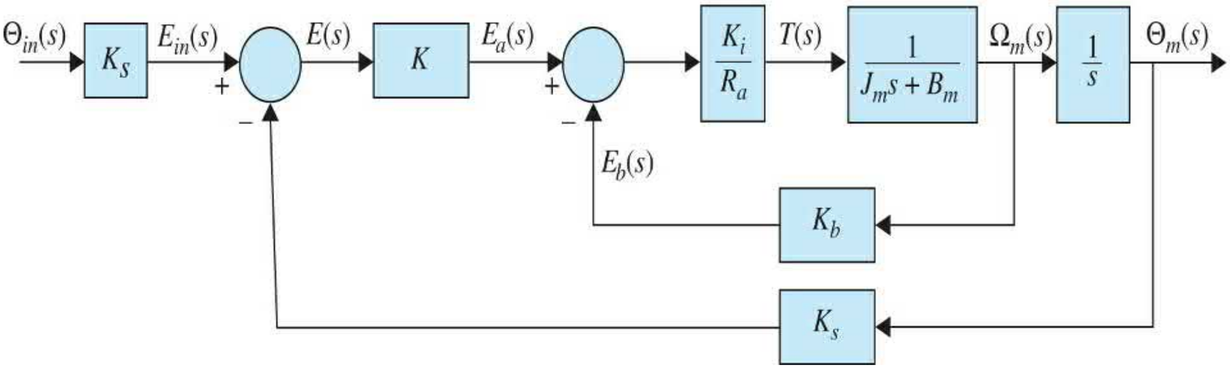


Figure 7-19 Block diagram of a position-control, armature-controlled dc motor.

Position Control of a DC Motor

Note that for the sake of proper comparison, we need to ensure that both input and output signals have the same units. As a result, the input voltage $E_{in}(s)$ has been scaled to position $\Theta_{in}(s)$ using the output sensor gain, K_s. So using a unit-step input, our objective for the motor shaft is to rotate 1 rad, as shown in Fig. 7-20. Note the disc connected to motor shaft is very thin (no inertia). In this case, for small L_a , *the* motor electrical time constant $\tau_e = (L_a/R_a)$ has been neglected. Hence, the simplified closed-loop transfer function is

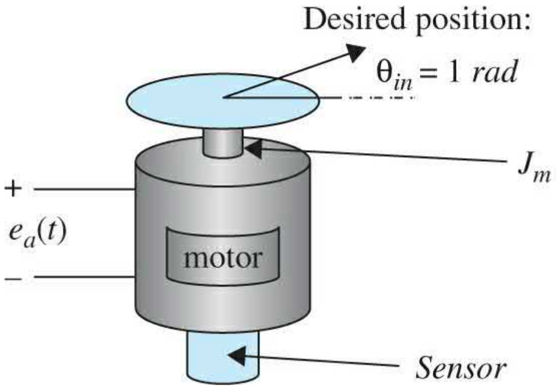


Figure 7-20 Desired rotation of an armature-controlled dc motor.

$$\frac{\Theta_m(s)}{\Theta_{in}(s)} = \frac{\frac{KK_iK_s}{R_aJ}}{s^2 + \left(\frac{R_aB_m + K_iK_b}{R_aJ}\right)s + \frac{KK_iK_s}{R_aJ}} = \frac{\omega_n^2}{(s^2 + 2\zeta\omega_n s + \omega_n^2)} \tag{7-53}$$

where K_s is the sensor gain, a potentiometer in this case with the gain $K_s = 1$. Since Eq. (7-53) is a second-order system, we have

$$2\zeta\omega_n = \frac{R_aB_m + K_iK_b}{R_aJ} \tag{7-54}$$

$$\omega_n^2 = \frac{KK_iK_s}{R_aJ} \tag{7-55}$$

For a given motor and position sensor, all parameters are known, and the only varying term is the amplifier gain *K*—the controller gain. Upon varying *K*, we can directly change ω_n, while $\sigma = \zeta\omega_n$ remains constant. As a result, ζ changes indirectly. For a positive *K*, regardless of the type of response (e.g., critically damped or underdamped), the final value of the system is $\theta_{fv} = 1$, which implies that the output will follow the input (recall we used a unit step input). Table 7-4 describes the motor performance for three values of *K*. Using Toolbox 7-5-1 the response of the motor for these cases is obtained and is shown in Fig. 7-21. As shown the actual PO and t_{max} exactly match the values obtained in Table 7-4, while the rise time and the settling time values in Table 7-4 are approximations.

TABLE 7-4 Motor Performance for Three Values of Controller Gain *K*

Controller Gain K	Damping Ratio ζ	Natural Frequency ω_n	Response
$K = 1.0$	$\zeta = 1.0$		Critically damped
$K = 2.0$	$\zeta = 0.707$	$\omega_n = 844.8$ rad/s	Underdamped
			$5\%\, t_s = \dfrac{4.5\zeta}{\omega_n} = 0.0038\text{ s} \quad \zeta > 0.69$
			Note: Settling time occurs before overshoot (PO < 5)
			$2\%\, t_s = \dfrac{4}{\zeta\omega_n} = 0.0067\text{ s} \quad 0 < \zeta < 0.9$
			$t_r = \dfrac{1 - 0.4167\zeta + 2.917\zeta^2}{\omega_n} = 0.0026\text{ s} \quad 0 < \zeta < 1$
			$t_{\max} = \dfrac{\pi}{\omega_n\sqrt{1-\zeta^2}} = 0.0053\text{ s}$
			$\text{PO} = 100e^{-\pi\zeta/\sqrt{1-\zeta^2}} = 4.3$
$K = 4.0$	$\zeta = 0.5$	$\omega_n = 1{,}194.8$ rad/s	Underdamped
			$5\%\, t_s = \dfrac{3.2}{\zeta\omega_n} = 0.0054\text{ s} \quad 0 < \zeta < 0.69$
			$2\%\, t_s = \dfrac{4}{\zeta\omega_n} = 0.0067\text{ s} \quad 0 < \zeta < 0.9$
			$t_r = \dfrac{1 - 0.4167\zeta + 2.917\zeta^2}{\omega_n} \quad 0 < \zeta < 1$
			$= 0.0013\text{ sec}$
			$t_{\max} = \dfrac{\pi}{\omega_n\sqrt{1-\zeta^2}} = 0.003\text{ s}$
			$\text{PO} = 100e^{-\pi\zeta/\sqrt{1-\zeta^2}} = 16.3$

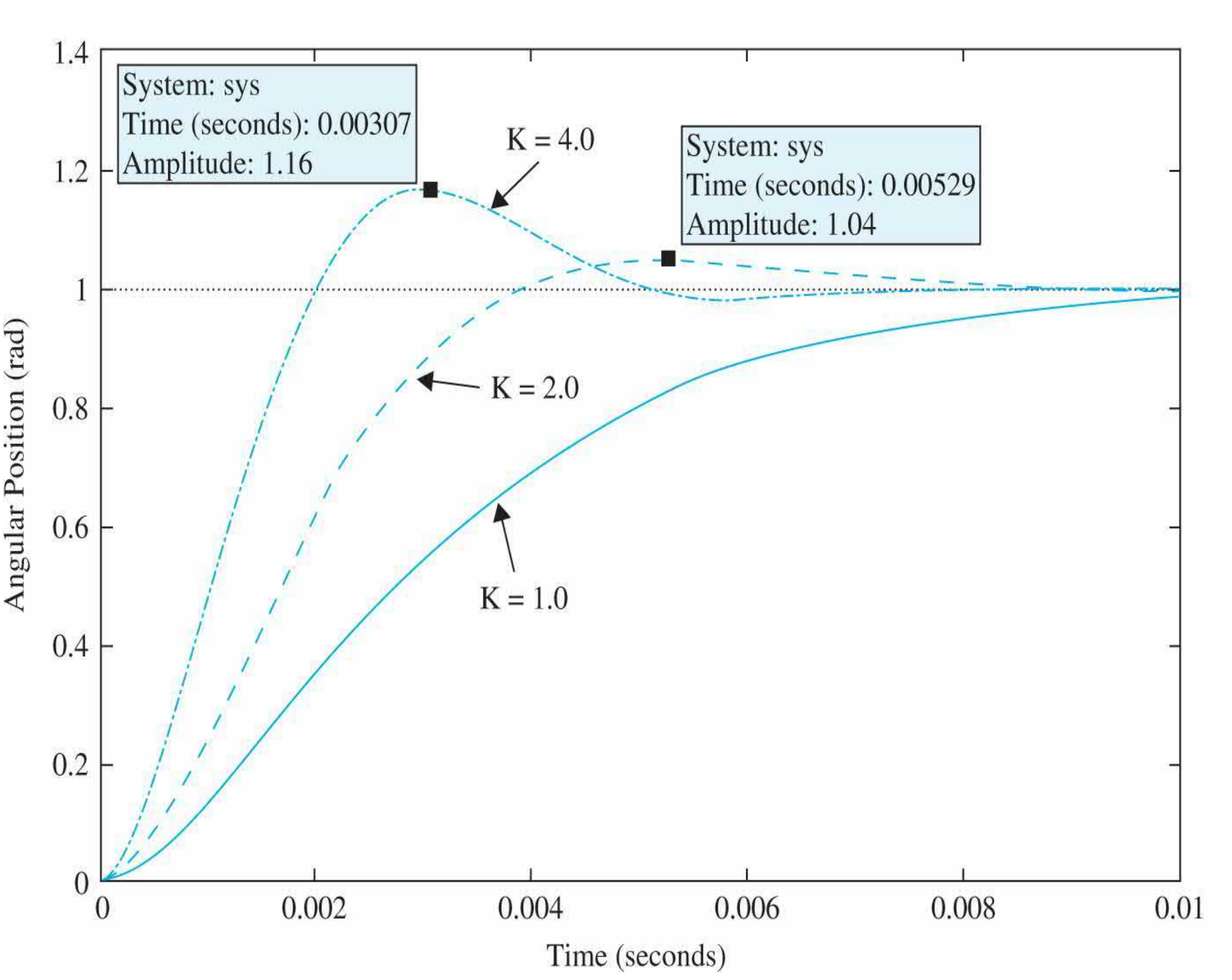


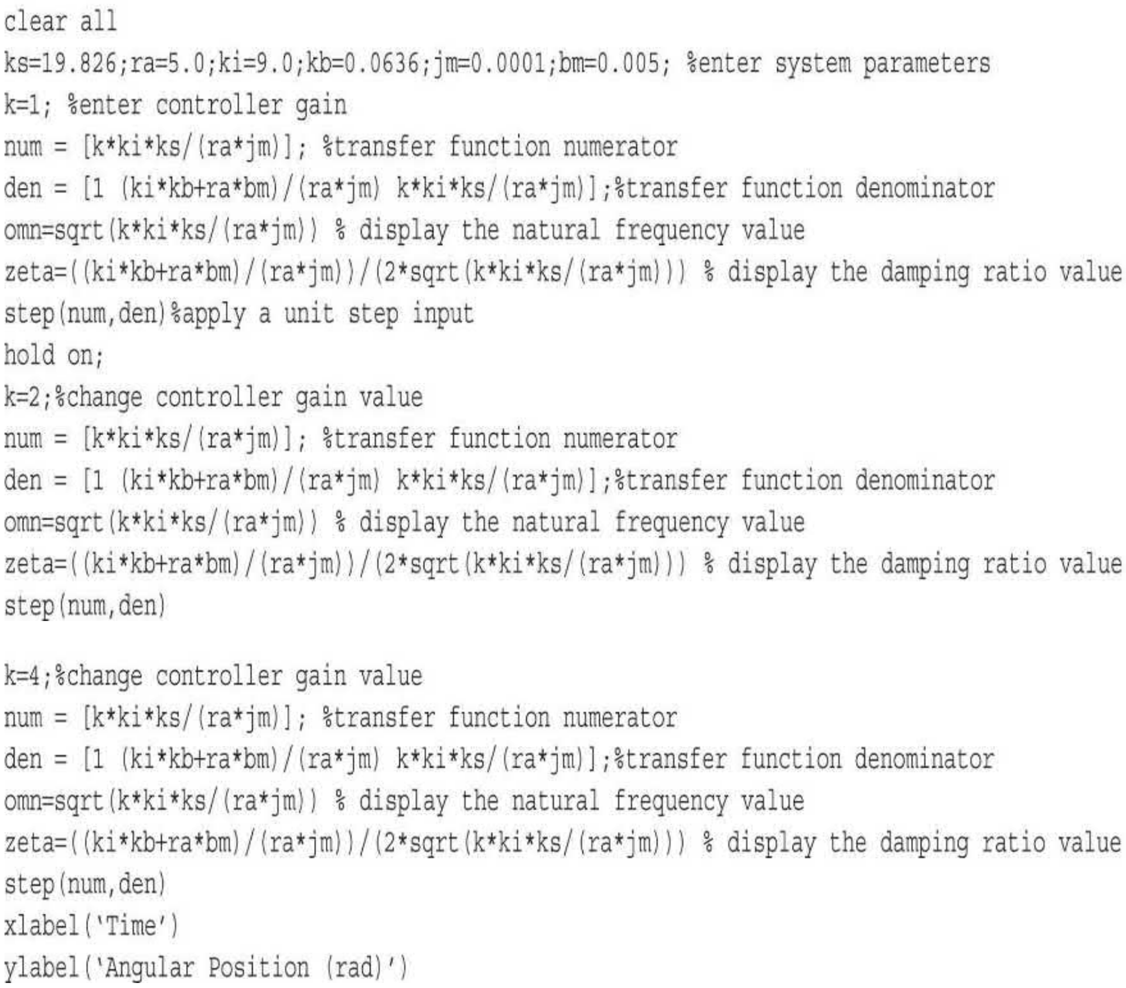
Figure 7-21 Position control response of dc motor in Example 7-5-1 to a unit-step input for three controller gain *K* values.

To find PO, rise time, and settling time using MATLAB, point at a desired location on the graph and right-click to display the x and y values.

Based on these results, we observe that as *K* increases, the damping ratio ζ decreases while the natural frequency ω_n increases. As a result with the increase in *K*, the system PO increases, resulting in a faster rise time t_r. In this case, the 5-percent settling time t_s increases with *K* in contrast to the *s*-plane observations in the previous section. This is because when ζ is greater than 0.69, the overshoot is less than 5 percent, and the settling time is less than t_{max}. For the 2-percent settling time, however, remains constant and greater than t_{max}, which is in line with the s-plane observations in the previous section.

Toolbox 7-5-1

Position response of Example 7-5-1 using MATLAB.



```
clear all
ks=19.826;ra=5.0;ki=9.0;kb=0.0636;jm=0.0001;bm=0.005; %enter system parameters
k=1; %enter controller gain
num = [k*ki*ks/(ra*jm)]; %transfer function numerator
den = [1 (ki*kb+ra*bm)/(ra*jm) k*ki*ks/(ra*jm)];%transfer function denominator
omn=sqrt(k*ki*ks/(ra*jm)) % display the natural frequency value
zeta=((ki*kb+ra*bm)/(ra*jm))/(2*sqrt(k*ki*ks/(ra*jm))) % display the damping ratio value
step(num,den)%apply a unit step input
hold on;
k=2;%change controller gain value
num = [k*ki*ks/(ra*jm)]; %transfer function numerator
den = [1 (ki*kb+ra*bm)/(ra*jm) k*ki*ks/(ra*jm)];%transfer function denominator
omn=sqrt(k*ki*ks/(ra*jm)) % display the natural frequency value
zeta=((ki*kb+ra*bm)/(ra*jm))/(2*sqrt(k*ki*ks/(ra*jm))) % display the damping ratio value
step(num,den)

k=4;%change controller gain value
num = [k*ki*ks/(ra*jm)]; %transfer function numerator
den = [1 (ki*kb+ra*bm)/(ra*jm) k*ki*ks/(ra*jm)];%transfer function denominator
omn=sqrt(k*ki*ks/(ra*jm)) % display the natural frequency value
zeta=((ki*kb+ra*bm)/(ra*jm))/(2*sqrt(k*ki*ks/(ra*jm))) % display the damping ratio value
step(num,den)
xlabel('Time')
ylabel('Angular Position (rad)')
```

7-6 STEADY-STATE ERROR

One of the objectives of most control systems is that the system output response follows a specific reference signal accurately in the steady state. For example, a pick and place robot arm must end up accurately in the desired position to pick or place an object (in App. A we provide an example of pick and place robotic arm). In the real world, because of friction and other imperfections and the natural composition of the system, the steady state of the output response seldom agrees exactly with the reference. Therefore, steady-state errors in control systems are almost unavoidable. In a design problem, one of the objectives is to keep the steady-state error to a minimum, or below a certain tolerable value, and at the same time the transient response must satisfy a certain set of specifications.

The accuracy requirement on control systems depends to a great extent on the control objectives of the system. For instance, the final position accuracy of an elevator would be far less stringent than the pointing accuracy on the control of the space telescopes. The accuracy of position control of such systems is often measured in microradians.

7-6-1 Definition of the Steady-State Error

Unity-Feedback Systems

Error in a unity-feedback control system is defined as the difference between **input** and **output**. For the closed-loop system shown in Fig. 7-22, the error of the system is defined as

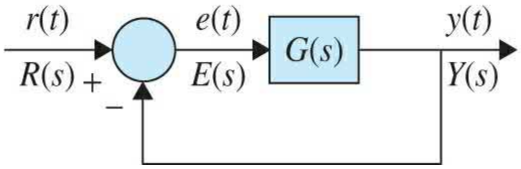


Figure 7-22 Error in a unity-feedback control system.

$$e(t) = r(t) - y(t) \tag{7-56}$$

where $r(t)$, the input, is the signal that the output $y(t)$ is to track. Note that all quantities in Eq. (7-56) have **same units** or **dimensions** (e.g., volts, meters, etc.). In this case, the input $r(t)$ is also called the **reference signal**, which is the desired value for the output. In the Laplace domain as shown in Fig. 7-23 the error is

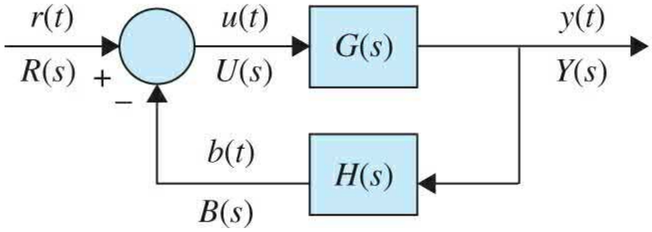


Figure 7-23 Nonunity-feedback control system.

$$E(s) = R(s) - Y(s) \tag{7-57}$$

or,

$$E(s) = \left(\frac{1}{1+G(s)}\right)R(s) \tag{7-58}$$

The **steady-state error** is defined as the value of the error at steady state, or the final value of the error. That is

$$e_{ss} = \lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) \tag{7-59}$$

> The **steady-state error** e_{ss} is defined as the value of error at steady state.

For a unity-feedback system, the steady-state error is obtained from Eq. (7-59) as

$$e_{ss} = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\left(\frac{1}{1+G(s)}\right)R(s) \tag{7-60}$$

Nonunity-Feedback Systems

Let us consider the nonunity-feedback system in Fig. 7-23, where $r(t)$ is the input (but not the reference signal), $u(t)$ is the actuating signal, $b(t)$ is the feedback signal, and $y(t)$ is the output. In this case, the error formula in Eq. (7-57) is not valid because the input and output may **not** have **same units** or **dimensions**. In order to establish a proper formula for the error, we must first obtain a clear picture of what the reference signal is. From the discussions in Chap. 3 and earlier on in this section, for a stable system, the steady-state output will be tracking the reference signal. The error of the system at all times is the difference between the reference signal and the output. In order to establish the reference signal, let us modify the system in Fig. 7-23 by first factoring out the feedback gain $H(s)$, as shown in Fig. 7-24. Error in the system is the difference between the output and the desired value of the output—or the **reference signal.** In this case, the reference signal in Laplace domain is $R(s)G_1(s)$, as shown in Fig. 7-24. The value of $G_1(s)$ is obviously is related to $1/H(s)$, and it may be obtained based on the time response characteristics of the original system in Fig. 7-23. Obviously, the reference signal reflects the desired value of the system output based on a given input, and it cannot contain additional **transient** behavior due to $H(s)$.

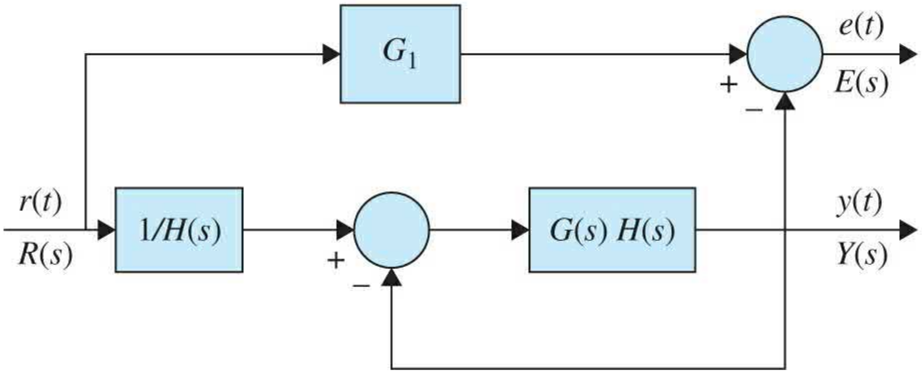


Figure 7-24 Error in a nonunity-feedback control system.

As it becomes clearer through the examples in the end of this section, there are two possible scenarios based on the value of $H(s)$.

$$\textbf{Case 1: } G_1(s) = \frac{1}{\lim\limits_{s \to 0} H(s)} = 1/H(0) = \text{Constant} \tag{7-61}$$

which means that $H(s)$ cannot have poles at $s = 0$. Hence, the reference signal becomes

$$\text{Reference signal} = R(s)[1/H(0)] = R(s)G_1(s) \tag{7-62}$$

Case 2: *H*(*s*) has *N*th-order zero at *s* = 0

$$G_1(s) = \left(\frac{1}{s^N}\right)\frac{1}{\lim\limits_{s\to 0} H(s)} = \frac{1}{s^N H(0)} \tag{7-63}$$

$$\text{Reference signal} = \frac{R(s)}{s^N H(0)} = R(s)G_1(s) \tag{7-64}$$

In both given cases, the error signal in Laplace transform domain becomes

$$E(s) = \text{Reference signal} - Y(s) = R(s)G_1(s) - Y(s) \tag{7-65}$$

or,

$$E(s) = \left(G_1(s) - \frac{Y(s)}{R(s)}\right)R(s) = \left(G_1(s) - \frac{G(s)}{1+G(s)H(s)}\right)R(s) \tag{7-66}$$

In the case of a nonunity feedback, the steady-state error is obtained from Eq. (7-65) as

$$e_{ss} = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\left(G_1(s) - \frac{G(s)}{1+G(s)H(s)}\right)R(s) \tag{7-67}$$

We can also rearrange the block diagram of Fig. 7-24 to arrive at a unity-feedback system to be able to utilize Eq. (7-60) as the definition of steady-state error. This can be done by arriving at an equivalent system with unity feedback that can model the system in Fig. 7-24. To do so we set the error equation for the nonunity system equal to the error in an equivalent unity-feedback system. That is

$$E(s)=\left(\frac{1}{1+G_{eq}(s)}\right)R(s)=\left(\frac{G_1(s)+G_1(s)G(s)H(s)-G(s)}{1+G(s)H(s)}\right)R(s) \tag{7-68}$$

For this to be valid

$$G_{eq}(s)=\left(\frac{[1+G(s)H(s)][1-G_1(s)]+G(s)}{G_1(s)[1+G(s)H(s)]-G(s)}\right) \tag{7-69}$$

where the equivalent system in Eq. (7-65) can now be represented in a unity-feedback form as in Fig. 7-25. If $H(s)$ = 1 then $G_1(s)$ = 1, which implies $G_{eq} = G$. As a result the two systems in Figs. 7-22 and 7-25 become the same.

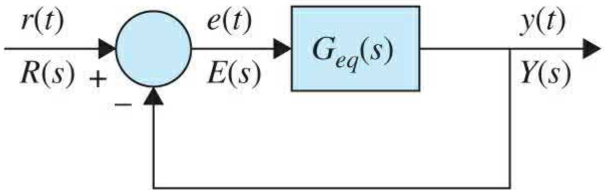


Figure 7-25 Equivalent unity-feedback control system representing the nonunity-feedback system in Fig. 7-23.

EXAMPLE 7-6-1 The forward-path and closed-loop transfer functions of the system shown in Fig. 7-22 are given next. The system is assumed to have unity feedback, so $H(s)$ = 1, and thus we use Eq. (7-60) for error calculation.

$$G(s)=\frac{5(s+1)}{s^2(s+12)(s+5)} \tag{7-70}$$

$$M(s)=\frac{G(s)}{1+G(s)}=\frac{5(s+1)}{s^4+17s^3+60s^2+5s+5} \tag{7-71}$$

The poles of $M(s)$ are all in the left-half s-plane. Thus, the system is stable. The steady-state errors due to the three basic types of inputs are evaluated as follows:

Unit-step input:

$$e_{ss} = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\left(\frac{1}{1+G(s)}\right)R(s) = \lim_{s\to 0} s\left(\frac{s^4+17s^3+60s^2}{s^4+17s^3+60s^2+5s+5}\right)\left(\frac{1}{s}\right) = 0 \tag{7-72a}$$

Unit-ramp input:

$$e_{ss} = \lim_{s\to 0} s\left(\frac{s^4+17s^3+60s^2}{s^4+17s^3+60s^2+5s+5}\right)\left(\frac{1}{s^2}\right) = 0 \tag{7-72b}$$

Unit-parabolic input:

$$e_{ss} = \lim_{s\to 0} s\left(\frac{s^4+17s^3+60s^2}{s^4+17s^3+60s^2+5s+5}\right)\left(\frac{1}{s^3}\right) = \frac{60}{5} = 12 \tag{7-72c}$$

EXAMPLE 7-6-2 Consider the nonunity-feedback system, shown in Fig. 7-23, which has the following transfer functions:

$$G(s) = \frac{1}{s^2(s+12)} \qquad H(s) = \frac{5(s+1)}{s+5} \tag{7-73}$$

Because $H(s)$ has no zeros at $s = 0$, we use Case 1 scenario given by Eq. (7-61) in error calculations—that is, $G_1(s) = 1/H(0) = 1$. So $r(t)$ in Fig. 7-23 is the reference signal. The closed-loop transfer function is

$$M(s) = \frac{Y(s)}{R(s)} = \frac{G(s)}{1+G(s)H(s)} = \frac{s+5}{s^4+17s^3+60s^2+5s+5} \tag{7-74}$$

Using Eq. (7-67) The steady-state errors of the system are calculated for the three basic types of inputs.

Unit-step input:

$$e_{ss} = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s[1 - M(s)]R(s) = \lim_{s\to 0} s\left(1 - \frac{s+5}{s^4+17s^3+60s^2+5s+5}\right)\left(\frac{1}{s}\right)$$

$$= \lim_{s\to 0} s\left(\frac{s^4+17s^3+60s^2+4s}{s^4+17s^3+60s^2+5s+5}\right)\left(\frac{1}{s}\right) = 0 \qquad (7\text{-}75)$$

Unit-ramp input:

$$e_{ss} = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s[1 - M(s)]R(s)$$

$$= \lim_{s\to 0} s\left(\frac{s^4+17s^3+60s^2+4s}{s^4+17s^3+60s^2+5s+5}\right)\left(\frac{1}{s^2}\right) = 0.8 \qquad (7\text{-}76)$$

Unit-parabolic input:

$$e_{ss} = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s[1 - M(s)]R(s)$$

$$= \lim_{s\to 0} s\left(\frac{s^4+17s^3+60s^2+4s}{s^4+17s^3+60s^2+5s+5}\right)\left(\frac{1}{s^3}\right) = \infty \qquad (7\text{-}77)$$

It would be important to also find the time response of the system to calculate the steady-state errors from the difference between the input and the output and compare them with the results just obtained. Applying the unit-step, unit-ramp, and unit-parabolic inputs to the system described by Eq. (7-74) and taking the inverse Laplace transform of *Y*(*s*), the outputs are

Unit-stepinput:

$$y(t) = 1 - 0.00056e^{-12.05t} - 0.0001381e^{-4.886t}$$

$$-0.9993e^{-0.0302t}\cos 0.2898t - 0.1301e^{-0.0302t}\sin 0.2898t \; t \geq 0 \qquad (7\text{-}78)$$

Thus, from the leading term the reference input is unit step, the steady-state value of *y*(*t*) is also unity, and therefore, the steady-state error is zero.

Unit-rampinput:

$$y(t) = t - 0.8 + 4.682\times10^{-5}e^{-12.05t} + 2.826\times10^{-5}e^{-4.886t}$$
$$+0.8e^{-0.0302t}\cos 0.2898t - 3.365e^{-0.0302t}\sin 0.2898t \quad t \ge 0 \tag{7-79}$$

Thus, the reference input is a unit ramp $tu_s(t)$, the steady-state portion of $y(t)$ is $t – 0.8$, and the steady-state error to a unit ramp is 0.8.

Unit-parabolic input:

$$y(t) = 0.5t^2 - 0.8t - 11.2 - 3.8842\times10^{-6}e^{-12.05t} - 5.784\times10^{-6}e^{-4.886t}$$
$$+11.2e^{-0.0302t}\cos 0.2898t + 3.9289e^{-0.0302t}\sin 0.2898t \quad t \ge 0 \tag{7-80}$$

Thus, the reference input is a unit parabolic input $t^2u_s(t)/2$, the steady-state portion of $y(t)$ is $0.5t^2 – 0.8t + 11.2$. Thus, the steady-state error is $0.8t + 11.2$, which becomes infinite as time goes to infinity.

Suppose we change the transfer function $H(s)$ so that

$$G(s) = \frac{1}{s^2(s+12)} \quad H(s) = \frac{10(s+1)}{s+5} \tag{7-81}$$

Then

$$G_1 = \frac{1}{\lim\limits_{s\to0} H(s)} = 1/2 \tag{7-82}$$

The closed-loop transfer function is

$$M(s) = \frac{Y(s)}{R(s)} = \frac{G(s)}{1+G(s)H(s)} = \frac{s+5}{s^4+17s^3+60s^2+10s+10} \tag{7-83}$$

The steady-state errors of the system due to the three basic types of inputs are calculated as follows:

Unit-stepinput:

Solving for the output using the $M(s)$ in Eq. (7-83), we get

$$y(t) = 0.5u_s(t) + \text{transient terms} \tag{7-84}$$

Thus, the steady-state value of $y(t)$ is 0.5, and because $K_H = 2$, the steady-state error due to a unit-step input is zero.

Unit-rampinput:

The unit-ramp response of the system is written as

$$y(t) = [0.5t - 0.4]u_s(t) + \text{transient terms} \tag{7-85}$$

Because the transient terms will die out as t approaches infinity, the steady-state error due to a unit-ramp input is 0.4.

Unit-parabolic input:

The unit-parabolic input is

$$y(t) = [0.25t^2 - 0.4t - 2.6]u_s(t) + \text{transient terms} \tag{7-86}$$

Thus, the steady-state error is $0.4t + 2.6$, which increases with time. ▲

EXAMPLE 7-6-3 Consider the nonunity-feedback system shown in Fig. 7-23 with the following transfer functions:

$$G(s) = \frac{1}{s^2(s+12)} \qquad H(s) = \frac{10s}{s+5} \tag{7-87}$$

where $H(s)$ has one zero at $s = 0$. Thus,

$$G_1(s) = \left(\frac{1}{s}\right)\frac{1}{\lim\limits_{s \to 0} H(s)} = \frac{1}{2s} \tag{7-88}$$

Hence,

$$\text{Reference signal } = \frac{R(s)}{2s} \tag{7-89}$$

The closed-loop transfer function is

$$M(s) = \frac{Y(s)}{R(s)} = \frac{s+5}{s^4 + 17s^3 + 60s^2 + 10s} \tag{7-90}$$

For a unit-step input, the steady-state error, from Eq. (7-67), is

Unit step input:

$$\begin{aligned} e_{ss} &= \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\left[\frac{1}{2s} - M(s)\right]R(s) \\ &= \lim_{s\to 0} s\left(\frac{1}{2s} - \frac{s+5}{s^4+17s^3+60s^2+10s}\right)\left(\frac{1}{s}\right) \\ &= \lim_{s\to 0} s\left(\frac{s^4+17s^3+58s^2}{2s^5+34s^4+120s^3+20s^2}\right)\left(\frac{1}{s}\right) = \frac{58}{20} = 2.9 \end{aligned} \tag{7-91}$$

To verify this result, we can find the unit-step response using the closed-loop transfer function in Eq. (7-90). The result is

$$y(t) = (0.5t - 2.9)u_s(t) + \text{transient terms} \tag{7-92}$$

From Eq. (7-92), the reference signal is considered to be $0.5tu_s(t) = 0.5t$, and the steady-state error is 2.9. Of course, in Laplace domain the reference signal is a ramp function $\frac{1}{2s^2}$ as initially selected in Eq. (7-89). ▲

EXAMPLE 7-6-4 The block diagram of the *speed control* dc motor system, discussed in Sec. 7-4-2, without a load for small motor electric-time constant is shown in Fig. 7-25. In this case, because $r(t)$ and $\omega_m(t)$ are not of the same dimension, following Case 1 in Eq. (7-61), we can arrive at the reference signal $r(t)/K_t$ for error calculation. This is equivalent to introducing angular speed $\omega_{in}(t)$ as the input, as shown in Fig. 7-26, and the gain K_t (tachometer gain in this case) so that $\omega_{in}(t) = r(t)/K_t$. This would also imply that the input and output have the same unit and dimension. As a result, the **reference signal** is the **desired speed** and not the input voltage $r(t)$.

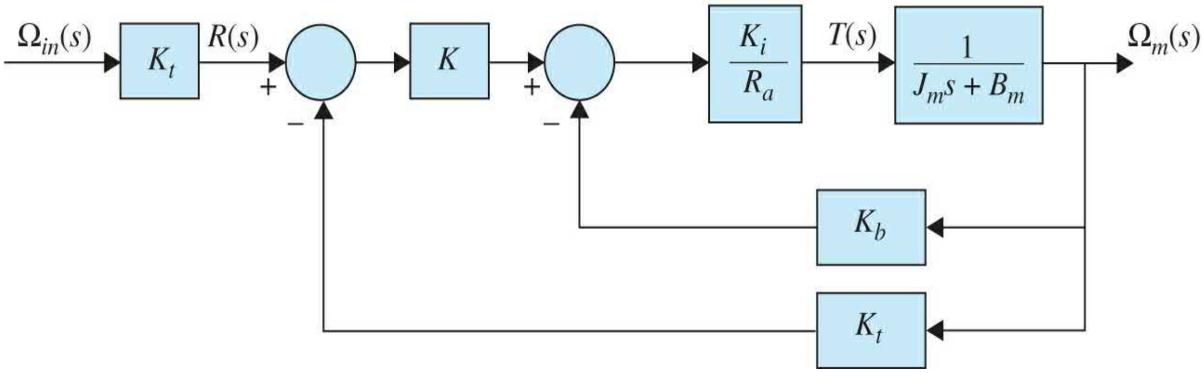


Figure 7-26 Speed control block diagram of a dc motor.

Speed Control of a DC Motor

Simplifying the block diagram in Fig. 7-26 we can arrive at the final unity-feedback representation of the system, as in Fig. 7-22, where the open loop transfer function of the system is

$$G(s) = \frac{\frac{K_t K_i K}{R_a J_m}}{s + \left(\frac{K_i K_b + R_a B_m}{R_a J_m}\right)} \tag{7-93}$$

The angular speed of the system can be obtained in terms of the input as

$$\Omega_m(s) = \frac{\dfrac{K_t K_i K}{R_a J_m}}{s + \left(\dfrac{K_i K_b + R_a B_m + K_t K_i K}{R_a J_m}\right)} \Omega_{in}(s) \tag{7-94}$$

where

$$\tau_c = \frac{R_a J_m}{K_i K_b + R_a B_m + K_t K_i K} \tag{7-95}$$

is the system time constant. Note also that the system in Eq. (7-94) is always **stable** for any $K > 0$. Finally, the time response of the system for the reference signal $\Omega_{in}(s) = 1/s$ is

$$\omega_m(t) = \frac{K_t K_i K}{K_i K_b + R_a B_m + K_t K_i K}(1 - e^{-t/\tau_c}) \tag{7-96}$$

The final value of angular speed output is

$$\omega_{fv} = \lim_{s\to 0} s\Omega_m(s) = \lim_{s\to 0} s\left(\frac{G(s)}{1+G(s)}\right)\frac{1}{s} = \frac{K_t K_i K}{K_i K_b + R_a B_m + K_t K_i K} \tag{7-97}$$

While the steady-state error is obtained as

$$e_{ss} = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\left(\frac{1}{1+G(s)}\right)\frac{1}{s} = \frac{K_i K_b + R_a B_m}{K_i K_b + R_a B_m + K_t K_i K} \tag{7-98}$$

For $H(s) = K_t = 1$ (this simply implies the sensor output is calibrated to show 1 rad/s is 1 V), selecting the system parameters from Example 7-4-1, we get

$$\omega_{fv} = \lim_{s\to 0} s\Omega_m(s) = \lim_{s\to 0} s\left(\frac{G(s)}{1+G(s)}\right)\frac{1}{s} = \frac{18{,}000K}{1194.8+18{,}000K} \qquad (7\text{-}100)$$

The final value of angular speed output is

$$\omega_{fv} = \lim_{s\to 0} s\Omega_m(s) = \lim_{s\to 0} s\left(\frac{G(s)}{1+G(s)}\right)\frac{1}{s} = \frac{18{,}000K}{1194.8+18{,}000K} \qquad (7\text{-}100)$$

While the steady-state error is obtained from Eq. (7-61) as

$$e_{ss} = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\left(\frac{1}{1+G(s)}\right)\frac{1}{s} = \frac{1194.8}{1194.8+18{,}000K} \qquad (7\text{-}101)$$

As shown in Table 7-5, for a unit step input (1 rad/s), increasing the controller gain *K* results in the reduction of both the steady-state error and system time constant values. The unit-step time responses of the system for different *K* values are shown in Fig. 7-27. Practically speaking there is a limit to how much we can increase *K* because of amplifier saturation or the applied voltage exceeding motor input capacity.

TABLE 7-5 System Time Constant and Steady-State Speed Error for Three Different Controller Gain Values—Given a Unit-Step Input

Controller Gain K	Steady-State Error e_{ss}	System Time Constant $\tau_c = \dfrac{R_aJ_m}{K_iK_b + R_aB_m + K_tK_iK}$
0.1	0.3990 rad/s	$\tau_c = 3.3391 \times 10^{-4}$ s
1.0	0.0622 rad/s	$\tau_c = 5.2097 \times 10^{-5}$ s
10.0	0.0066 rad/s	$\tau_c = 5.5189 \times 10^{-6}$ s

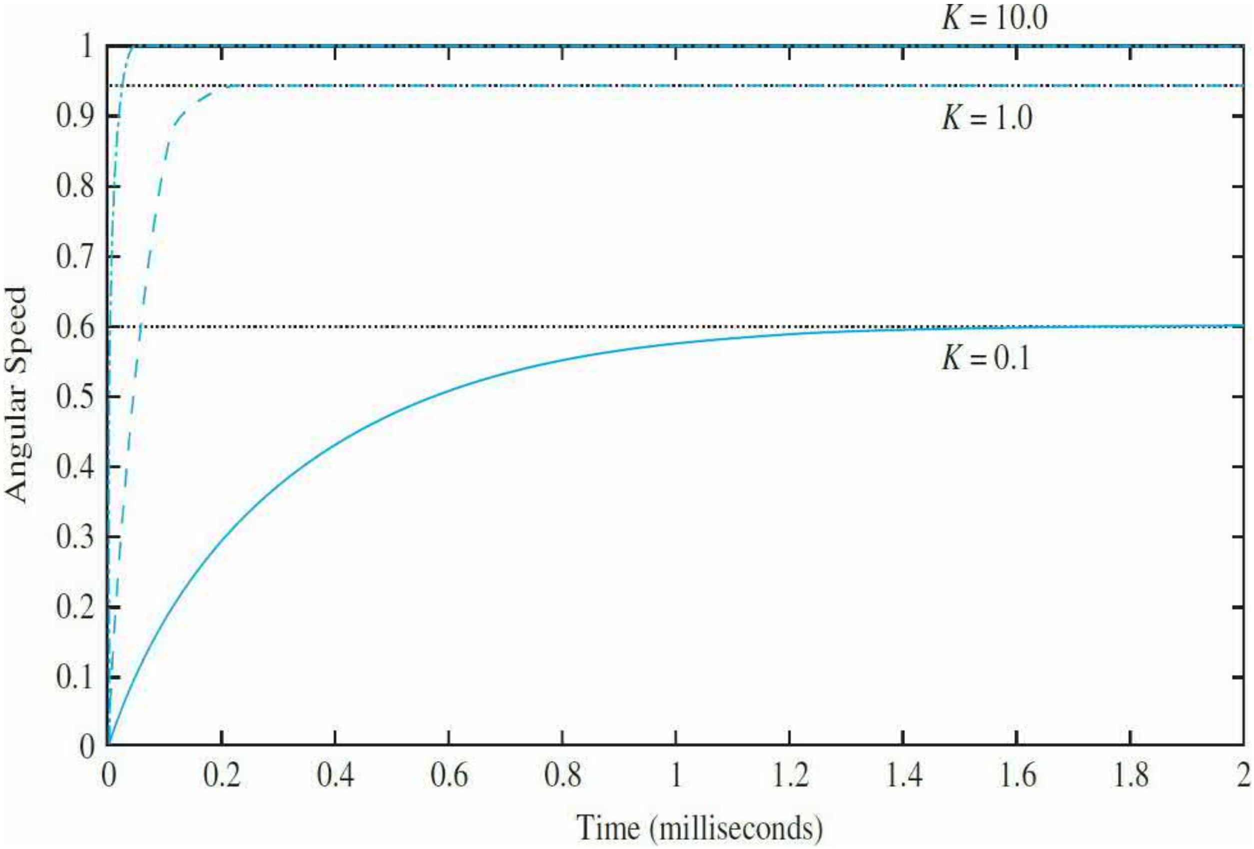
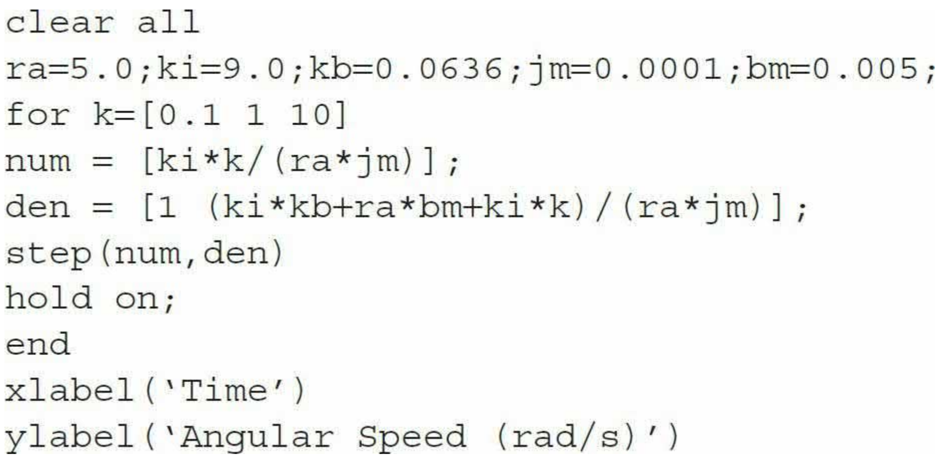


Figure 7-27 System time response for three different controller gain values when $K_t = 1$.

Toolbox 7-6-1

Speed response of Example 7-6-4 using MATLAB.



```
clear all
ra=5.0;ki=9.0;kb=0.0636;jm=0.0001;bm=0.005;
for k=[0.1 1 10]
num = [ki*k/(ra*jm)];
den = [1 (ki*kb+ra*bm+ki*k)/(ra*jm)];
step(num,den)
hold on;
end
xlabel(‘Time’)
ylabel(‘Angular Speed (rad/s)’)
```

7-6-2 Steady-State Error in Systems with a Disturbance

Not all system errors are defined with respect to the response due to the input. Figure 7-28 shows a unity-feedback system with a disturbance *D*(*s*), in addition to the input *R*(*s*). The output due to *D*(*s*) acting alone may also be considered an error. As discussed in Sec. 4-1-4, disturbance usually adversely affects the performance of the control system by placing a burden on the controller/actuator components. Before designing a proper controller for the system, it is always important to learn the effects of *D*(*s*) on the system.

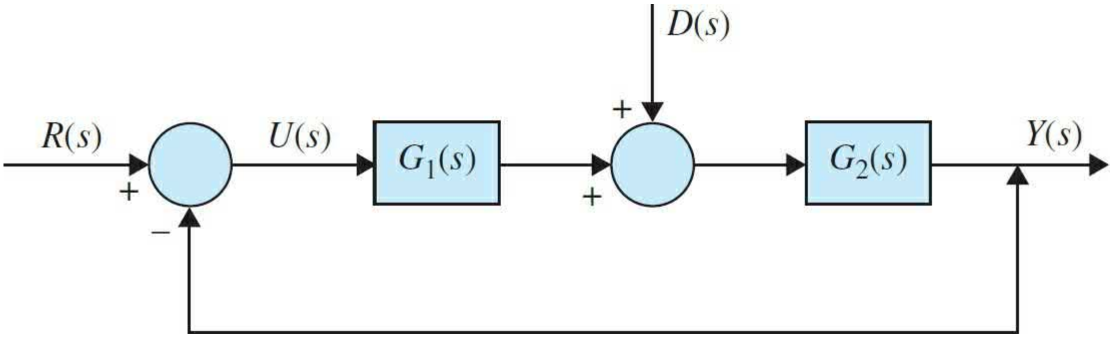


Figure 7-28 Block diagram of a system undergoing disturbance.

For the system in Fig. 7-28, using the principle of superposition, the output can be written in terms of the input *R*(*s*) and disturbance *D*(*s*) as

$$Y_{\text{total}} = \left.\frac{Y(s)}{R(s)}\right|_{D=0} R(s) + \left.\frac{Y(s)}{D(s)}\right|_{R=0} D(s)$$

$$Y(s) = \frac{G_1G_2}{1+G_1G_2}R(s) + \frac{-G_2}{1+G_1G_2}D(s) \qquad (7\text{-}102)$$

and the error is

$$E(s) = \frac{1}{1+G_1G_2}R(s) + \frac{-G_2}{1+G_1G_2}D(s) \qquad (7\text{-}103)$$

The **steady-state error** of the system is defined as

$$e_{ss} = \lim_{s \to 0} sE(s) = \lim_{s \to 0} s \frac{1}{1+G_1G_2} R(s) + \lim_{s \to 0} s \frac{-G_2}{1+G_1G_2} D(s) \tag{7-104}$$

where

$$e_{ss}(R) = \lim_{s \to 0} s \frac{1}{1+G_1G_2} R(s) \tag{7-105}$$

and

$$e_{ss}(D) = \lim_{s \to 0} s \frac{-G_2}{1+G_1G_2} D(s) \tag{7-106}$$

are the steady-state errors due to the reference input and disturbance, respectively.

Observations

$e_{ss}(R)$ and $e_{ss}(D)$ have the same denominators if the disturbance signal is in the forward path. The negative sign in the numerator of $e_{ss}(D)$ shows that, at steady state, the disturbance signal adversely affects the performance of the system. Naturally, to compensate for this burden, the control system has to alter the system performance at steady state.

EXAMPLE 7-6-5 Follow up to Example 7-6-4, we now add a disturbance torque *TL* to the motor speed control system as shown in Fig. 7-29. Note that in order to use Eq. (7-60) for steady-state error calculations, the gain K_t is moved to the forward path to create a unity-feedback system—compare the block diagram in Fig. 7-29 to Fig. 7-26. Simplifying the block diagram in Fig. 7-29, we get

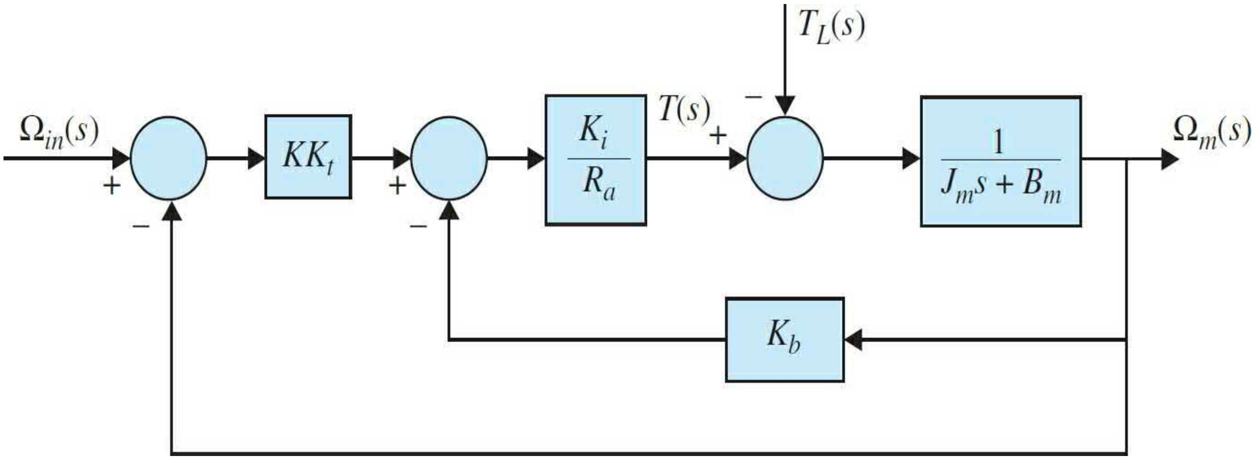


Figure 7-29 Speed control block diagram of a dc motor.

$$\Omega_m(s) = \frac{\dfrac{K_tK_iK}{R_aJ_m}}{s+\left(\dfrac{K_iK_b+R_aB_m+K_tK_iK}{R_aJ_m}\right)}\Omega_{in}(s) - \frac{\dfrac{1}{J_m}}{s+\left(\dfrac{K_iK_b+R_aB_m+K_tK_iK}{R_aJ_m}\right)}T_L(s) \qquad (7\text{-}107)$$

Speed Control of a DC Motor with a Disturbance

For a unit step input $\Omega_{in} = 1/s$ and a unit disturbance torque $T_L = 1/s$, the output becomes

$$\omega_m(t) = \frac{KK_iK_t}{R_aJ_m}\tau_c(1-e^{-t/\tau_c}) - \frac{\tau_c}{J_m}(1-e^{-t/\tau_c}) \qquad (7\text{-}108)$$

where $\tau_c = \dfrac{R_aJ_m}{K_iK_b+R_aB+K_tK_iK}$ is the system time constant. The steady-state response and the steady-state errors in this case are

$$\omega_{fv} = \left(\frac{KK_iK_t}{K_iK_b + R_aB_m + K_tK_iK} - \frac{R_a}{K_iK_b + R_aB_m + K_tK_iK} \right) \tag{7-108a}$$

$$e_{ss}(\Omega_{in}) = \frac{K_iK_b + R_aB_m}{K_iK_b + R_aB_m + K_iK} \tag{7-108b}$$

$$e_{ss}(T_L) = \frac{R_a}{K_iK_b + R_aB_m + KK_iK} \tag{7-108c}$$

As in the previous Example 7-6-4, as *K* increases, the steady-state error due to both input and disturbance signals decrease. ▲

7-6-3 Types of Control Systems: Unity-Feedback Systems

The steady-state error e_{ss} depends on the type of the control system.

Consider that a control system with unity feedback can be represented by or simplified to the block diagram with *H*(*s*) = 1 in Fig. 7-22. The steady-state error of the system is written as

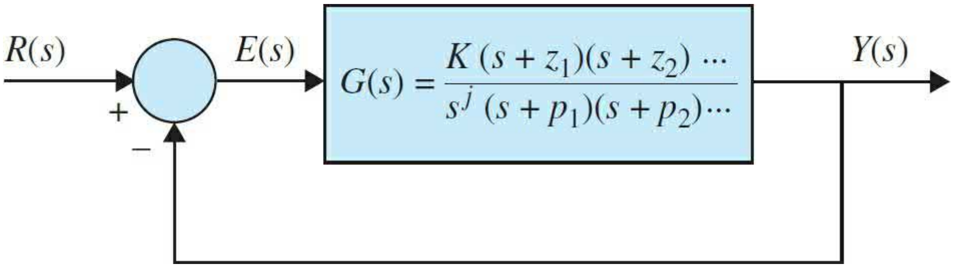
$$\begin{aligned} e_{ss} &= \lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) \\ &= \lim_{s\to 0} \frac{sR(s)}{1+G(s)} \end{aligned} \tag{7-110}$$

Clearly, e_{ss} depends on the characteristics of *G*(*s*). More specifically, we can show that e_{ss} depends on the number of poles *G*(*s*) has at *s* = 0. This number is known as the type of the control system, or simply, **system type**.

Let us formalize the system type by referring to the form of the forward-path transfer function *G*(*s*). In general, *G*(*s*) can be expressed for convenience as

$$G(s) = \frac{K(s+z_1)(s+z_2)\cdots}{s^j(s+p_1)(s+p_2)\cdots} \tag{7-111}$$

the system type refers to the order of the pole of $G(s)$ at $s = 0$. Thus, the closed-loop system in Fig. 7-30 having the forward-path transfer function of Eq. (7-111) is type j, where $j = 0, 1, 2, \ldots$. The total number of terms in the numerator and the denominator and the values of the coefficients are not important to the system type, as system type refers only to the number of poles $G(s)$ has at $s = 0$. The following example illustrates the system type with reference to the form of $G(s)$.



R(s)
E(s)
+
−
$G(s) = \dfrac{K(s+z_1)(s+z_2)\cdots}{s^j(s+p_1)(s+p_2)\cdots}$
Y(s)

Figure 7-30 A unity-feedback control system used to define system type.

EXAMPLE 7-6-6 For the following transfer functions, in the block diagram shown in Fig. 7-30, the system types are defined as

$$G(s) = \frac{K(1+0.5s)}{s(1+s)(1+2s)(1+s+s^2)} \quad \text{type 1} \tag{7-112}$$

$$G(s) = \frac{K(1+2s)}{s^3} \quad \text{type 3} \tag{7-113}$$

Now let us investigate the effects of the types of inputs on the steady-state error. We shall consider only the step, ramp, and parabolic inputs. ▲

7-6-4 Error Constants

Steady-State Error of System with a Step-Function Input

For the unity-feedback control system in Fig. 7-22, when the input $r(t)$ is a step function with magnitude R, $R(s) = R/s$, the steady-state error is written from Eq. (7-60),

$$e_{ss} = \lim_{s\to 0} \frac{sR(s)}{1+G(s)} = \lim_{s\to 0} \frac{R}{1+G(s)} = \frac{R}{1+\lim_{s\to 0} G(s)} \tag{7-114}$$

For convenience, we define

$$K_p = \lim_{s\to 0} G(s) \tag{7-115}$$

as the **step-error constant**. Then Eq. (7-114) becomes

$$e_{ss} = \frac{R}{1+K_p} \tag{7-116}$$

A typical e_{ss} due to a step input when K_p is finite and nonzero is shown in Fig. 7-31. We see from Eq. (7-116) that, for e_{ss} to be zero, when the input is a step function, K_p must be infinite. If $G(s)$ is described by Eq. (7-111), we see that, for K_p to be infinite, j must be at least equal to unity; that is, $G(s)$ must have at least one pole at $s = 0$. Therefore, we can summarize the steady-state error due to a step function input as follows:

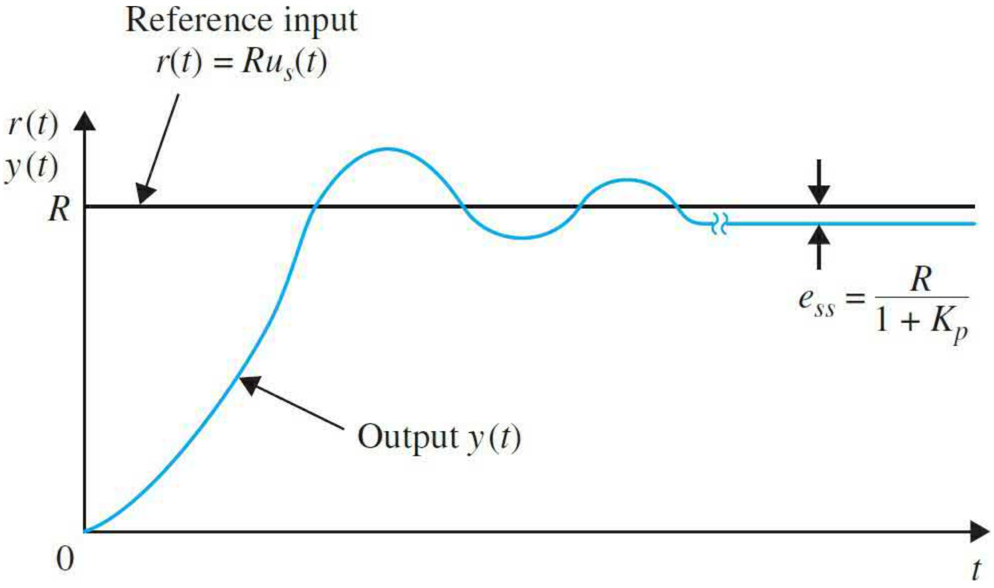


Figure 7-31 Typical steady-state error due to a step input.

$$\text{Type 0 system: } e_{ss} = \frac{R}{1+K_p} = \text{constant}$$

$$\text{Type 1 or higher system: } e_{ss} = 0$$

Steady-State Error of System with a Ramp-Function Input

For the unity-feedback control system in Fig. 7-22, when the input to is a ramp-function with magnitude *R*,

$$r(t) = Rtu_s(t) \tag{7-117}$$

where *R* is a real constant, the Laplace transform of *r*(*t*) is

$$R(s) = \frac{R}{s^2} \tag{7-118}$$

The steady-state error is written using Eq. (7-60),

$$e_{ss} = \lim_{s\to 0} \frac{R}{s+sG(s)} = \frac{R}{\lim_{s\to 0} sG(s)} \tag{7-119}$$

We define the ramp-error constant as

$$K_v = \lim_{s\to 0} sG(s) \tag{7-120}$$

Then, Eq. (7-119) becomes

$$e_{ss} = \frac{R}{K_v} \tag{7-121}$$

which is the steady-state error when the input is a ramp function. A typical e_{ss} due to a ramp input when K_v is finite and nonzero is illustrated in Fig. 7-32.

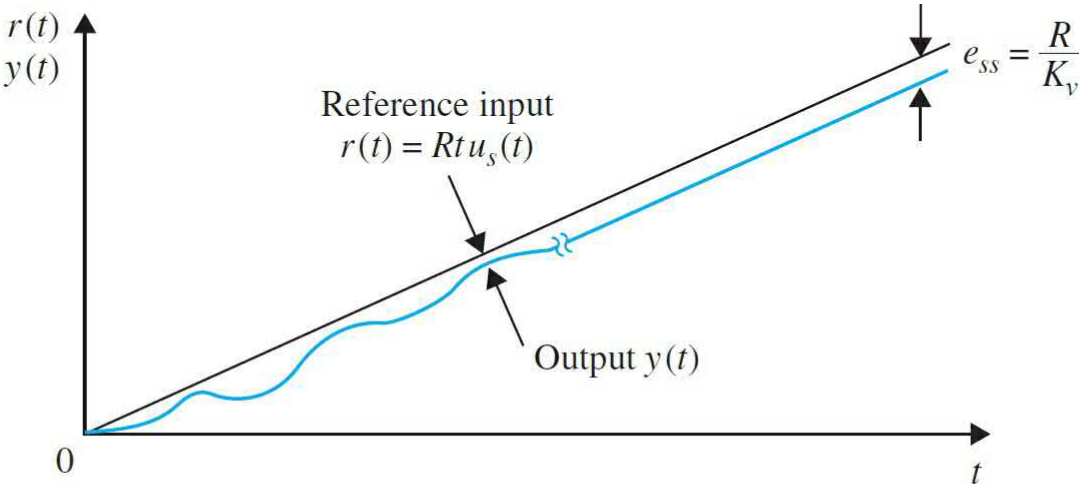


Figure 7-32 Typical steady-state error due to a ramp-function input.

Equation (7-121) shows that, for e_{ss} to be zero when the input is a ramp function, K_v must be infinite. Using Eqs. (7-120) and (7-111), we obtain

$$K_v = \lim_{s\to 0} sG(s) = \lim_{s\to 0} \frac{K}{s^{j-1}} \quad j = 0, 1, 2, \ldots \tag{7-122}$$

Thus, for K_v to be infinite, j must be at least equal to 2, or the system must be of type 2 or higher. The following conclusions may be stated with regard to the steady-state error of a system with ramp input:

Type 0 system: $e_{ss} = \infty$

Type 1 system: $e_{ss} = \dfrac{R}{K_v} = \text{constant}$

Type 2 system: $e_{ss} = 0$

Steady-State Error of System with a Parabolic-Function Input

When the input is described by the standard parabolic form

$$r(t) = \frac{Rt^2}{2} u_s(t) \tag{7-123}$$

the Laplace transform of $r(t)$ is

$$R(s) = \frac{R}{s^3} \tag{7-124}$$

The steady-state error of the system in Fig. 7-22 is

$$e_{ss} = \frac{R}{\lim_{s \to 0} s^2 G(s)} \tag{7-125}$$

A typical e_{ss} of a system with a nonzero and finite K_a due to a parabolic-function input is shown in Fig. 7-33.

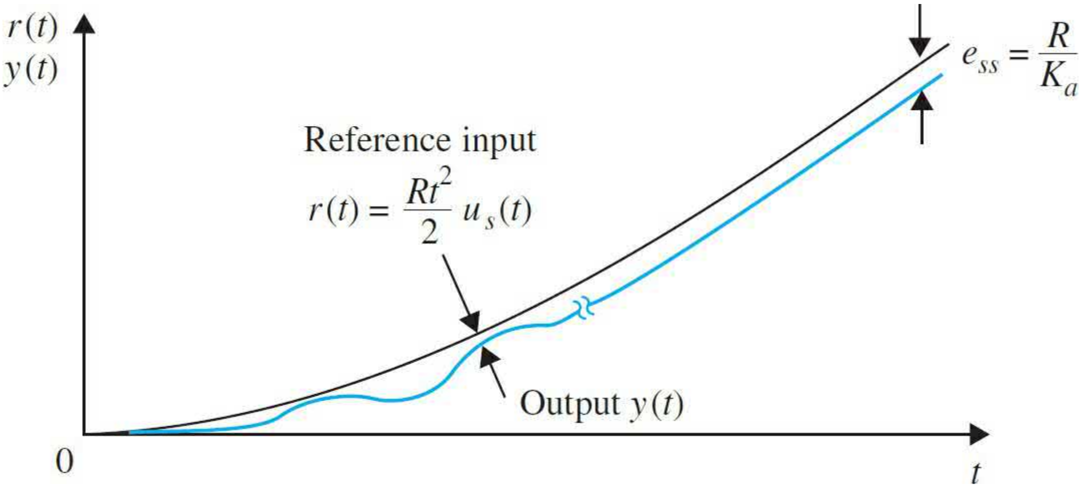


Figure 7-33 Typical steady-state error due to a parabolic-function input.

Defining the parabolic-error constant as

$$K_a = \lim_{s \to 0} s^2 G(s) \tag{7-126}$$

the steady-state error becomes

$$e_{ss} = \frac{R}{K_a} \tag{7-127}$$

Following the pattern set with the step and ramp inputs, the steady-state error due to the parabolic input is zero if the system is of type 3 or greater. The following conclusions are made with regard to the steady-state error of a system with parabolic input:

Type 0 system: $e_{ss} = \infty$

Type 1 system: $e_{ss} = \infty$

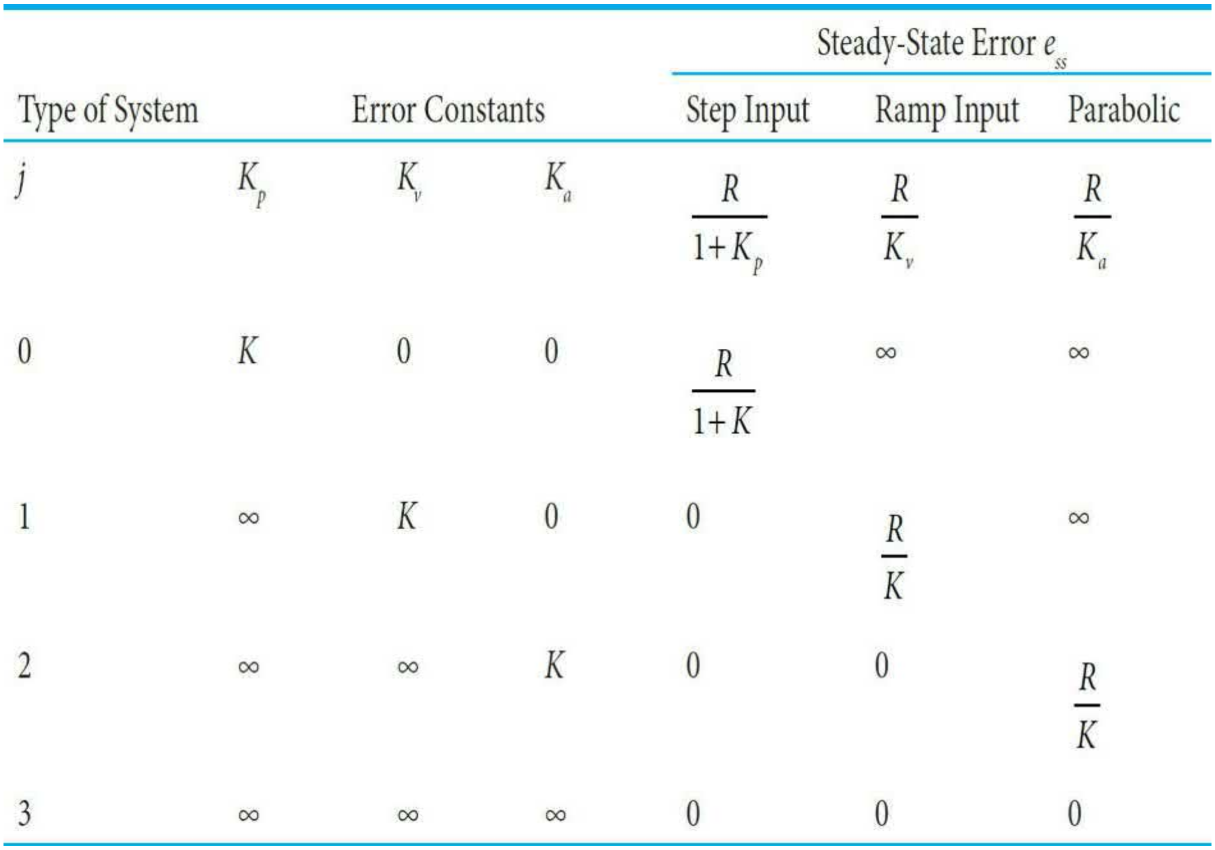
Type 2 system: $e_{ss} = \dfrac{R}{K_a} = \text{constant}$

Type 3 or higher system: $e_{ss} = 0$

Obviously, for these results to be valid, the closed-loop system must be **stable**.

By using the method described, the steady-state error of any linear closed-loop system subject to an input with an order higher than the parabolic function can also be derived if necessary. As a summary of the error analysis, Table 7-6 shows the relations among the error constants, the types of systems with reference to Eq. (7-111), and the input types.

TABLE 7-6 Summary of the Steady-State Errors Due to Step-, Ramp-, and Parabolic-Function Inputs for Unity-Feedback Systems



				Steady-State Error e_{ss}		
Type of System	Error Constants			Step Input	Ramp Input	Parabolic
j	K_p	K_v	K_a	$\dfrac{R}{1+K_p}$	$\dfrac{R}{K_v}$	$\dfrac{R}{K_a}$
0	K	0	0	$\dfrac{R}{1+K}$	∞	∞
1	∞	K	0	0	$\dfrac{R}{K}$	∞
2	∞	∞	K	0	0	$\dfrac{R}{K}$
3	∞	∞	∞	0	0	0

As a summary, the following points should be noted when applying the error-constant analysis just presented.

1. The step-, ramp-, or parabolic-error constants are significant for the error analysis only when the input signal is a step function, ramp function, or parabolic function, respectively.
2. Because the error constants are defined with respect to the forward-path transfer function $G(s)$, the method is applicable to only the system configuration shown in Fig. 7-22 with a unity feedback. Because the error analysis relies on the use of the final-value theorem of the Laplace transform, it is important to first check the stability of the system to see if $sE(s)$ has any poles on the $j\omega$-axis or in the right-half s-plane.
3. The steady-state error properties summarized in Table 7-6 are for systems with unity feedback only.
4. The steady-state error of a system with an input that is a linear combination of the three basic types of inputs can be determined by superimposing the errors due to each input component.
5. When the system configuration differs from that of Fig. 7-22 with

H(*s*) = 1, we can either simplify the system to the form of Fig. 7-22 or establish the error signal and apply the final-value theorem. The error constants defined here may or may not apply, depending on the individual situation.

When the steady-state error is infinite, that is, when the error increases continuously with time, the error-constant method does not indicate how the error varies with time. This is one of the disadvantages of the error-constant method. The error-constant method also does not apply to systems with inputs that are sinusoidal, since the final-value theorem cannot be applied. The following examples illustrate the utility of the error constants and their values in the determination of the steady-state errors of linear control systems with unity feedback.

EXAMPLE 7-6-7 Consider that the system shown in Fig. 7-22, that is, ***H(s)* = *1***, has the following transfer functions. The error constants and steady-state errors are calculated for the three basic types of inputs using the error constants.

a. $G(s) = \dfrac{K(s+3.15)}{s(s+1.5)(s+0.5)}$ $H(s)=1$ Type 1 system

$$K_p = \infty \; e_{ss} = \frac{R}{1+K_p} = 0$$

Step input: Step-error constant

Ramp input: Ramp-error constant

$$K_v = 4.2K \; e_{ss} = \frac{R}{K_v} = \frac{R}{4.2K}$$

Parabolic input: Parabolic-error constant

$$K_a = 0 \; e_{ss} = \frac{R}{K_a} = \infty$$

These results are valid only if the value of *K* stays within the range that corresponds to a stable closed-loop system, which is 0 < *K* < 1.304

b. $G(s) = \dfrac{K}{s^2(s+12)}$ $H(s)=1$ Type 2 system

The closed-loop system is unstable for all values of *K*, and error analysis is meaningless.

$$\textbf{c.}\quad G(s)=\frac{5(s+1)}{s^2(s+12)(s+5)}\quad H(s)=1\quad \text{Type 2 system}$$

We can show that the closed-loop system is stable. The steady-state errors are calculated for the three basic types of inputs.

Step input: Step-error constant: $K_p = \infty \; e_{ss} = \dfrac{R}{1+K_p} = 0$

Ramp input: Ramp-error constant: $K_v = \infty \; e_{ss} = \dfrac{R}{K_v} = 0$

Parabolic input: Parabolic-error constant:

$$K_a = 1/12 \; e_{ss} = \frac{R}{K_a} = 12R$$

7-6-5 Steady-State Error Caused by Nonlinear System Elements

In many instances, steady-state errors of control systems are attributed to some nonlinear system characteristics such as nonlinear friction or dead zone. For instance, in real-life applications, an amplifier used in a control system may have the input-output characteristics shown in Fig. 7-44. Then, when the amplitude of the amplifier input signal falls within the dead zone, the output of the amplifier would be zero, and the control would not be able to correct the error if any exists. Dead-zone nonlinearity characteristics shown in Fig. 7-34 are not limited to amplifiers. The flux-to-current relation of the magnetic field of an electric motor may exhibit a similar characteristic. As the current of the motor falls below the dead zone D, no magnetic flux, and, thus, no torque will be produced by the motor to move the load.

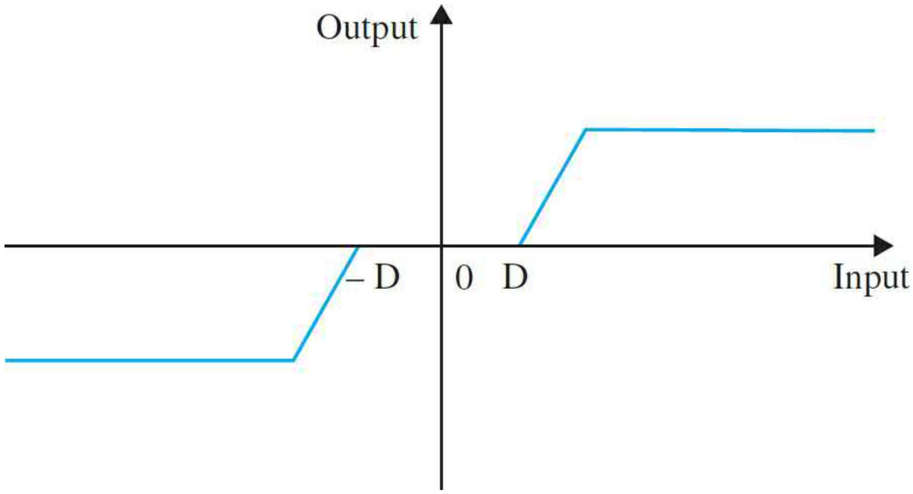


Figure 7-34 Typical input-output characteristics of an amplifier with dead zone and saturation.

The output signals of digital components used in control systems, such as a microprocessor, can take on only discrete or quantized levels. This property is illustrated by the quantization characteristics shown in Fig. 7-35. When the input to the quantizer is within ±$q/2$, the output is zero, and the system may generate an error in the output whose magnitude is related to ±$q/2$. This type of error is also known as the quantization error in digital control systems.

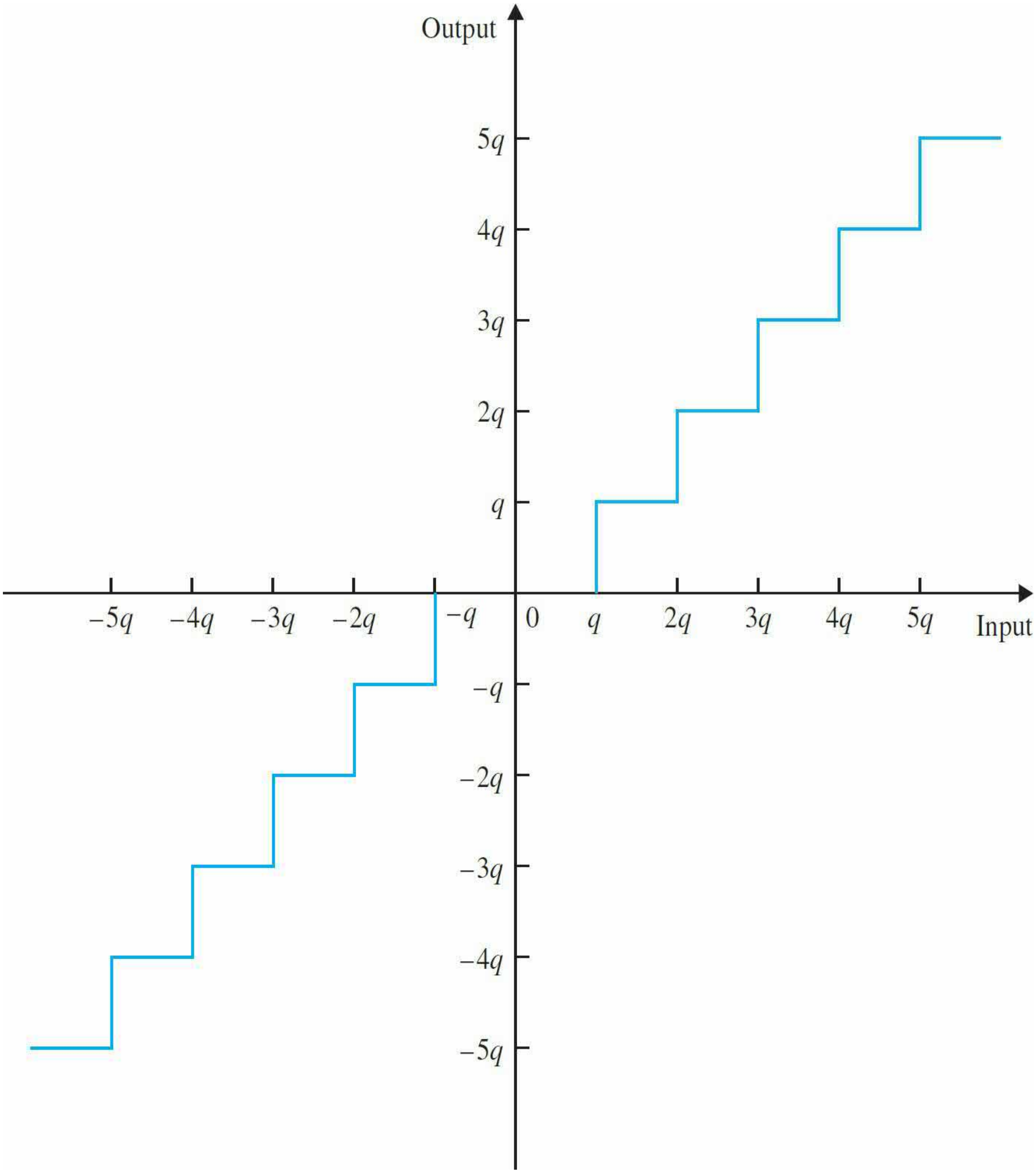


Figure 7-35 Typical input-output characteristics of a quantizer.

When the control of physical objects is involved, friction is almost always present. Coulomb friction is a common cause of steady-state position errors in control systems. Figure 7-36 shows a restoring-torque-versus-position curve of a control system. The torque curve typically could be generated by a step motor or a switched-reluctance motor or from a closed-loop system with a position encoder. Point 0 designates a stable equilibrium point on the torque curve, as well as the other periodic intersecting points along the axis, where the slope on the torque curve is negative. The torque on either side of point 0 represents a restoring torque that tends to return the output to the equilibrium point when some angular-displacement disturbance takes place. When there is no friction, the position error should be zero because there is always a restoring torque so long as the position is not at the stable equilibrium point. If the rotor of the motor sees a Coulomb friction torque T_F, then the motor torque must first overcome this frictional torque before producing any motion. Thus, as the motor torque falls below T_F as the rotor position approaches the stable equilibrium point, it may stop at any position inside the error band bounded by $\pm\theta_e$, as shown in Fig. 7-36.

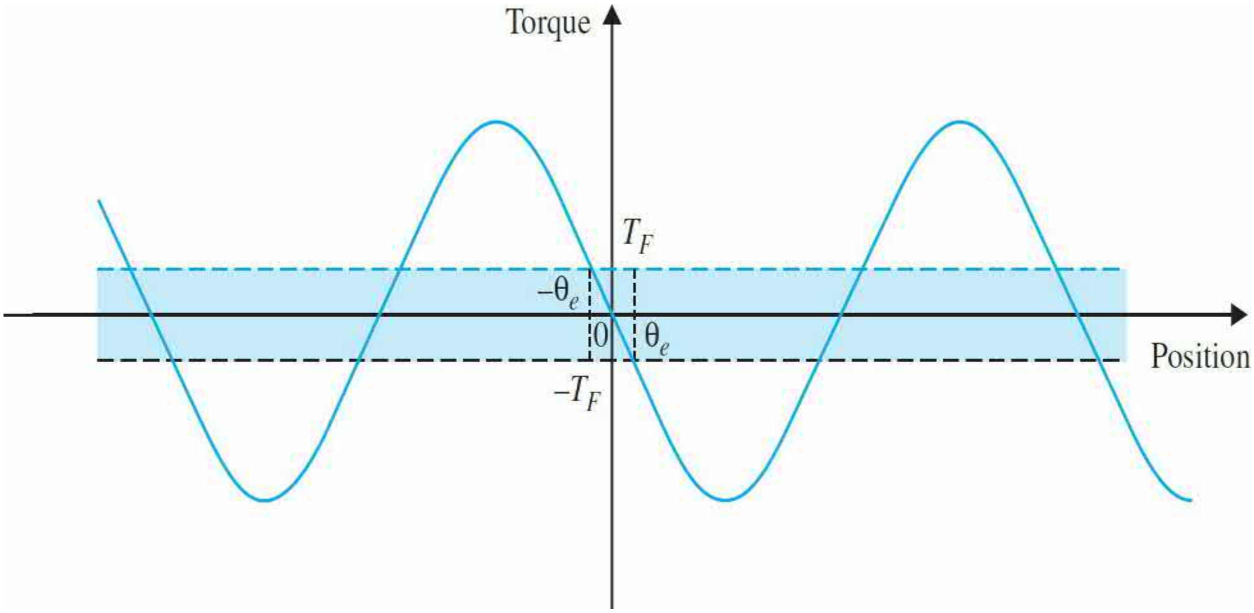


Figure 7-36 Torque-angle curve of a motor or closed-loop system with Coulomb friction.

Although it is relatively simple to comprehend the effects of nonlinearities on errors and to establish maximum upper bounds on the error magnitudes, it is difficult to establish general and closed-form solutions for nonlinear systems. Usually, exact and detailed analysis of errors in nonlinear control systems can be carried out only by computer simulations.

Therefore, we must realize that there are no error-free control systems in the real world, and, because all physical systems have nonlinear characteristics of one form or another, steady-state errors can be reduced but never completely eliminated.

7-7 BASIC CONTROL SYSTEMS AND EFFECTS OF ADDING POLES AND ZEROS TO TRANSFER FUNCTIONS

In all previous examples of control systems we have discussed thus far, the controller has been typically a simple amplifier with a constant gain *K*. This type of control action is formally known as **proportional control** because the control signal at the output of the controller is simply related to the input of the controller by a proportional constant.

Intuitively, one should also be able to use the derivative or integral of the input signal, in addition to the proportional operation. Therefore, we can consider a more general continuous-data controller to be one that contains such components as adders or summers (addition or subtraction), amplifiers, attenuators, differentiators, and integrators—see Sec. 6-1 and Chap. 11 for more details. For example, one of the best-known controllers used in practice is the PID controller, which stands for **proportional**, **integral**, and **derivative**. The integral and derivative components of the PID controller have individual performance implications, and their applications require an understanding of the basics of these elements.

All in all, what these controllers do is *add additional poles and zeros* to the open- or closed-loop transfer function of the overall system. As a result, it is important to appreciate the effects of adding poles and zeros to a transfer function first. We show that—although the roots of the characteristic equation of the system, which are the poles of the closed-loop transfer function, affect the transient response of linear time-invariant control systems, particularly the stability—the zeros of the transfer function are also important. Thus, the addition of poles and zeros and/or cancellation of undesirable poles and zeros of the transfer function often are necessary in achieving satisfactory time-domain performance of control systems.

In this section, we show that the addition of poles and zeros to forward-path and closed-loop transfer functions has varying effects on the transient response of the closed-loop system.

7-7-1 Addition of a Pole to the Forward-Path Transfer Function: Unity-Feedback Systems

To study the general effect of the addition of a pole, and its relative location, to a forward-path transfer function of a unity-feedback system, consider the transfer function

$$G(s) = \frac{\omega_n^2}{s(s+2\zeta\omega_n)(1+T_p s)} \tag{7-128}$$

The pole at $s = -1/T_p$ is considered to be added to the prototype second-order transfer function. The closed-loop transfer function is written as

$$M(s) = \frac{Y(s)}{R(s)} = \frac{G(s)}{1+G(s)} = \frac{\omega_n^2}{T_p s^3 + (1+2\zeta\omega_n T_p)s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{7-129}$$

Table 7-7 shows the poles of the closed-loop system when $\omega_n = 1$, $\zeta = 1$ and $T_p = 0$, 1, 2, and 5. As the value of T_p increases, the **open loop** added pole at $-1/T_p$ **moves closer to the origin** in the *s*-plane, causing the closed-loop system to have a pair of complex conjugate poles that move toward the origin. Figure 7-37 illustrates the unit-step responses of the closed-loop system. As the value of T_p increases, the open loop added pole at $-1/T_p$ **moves closer to the origin** in the *s*-plane, and the **maximum overshoot increases**. These responses also show that the added pole **increases the rise time** of the step response.

TABLE 7-7 The Poles of the Closed-Loop System in Eq. (7-129) when $\omega_n = 1$, $\zeta = 1$ and $T_p = 0$, 1, 2, and 5

T_p	Poles		
0	Prototype Second-Order System	$s_1 = -1$	$s_2 = -1$
1	$s_1 = -2.32$	$s_2 = -0.34 + j\,0.56$	$s_3 = -0.34 - j\,0.56$
2	$s_1 = -2.14$	$s_2 = -0.18 + j\,0.45$	$s_3 = -0.18 - j\,0.45$
5	$s_1 = -2.05$	$s_2 = -0.07 + j\,0.30$	$s_3 = -0.07 - j\,0.30$

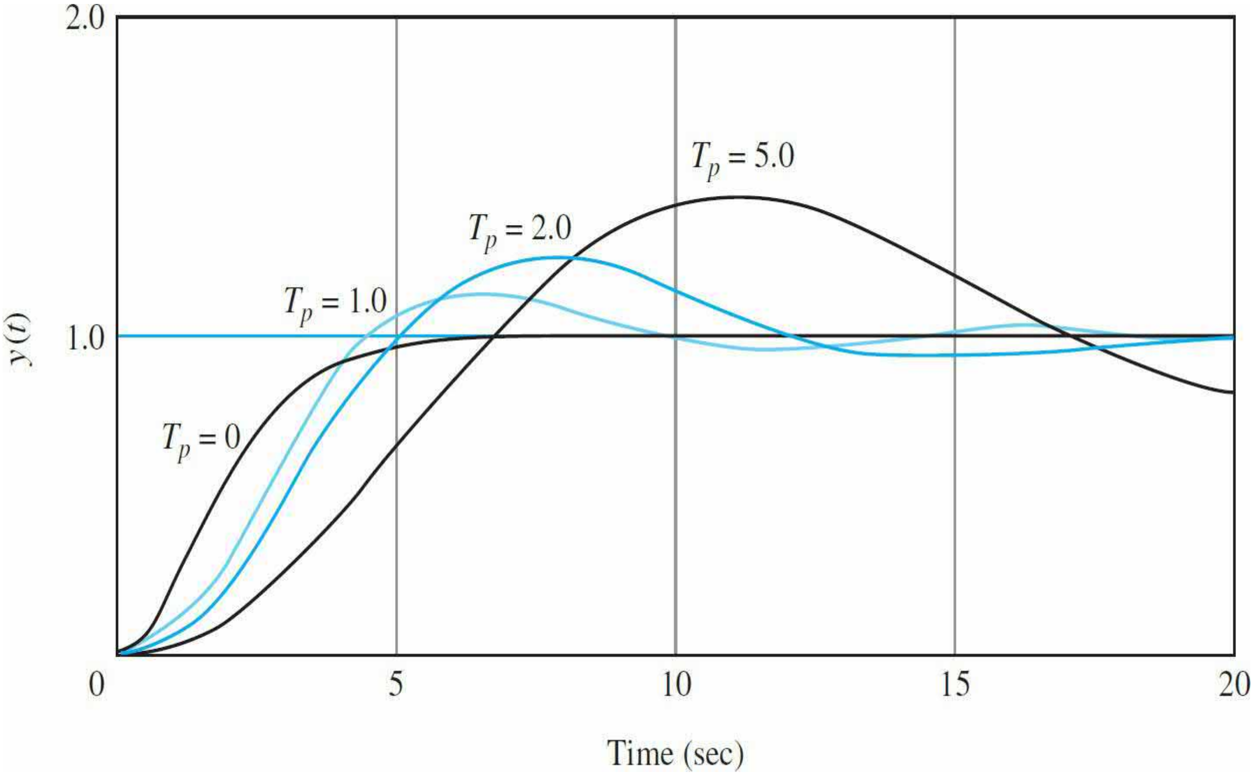


Figure 7-37 Unit-step responses of the system with the closed-loop transfer function in Eq. (7-129): $\zeta = 1$, $\omega_n = 1$ and $T_p = 0, 1, 2,$ and 5.

The same conclusion can be drawn from the unit-step responses of Fig. 7-38, which are obtained for $\omega_n = 1$, $\zeta = 0.25$ and $T_p = 0, 0.2, 0.667,$ and 1.0. In this case, when T_p is greater than 0.667, the amplitude of the unit-step response increases with time, and the system is **unstable**.

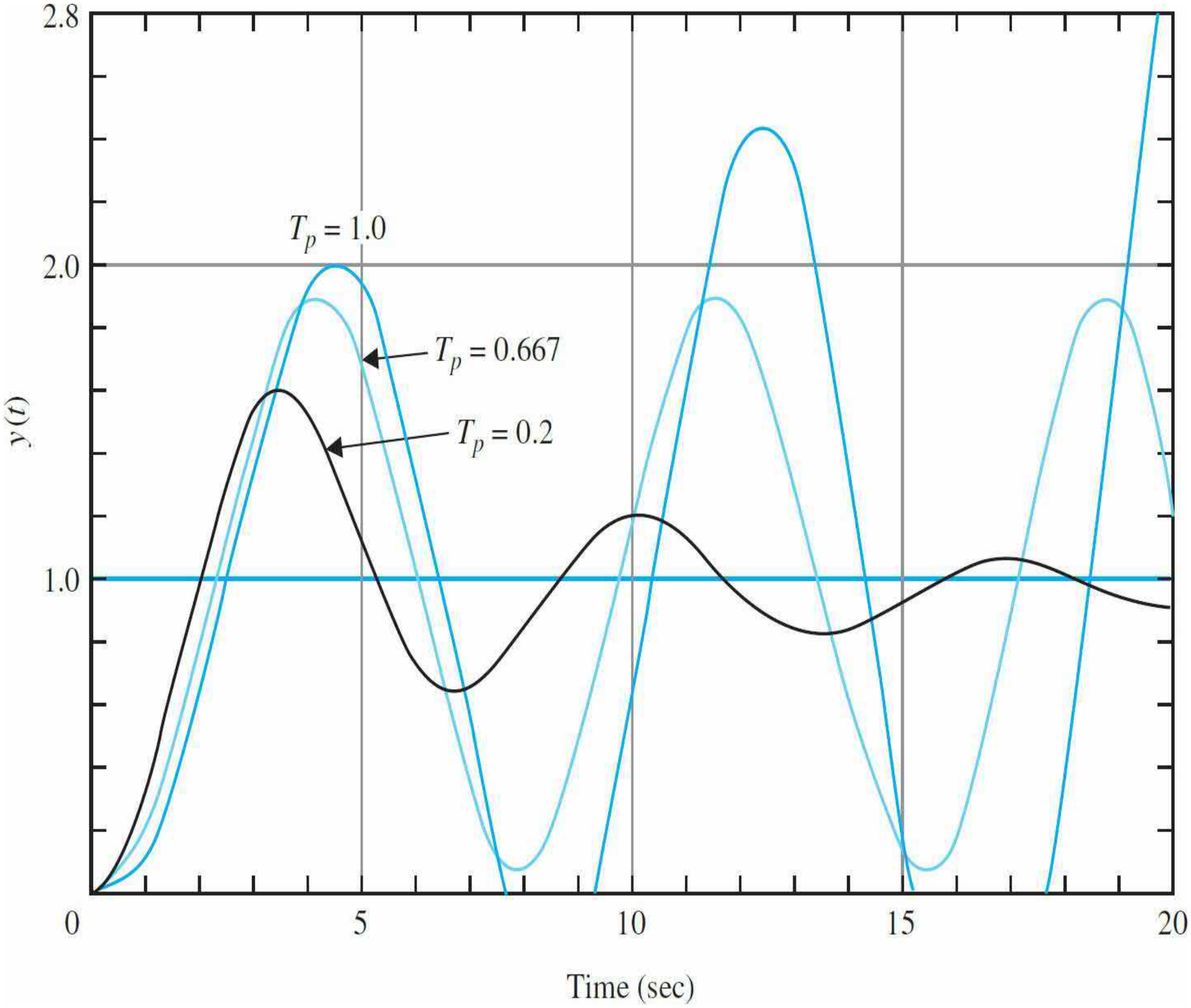


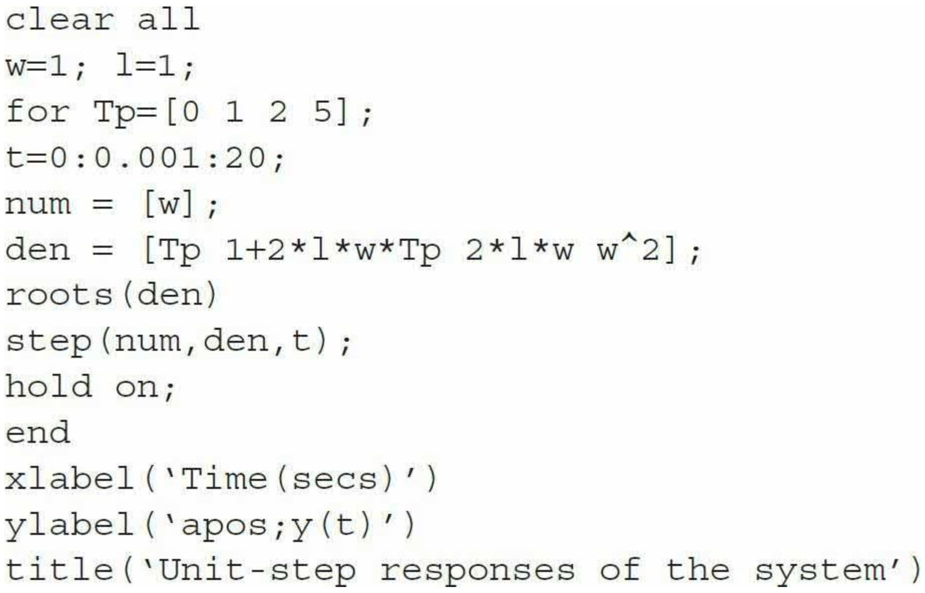
Figure 7-38 Unit-step responses of the system with the closed-loop transfer function in Eq. (7-129): $\zeta = 0.25$, $\omega_n = 1$ and $T_p = 0, 0.2, 0.667$, and 1.0.

In general, *addition of a pole to the forward-path transfer function generally has the effect of increasing the maximum overshoot of the closed-loop system*.

For a more detailed treatment of this subject, please refer to the Case Study in Sec. 7-9.

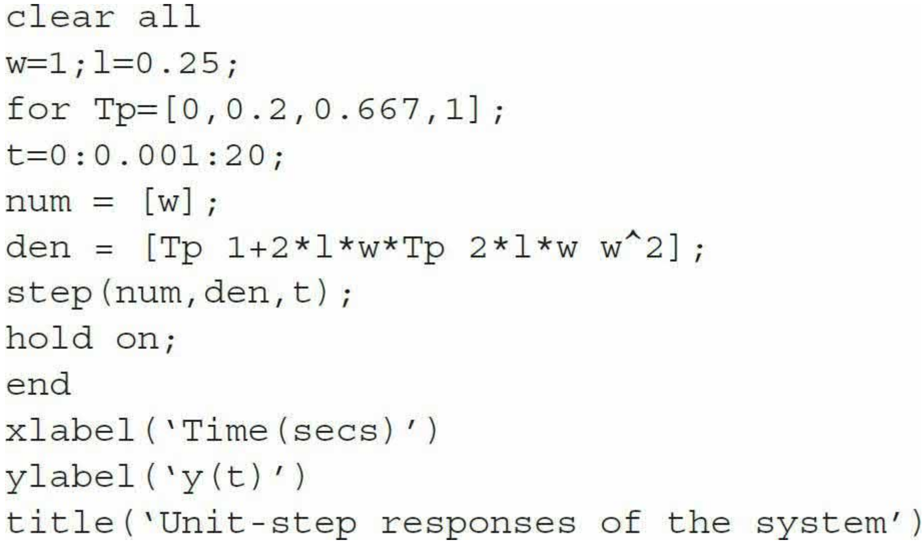
Toolbox 7-7-1

The corresponding responses for Fig. 7-37 are obtained by the following sequence of MATLAB functions:



```
clear all
w=1; l=1;
for Tp=[0 1 2 5];
t=0:0.001:20;
num = [w];
den = [Tp 1+2*l*w*Tp 2*l*w w^2];
roots(den)
step(num,den,t);
hold on;
end
xlabel('Time(secs)')
ylabel('apos;y(t)')
title('Unit-step responses of the system')
```

The corresponding responses for Fig. 7-38 are obtained by the following sequence of MATLAB functions



```
clear all
w=1;l=0.25;
for Tp=[0,0.2,0.667,1];
t=0:0.001:20;
num = [w];
den = [Tp 1+2*l*w*Tp 2*l*w w^2];
step(num,den,t);
hold on;
end
xlabel('Time(secs)')
ylabel('y(t)')
title('Unit-step responses of the system')
```

7-7-2 Addition of a Pole to the Closed-Loop Transfer Function

Because the poles of the closed-loop transfer function are roots of the characteristic equation, they control the transient response of the system directly. Consider the closed-loop transfer function

$$M(s)=\frac{Y(s)}{R(s)}=\frac{\omega_n^2}{(s^2+2\zeta\omega_n s+\omega_n^2)(1+T_p s)} \qquad (7\text{-}130)$$

where the term $(1 + T_p s)$ is added to a prototype second-order transfer function. Table 7-8 shows the poles of the closed-loop system when $\omega_n = 1$, $\zeta = 1$ and $T_p = 0, 0.5, 1, 2$, and 5. As the value of T_p increases, the **closed-loop** added pole at $-1/T_p$ **moves closer to the origin** in the *s*-plane. Figure 7-39 illustrates the unit-step response of the system. As the pole at $s = -1/T_p$ is **moved toward the origin** in the *s*-plane, **the rise time increases** and the **maximum overshoot decreases**. Thus, as far as the overshoot is concerned, adding a pole to the closed-loop transfer function has just the opposite effect to that of adding a pole to the forward-path transfer function.

TABLE 7-8 The Poles of the Closed-Loop System in Eq. (7-130) when $\omega_n = 1$, $\zeta = 1$ and $T_p = 0, 0.5, 1, 2$, and 5

T_p	Poles		
0	Prototype Second-Order System	$s_1 = -0.5 + j\,0.87$	$s_2 = -0.5 - j\,0.87$
0.5	$s_1 = -2$	$s_2 = -0.5 + j\,0.87$	$s_3 = -0.5 - j\,0.87$
1	$s_1 = -1$	$s_2 = -0.5 + j\,0.87$	$s_3 = -0.5 - j\,0.87$
2	$s_1 = -0.5$	$s_2 = -0.5 + j\,0.87$	$s_3 = -0.5 - j\,0.87$
4	$s_1 = -0.25$	$s_2 = -0.5 + j\,0.87$	$s_3 = -0.5 - j\,0.87$

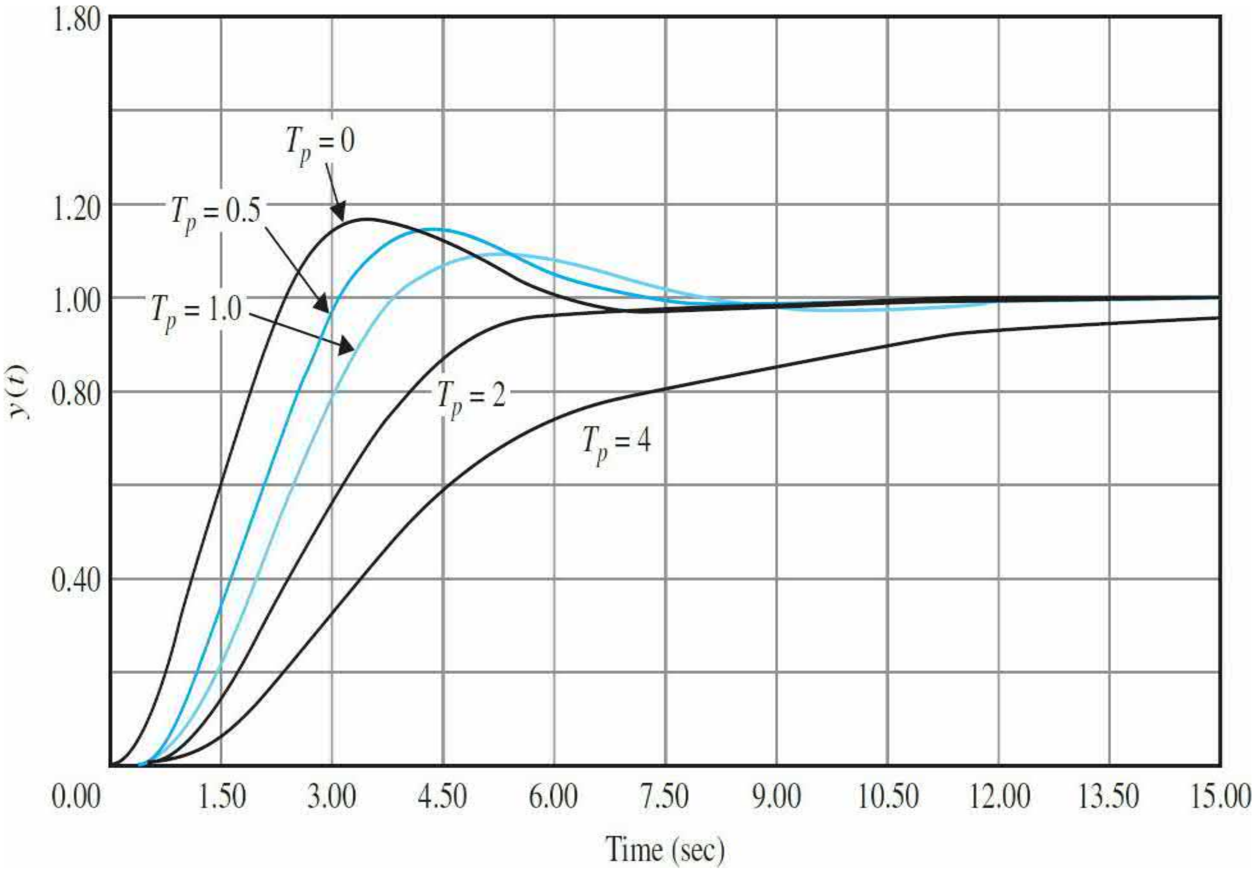
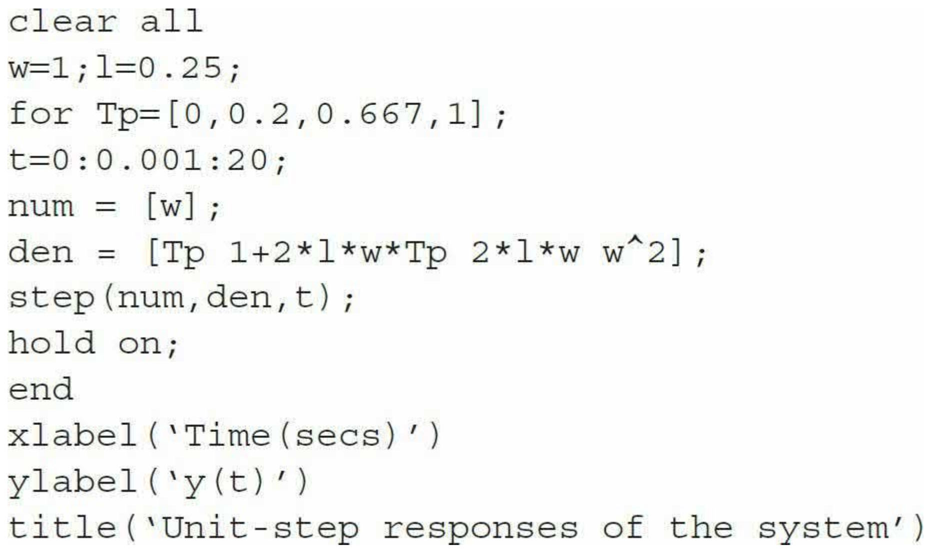


Figure 7-39 Unit-step responses of the system with the closed-loop transfer function in Eq. (7-130): $\zeta = 0.5$, $\omega_n = 1$ and $T_p = 0, 0.5, 1.0, 2.0$, and 4.0.

For $T_p < 2$, the real pole value is less than the real portion of the complex conjugate poles. While for $T_p > 2$, it is closer to origin than the complex poles. As we will discuss later in Sec. 7-8, in the latter case the real pole is less dominant than the complex poles, while in the former case it overpowers the response causing the overshoot to diminish.

Toolbox 7-7-2

The corresponding responses for Fig. 7-39 are obtained by the following sequence of MATLAB functions:



```
clear all
w=1;l=0.25;
for Tp=[0,0.2,0.667,1];
t=0:0.001:20;
num = [w];
den = [Tp 1+2*l*w*Tp 2*l*w w^2];
step(num,den,t);
hold on;
end
xlabel('Time(secs)')
ylabel('y(t)')
title('Unit-step responses of the system')
```

7-7-3 Addition of a Zero to the Closed-Loop Transfer Function

Consider the following closed-loop transfer function with an added zero:

$$M(s)=\frac{Y(s)}{R(s)}=\frac{\omega_n^2(1+T_z s)}{(s^2+2\zeta\omega_n s+\omega_n^2)} \tag{7-131}$$

Table 7-9 shows the roots of the system when $\omega_n = 1$, $\zeta = 0.5$ and $T_z = 0, 1, 2, 3, 6,$ and 10. As the value of T_z increases, the **closed-loop** added zero at $-1/T_z$ **moves closer to the origin** in the *s*-plane. Figure 7-40 shows the unit-step responses of the closed-loop system. In this case, we see that **adding a zero** to the closed-loop transfer function **decreases the rise time** and **increases the maximum overshoot** of the step response.

TABLE 7-9 The Roots of the Closed-Loop System in Eq. (7-131) when $\omega_n = 1$, $\zeta = 0.5$ and $T_z = 0, 1, 2, 3, 6,$ and 10

T_z	Zero	Poles	
0	Prototype Second-Order System	$s_1 = -0.5 + j\,0.87$	$s_2 = -0.5 - j\,0.87$
0.5	$s_1 = -1$	$s_2 = -0.5 + j\,0.87$	$s_3 = -0.5 - j\,0.87$
1	$s_1 = -0.5$	$s_2 = -0.5 + j\,0.87$	$s_3 = -0.5 - j\,0.87$
3	$s_1 = -0.33$	$s_2 = -0.5 + j\,0.87$	$s_3 = -0.5 - j\,0.87$
6	$s_1 = -0.17$	$s_2 = -0.5 + j\,0.87$	$s_3 = -0.5 - j\,0.87$
10	$s_1 = -0.10$	$s_2 = -0.5 + j\,0.87$	$s_3 = -0.5 - j\,0.87$

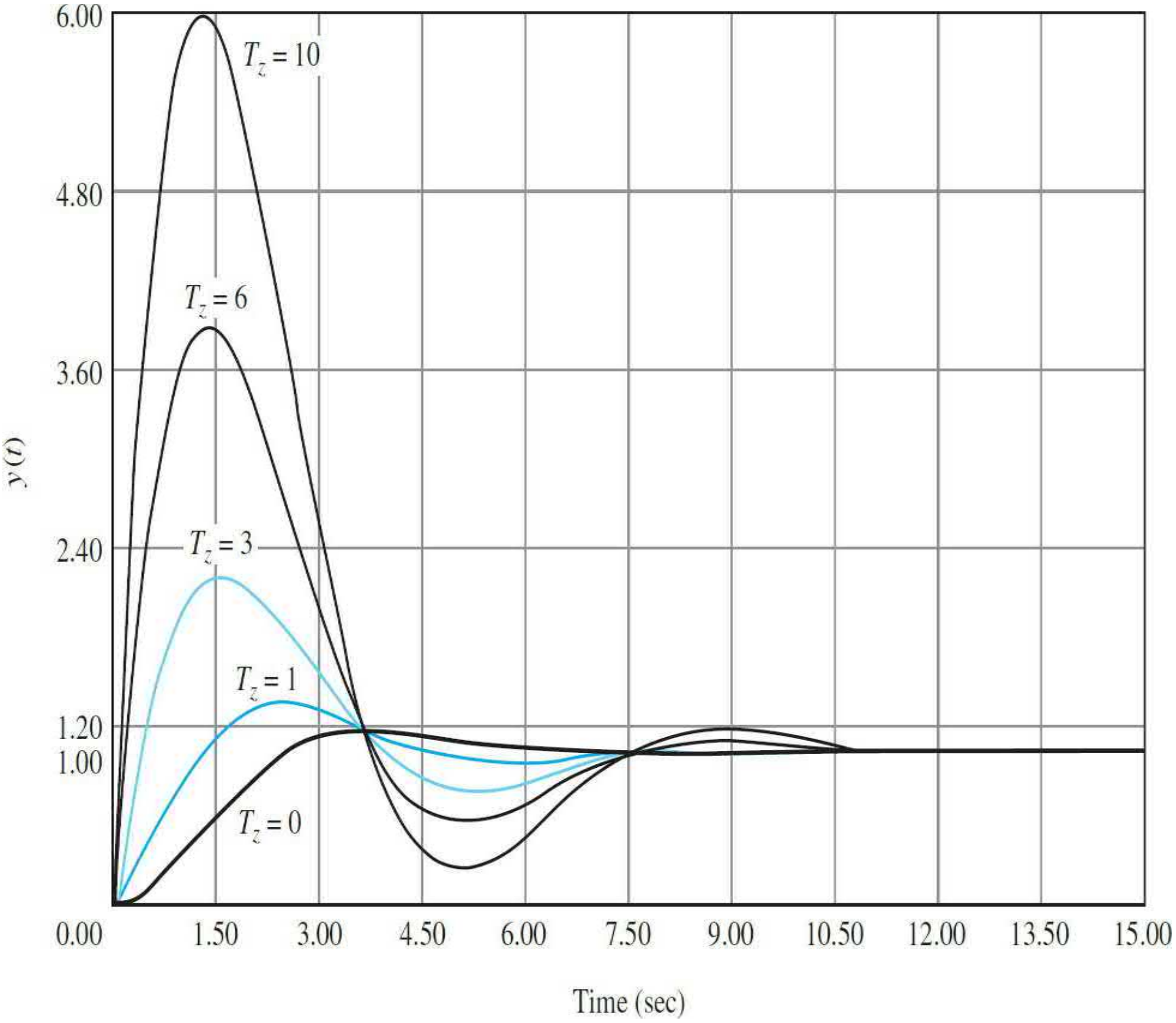


Figure 7-40 Unit-step responses of the system with the closed-loop

transfer function in Eq. (7-131): T_z = 0, 1, 2, 3, 6, and 10.

For $T_z < 1$, the zero value is less than the real portion of the complex conjugate poles. While for $T_z > 1$, it is closer to origin than the complex poles. As in the previous section, in the latter case the zero is less dominant than the complex poles, while in the former case it overpowers the response causing more overshoot.

We can analyze the general case by writing Eq. (7-131) as

$$M(s)=\frac{Y(s)}{R(s)}=\frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}+\frac{T_z\omega_n^2 s}{s^2+2\zeta\omega_n s+\omega_n^2} \qquad (7\text{-}132)$$

For a unit-step input, let the output response that corresponds to the first term of the right side of Eq. (7-132) be $y_1(t)$. Then, the total unit-step response is

$$y(t)=y_1(t)+T_z\frac{dy_1(t)}{dt} \qquad (7\text{-}133)$$

Figure 7-41 shows why the addition of the zero at s = –1/T_c reduces the rise time and increases the maximum overshoot, according to Eq. (7-133). In fact, as T_z approaches infinity, the maximum overshoot also approaches infinity, and yet the system is still **stable** as long as the overshoot is finite and ζ is positive.

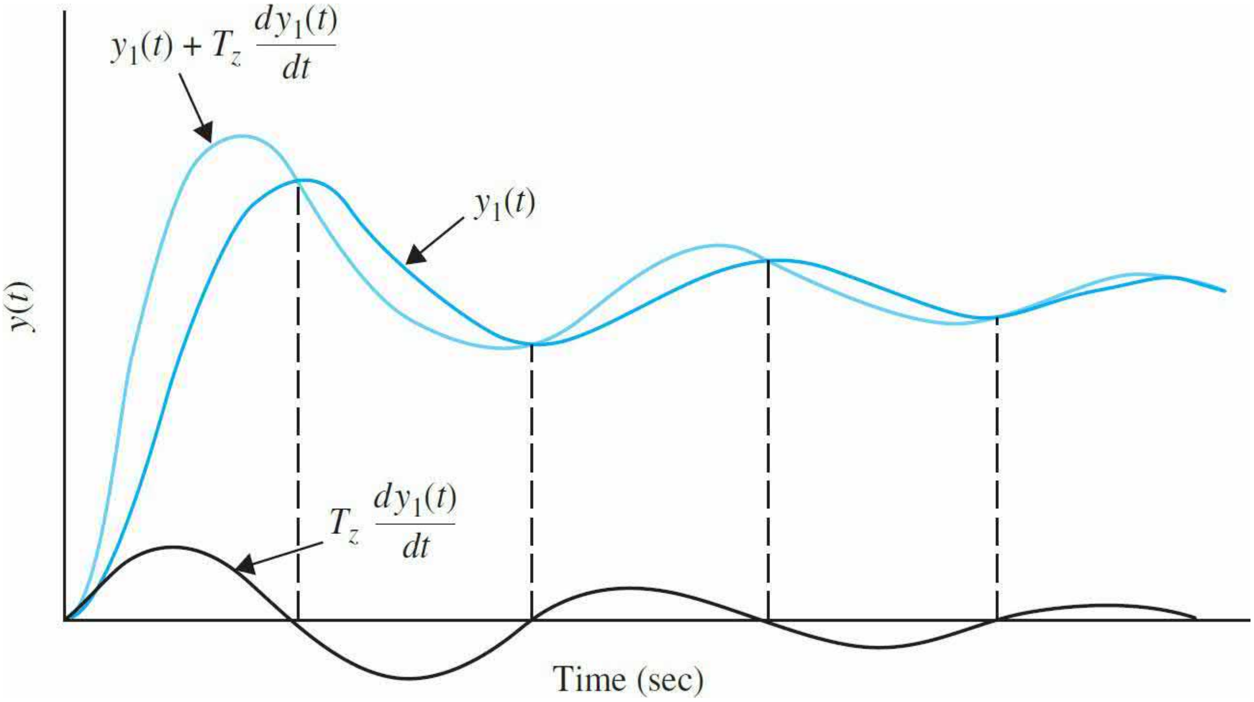


Figure 7-41 Unit-step responses showing the effect of adding a zero to the closed-loop transfer function.

7-7-4 Addition of a Zero to the Forward-Path Transfer Function: Unity-Feedback Systems

Let us consider that a zero at $-1/T_z$ is added to the forward-path transfer function of a third-order system, so

$$G(s) = \frac{6(1+T_z s)}{s(s+1)(s+2)} \tag{7-134}$$

The closed-loop transfer function is

$$M(s) = \frac{Y(s)}{R(s)} = \frac{6(1+T_z s)}{s^3 + 3s^2 + (2+6T_z)s + 6} \tag{7-135}$$

The difference between this case and that of adding a zero to the closed-loop transfer function is that, in the present case, not only the term $(1 + T_z s)$ appears in the numerator of $M(s)$, but the denominator of $M(s)$ also contains T_z. The term $(1 + T_z s)$ in the numerator of $M(s)$ increases the maximum overshoot, but T_z appears in the coefficient of the s term in the denominator, which has the effect of improving damping, or reducing the maximum overshoot. Figure 7-42 illustrates the unit-step responses when $T_z = 0, 0.2, 0.5, 2.0, 5.0$, and 10.0. Notice that, when $T_z = 0$, the closed-loop system is on the verge of becoming unstable. When $T_z = 0.2$ and 0.5, the maximum overshoots are reduced, mainly because of the improved damping. As T_z increases beyond 2.0, although the damping is still further improved, the $(1 + T_z s)$ term in the numerator becomes more dominant, so the maximum overshoot actually becomes greater as T_z is increased further.

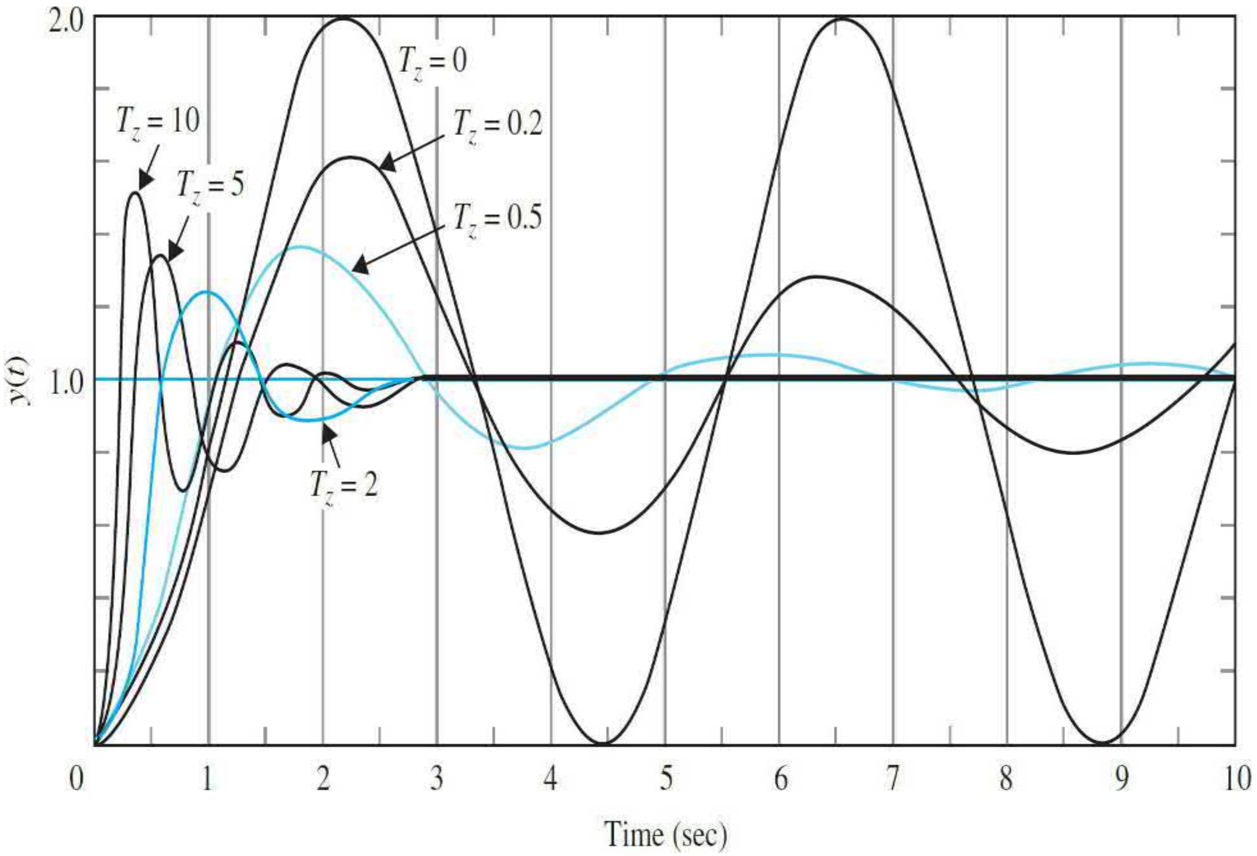
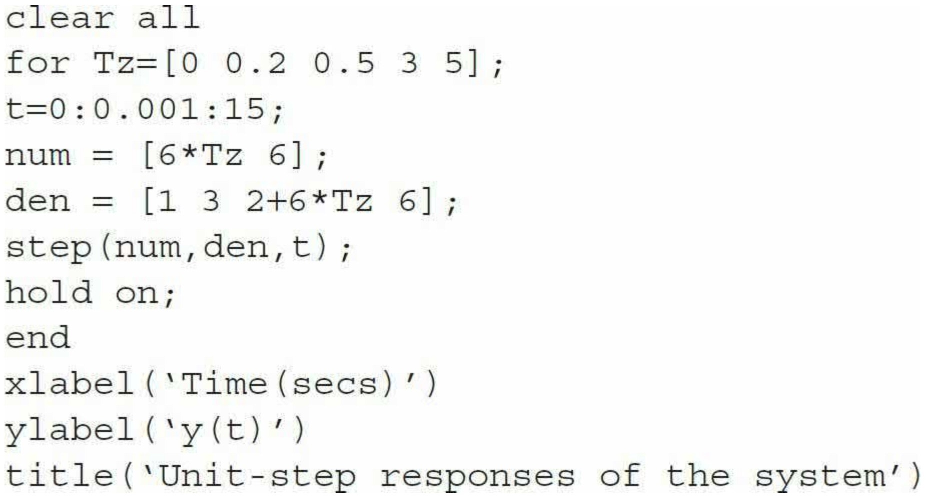


Figure 7-42 Unit-step responses of the system with the closed-loop transfer function in Eq. (7-135): $T_z = 0, 0.2, 0.5, 2.0, 5.0$, and 10.0.

An important finding from these discussions is that, although the characteristic equation roots are generally used to study the relative damping and relative stability of linear control systems, the zeros of the transfer function should not be overlooked in their effects on the transient performance of the system. See Example 7-7-1 in the following section for another treatment of this case.

Toolbox 7-7-3

The corresponding responses for Fig. 7-42 are obtained by the following sequence of MATLAB functions:



```
clear all
for Tz=[0 0.2 0.5 3 5];
t=0:0.001:15;
num = [6*Tz 6];
den = [1 3 2+6*Tz 6];
step(num,den,t);
hold on;
end
xlabel(‘Time(secs)’)
ylabel(‘y(t)’)
title(‘Unit-step responses of the system’)
```

7-7-5 Addition of Poles and Zeros: Introduction to Control of Time Response

In practice we can control the response of a system by adding poles and zeros or a simple amplifier with a constant gain *K* to its transfer function. So far in this chapter, we have discussed the effect of adding a simple gain in the time response—that is, proportional control. In this section, we look at controllers that include derivative or integral of the input signal in addition to the proportional operation.

EXAMPLE 7-7-1 Consider the second-order model

$$G_p(s) = \frac{2}{s(s+2)} = \frac{\omega_n^2}{s(s+2\zeta\omega_n)} \qquad (7\text{-}136)$$

where ω_n = 1.414 rad/s and ζ = 0.707. The forward-path transfer function has two poles at 0 and –2. Figure 7-43 shows the block diagram of the system. The series controller, which adds a zero to the forward path transfer function, is a proportional-derivative (PD) type with the transfer function

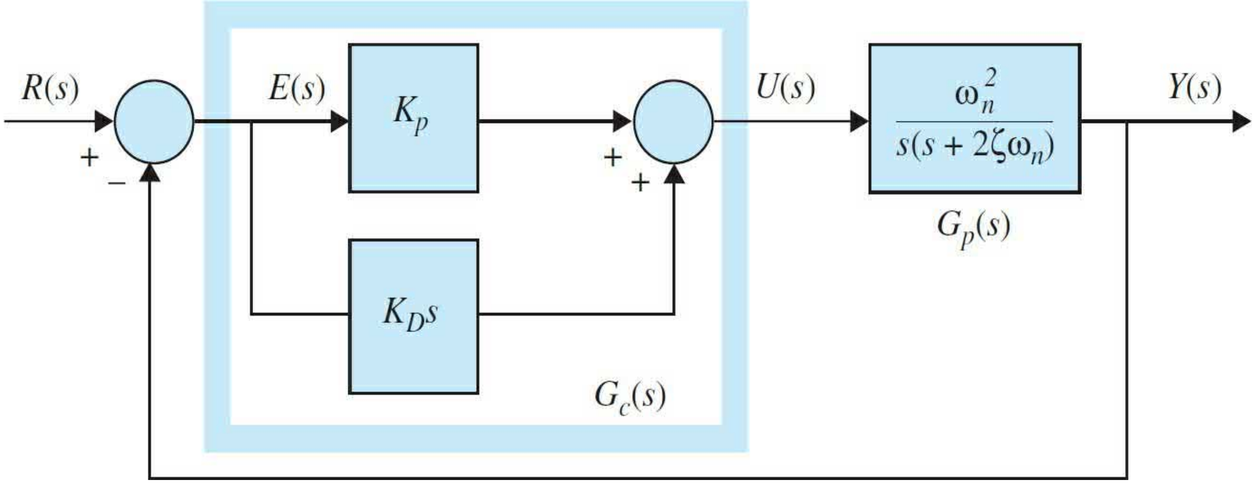


Figure 7-43 Control system with PD controller.

$$G_c(s) = K_P + K_D s \tag{7-137}$$

In this case, the forward-path transfer function of the compensated system is

$$G(s) = \frac{Y(s)}{E(s)} = G_c(s)G_p(s) = \frac{\omega_n^2(K_P + K_D s)}{s(s+2\zeta\omega_n)} \tag{7-138}$$

which shows that the PD control is equivalent to adding a simple zero at $s = -K_p/K_D$ to the forward-path transfer function. *Note that this controller does not affect the system type, and it only alters the transient response of the system.*

Rewriting the transfer function of the PD controller as

$$G_c(s) = (K_P + K_D s) = K_P(1 + T_z s) \tag{7-139}$$

where

$$T_z = K_D / K_P \tag{7-140}$$

The forward-path transfer function of the system becomes

$$G(s) = \frac{Y(s)}{E(s)} = \frac{2K_P(1+T_z s)}{s(s+2)} \tag{7-141}$$

The closed-loop transfer function is

$$\frac{Y(s)}{R(s)} = \frac{2K_P(1+T_z s)}{s^2 + (2+2K_P T_z)s + 2K_P} \tag{7-142}$$

We should quickly point out that Eq. (7-142) no longer represents a prototype second-order system, since the transient response is also affected by the zero of the transfer function at $s = –K_p/K_D$ —refer to Sec. 7-7-4 for more discussions.

Let us now examine how the controller gains K_P and K_D affect the response of the system. Obviously, because there are two controller gains that we can vary, this process is not unique. In Chap. 11, we discuss this subject in more depth. In this example, we provide a simple approach by examining the how K_P and K_D affect the poles and zero of the system. Let’s fix the value of the zero at an arbitrary location to the left of forward-path transfer function poles at 0 and −2. If T_z is too small, the system in Eq. (7-142) converges to a prototype second-order transfer function. That is

$$\frac{Y(s)}{R(s)} = \lim_{T_z \to 0} \frac{2K_P(1+T_z s)}{s^2 + (2+2K_P T_z)s + 2K_P} = \frac{2K_P}{s^2 + 2s + 2K_P} \tag{7-143}$$

This is simply saying a large negative zero, will have a minimal effect on the system transient response. In order to better examine the effect of the zero, let’s choose the zero $s = –1/T_z = –2.5$, implying $T_z = 0.4$. The poles may be obtained from the characteristic equation:

$$s^2+(2+2K_PT_z)s+2K_P=0 \tag{7-144}$$

or

$$s_{1,2}=-1-K_PT_z\pm\sqrt{(1+K_PT_z)^2-2K_P} \tag{7-145}$$

> Hence, addition of a zero to the forward path transfer function impacted the overshoot and rise time of the system. If the zero is more dominant than the poles, the overshoot increases, despite a decrease in the oscillations, while the rise time also decreases. If we select the zero farther to the left of the *s*-plane, its effect will become less dominant.

Table 7-10 shows selected pole values when K_P varies from 0 to 7. The results are also plotted in the *s*-plane in a graph that is better known as the root locus of the system—see Fig. 7-44. The root locus is essentially the graphical representation of the system zero and roots of Eq. (7-145) for all K_P values. As shown, as K_P varies the poles of the system move together and meet at $s = -1.38$ for $K_P = 0.9549$. Then the poles become complex and encircle the zero at −2.5. They meet again at $s = -3.64$ for $K_P = 6.5463$. Beyond this point, one pole moves toward the zero at −2.5 while the other one moves to the left. Finally, as $K_p \to \infty$, $s_1 \to \infty$, and $s_2 \to -2.5$.

TABLE 7-10 The Roots of the Closed-Loop System in Eq. (7-142) when $T_z = 0.4$ and K_P varies from 0 to 7

K_p	Zero	Poles	
0	−2.5	0	−2
0.9549	−2.5	−1.38	−1.38
1	−2.5	$-1.4 + j\,0.2$	$-1.4 - j\,0.2$
3	−2.5	$-2.2 + j\,1.08$	$-2.2 - j\,1.08$
4	−2.5	$-2.6 + j\,1.11$	$-2.6 - j\,1.11$
5	−2.5	$-3 + j$	$-3 - j$
6.5	−2.5	$-3.6 + j\,0.2$	$-3.6 - j\,0.2$
6.5463	−2.5	−3.64	−3.64
6.6	−2.5	−3.86	−3.42

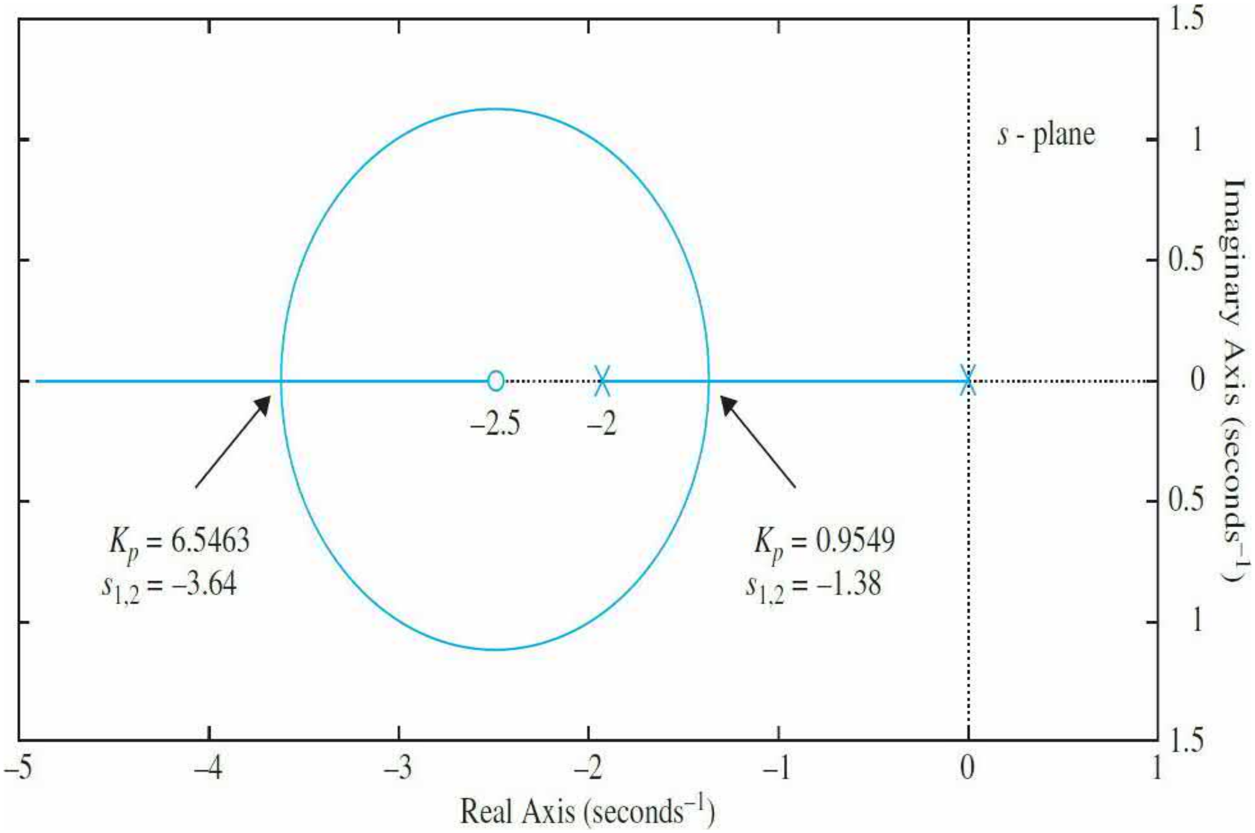
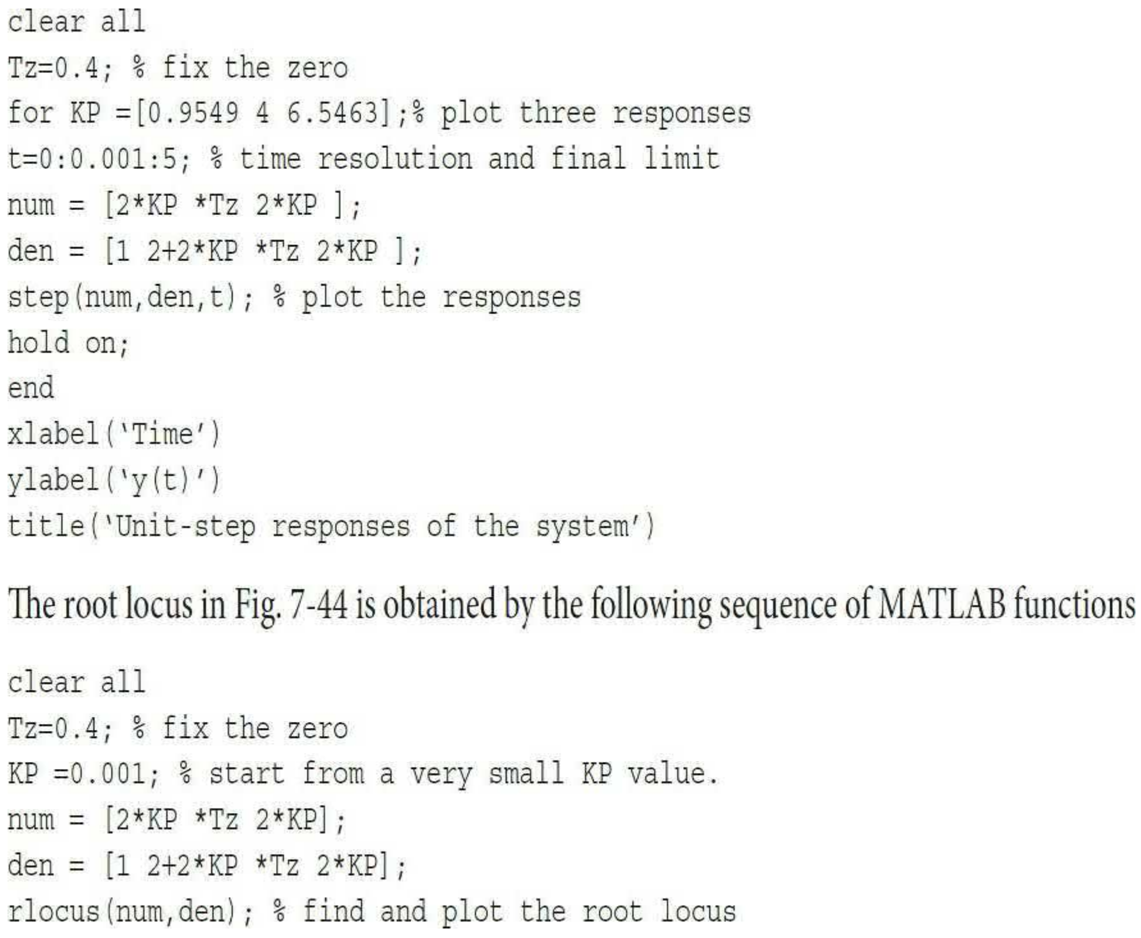


Figure 7-44 The root locus representing the zero and poles of Eq. (7-142) for K_p varying from 0 to ∞ and $T_z = 0.4$.

By looking at the root locus, and based on the previous discussions, at K_P = 0.9549, the two poles at −1.38 dominate the effect of the zero at −2.5, and we expect to observe a critically damped type response. At K_P = 6.5463, however, the zero at −2.5 dominates the two poles at −3.64. As a result, we expect to see more overshoot and a faster rise time—see Secs. 7-7-3 and 7-7-4. Between these values, the system can exhibit oscillatory response, depending on the effect of the zero at −2.5. Examining the unit-step response of the system in Eq. (7-142) for selected values of K_P = 0.9549, 1, and 6.5463 confirms our *s*-plane assessment. It is important to note that the zero's effect in this case appears significant enough to suppress the oscillatory nature of the system even when the poles are complex.

Toolbox 7-7-4

The corresponding responses for Fig. 7-45 are obtained by the following sequence of MATLAB functions:



```
clear all
Tz=0.4; % fix the zero
for KP =[0.9549 4 6.5463];% plot three responses
t=0:0.001:5; % time resolution and final limit
num = [2*KP *Tz 2*KP ];
den = [1 2+2*KP *Tz 2*KP ];
step(num,den,t); % plot the responses
hold on;
end
xlabel(‘Time’)
ylabel(‘y(t)’)
title(‘Unit-step responses of the system’)
```

The root locus in Fig. 7-44 is obtained by the following sequence of MATLAB functions

```
clear all
Tz=0.4; % fix the zero
KP =0.001; % start from a very small KP value.
num = [2*KP *Tz 2*KP];
den = [1 2+2*KP *Tz 2*KP];
rlocus(num,den); % find and plot the root locus
```

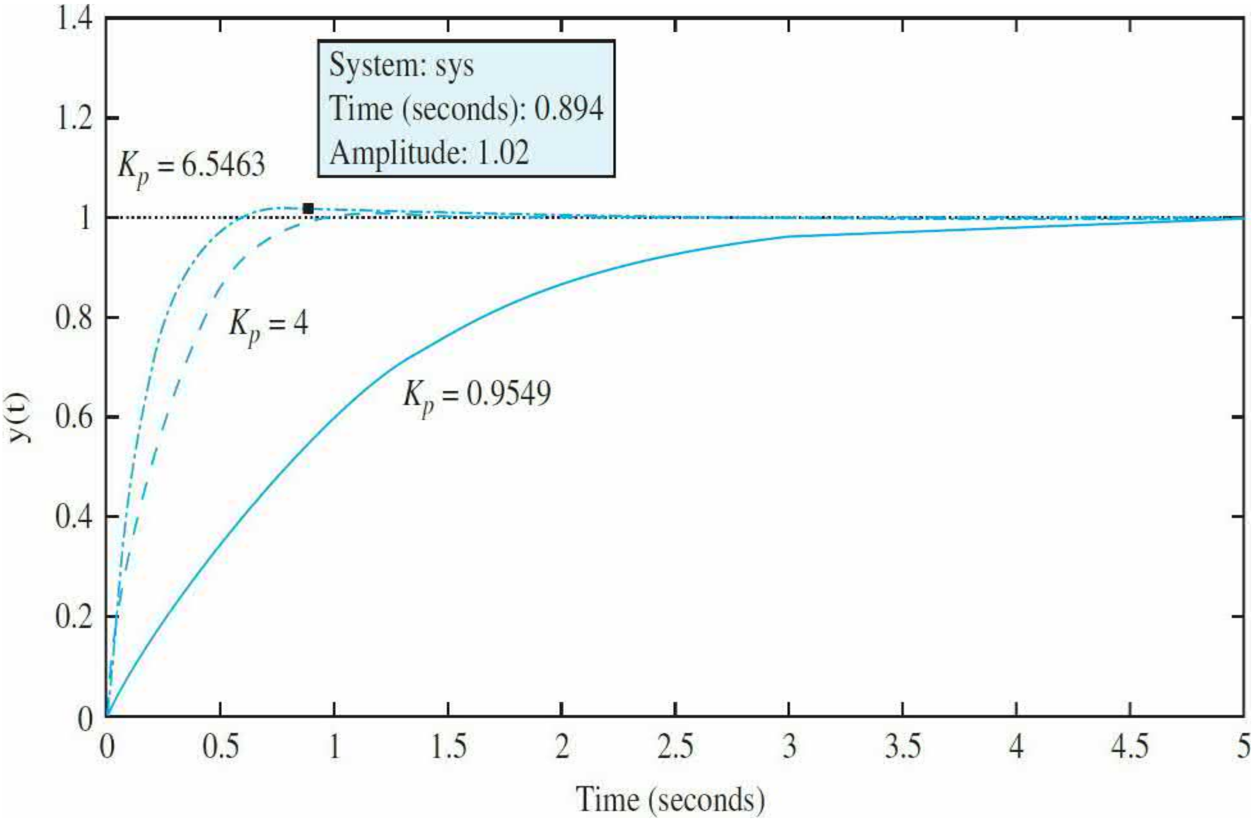


Figure 7-45 Unit-step response of Eq. (7-142) for T_z = 0.4 and three K_P values.

EXAMPLE 7-7-2 Consider the following second-order plant

$$G_p(s) = \frac{2}{(s+1)(s+2)} \tag{7-146}$$

Figure 7-46 illustrates the block diagram of the system with a series PI controller. Using the circuit elements given in Table 7-1 in Chap. 6, the transfer function of the PI controller is

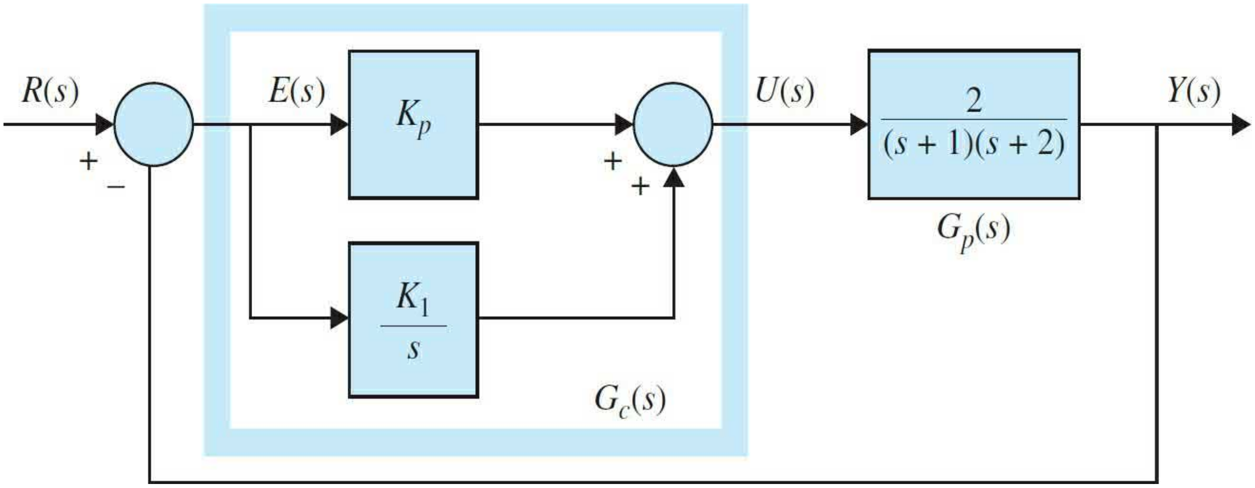


Figure 7-46 Control system with PI controller.

$$G_c(s)=K_P+\frac{K_I}{s} \tag{7-147}$$

which adds a zero and a pole to the forward path transfer function. The addition of a pole at s = 0, changes the system to a type 1 system, and as a result, eliminates the steady-state error to a step input. The forward-path transfer function of the compensated system is

$$G(s)=G_c(s)G_p(s)=\frac{2K_P(s+K_I/K_P)}{s(s+1)(s+2)}=\frac{2K_P(s+K_I/K_P)}{s^3+3s^2+2s} \tag{7-148}$$

Our **design criteria** in this case are zero steady error and a PO of 4.33 to a unit step input.

The closed-loop transfer function is

$$\frac{Y(s)}{R(s)}=\frac{2K_P(s+K_I/K_P)}{s^3+3s^2+2(1+K_P)s+2K_I} \tag{7-149}$$

where the characteristic equation of the closed-loop system is

$$s^3+3s^2+2(1+K_P)s+2K_I=0 \tag{7-150}$$

From Routh-Hurwitz stability test, the system is stable for $0 < K_I/K_p < 13.5$. This means that the zero of $G(s)$ at $s = -K_I/K_p$ cannot be placed too far to the left in the left-half s-plane, or the system will be unstable (see Fig. 7-47 for controller pole and zero locations). Hence, *when a type 0 system is converted to type 1 using a PI controller, the steady-state error due to a step input is always zero if the closed-loop system is stable.*

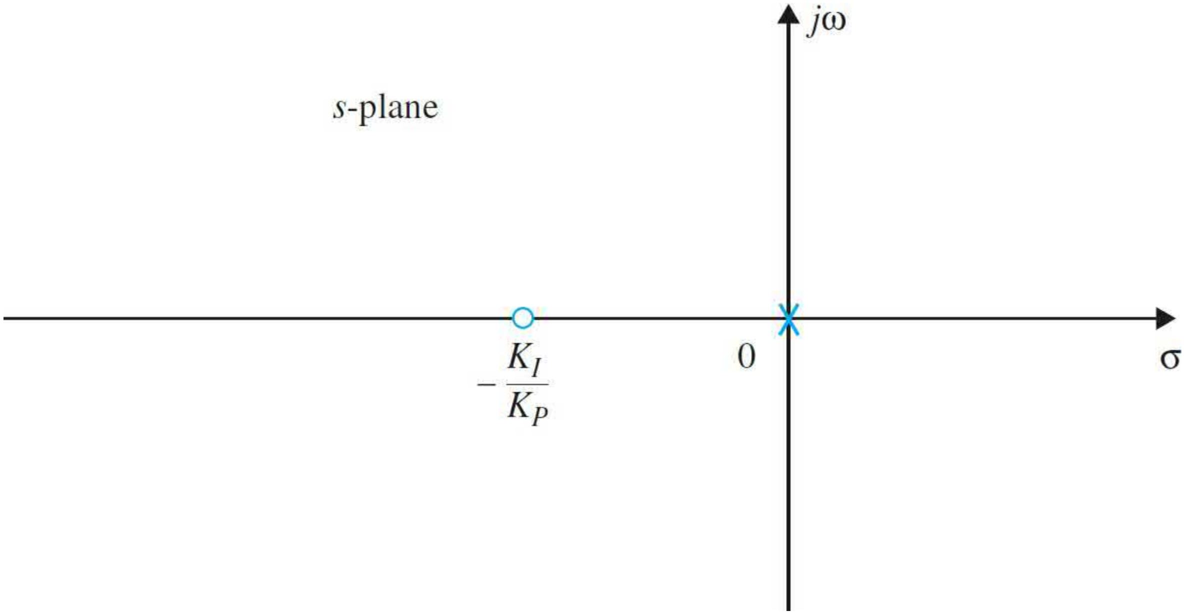


Figure 7-47 Pole-zero configuration of a PI controller.

A viable method of designing the PI control is to select the zero at $s = -K_I/K_p$ so that it is relatively close to the origin and away from the most significant poles of the process; the values of K_P and K_I should be relatively small.

The system in Fig. 7-46, with the forward-path transfer function in Eq. (7-148), will now have a zero steady-state error when the reference input is a step function. However, because the system is now third order, *it may be less stable* than the original second-order system or even it may become *unstable* if the parameters K_P and K_I

are not properly chosen. The problem is then to choose the proper combination of K_p and K_I so that the transient response is satisfactory.

Let us place the zero of the controller at $-K_I/K_p$ relatively close to the origin. For the present case, the most significant pole of $G_p(s)$ is at –1. Thus, K_I/K_p should be chosen so that the following condition is satisfied:

$$\frac{K_I}{K_P} < 1 \tag{7-151}$$

As a start point in designing of the controller, considering the condition in Eq. (7-151), Eq. (7-149) may loosely be approximated by

$$G(s) \cong \frac{2K_P}{s^2 + 3s + 2 + 2K_P} \tag{7-152}$$

where the term K_I/K_P in the numerator and K_I in the denominator are neglected.

As a design criterion, we require a desired percent maximum overshoot value of 4.3 for a unit-step input, which utilizing Eq. (7-40) results in a relative damping ratio of 0.707. From the denominator of Eq. (7-152) compared with a prototype second-order system, we get the natural frequency value of ω_n = 2.1213 rad/s and the required proportional gain of K_p = 1.25 — see Fig. 7-48 for the time response. Using K_p = 1.25, let us now examine the time response of the third-order system in Eq. (7-149). As shown in Fig. 7-48, if K_I is too small, 0.625 in this case, the system time response is slow and the desired zero steady-state error requirement is not met fast enough. Upon increasing K_I to 1.125, the desired response is met, as shown in Fig. 7-48. In this case, the controller zero still meets the condition in Eq. (7-151).

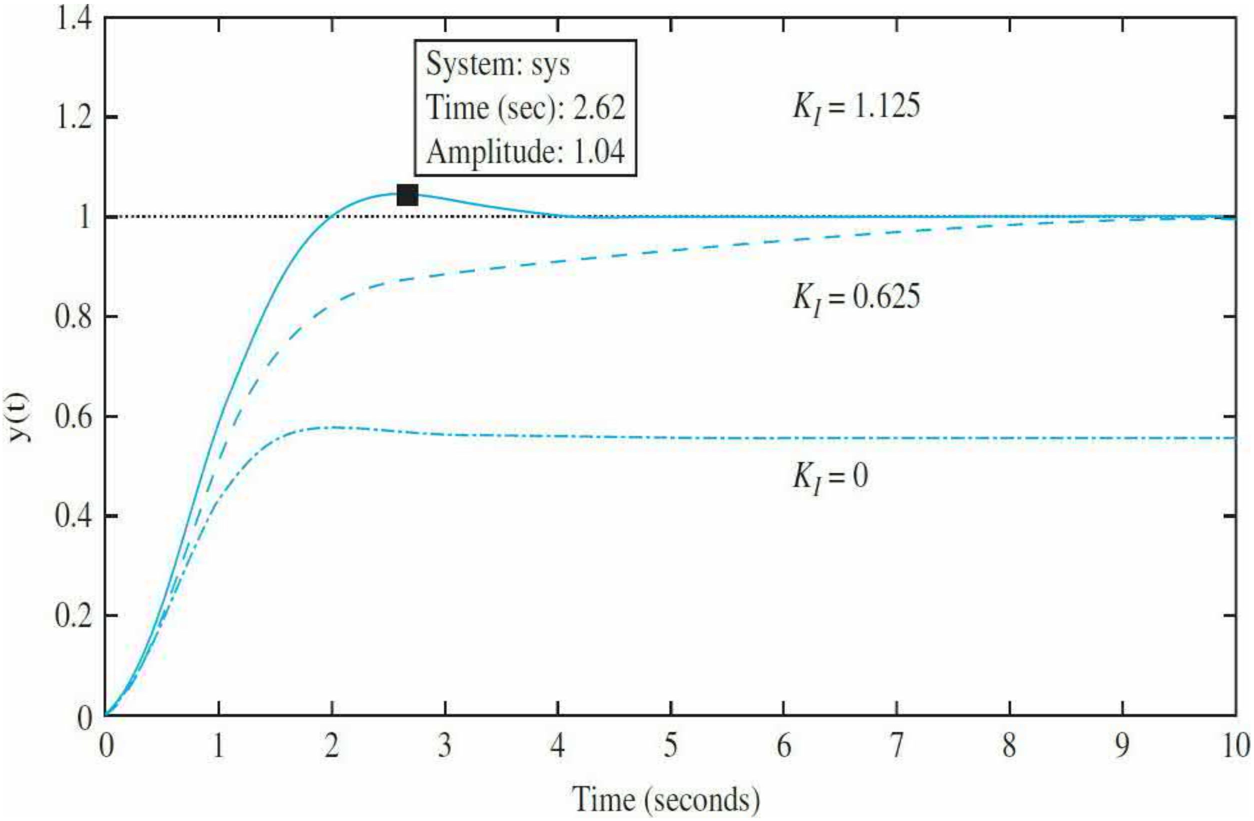
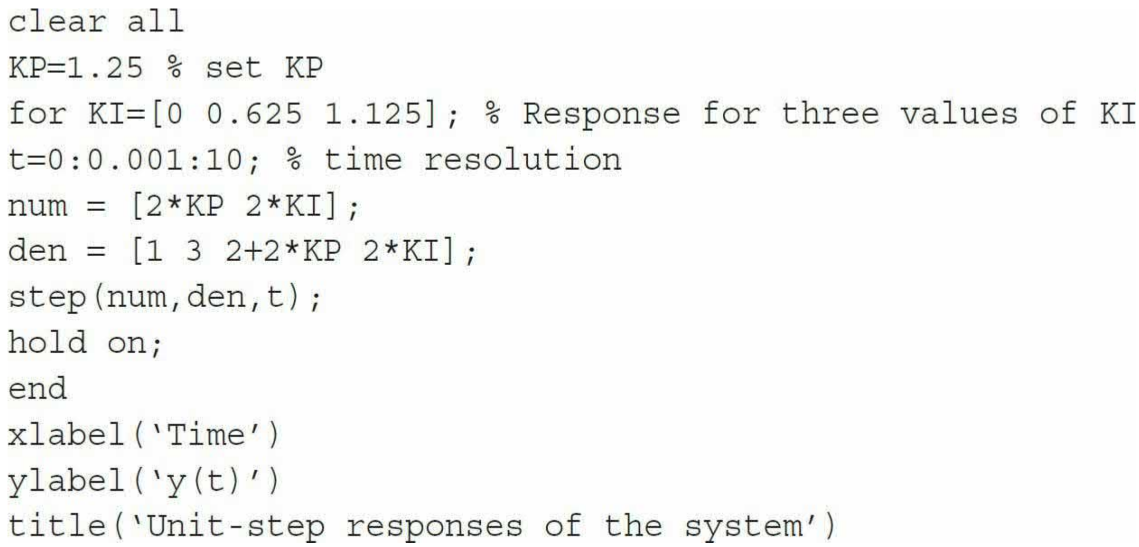


Figure 7-48 Unit-step response of Eq. (7-149) for three K_I values, when $K_P = 1.25$.

Hence, in this case addition of a controller pole at $s = 0$ eliminated the steady-state error, while the controller zero impacted the transient response to meet the desired PO requirement.

Toolbox 7-7-5

The corresponding responses for Fig. 7-48 are obtained by the following sequence of MATLAB functions:



```
clear all
KP=1.25 % set KP
for KI=[0 0.625 1.125]; % Response for three values of KI
t=0:0.001:10; % time resolution
num = [2*KP 2*KI];
den = [1 3 2+2*KP 2*KI];
step(num,den,t);
hold on;
end
xlabel('Time')
ylabel('y(t)')
title('Unit-step responses of the system')
```

7-8 DOMINANT POLES AND ZEROS OF TRANSFER FUNCTIONS

From the discussions given in the preceding sections, it becomes apparent that the location of the poles and zeros of a transfer function in the *s*-plane greatly affects the transient response of the system. For analysis and design purposes, it is important to sort out the poles that have a dominant effect on the transient response and call these the dominant poles.

Because most control systems in practice are of orders higher than two, it would be useful to establish guidelines on the approximation of high-order systems by lower-order ones insofar as the transient response is concerned. In design, we can use the dominant poles to control the dynamic performance of the system, whereas the insignificant poles are used for the purpose of ensuring that the controller transfer function can be realized by physical components.

For all practical purposes, we can divide the *s*-plane into regions in which the dominant and insignificant poles can lie, as shown in Fig. 7-40. We intentionally do not assign specific values to the coordinates, since these are all relative to a given system.

The poles that are *close* to the imaginary axis in the left-half *s*-plane give rise to transient responses that will decay relatively slowly, whereas the poles that are *far away* from the axis (relative to the dominant poles) correspond to fast-decaying time responses. The distance *D* between the dominant region and the least significant region shown in Fig. 7-49 will be subject to discussion. The question is: How large a pole is considered to be really large? It has been recognized in practice and in the literature that if the magnitude of the real part of a pole is at least **5 to 10 times** that of a dominant pole or a pair of complex dominant poles, then the pole may be regarded as insignificant insofar as the transient response is concerned. The zeros that are *close* to the imaginary axis in the left-half *s*-plane affect the transient responses more significantly, whereas the zeros that are *far away* from the axis (relative to the dominant poles) have a smaller effect on the time response.

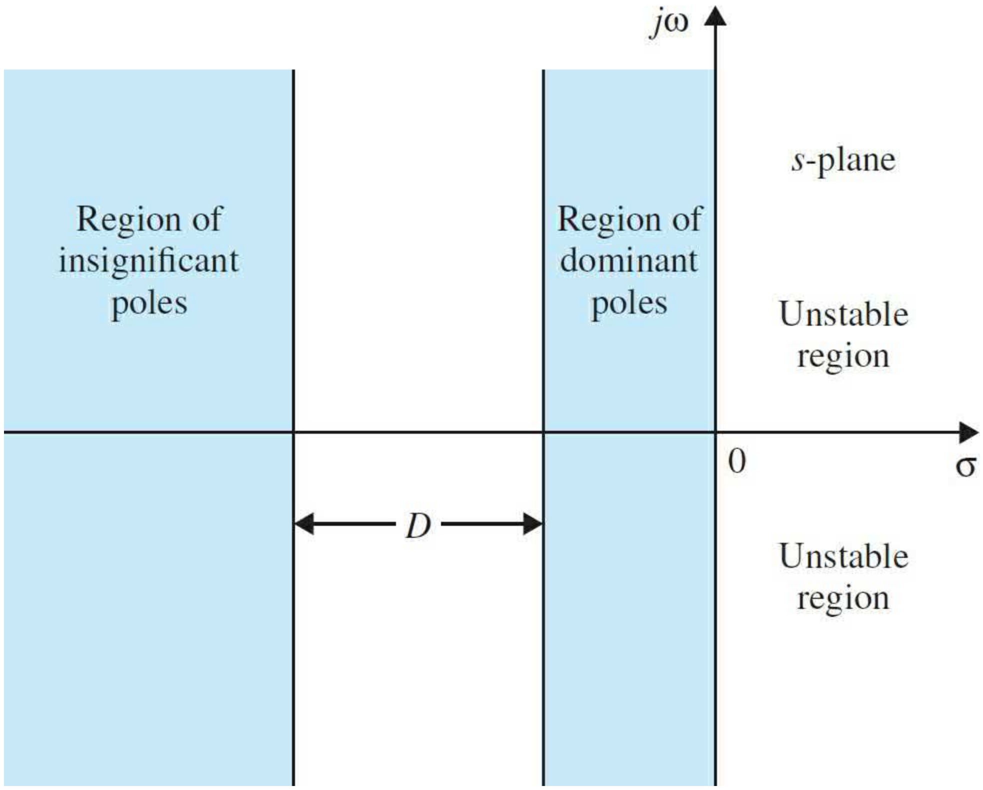


Figure 7-49 Regions of dominant and insignificant poles in the *s*-plane.

We must point out that the regions shown in Fig. 7-49 are selected merely for the definitions of dominant and insignificant poles. For **controller design** purposes, such as in pole-placement design, the dominant poles and the insignificant poles should most likely be located in the tinted regions in Fig. 7-50. Again, we do not show any absolute coordinates, except that the desired region of the dominant poles is centered around the line that corresponds to $\zeta = 0.707$. It should also be noted that, while designing, we cannot place the insignificant poles arbitrarily far to the left in the *s*-plane or these may require unrealistic system parameter values when the pencil-and-paper design is implemented by physical components.

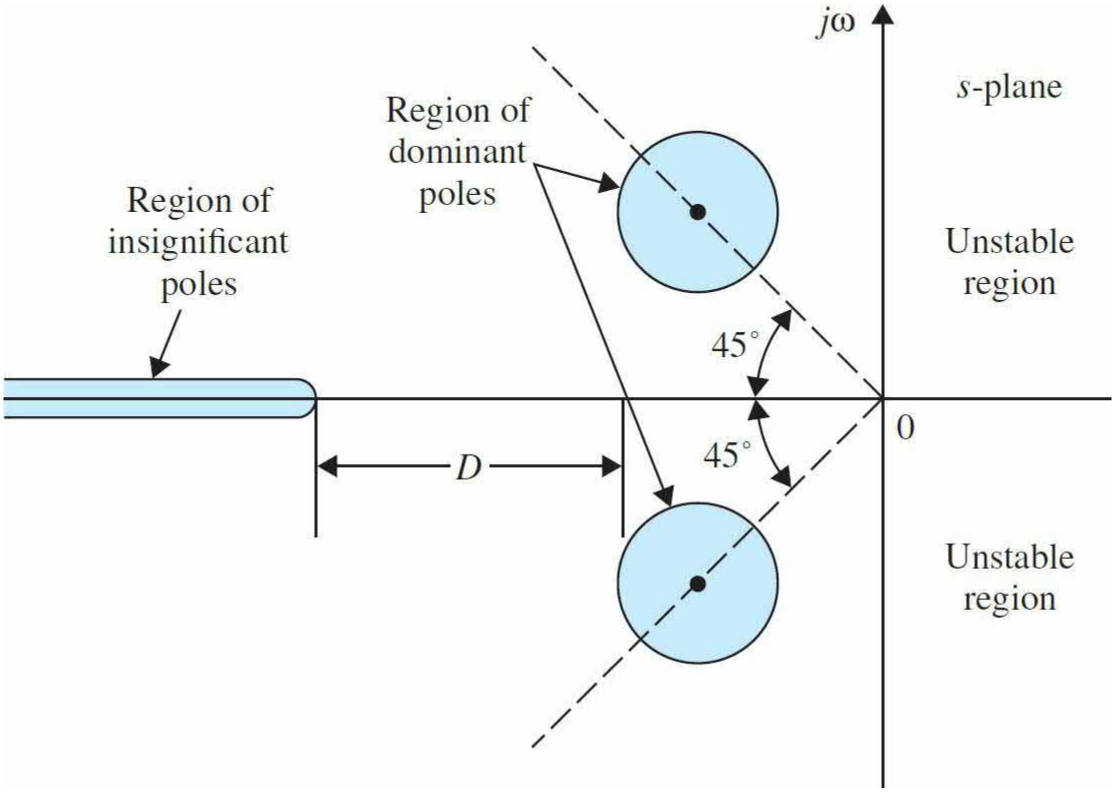


Figure 7-50 Regions of dominant and insignificant poles in the *s*-plane for design purposes.

7-8-1 Summary of Effects of Poles and Zeros

Based on previous observations, we can summarize the following:

1. Complex-conjugate poles of the closed-loop transfer function lead to a step response that is underdamped. If all system poles are real, the step response is overdamped. However, zeros of the closed-loop transfer function may cause overshoot even if the system is overdamped.
2. The response of a system is dominated by those poles closest to the origin in the *s*-plane. Transients due to those poles, which are farther to the left, decay faster.

3. The farther to the left in the *s*-plane the system's dominant poles are, the faster the system will respond and the greater its bandwidth will be.
4. The farther to the left in the *s*-plane the system's dominant poles are, the more expensive it will be and the larger its internal signals will be. While this can be justified analytically, it is obvious that striking a nail harder with a hammer drives the nail in faster but requires more energy per strike. Similarly, a sports car can accelerate faster, but it uses more fuel than an average car.
5. When a pole and zero of a system transfer function nearly cancel each other, the portion of the system response associated with the pole will have a small magnitude.

7-8-2 The Relative Damping Ratio

When a system is higher than the second order, we can no longer strictly use the damping ratio ζ and the natural undamped frequency ω_n, which are defined for the prototype second-order systems. However, if the system dynamics can be accurately represented by a pair of complex-conjugate dominant poles, then we can still use ζ and ω_n to indicate the dynamics of the transient response, and the damping ratio in this case is referred to as the relative damping ratio of the system. For example, consider the closed-loop transfer function

$$M(s)=\frac{Y(s)}{R(s)}=\frac{20}{(s+10)(s^2+2s+2)} \tag{7-153}$$

The pole at $s = -10$ is 10 times the real part of the complex conjugate poles, which are at $-1 \pm j1$. We can refer to the relative damping ratio of the system as 0.707.

7-8-3 The Proper Way of Neglecting the Insignificant Poles with Consideration of the Steady-State Response

Thus far, we have provided guidelines for neglecting insignificant poles of a transfer function from the standpoint of the transient response. However, going through with the mechanics, the steady-state performance must also be considered. Let us consider the transfer function in Eq. (7-153); the pole at −10 can be neglected from the transient standpoint. To do this, we should first express Eq. (7-153) as

$$M(s) = \frac{20}{10(s/10+1)(s^2+2s+2)} \tag{7-154}$$

Then we reason that |s/10| ≪ 1 when the absolute value of *s* is much smaller than 10 because of the dominant nature of the complex poles. The term *s*/10 can be neglected when compared with 1. Then, Eq. (7-154) is approximated by

$$M(s) \cong \frac{20}{10(s^2+2s+2)} \tag{7-155}$$

This way, the steady-state performance of the third-order system will not be affected by the approximation. In other words, the third-order system described by Eq. (7-153) and the second-order system approximated by Eq. (7-155) all have a final value of unity when a unit-step input is applied. On the other hand, if we simply throw away the term (*s* + 10) in Eq. (7-153), the approximating second-order system will have a steady-state value of 5 when a unit-step input is applied, which is incorrect.

7-9 CASE STUDY: TIME-DOMAIN ANALYSIS OF A POSITION-CONTROL SYSTEM

Due to the requirements of improved response and reliability, the surfaces of modern aircraft are controlled by electric actuators with electronic controls. Consider the system in Fig. 7-51. The purpose of the system considered here is to control the positions of the fins of an airplane. The so-called fly-by-wire control system implies that the attitude of aircraft is no longer controlled by mechanical linkages. Figure 7-51 illustrates the controlled surfaces and the block diagram of one axis of such a position-control system. Figure 7-52 shows the analytical block diagram of the system using the dc-motor model given in Fig. 7-51. The system is simplified to the extent that saturation of the amplifier gain and motor torque, gear backlash, and shaft compliances have all been neglected. (When you get into the real world, some of these nonlinear effects should be incorporated into the mathematical model to come up with a better controller design that works in reality.)

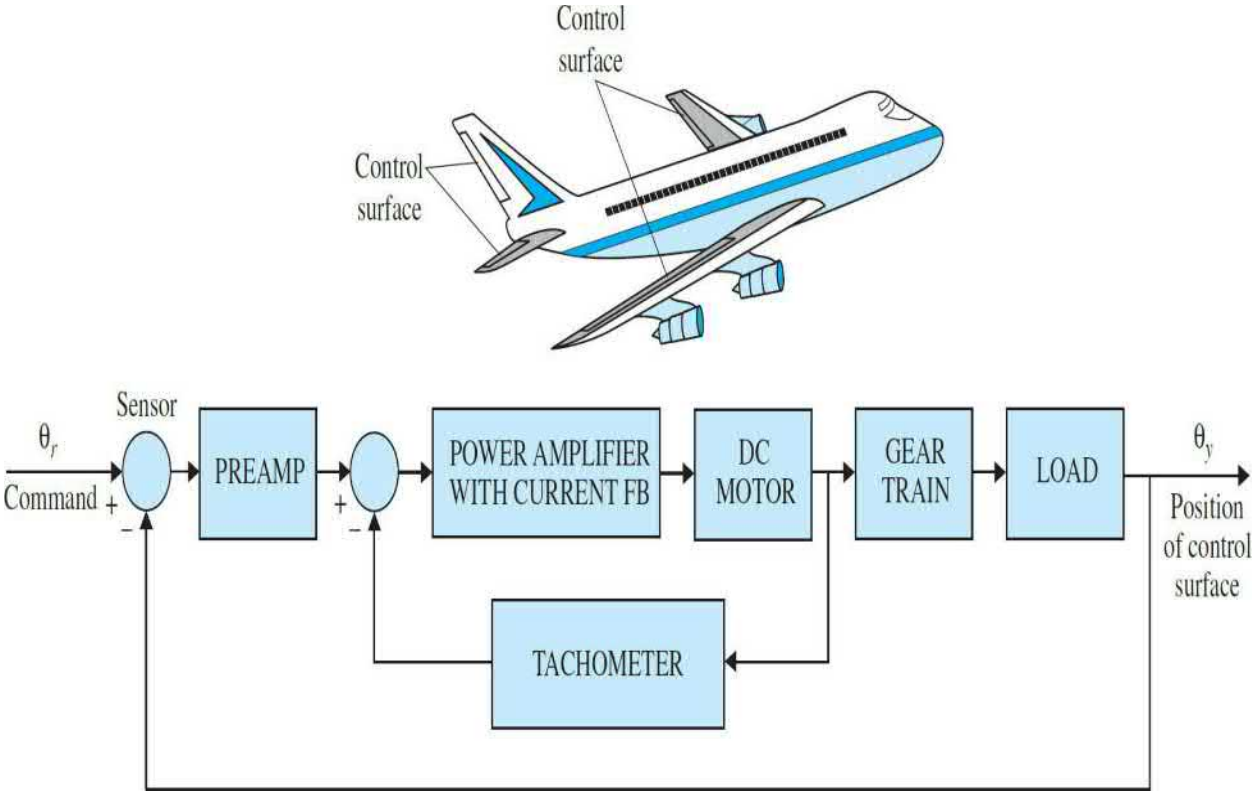


Figure 7-51 Block diagram of an attitude-control system of an aircraft.

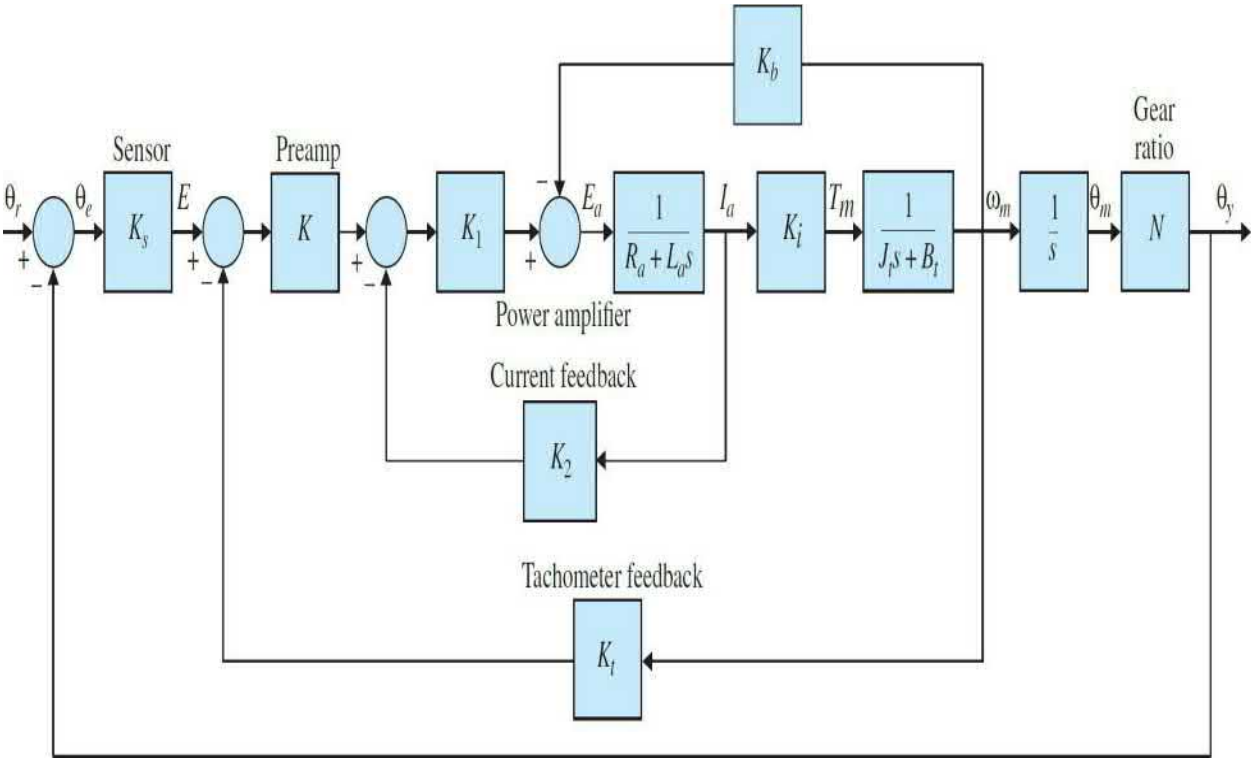
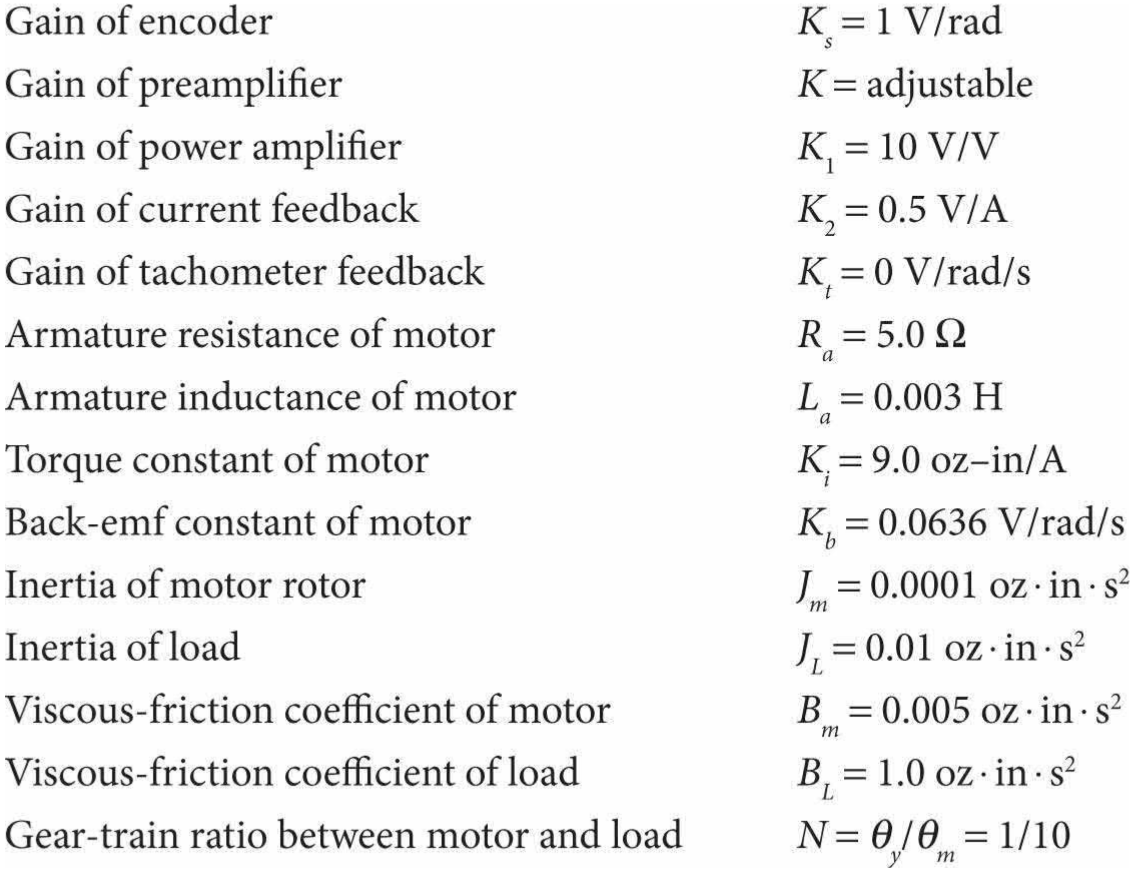


Figure 7-52 Transfer-function block diagram of the system shown in Fig. 7-51.

The objective of the system is to have the output of the system, $\theta_y(t)$, follow the input, $\theta_r(t)$. The following system parameters are given initially:



Gain of encoder	$K_s = 1$ V/rad
Gain of preamplifier	$K =$ adjustable
Gain of power amplifier	$K_1 = 10$ V/V
Gain of current feedback	$K_2 = 0.5$ V/A
Gain of tachometer feedback	$K_t = 0$ V/rad/s
Armature resistance of motor	$R_a = 5.0\ \Omega$
Armature inductance of motor	$L_a = 0.003$ H
Torque constant of motor	$K_i = 9.0$ oz–in/A
Back-emf constant of motor	$K_b = 0.0636$ V/rad/s
Inertia of motor rotor	$J_m = 0.0001\ \text{oz}\cdot\text{in}\cdot\text{s}^2$
Inertia of load	$J_L = 0.01\ \text{oz}\cdot\text{in}\cdot\text{s}^2$
Viscous-friction coefficient of motor	$B_m = 0.005\ \text{oz}\cdot\text{in}\cdot\text{s}^2$
Viscous-friction coefficient of load	$B_L = 1.0\ \text{oz}\cdot\text{in}\cdot\text{s}^2$
Gear-train ratio between motor and load	$N = \theta_y/\theta_m = 1/10$

Because the motor shaft is coupled to the load through a gear train with a gear ratio of N, $\theta_y = N\theta_m$, the total inertia and viscous-friction coefficient seen by the motor are

$$J_t = J_m + N^2 J_L = 0.0001 + 0.01/100 = 0.0002\ \text{oz}\cdot\text{in}\cdot\text{s}^2$$

$$B_t = B_m + N^2 B_L = 0.005 + 1/100 = 0.015\ \text{oz}\cdot\text{in}\cdot\text{s}$$

respectively. The forward-path transfer function of the unity-feedback system is written from Fig. 7-52 by applying the SFG gain formula:

$$G(s) = \frac{\Theta_y(s)}{\Theta_e(s)}$$

$$= \frac{K_s K_1 K_i K N}{s[L_a J_t s^2 + (R_a J_t + L_a B_t + K_1 K_2 J_t)s + R_a B_t + K_1 K_2 B_t + K_i K_b + K K_1 K_t K_i]} \qquad (5\text{-}157)$$

The system is of the third order, since the highest-order term in $G(s)$ is s^3.

The electrical time constant of the amplifier-motor system is

$$\tau_a = \frac{L_a}{R_a + K_1 K_2} = \frac{0.003}{5+5} = 0.0003\,\text{s} \tag{7-158}$$

The mechanical time constant of the motor-load system is

$$\tau_t = \frac{J_t}{B_t} = \frac{0.0002}{0.015} = 0.01333\ \text{s} \tag{7-159}$$

Because the electrical time constant is much smaller than the mechanical time constant, on account of the low inductance of the motor, we can perform an initial approximation by neglecting the armature inductance L_a. The result is a second-order approximation of the third-order system. Later we will show that this is not the best way of approximating a high-order system by a low-order one. The forward-path transfer function is now

$$\begin{aligned} G(s) &= \frac{K_s K_1 K_i K N}{s[(R_a J_t + K_1 K_2 J_t)s + R_a B_t + K_1 K_2 B_t + K_i K_b + K K_1 K_i K_t]} \\ &= \frac{\dfrac{K_s K_1 K_i K N}{R_a J_t + K_1 K_2 J_t}}{s\left(s + \dfrac{R_a B_t + K_1 K_2 B_t + K_i K_b + K K_1 K_i K_t}{R_a J_t + K_1 K_2 J_t}\right)} \end{aligned} \tag{7-160}$$

Substituting the system parameters in the last equation, we get

$$G(s) = \frac{4500K}{s(s+361.2)} \tag{7-161}$$

The closed-loop transfer function of the unity-feedback control system is

$$\frac{\Theta_y(s)}{\Theta_r(s)} = \frac{4500K}{s^2 + 361.2s + 4500K} \tag{7-162a}$$

Comparing Eq. (7-162) with the prototype second-order transfer function of Eq. (7-18), we have

$$\begin{cases} \omega_n = \sqrt{\dfrac{K_s K_1 K_i K N}{R_a J_t + K_1 K_2 J_t}} = \sqrt{4500K} \text{ rad/s} \\ \zeta = \dfrac{R_a B_t + K_1 K_2 B_t + K_i K_b + K K_1 K_i K_t}{2\sqrt{K_s K_1 K_i K N (R_a J_t + K_1 K_2 J_t)}} = \dfrac{2.692}{\sqrt{K}} \end{cases} \tag{7-162b}$$

Thus, we see that the natural frequency ω_n is proportional to the square root of the amplifier gain K, whereas the damping ratio ζ is inversely proportional to $\sqrt{K}$.

7-9-1 Unit-Step Transient Response

For the characteristic equation of Eq. (7-162), the roots are

$$s_1 = -180.6 + \sqrt{32616 - 4500K} \tag{7-163}$$

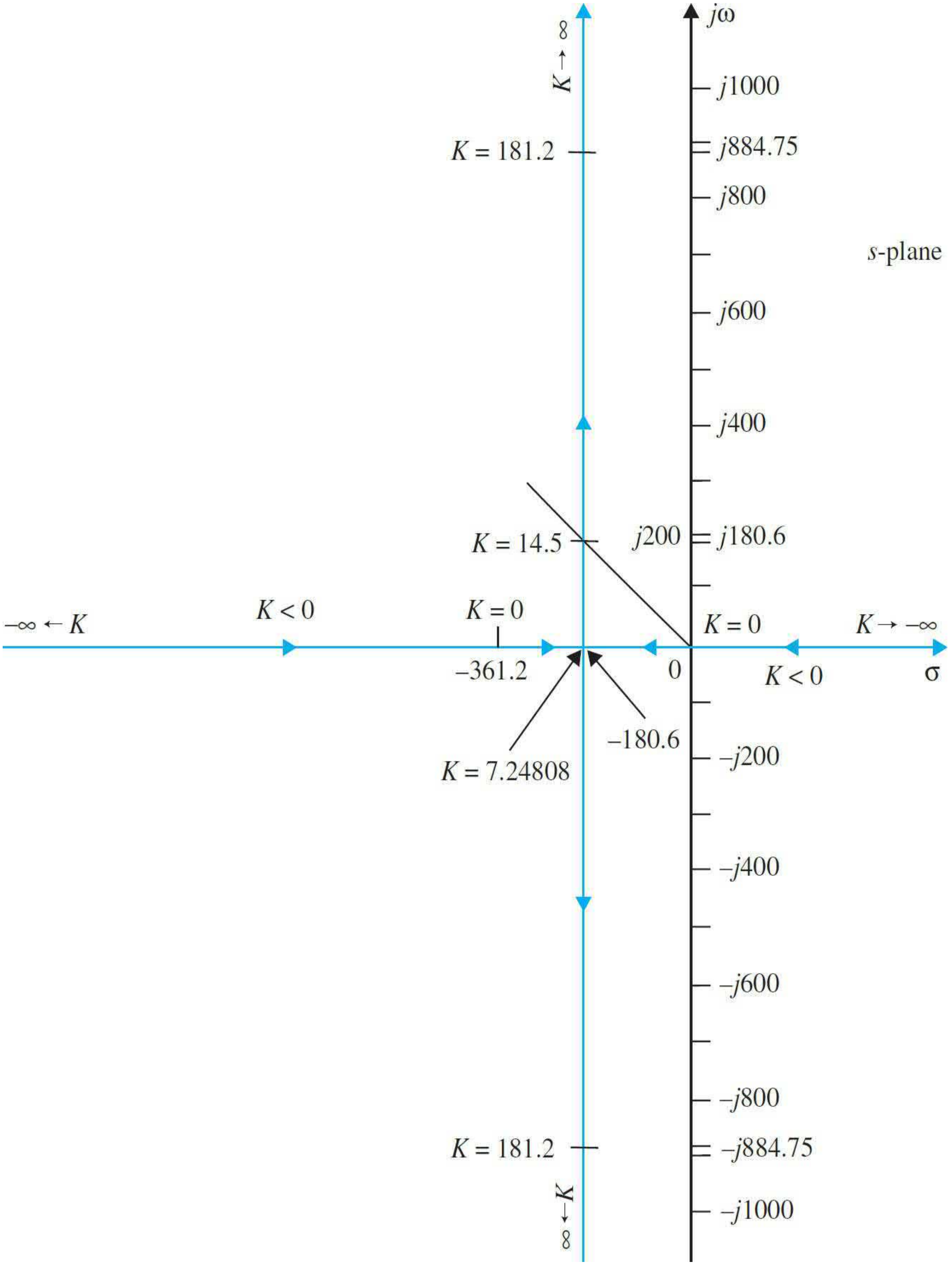
$$s_2 = -180.6 - \sqrt{32616 - 4500K} \tag{7-164}$$

For K = 7.24808, 14.5, and 181.2, the roots of the characteristic equation are tabulated as follows:

$$\begin{array}{lll} K = 7.24808: & s_1 = s_2 = -180.6 & \\ K = 14.5: & s_1 = -180.6 + j180.6 & s_2 = -180.6 - j180.6 \\ K = 181.2: & s_1 = -180.6 + j884.7 & s_2 = -180.6 + j884.7 \end{array}$$

These roots are marked as shown in Fig. 7-30. The trajectories of the two characteristic equation roots when K varies continuously from –∞ to ∞ are also shown in Fig. 7-30. These root trajectories are called the **root loci** of Eq. (7-135) and are used extensively for the analysis and design of linear control systems.

From Eqs. (7-163) and (7-164), we see that the two roots are real and negative for values of K between 0 and 7.24808. This means that the system is overdamped, and the step response will have no overshoot for this range of *K*. For values of *K* greater than 7.24808, the natural undamped frequency will increase with $\sqrt{K}$. When *K* is negative, one of the roots is positive, which corresponds to a time response that increases monotonically with time, and the system is unstable. The dynamic characteristics of the transient step response as determined from the root loci of Fig. 7-53 are summarized as follows:



jω
K → ∞
j1000
K = 181.2
j884.75
j800
s-plane
j600
j400
K = 14.5
j200
j180.6
−∞ ← K
K < 0
K = 0
K = 0
K → −∞
−361.2
0
K < 0
σ
−180.6
K = 7.24808
−j200
−j400
−j600
−j800
K = 181.2
−j884.75
K → ∞
−j1000

Figure 7-53 Root loci of the characteristic equation in Eq. (7-162) as *K* varies.

Amplifier Gain Dynamics	Characteristic Equation Roots	System
$0 < K < 7.24808$	Two negative distinct real roots	Overdamped ($\zeta > 1$)
$K = 7.24808$	Two negative equal real roots	Critically damped ($\zeta = 1$)
$7.24808 < K < \infty$	Two complex-conjugate roots with negative real parts	Underdamped ($\zeta < 1$)
$-\infty < K < 0$	Two distinct real roots, one positive and one negative	Unstable system ($\zeta < 0$)

Using a test unit-step input, we can characterize the time-domain performance of the system in terms of the maximum overshoot, rise time, delay time, and settling time. Let the reference input be a unit-step function $\theta_r(t) = u_s(t)$ rad; then $\Theta_r(s) = 1/s$. The output of the system with zero initial conditions, for the three values of *K* indicated, is

***K* = 7.248 ($\zeta \cong 1.0$):**

$$\theta_y(t) = (1 - 151e^{-180t} + 150e^{-181.2t})u_s(t) \tag{7-165}$$

***K* = 14.5 ($\zeta = 0.707$):**

$$\theta_y(t) = (1 - e^{-180.6t}\cos 180.6t - 0.9997e^{-180.6t}\sin 180.6t)u_s(t) \tag{7-166}$$

***K* = 181.17($\zeta = 0.2$):**

$$\theta_y(t) = (1 - e^{-180.6t}\cos 884.7t - 0.2041e^{-180.6t}\sin 884.7t)u_s(t) \tag{7-167}$$

The three responses are plotted as shown in Fig. 7-54. Table 7-11 gives the comparison of the characteristics of the three unit-step responses for the three values of *K* used. When *K* = 181.17, ζ = 0.2, the system is lightly damped, and the maximum overshoot is 52.7 percent, which is excessive. When the value of *K* is set at 7.248, ζ is very close to 1.0, and the system is almost critically damped. The unit-step response does not have any overshoot or oscillation. When *K* is set at 14.5, the damping ratio is 0.707, and the overshoot is 4.3 percent.

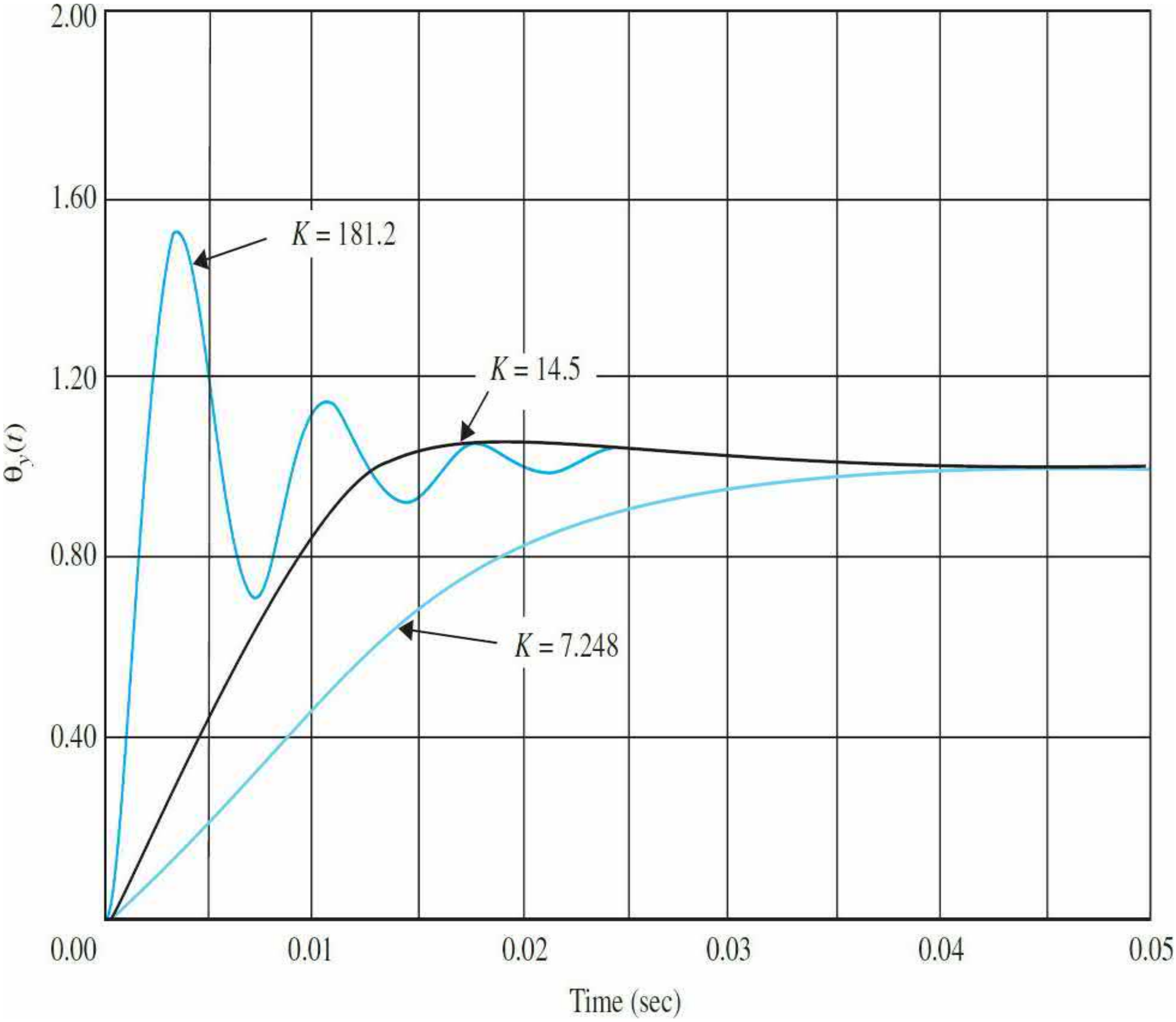
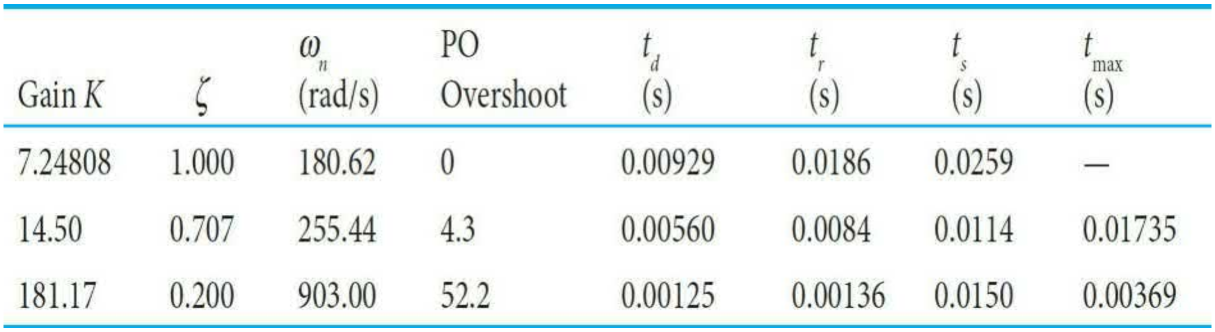


Figure 7-54 Unit-step responses of the attitude-control system in Fig. 7-52; $L_a = 0$.

TABLE 7-11 Comparison of the Performance of the Second-Order Position-Control System with the Gain *K* Values



Gain K	ζ	ω_n (rad/s)	PO Overshoot	t_d (s)	t_r (s)	t_s (s)	$t_{\max}$ (s)
7.24808	1.000	180.62	0	0.00929	0.0186	0.0259	—
14.50	0.707	255.44	4.3	0.00560	0.0084	0.0114	0.01735
181.17	0.200	903.00	52.2	0.00125	0.00136	0.0150	0.00369

Toolbox 7-9-1

Figure 7-54 responses may be obtained by the following sequence of MATLAB functions:

```
% Unit-Step Transient Response
for k=[7.248,14.5,181.2]
num = [4500*k];
den = [1 361.2 4500*k];
step(num,den)
hold on;
end
xlabel('Time(secs)')
ylabel('Amplitude')
title('Closed-Loop Step')
```

7-9-2 The Steady-State Response

Because the forward-path transfer function in Eq. (7-161) has a simple pole at s = 0, the system is of type 1. This means that the steady-state error of the system is zero for all positive values of K when the input is a step function. Substituting Eq. (7-161) into Eq. (7-115), the step-error constant is

$$K_p = \lim_{s \to 0} \frac{4500K}{s(s+361.2)} = \infty \tag{7-168}$$

Thus, the steady-state error of the system due to a step input, as given by Eq. (7-116), is zero. The unit-step responses in Fig. 7-54 verify this result.

The zero-steady-state condition is achieved because only viscous friction is considered in the simplified system model. In the practical case, Coulomb friction is almost always present, so the steady-state positioning accuracy of the system can never be perfect.

7-9-3 Time Response of a Third-Order System—Electrical Time Constant Not Neglected

In the preceding section, we have shown that the prototype second-order system, obtained by neglecting the armature inductance, is always stable for all positive values of *K*. It is not difficult to prove that, in general, all second-order systems with positive coefficients in the characteristic equations are stable.

Let us investigate the performance of the position-control system with the armature inductance L_a = 0.003 H. The forward-path transfer function of Eq. (7-160) becomes

$$G(s)=\frac{1.5\times10^{7}K}{s(s^{2}+3408.3s+1{,}204{,}000)}$$

$$=\frac{1.5\times10^{7}K}{s(s+400.26)(s+3008)} \qquad (7\text{-}169)$$

The closed-loop transfer function is

$$\frac{\Theta_y(s)}{\Theta_r(s)}=\frac{1.5\times10^{7}K}{s^{3}+3408.3s^{2}+1{,}204{,}000s+1.5\times10^{7}K} \qquad (7\text{-}170)$$

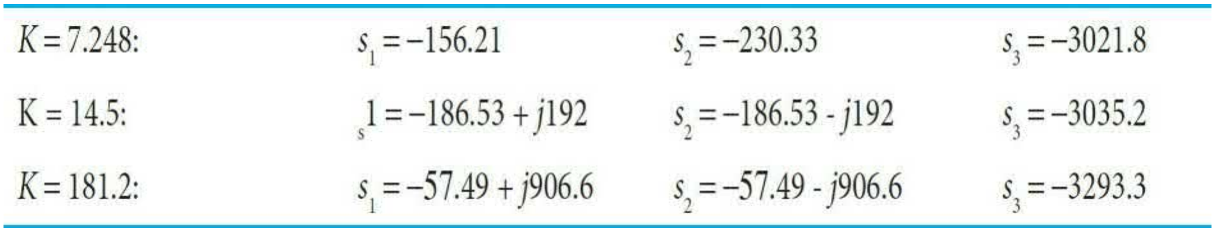
The system is now of the third order, and the characteristic equation is

$$s^{3}+3408.3s^{2}+1{,}204{,}000s+1.5\times10^{7}K=0 \qquad (7\text{-}171)$$

By using the Routh-Hurwitz criterion to Eq. (7-171), we can see that at *K* = 273.57, the third-order system becomes marginally stable with two poles at $s_{1,2}$ = ±1097.3. This is a clear difference with the second-order system approximation in Eq. (7-162), which is stable for all positive *K* values. As a result, as discussed in detail in the next section, we do expect for some values of K the negligible electrical time constant approximation not to be valid.

7-9-4 Unit-Step Transient Response

The roots of the characteristic equation are tabulated for the three values of K used earlier for the second-order system:



$K = 7.248$:	$s_1 = -156.21$	$s_2 = -230.33$	$s_3 = -3021.8$
K = 14.5:	${}_s1 = -186.53 + j192$	$s_2 = -186.53 - j192$	$s_3 = -3035.2$
$K = 181.2$:	$s_1 = -57.49 + j906.6$	$s_2 = -57.49 - j906.6$	$s_3 = -3293.3$

Comparing these results with those of the approximating second-order system, we see that, when K = 7.428, the second-order system is critically damped, whereas the third-order system has three distinct real roots, and the system is slightly overdamped. The root at s_3 = -3021.8 corresponds to a time constant of $\tau = 1/s^3$ = 0.33 ms, which is more than 13 times faster than the next fastest time constant because of the pole at –230.33. Thus, the transient response due to the pole at –3021.8 decays rapidly, and the pole can be neglected from the transient standpoint. The output transient response is dominated by the two roots at –156.21 and –230.33. This analysis is verified by writing the transformed output response as

$$\Theta_y(s) = \frac{10.87\times 10^7}{s(s+156.21)(s+230.33)(s+3021.8)} \qquad (7\text{-}172)$$

Taking the inverse Laplace transform of the last equation, we get

$$\theta_y(t) = (1 - 3.28e^{-156.21t} + 2.28e^{-230.33t} - 0.0045e^{-3021.8t})u_s(t) \qquad (7\text{-}173)$$

The last term in Eq. (7-173), which is due to the root at –3021.8, decays to zero very rapidly. Furthermore, the magnitude of the term at t = 0 is very small compared to the other two transient terms. This simply demonstrates that, in general, the contribution of roots that lie relatively far to the left in the s-plane to the transient response will be small. The roots that are closer to the imaginary axis will dominate the transient response, and these are defined as the **dominant roots** of the characteristic equation or of the system. In this case, the second-order system in Eq. (7-162) is a good approximation of the third-order system in Eq. (7-170).

When *K* = 14.5, the second-order system has a damping ratio of 0.707 because the real and imaginary parts of the two characteristic equation roots are identical. For the third-order system, recall that the damping ratio is strictly not defined. However, because the effect on transient of the root at –3021.8 is negligible, the two roots that dominate the transient response correspond to a damping ratio of 0.697. Thus, for *K* = 14.5, the second-order approximation by setting L_a to zero is not a bad one. It should be noted, however, that the fact that the second-order approximation is justified for *K* = 14.5 does not mean that the approximation is valid for all values of *K*.

When *K* = 181.2, the two complex-conjugate roots of the third-order system again dominate the transient response, and the equivalent damping ratio due to the two roots is only 0.0633, which is much smaller than the value of 0.2 for the second-order system. Thus, we see that the justification and accuracy of the second-order approximation diminish as the value of *K* is increased.

Figure 7-55 illustrates the root loci of the third-order characteristic equation of Eq. (7-171) as *K* varies. When *K* = 181.2, the real root at –3293.3 still contributes little to the transient response, but the two complex-conjugate roots at –57.49 ± *j*906.6 are much closer to the *jω*-axis than those of the second-order system for the same *K*, which are at –180.6 ± j884.75. This explains why the third-order system is a great deal less stable than the second-order system when *K* = 181.2.

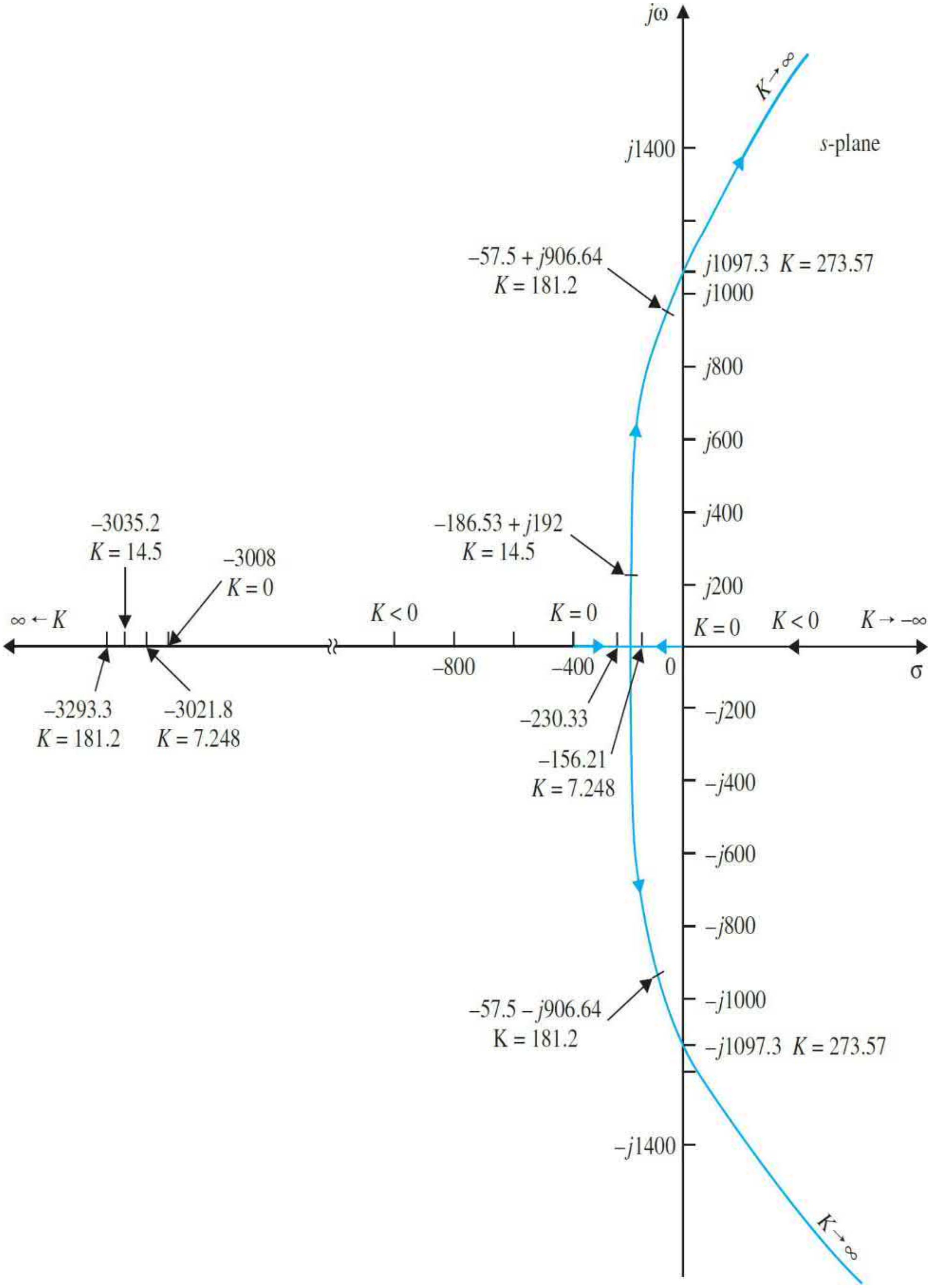

jω
s-plane
K → ∞
j1400
−57.5 + j906.64
K = 181.2
j1097.3 K = 273.57
j1000
j800
j600
−186.53 + j192
K = 14.5
j400
j200
−3035.2
K = 14.5
−3008
K = 0
∞ ← K
K < 0
K = 0
K = 0
K < 0
K → −∞
−800
−400
0
σ
−3293.3
K = 181.2
−3021.8
K = 7.248
−230.33
−156.21
K = 7.248
−j200
−j400
−j600
−j800
−57.5 − j906.64
K = 181.2
−j1000
−j1097.3 K = 273.57
−j1400
K → ∞

Figure 7-55 Root loci of the third-order attitude-control system.

From the Routh-Hurwitz criterion, the marginal value of *K* for stability is found to be 273.57. With this critical value of *K*, the closed-loop transfer function becomes

$$\frac{\Theta_y(s)}{\Theta_r(s)} = \frac{1.0872\times 10^8}{(s+3408.3)(s^2+1.204\times 10^6)} \tag{7-174}$$

The roots of the characteristic equation are at *s* = –3408.3, –j1097.3, and *j*1097.3. These points are shown on the root loci in Fig. 7-55.

The unit-step response of the system when *K* = 273.57 is

$$\theta_y(t) = [1 - 0.094e^{-3408.3t} - 0.952\sin(1097.3t + 72.16°)]u_s(t) \tag{7-175}$$

The steady-state response is an undamped sinusoid with a frequency of 1097.3 rad/s, and the system is said to be marginally stable. When *K* is greater than 273.57, the two complex-conjugate roots will have positive real parts, the sinusoidal component of the time response will increase with time, and the system is **unstable**. Thus, we see that the third-order system is capable of being unstable, whereas the second-order system obtained with L_a = 0 is stable for all finite positive values of *K*.

Figure 7-56 shows the unit-step responses of the third-order system for the three values of *K* used. The responses for *K* = 7.248 and *K* = 14.5 are very close to those of the second-order system with the same values of *K* that are shown in Fig. 7-54. However, the two responses for *K* = 181.2 are quite different.

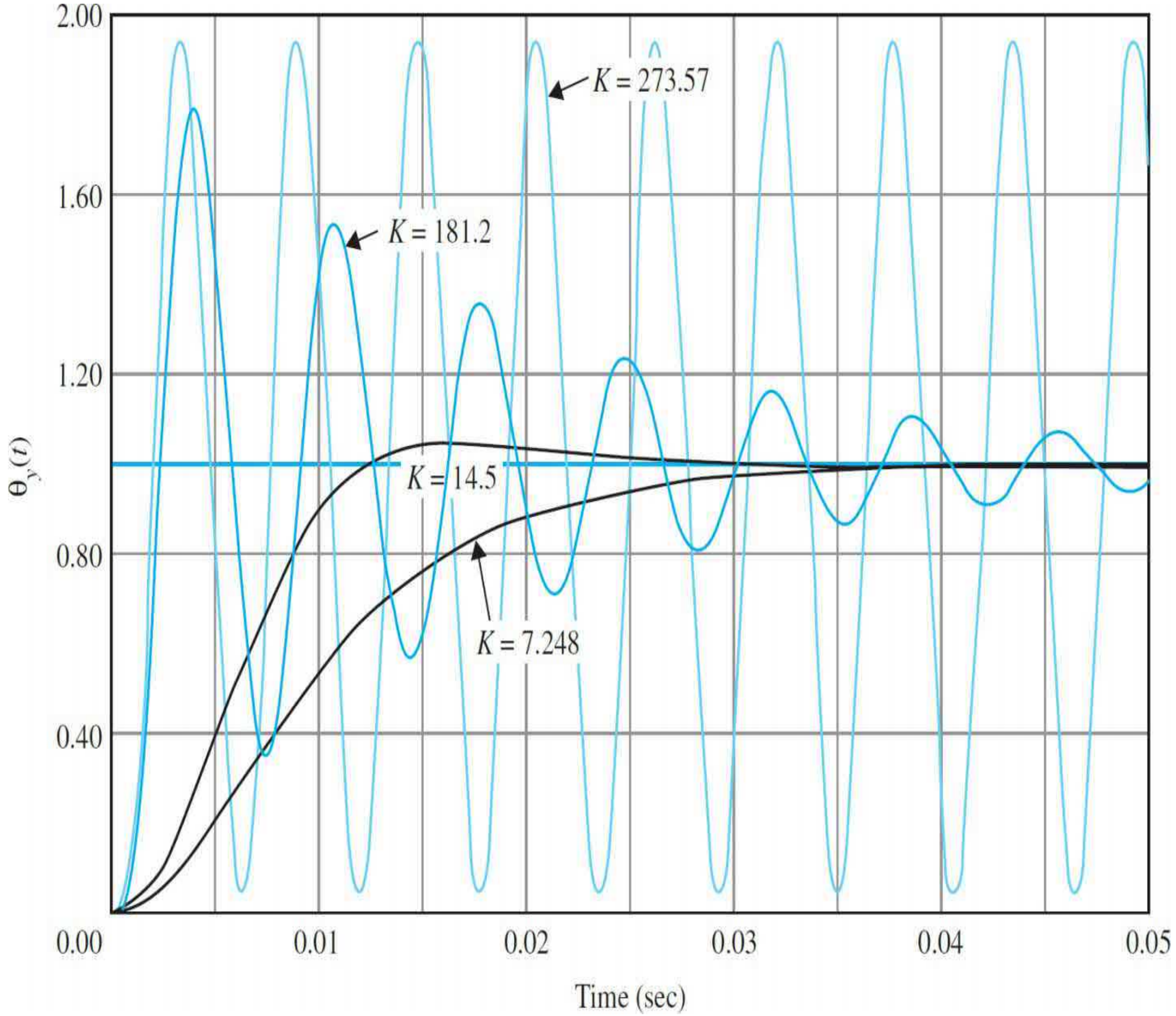
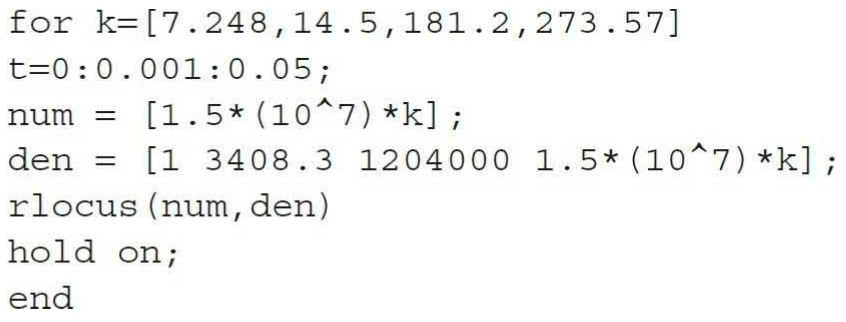


Figure 7-56 Unit-step responses of the third-order attitude-control system.

Toolbox 7-9-2

The root locus plot in Fig. 7-56 is obtained by the following MATLAB commands:



```
for k=[7.248,14.5,181.2,273.57]
t=0:0.001:0.05;
num = [1.5*(10^7)*k];
den = [1 3408.3 1204000 1.5*(10^7)*k];
rlocus(num,den)
hold on;
end
```

Final Thoughts

When the motor inductance is restored, the system is of the third order, and its apparent effect is addition of a pole to the forward-path transfer function. For small *K* values, the additional pole of the third-order system is far to the left of the *s*-plane, so its effect is small. However, as the value of *K* gets larger, the new pole of *G*(*s*) essentially “pushes” and “bends” the complex-conjugate portion of the root loci of the second-order system toward the right-half *s*-plane. The third-order system can now become unstable for large amplifier gain *K* values.

7-9-5 Steady-State Response

From Eq. (7-169), we see that, when the inductance is restored, the third-order system is still of type 1. The value of K_p is still the same as that given in Eq. (7-168). Thus, the inductance of the motor does not affect the steady-state performance of the system, provided that the system is stable. This is expected, since L_a affects only the rate of change and not the final value of the motor current.

7-10 THE CONTROL LAB: INTRODUCTION TO LEGO MINDSTORMS NXT MOTOR—POSITION CONTROL

Continuing our work from Sec. 6-6, now that the motor parameters have been measured, they can be further fine-tuned by comparing the simulated position response to the actual position response of the motor. See App. D for more details.

No-Load Position Response

Hence, using the simplified closed-loop transfer function, similar to Example 7-5-1,

$$\frac{\Theta_m(s)}{\Theta_{in}(s)} = \frac{\dfrac{K_P K_i K_s}{R_a J}}{s^2 + \left(\dfrac{R_a B_m + K_i K_b}{R_a J}\right)s + \dfrac{K_P K_i K_s}{R_a J}} = \frac{\omega_n^2}{(s^2 + 2\zeta\omega_n s + \omega_n^2)} \tag{7-176}$$

where K_s is the sensor gain, calibrated to $K_s = 1$ (i.e., 1 V = 1 rad). The closed-loop position response of the motor with no load is simulated for a step input of 160 degrees or 5.585 rad. The results are shown below in Fig. 7-57 for multiple proportional control gains, K_P.

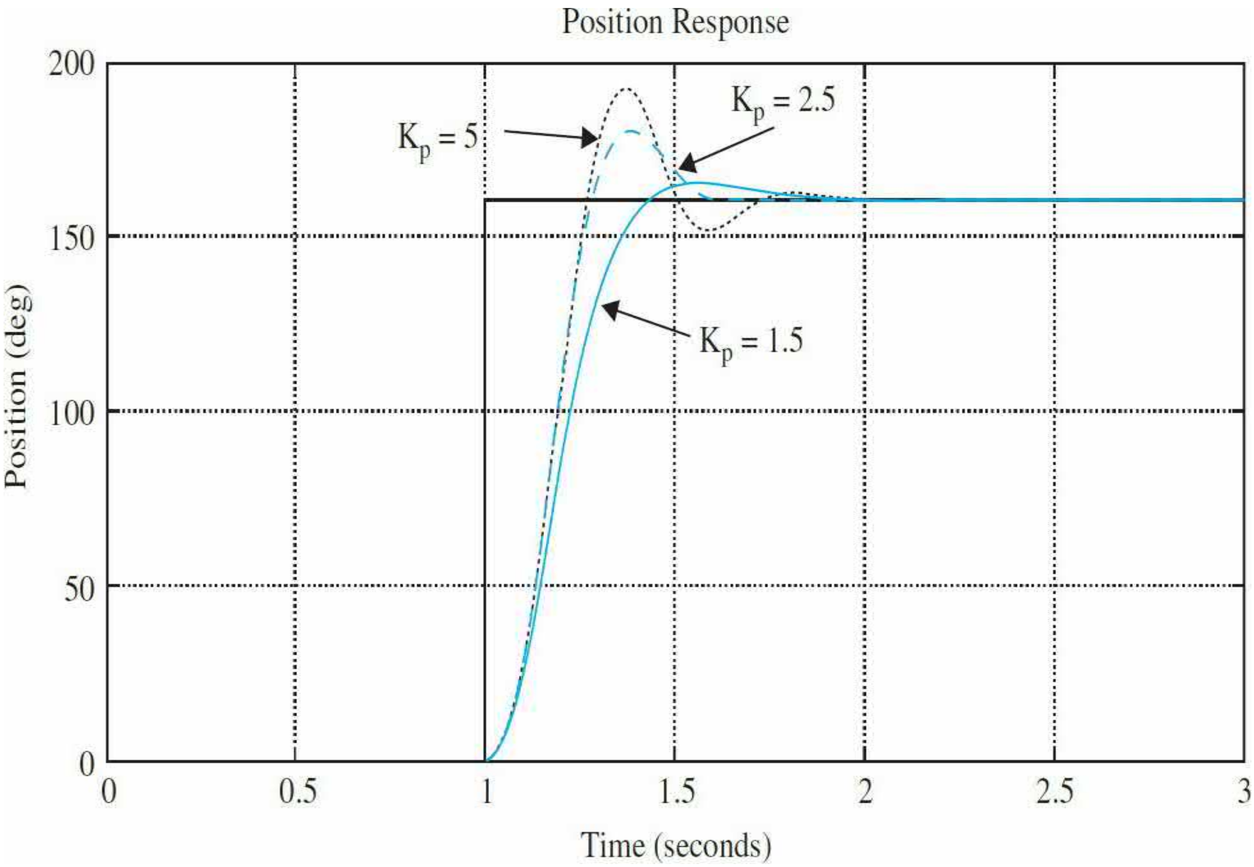


Figure 7-57 Simulated no-load closed-loop position response results for multiple K_P gains.

Next, the closed-loop position response of the NXT motor is found. The results are shown in Fig. 7-58 for multiple K_P gains.

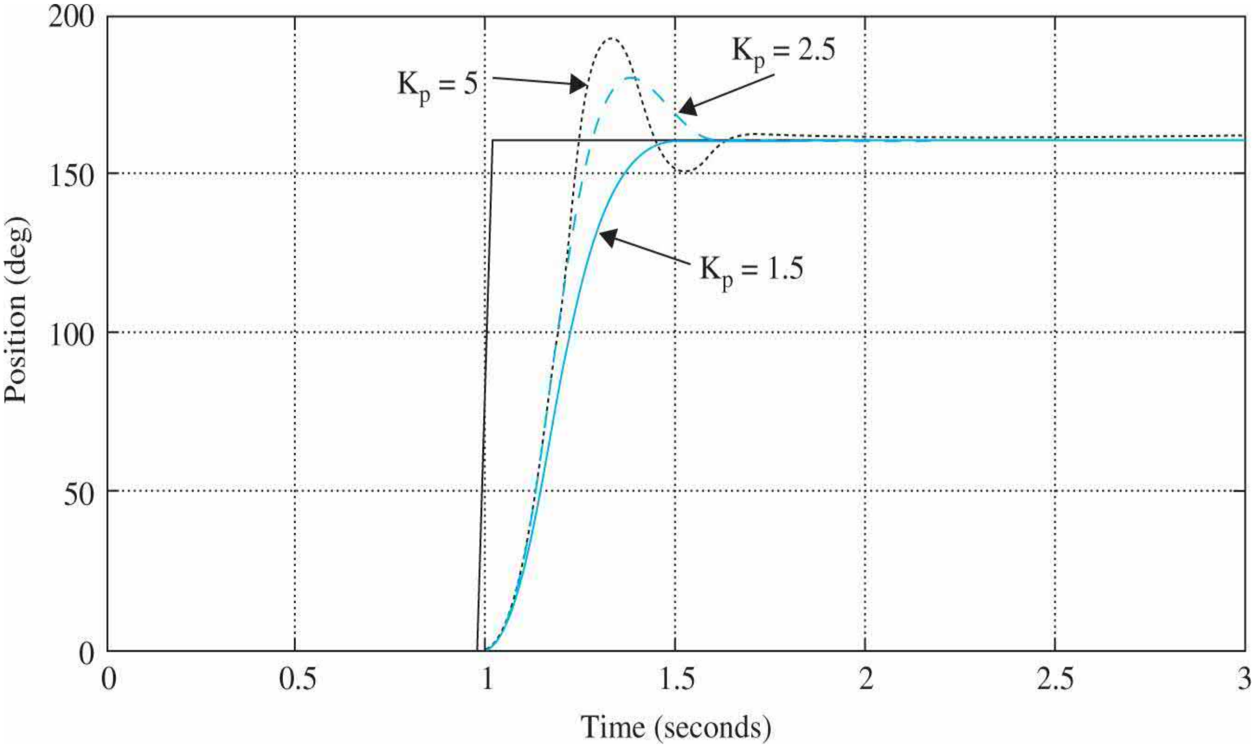
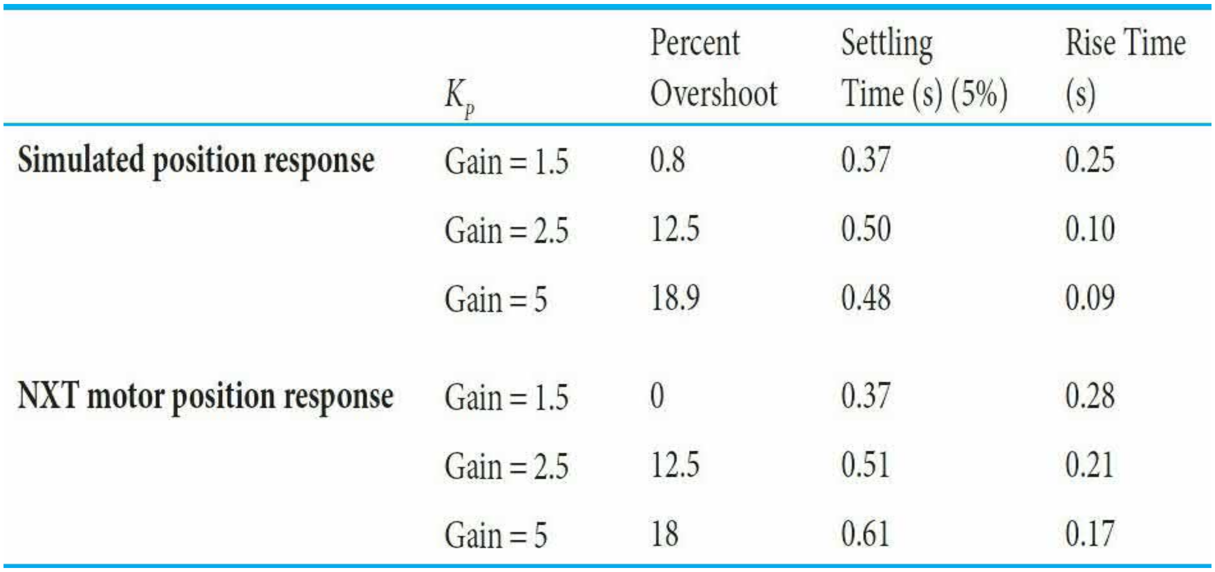


Figure 7-58 NXT motor no-load closed-loop position response results for multiple K_P gains.

Both the model response and actual motor response performance specifications are measured and tabulated in Table 7-12. By analyzing the results in Table 7-12, the system model matches the behavior of the actual motor, and no further fine tuning is necessary.

TABLE 7-12 No-Load Closed-Loop Position Response Performance Specifications Comparison



	K_p	Percent Overshoot	Settling Time (s) (5%)	Rise Time (s)
Simulated position response	Gain = 1.5	0.8	0.37	0.25
	Gain = 2.5	12.5	0.50	0.10
	Gain = 5	18.9	0.48	0.09
NXT motor position response	Gain = 1.5	0	0.37	0.28
	Gain = 2.5	12.5	0.51	0.21
	Gain = 5	18	0.61	0.17

Robotic Arm Position Response

Next, the closed-loop position response of the NXT motor with the robotic arm and payload is found. The results are shown in Fig. 7-59 for multiple *KP* gains. Note because of the backlash in the gearbox, the final value is not always 160 degrees.

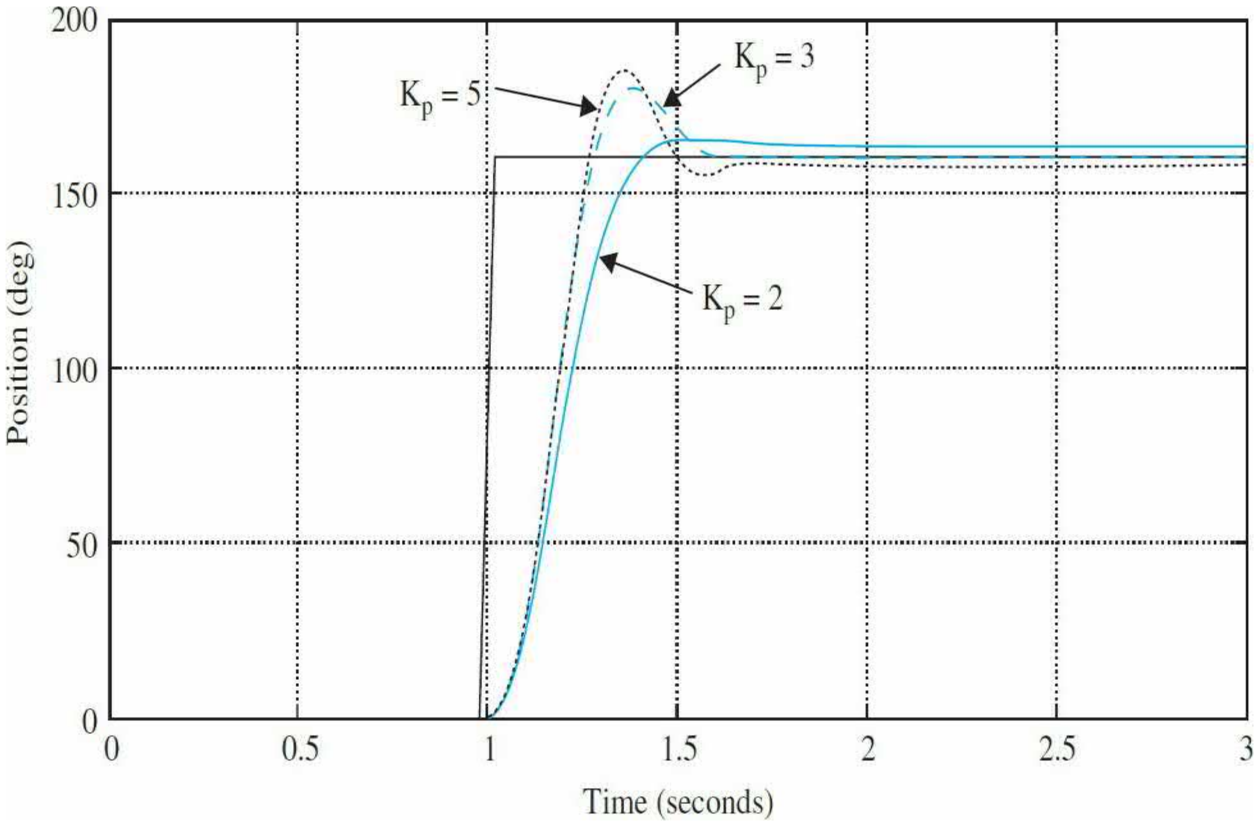
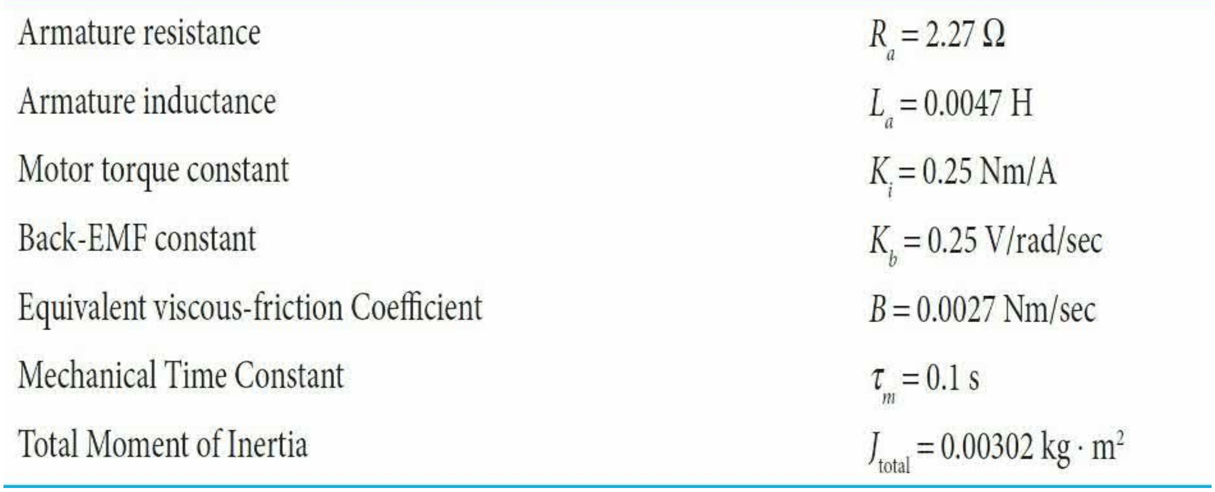


Figure 7-59 Robotic arm with payload closed-loop position response results for multiple *KP* gains.

The measured parameter values when the robotic arm is attached to the motor are shown in Table 7-13. Notice that the total inertia and viscous damping coefficient are higher than the no-load case found in Sec. 6-6. Next, the closed-loop position response of the motor with the robotic arm and payload is simulated for $K_P = 3$. Note that the voltage is saturated to half its maximum value (~±2.25 V) to slow down the robotic arm in all the following tests. The simulation result is compared to the corresponding experimental response, as shown in Fig. 7-59.

TABLE 7-13 Robotic Arm and Payload Experimental Parameters



Armature resistance	$R_a = 2.27\ \Omega$
Armature inductance	$L_a = 0.0047$ H
Motor torque constant	$K_i = 0.25$ Nm/A
Back-EMF constant	$K_b = 0.25$ V/rad/sec
Equivalent viscous-friction Coefficient	$B = 0.0027$ Nm/sec
Mechanical Time Constant	$\tau_m = 0.1$ s
Total Moment of Inertia	$J_{\text{total}} = 0.00302\ \text{kg} \cdot \text{m}^2$

Comparing the two responses, as shown in Fig. 7-60, it is clear that the arm/payload model requires some fine tuning. In order to improve model accuracy, the system overshoot, rise time and settling time must all decrease. To achieve this task, let us examine the mathematical model of the system. Note that the parameter identification discussed in this section assumes a second-order model for the position response because the motor electric-time constant $\tau_e = L_a/R_a = 0.002$ sec is very small. Hence, similar to Example 7-5-1, the simplified closed-loop transfer function is

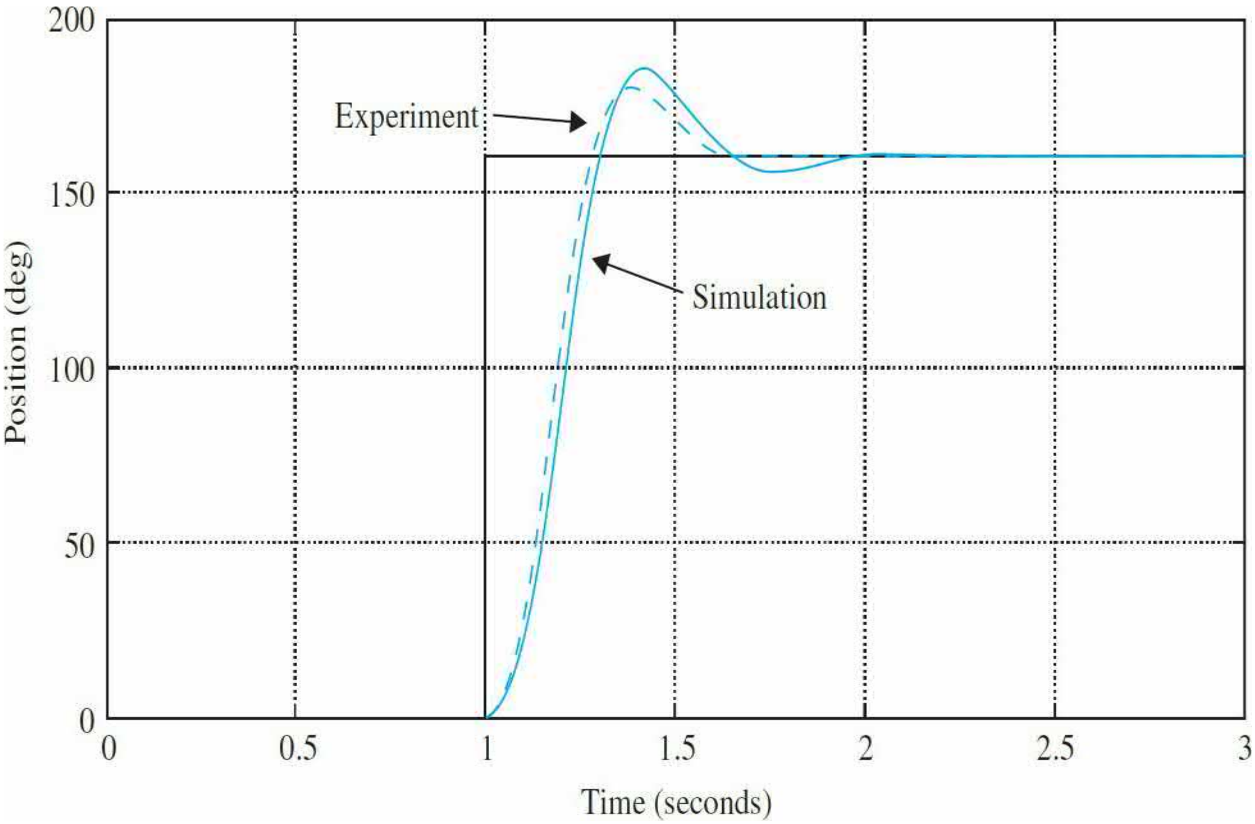


Figure 7-60 Comparison of simulated and experimental robotic arm with payload closed-loop position response for $K_P = 3$.

$$\frac{\Theta_{\text{arm/payload}}(s)}{\Theta_{in}(s)} = \frac{\dfrac{K_P K_i K_s}{R_a J_{\text{total}}}}{s^2 + \left(\dfrac{R_a B + K_i K_b}{R_a J_{\text{total}}}\right)s + \dfrac{K_P K_i K_s}{R_a J_{\text{total}}}} = \frac{\omega_n^2}{(s^2 + 2\zeta\omega_n s + \omega_n^2)} \tag{7-177}$$

where K_s is the sensor gain, calibrated to $K_s = 1$. Since Eq. (7-177) is a second-order system, we have

$$2\zeta\omega_n = \frac{R_a B + K_i K_b}{R_a J_{\text{total}}} \tag{7-178a}$$

$$\omega_n = \sqrt{\frac{K_P K_i}{R_a J_{\text{total}}}} \tag{7-178b}$$

Combining Eqs. (7-177) and (7-178), we get

$$\zeta = \frac{R_a B + K_i K_b}{2\sqrt{K_P K_i R_a J_{\text{total}}}} \tag{7-179}$$

The system poles are

$$s_{1,2} = -\frac{R_a B + K_i K_b}{2R_a J_{\text{total}}} \pm \sqrt{\left(\frac{R_a B + K_i K_b}{2R_a J_{\text{total}}}\right)^2 - \frac{K_P K_i}{R_a J_{\text{total}}}} \tag{7-180}$$

Considering Eqs. (7-178) through (7-180), and using the parameter values in Table 7-8 and $K_P = 3$, we have

$$\omega_n = 10.45 \text{ rad/s} \tag{7-181}$$

$$\zeta = 0.478 \tag{7-182}$$

$$s_{1,2} = -5 \pm j9.18 \tag{7-183}$$

Further, upon examining Fig. 7-18, which describes the effects of moving a second-order system poles (in the *s*-plane) on its time response performance, moving the two poles in Eq. (7-180) horizontally to the left should increase ζ while reducing system overshoot, rise time, and settling time. To do so the first term in Eq. (7-180) should increase while the second term should be held a constant. Having these two conditions met simultaneously may be a tedious task. So without bothering with rigorous mathematical expressions, we resort to trial and error and reduce the value of J_{total} while checking the overall response. Alternatively, you can vary B or both J_{total} and B, simultaneously. Variation of J_{total} seems to be the best choice for fine tuning, as our confidence levels for J_{total} was not high, see discussions in Secs. 6-6-3 and 6-6-4.

The best response for $K_P = 3$ may be achieved for $J_{total} = 0.00273 \text{ kg} \cdot \text{m}^2$, as shown in Fig. 7-61. For this parameter choice from Eqs. (7-181) to (7-183), we have

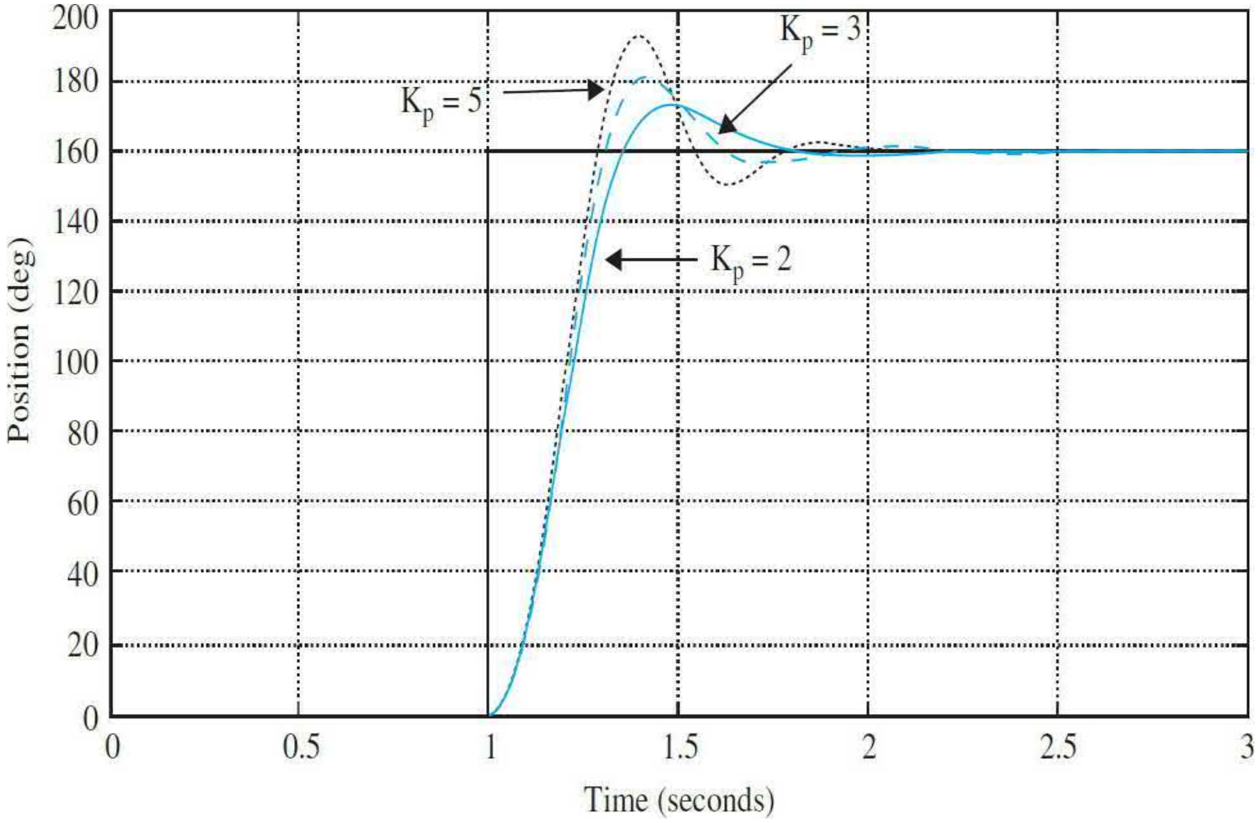


Figure 7-61 Simulated robotic arm with payload closed-loop position response results for multiple K_P gains.

$$\omega_n = 11.0 \text{ rad/s} \tag{7-184}$$

$$\zeta = 0.503 \tag{7-185}$$

$$s_{1,2} = -5.54 \pm j9.03 \tag{7-186}$$

Where the poles in Eq. (7-186), in comparison to Eq. (7-183) have moved to the left while both z and ω_n have increased. From Eq. (7-42), the PO is expected to decrease. Also from Eqs. (7-46) and (7-50), the rise time and settling times are also expected to decrease. For the second-order model using performance specification formulas, we get

$$\text{Percent maximum overshoot} = 100e^{-\pi\zeta/\sqrt{1-\zeta^2}} = 16 \tag{7-187}$$

$$t_{\max} = \frac{\pi}{\omega_n\sqrt{1-\zeta^2}} = 0.33 \text{ s} \qquad (7\text{-}188)$$

$$t_r = \frac{1-0.4167\zeta+2.917\zeta^2}{\omega_n} = 0.14 \text{ s} \qquad (7\text{-}189)$$

$$5 \text{ percent settling time: } t_s \cong \frac{3.2}{\zeta\omega_n} = 0.58 \text{ s} \qquad (7\text{-}190)$$

The performance specifications of the system, shown in Table 7-14, for $K_P = 3$ are in line with these expectations. The minor discrepancies between the calculated values, Eqs. (7-188) and (7-189), and the simulation measurements in Table 7-14 are obviously attributed to the differences between the second- and third-order models—the simulation software used here (Simulink; see App. D) considers the third-order model. Note also since our system is nonlinear in reality, we should not expect the simulation response to closely match that of the experiment for other controller gain values. From Table 7-14, we see this is in fact the case.

TABLE 7-14 Robotic Arm Closed-Loop Position Response Performance Specification Comparison

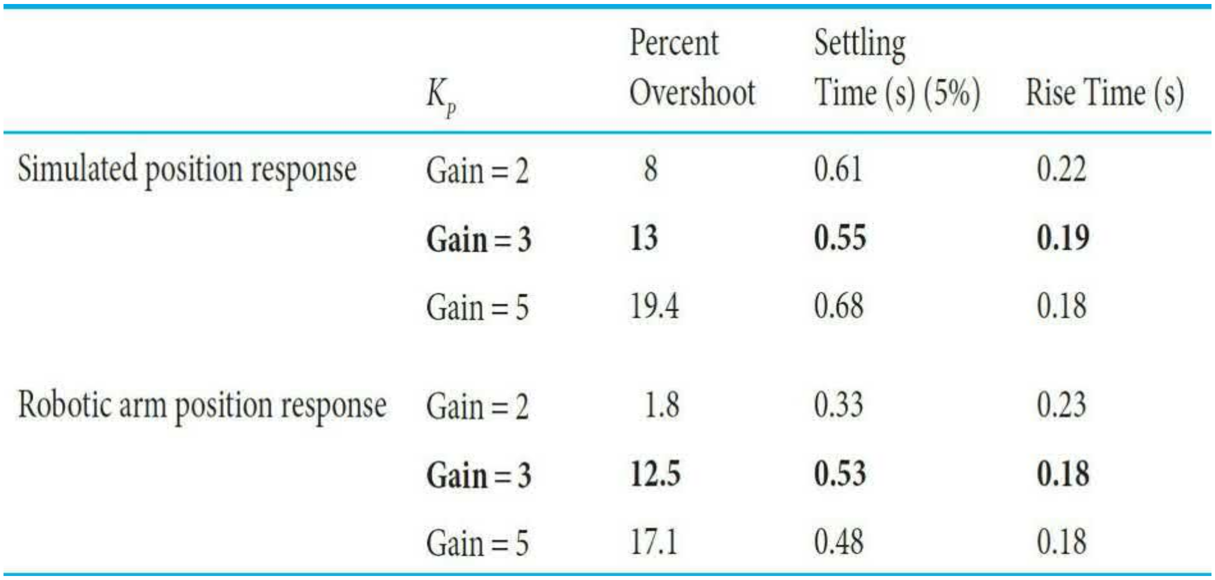
	K_p	Percent Overshoot	Settling Time (s) (5%)	Rise Time (s)
Simulated position response	Gain = 2	8	0.61	0.22
	Gain = 3	**13**	**0.55**	**0.19**
	Gain = 5	19.4	0.68	0.18
Robotic arm position response	Gain = 2	1.8	0.33	0.23
	Gain = 3	**12.5**	**0.53**	**0.18**
	Gain = 5	17.1	0.48	0.18

Do not forget that the PO for the experimental system is measured from the response final value, using Eq. (7-41). For example for $K_p = 2$, the NXT motor (with arm and payload) response final and peak values are at 164 and 167 degrees, respectively. So

$$PO = 100\left(\frac{167-164}{164}\right) = 1.8 \tag{7-191}$$

At this point you may wish to further fine tune system parameters, or decide that the model is good enough. For all practical purposes, the parameter values shown in Table 7-15 appear reasonable, and we should stop the fine tuning process.

TABLE 7-15 Fine-Tuned Robotic Arm and Payload Experimental Parameters



	K_p	Percent Overshoot	Settling Time (s) (5%)	Rise Time (s)
Simulated position response	Gain = 2	8	0.61	0.22
	Gain = 3	**13**	**0.55**	**0.19**
	Gain = 5	19.4	0.68	0.18
Robotic arm position response	Gain = 2	1.8	0.33	0.23
	Gain = 3	**12.5**	**0.53**	**0.18**
	Gain = 5	17.1	0.48	0.18

In the end, now that we have a *good enough* model of the system, we can design different type of controllers for this system. In App. D, we will provide labs that allow you to further compare the real-life motor characteristics with simulation using MATLAB and Simulink software.

7-11 SUMMARY

This chapter was devoted to the time-domain analysis of linear continuous-data control systems. The time response of control systems is divided into the transient and the steady-state responses. The transient response is characterized by such criteria as the maximum overshoot, rise time, delay time, and settling time, and such parameters as damping ratio, natural undamped frequency, and time constant. The analytical expressions of these parameters can all be related to the system parameters simply if the transfer function is of the second-order prototype. For other type second-order systems or for higher-order systems, the analytical relationships between the transient parameters and the system constants are more difficult to determine. Computer simulations are recommended for these systems. Examples of speed response and position control of a motor were used to better demonstrate the topic.

The steady-state error is a measure of the accuracy of the system as time approaches infinity. When the system has unity feedback for the step, ramp, and parabolic inputs, the steady-state error is characterized by the error constants *Kp*, *Kv*, and *Ka*, respectively, as well as the system type. When applying the steady-state error analysis, the final-value theorem of the Laplace transform is the basis; it should be ascertained that the closed-loop system is stable or the error analysis will be invalid. The error constants are not defined for systems with nonunity feedback. For nonunity-feedback systems, a method of determining the steady-state error was introduced by using the closed-loop transfer function. Examples of speed control of a motor were used to better demonstrate the topic.

Time-domain analysis of a position-control case system was conducted. The transient and steady-state analyses were carried out first by approximating the system as a second-order system. The effect of varying the amplifier gain *K* on the transient and steady-state performance was demonstrated. The concept of the root-locus technique was introduced, and the system was then analyzed as a third-order system. It was shown that the second-order approximation was accurate only for low values of *K*.

The effects of adding poles and zeros to the forward-path and closed-loop transfer functions were demonstrated. The dominant poles of transfer functions were also discussed. This established the significance of the location of the poles of the transfer function in the *s*-plane and under what conditions the insignificant poles (and zeros) could be neglected with regard to the transient response.

Later in the chapter, simple controllers—namely the PD, PI, and PID—

were introduced. Designs were carried out in the time-domain (and *s*-domain). The time-domain design may be characterized by specifications such as the relative damping ratio, maximum overshoot, rise time, delay time, settling time, or simply the location of the characteristic-equation roots, keeping in mind that the zeros of the system transfer function also affect the transient response. The performance is generally measured by the step response and the steady-state error.

REFERENCES

1. J. C. Willems and S. K. Mitter, “Controllability, Observability, Pole Allocation, and State Reconstruction,” *IEEE Trans. Automatic Control*, Vol. AC-16 pp. 582–595, Dec. 1971.
2. H. W. Smith and E. J. Davison, “Design of Industrial Regulators,” *Proc. IEE (London)*, Vol. 119**,** PP. 1210–1216, AUG. 1972.
3. F. N. Bailey and S. Meshkat, “Root Locus Design of a Robust Speed Control,” *Proc. Incremental Motion Control Symposium*, pp. 49–54, June 1983.
4. M. Vidyasagar, “On Undershoot and Nonminimum Phase Zeros,” *IEEE Trans. Automatic Control*, Vol. AC-31, p. 440, May 1986.
5. T. Norimatsu and M. Ito, “On the Zero Non-Regular Control System,” *J. Inst. Elec. Eng. Japan*, Vol. 81, pp. 567–575, 1961.
6. K. Ogata, *Modern Control Engineering*, 4th Ed., Prentice Hall, NJ, 2002.
7. G. F. Franklin and J. D. Powell, *Feedback Control of Dynamic Systems*, 5th Ed., Prentice-Hall, NJ, 2006.
8. J. J. Distefano, III, A. R. Stubberud, and I. J. Williams, *Schaum’s Outline of Theory and Problems of Feedback and Control Systems*, 2nd Ed. New York; McGraw-Hill, 1990.
9. F. Golnaraghi and B. C. Kuo, *Automatic Control Systems*, 9th Ed. 2009.
10. Retrieved February 24, 2012, from http://www.philohome.com/nxtmotor/nxtmotor.htm.
11. LEGO Education. (n.d.) LEGO® MINDSTORMS Education NXT User Guide. Retrieved March 07, 2012, from http://education.lego.com/downloads/?q={02FB6AC1-07B0-4E1A-862D-7AE2DBC88F9E}.

12. Paul Oh. (n.d.) NXT Motor Characteristics: Part 2—Electrical Connections. Retrieved March 07, 2012, from http://www.pages.drexel.edu/~pyo22/mem380Mechatronics2Spring2(2011/week09/lab/mechatronics2-LabNxtMotorCharacteristics-Part02.pdf

13. Mathworks In. (n.d.) Simulink Getting Started Guide. Retrieved April 1, 2012, from http://www.mathworks.com/access/helpdesk/help/pdf_doc/simulink/sl

PROBLEMS

In addition to using the conventional approaches, use MATLAB to solve the problem in this chapter.

7-1. A pair of complex-conjugate poles in the *s*-plane is required to meet the various specifications that follow. for each specification, sketch the region in the *s*-plane in which the poles should be located.

(a)	$\zeta \geq 0.0707$	$\omega_n \geq 2$ rad/s	(Positive damping)
(b)	$0 \leq \zeta \leq 0.707$	$\omega_n \leq 2$ rad/s	(Positive damping)
(c)	$\zeta \leq 0.5$	$1 \leq \omega_n \leq 5$ rad/s	(Positive damping)
(d)	$0.5 \leq \zeta \leq 0.707$	$\omega_n \leq 5$ rad/s	(Positive and negative damping)

7-2. Determine the type of the following unity-feedback systems for which the forward-path transfer functions are given.

(a) $$G(s) = \frac{K}{(1+s)(1+10s)(1+20s)}$$

(b) $$G(s) = \frac{10e^{-0.2s}}{(1+s)(1+10s)(1+20s)}$$

(c) $$G(s) = \frac{10(s+1)}{s(s+5)(s+6)}$$

(d) $$G(s) = \frac{100(s-1)}{s^2(s+5)(s+6)^2}$$

(e) $$G(s) = \frac{10(s+1)}{s^3(s^2+5s+5)}$$

(f) $$G(s) = \frac{100}{s^3(s+2)^2}$$

(g) $$G(s) = \frac{5(s+2)}{s^2(s+4)}$$

(h) $$G(s) = \frac{8(s+1)}{(s^2+2s+3)(s+1)}$$

7-3. Determine the step, ramp, and parabolic error constants of the following unity-feedback control systems. The forward-path transfer functions are given.

(a) $$G(s) = \frac{1000}{(1+0.1s)(1+10s)}$$

(b) $$G(s) = \frac{100}{s(s^2+10s+100)}$$

(c) $$G(s) = \frac{K}{s(1+0.1s)(1+0.5s)}$$

(d) $$G(s) = \frac{100}{s^2(s^2+10s+100)}$$

(e) $$G(s) = \frac{1000}{s(s+10)(s+100)}$$

(f) $$G(s) = \frac{K(1+2s)(1+4s)}{s^2(s^2+s+1)}$$

7-4. For the unity-feedback control systems described in Prob. 7-2, determine the steady-state error for a unit-step input, a unit-ramp input, and a parabolic input, $(t^2/2)u_s(t)$. Check the stability of the system before applying the final-value theorem.

7-5. The following transfer functions are given for a single-loop nonunity-feedback control system. Find the steady-state errors due to a unit-step input, a unit-ramp input, and a parabolic input $(t^2/2)u_s(t)$.

(a) $G(s)=\dfrac{1}{(s^2+s+2)}\quad H(s)=\dfrac{1}{(s+1)}$

(b) $G(s)=\dfrac{1}{s(s+5)}\quad H(s)=5$

(c) $G(s)=\dfrac{1}{s^2(s+10)}\quad H(s)=\dfrac{s+1}{s+5}$

(d) $G(s)=\dfrac{1}{s^2(s+12)}\quad H(s)=5(s+2)$

7-6. Find the steady-state errors of the following single-loop control systems for a unit-step input, a unit-ramp input, and a parabolic input, $(t^2/2)u_s(t)$. For systems that include a parameter K, find its value so that the answers are valid.

(a) $M(s)=\dfrac{s+4}{s^4+16s^3+48s^2+4s+4},K_H=1$

(b) $M(s)=\dfrac{K(s+3)}{s^3+3s^2+(K+2)s+3K},K_H=1$

(c) $M(s)=\dfrac{s+5}{s^4+15s^3+50s^2+10s},H(s)=\dfrac{10s}{s+5}$

(d) $M(s)=\dfrac{K(s+5)}{s^4+17s^3+60s^2+5Ks+5K},K_H=1$

7-7. The output of the system shown in Fig. 7P-8 has a transfer function Y/X. Find the poles and zeros of the closed-loop system and the system type.

7-8. Find the position, velocity, and acceleration error constants for the system given in Fig. 7P-8.

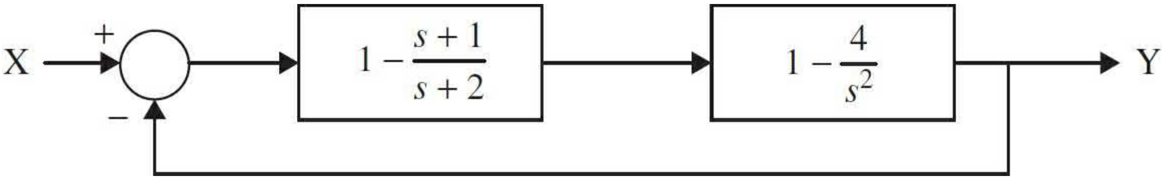


Figure 7P-8

7-9. Find the steady-state error for Prob. 7-8 for (a) a unit-step input, (b) a unit-ramp input, and (c) a unit-parabolic input.

7-10. Repeat Prob. 7-8 for the system given in Fig. 7P-10.

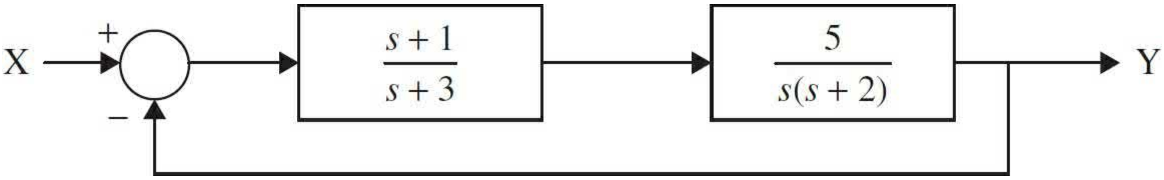


Figure 7P-10

7-11. Find the steady-state error of the system given in Prob. 7-10 when the input is

$$X = \frac{5}{2s} - \frac{3}{s^2} + \frac{4}{s^3}$$

7-12. Find the rise time of the following first-order system:

$$G(s) = \frac{1-k}{s-k} \quad \text{with} |k| < 1$$

7-13. The block diagram of a control system is shown in Fig. 7P-13. Find the step-, ramp-, and parabolic-error constants. The error signal is defined to be $e(t)$. Find the steady-state errors in terms of K and K_t when the following inputs are applied. Assume that the system is stable.

(a) $r(t) = u_s(t)$

(b) $r(t) = tu_s(t)$

(c) $r(t) = (t^2/2)u_s(t)$

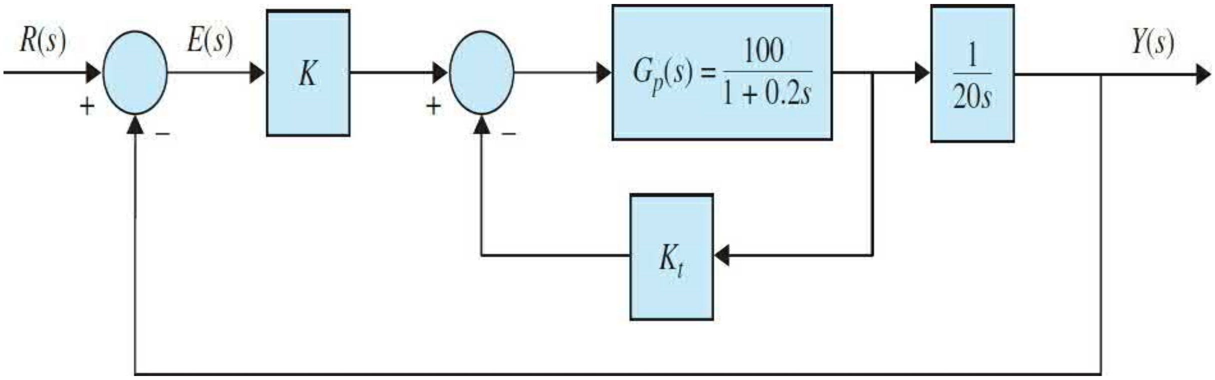


Figure 7P-13

7-14. Repeat Prob. 7-13 when the transfer function of the process is, instead,

$$G_p(s) = \frac{100}{(1+0.1s)(1+0.5s)}$$

What constraints must be made, if any, on the values of K and K_t so that the answers are valid? Determine the minimum steady-state error that can be achieved with a unit-ramp input by varying the values of K and K_t.

7-15. For the position-control system shown in Fig. 3P-7, determine the following.

(a) Find the steady-state value of the error signal $\theta_e(t)$ in terms of the system parameters when the input is a unit-step function.

(b) Repeat part (a) when the input is a unit-ramp function. Assume that the system is stable.

7-16. The block diagram of a feedback control system is shown in Fig. 7P-16. The error signal is defined to be $e(t)$.

(a) Find the steady-state error of the system in terms of K and K_t when the input is a unit-ramp function. Give the constraints on the values of K and K_t so that the answer is valid. Let $n(t) = 0$ for this part.

(b) Find the steady-state value of $y(t)$ when $n(t)$ is a unit-step function. Let $r(t) = 0$. Assume that the system is stable.

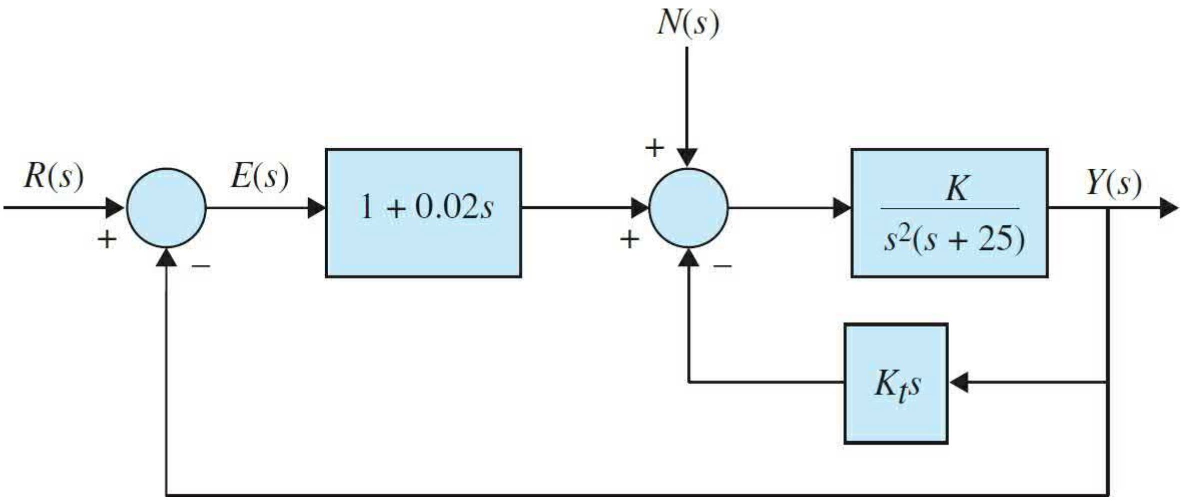


Figure 7P-16

7-17. The block diagram of a linear control system is shown in Fig. 7P-17, where $r(t)$ is the reference input and $n(t)$ is the disturbance.

(a) Find the steady-state value of $e(t)$ when $n(t) = 0$ and $r(t) = tu_s(t)$. Find the conditions on the values of α and K so that the solution is valid.

(b) Find the steady-state value of $y(t)$ when $r(t) = 0$ and $n(t) = u_s(t)$.

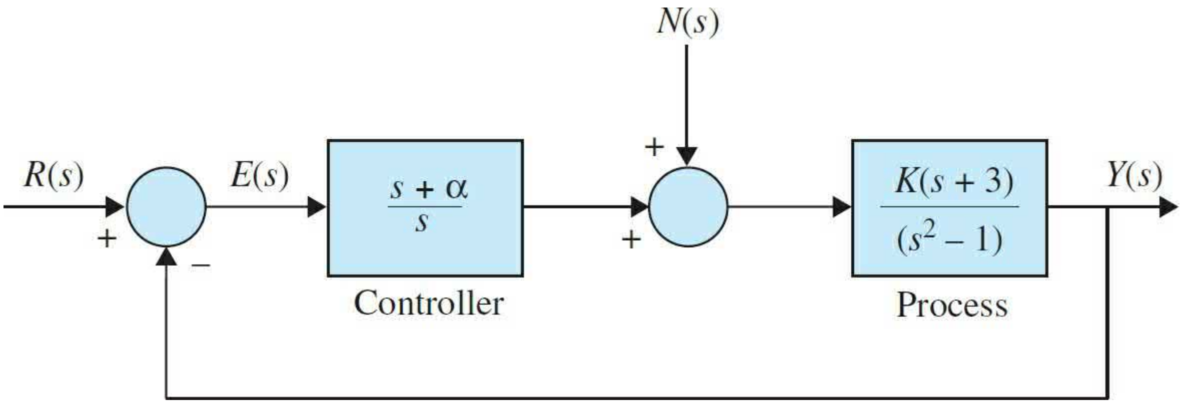


Figure 7P-17

7-18. The unit-step response of a linear control system is shown in Fig. 7P-18. Find the transfer function of a second-order prototype system to model the system.

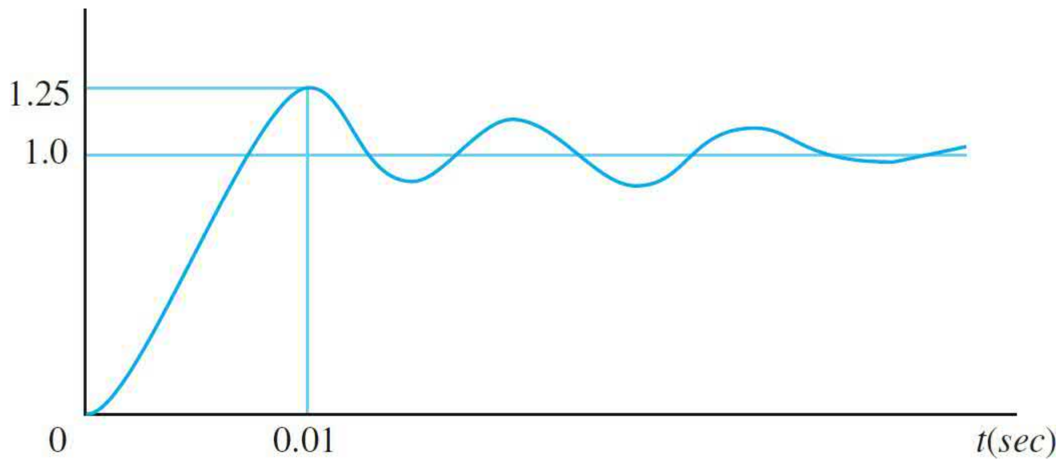


Figure 7P-18

7-19. For the control system shown in Fig. 7P-13, find the values of K and K_t so that the maximum overshoot of the output is approximately 4.3 percent and the rise time t_r is approximately 0.2 s. Use Eq. (7-98) for the rise-time relationship. Simulate the system with any time-response simulation program to check the accuracy of your solutions.

7-20. Repeat Prob. 7-19 with a maximum overshoot of 10 percent and a rise time of 0.1 s.

7-21. Repeat Prob. 7-19 with a maximum overshoot of 20 percent and a rise time of 0.05 s.

7-22. For the control system shown in Fig. 7P-13, find the values of K and K_t so that the maximum overshoot of the output is approximately 4.3 percent and the delay time t_d is approximately 0.1 s. Use Eq. (7-96) for the delay-time relationship. Simulate the system with a computer program to check the accuracy of your solutions.

7-23. Repeat Prob. 7-22 with a maximum overshoot of 10 percent and a delay time of 0.05 s.

7-24. Repeat Prob. 7-22 with a maximum overshoot of 20 percent and a delay time of 0.01 s.

7-25. For the control system shown in Fig. 7P-13, find the values of K and K_t so that the damping ratio of the system is 0.6 and the settling time of the unit-step response is 0.1 s. Use Eq. (7-102) for the settling time relationship. Simulate the system with a computer program to check the accuracy of your results.

7-26. (a) Repeat Prob. 7-25 with a maximum overshoot of 10 percent and a settling time of 0.05 s. (b) Repeat Prob. 7-25 with a maximum overshoot of 20 percent and a settling time of 0.01 s.

7-27. Repeat Prob. 7-25 with a damping ratio of 0.707 and a settling time of 0.1 s. Use Eq. (7-103) for the settling time relationship.

7-28. The forward-path transfer function of a control system with unity feedback is

$$G(s)=\frac{K}{s(s+a)(s+30)}$$

where a and K are real constants.

(a) Find the values of a and K so that the relative damping ratio of the complex roots of the characteristic equation is 0.5 and the rise time of the unit-step response is approximately 1 s. Use Eq. (7-98) as an approximation of the rise time. With the values of a and K found, determine the actual rise time using computer simulation.

(b) With the values of a and K found in part (a), find the steady-state errors of the system when the reference input is (i) a unit-step function and (ii) a unit-ramp function.

7-29. The block diagram of a linear control system is shown in Fig. 7P-29.

(a) By means of trial and error, find the value of K so that the characteristic equation has two equal real roots and the system is stable. You may use any root-finding computer program to solve this problem.

(b) Find the unit-step response of the system when K has the value found in part (a). Use any computer simulation program for this. Set all the initial conditions to zero.

(c) Repeat part (b) when $K = –1$. What is peculiar about the step response for small t, and what may have caused it?

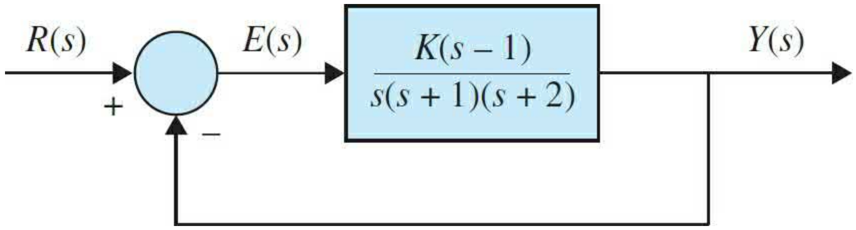


Figure 7P-29

7-30. A controlled process is represented by the following dynamic equations:

$$\frac{dx_1(t)}{dt} = -x_1(t) + 5x_2(t)$$

$$\frac{dx_2(t)}{dt} = -6x_1(t) + u(t)$$

$$y(t) = x_1(t)$$

The control is obtained through state feedback with

$$u(t) = -k_1 x_1(t) - k_2 x_2(t) + r(t)$$

where k_1 and k_2 are real constants, and $r(t)$ is the reference input.

(a) Find the locus in the k_1-versus-k_2 plane (k_1 = vertical axis) on which the overall system has a natural undamped frequency of 10 rad/s.

(b) Find the locus in the k_1-versus-k_2 plane on which the overall system has a damping ratio of 0.707.

(c) Find the values of k_1 and k_2 such that $\zeta = 0.707$ and $\omega_n = 10$ rad/sec.

(d) Let the error signal be defined as $e(t) = r(t) – y(t)$. Find the steady-state error when $r(t) = us(t)$ and k_1 and k_2 are at the values found in part (c).

(e) Find the locus in the k_1-versus-k_2 plane on which the steady-state error due to a unit-step input is zero.

7-31. The block diagram of a linear control system is shown in Fig. 7P-31. Construct a parameter plane of K_p versus K_d (K_p is the vertical axis), and show the following trajectories or regions in the plane.

(a) Unstable and stable regions

(b) Trajectories on which the damping is critical ($\zeta = 1$)

(c) Region in which the system is overdamped ($\zeta = 1$)

(d) Region in which the system is underdamped ($\zeta = 1$)

(e) Trajectory on which the parabolic-error constant K_a is 1000 s^{-2}

(f) Trajectory on which the natural undamped frequency ω_n is 50 rad/s

(g) Trajectory on which the system is either uncontrollable or unobservable (hint: look for pole-zero cancellation)

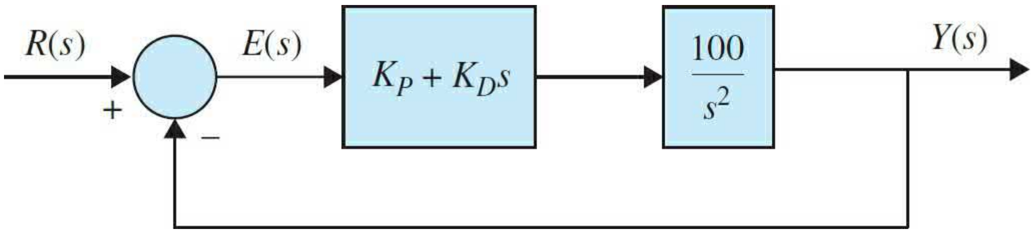


Figure 7P-31

7-32. The block diagram of a linear control system is shown in Fig. 7P-32. The fixed parameters of the system are given as $T = 0.1$, $J = 0.01$, and $K_i = 10$.

(a) When $r(t) = tu_s(t)$ and $T_d(t) = 0$, determine how the values of K and K_t affect the steady-state value of $e(t)$. Find the restrictions on K and K_t so that the system is stable.

(b) Let $r(t) = 0$. Determine how the values of K and K_t affect the steady-state value of $y(t)$ when the disturbance input $T_d(t) = u_s(t)$.

(c) Let $K_t = 0.01$ and $r(t) = 0$. Find the minimum steady-state value of $y(t)$ that can be obtained by varying K, when $T_d(t)$ is a unit-step function. Find the value of this K. From the transient standpoint, would you operate the system at this value of K? Explain.

(d) Assume that it is desired to operate the system with the value of K as selected in part (c). Find the value of K_t so that the complex roots of the characteristic equation will have a real part of −2.5. Find all three roots of the characteristic equation.

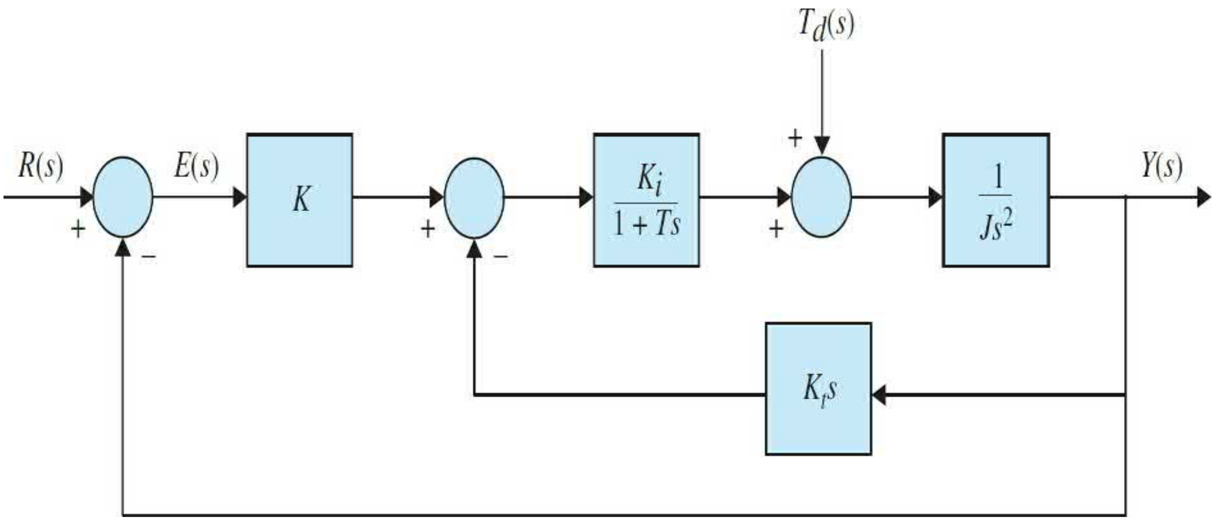


Figure 7P-32

7-33. Consider a second-order unity-feedback system with $\zeta = 0.6$ and $\omega_n = 5$ rad/s. Calculate the rise time, peak time, maximum overshoot, and settling time when a unit-step input is applied to the system.

7-34. Figure 7P-34 shows the block diagram of a servomotor. Assume $J = 1$ kg-m^2 and $B = 1$ N · m/rad/s. If the maximum overshoot of the unit-step input and the peak time are 0.2 and 0.1 s, respectively,

(a) Find its damping ratio and natural frequency.

(b) Find the gain K and velocity feedback K_f . Also, calculate the rise time and settling time.

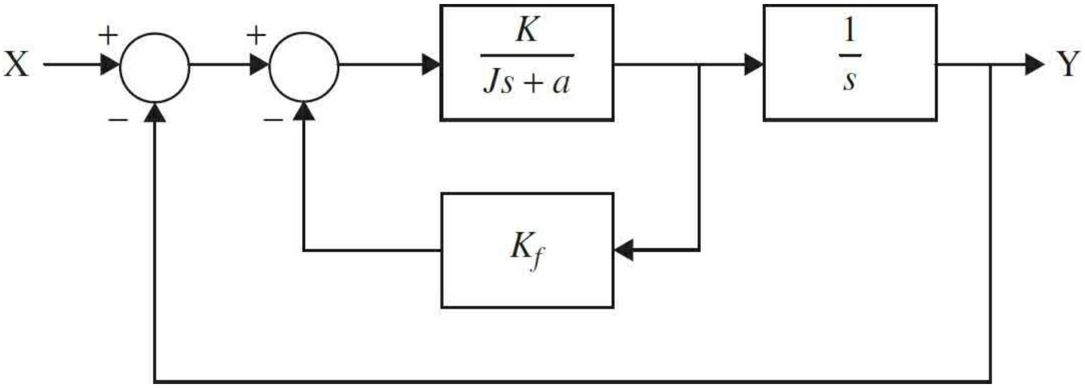


Figure 7P-34

7-35. Find the unit-step response of the following systems assuming zero initial conditions:

(a)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & -1 \\ 6.5 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

(b)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \qquad y_1 = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]u$$

(c)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} u \qquad y = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

7-36. Use MATLAB to solve Prob. 7-35.

7-37. Find the impulse response of the given systems in Prob. 7-35.

7-38. Use MATLAB to solve Prob. 7-37.

7-39. Figure 7P-39 shows a mechanical system.

(a) Find the differential equation of the system.

(b) Use MATLAB to find the unit-step input response of the system.

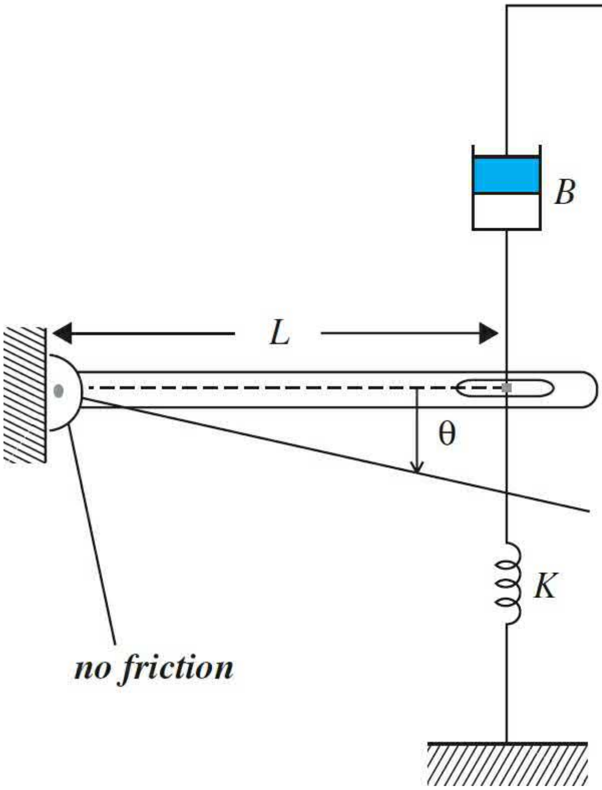


Figure 7P-39

7-40. The dc-motor control system for controlling a printwheel described in Prob. 4-49 has the forward-path transfer function

$$G(s) = \frac{\Theta_o(s)}{\Theta_e(s)} = \frac{nK_sK_iK_LK}{\Delta(s)}$$

$$\begin{aligned}\text{where } \Delta(s) = s\big[&L_aJ_mJ_Ls^4 + J_L(R_aJ_m + B_mL_a)s^3 \\ &+ (n^2K_LL_aJ_L + K_LL_aJ_m + K_iK_bJ_L + R_aB_mJ_L)s^2 \\ &+ (n^2R_aK_LJ_L + R_aK_LJ_m + B_mK_LL_a)s + R_aB_mK_L + K_iK_bK_L\big]\end{aligned}$$

where K_i = 9 oz-in./A, K_b = 0.636 V/rad/s, R_a = 5 Ω, L_a = 1 mH, K_s = 1 V/rad, n = 1/10, $J_m = J_L$ = 0.001 oz·in·s^2, and $B_m \cong 0$. The characteristic equation of the closed-loop system is

$$\Delta(s) + nK_sK_iK_LK = 0$$

(a) Let K_L = 10,000 oz·in·rad. Write the forward-path transfer function $G(s)$ and find the poles of $G(s)$. Find the critical value of K for the closed-loop system to be stable. Find the roots of the characteristic equation of the closed-loop system when K is at marginal stability.

(b) Repeat part (a) when K_L = 1000 oz·in/rad.

(c) Repeat part (a) when $K_L = \infty$; that is, the motor shaft is rigid.

(d) Compare the results of parts (a), (b), and (c), and comment on the effects of the values of K_L on the poles of $G(s)$ and the roots of the characteristic equation.

7-41. The block diagram of the guided-missile attitude-control system described in Prob. 4-20 is shown in Fig. 7P-41. The command input is $r(t)$, and $d(t)$ represents disturbance input. The objective of this problem is to study the effect of the controller $G_c(s)$ on the steady state and transient responses of the system.

(a) Let $G_c(s) = 1$. Find the steady-state error of the system when $r(t)$ is a unit-step function. Set $d(t) = 0$.

(b) Let $G_c(s) = (s + \alpha)/s$. Find the steady-state error when $r(t)$ is a unit-step function.

(c) Obtain the unit-step response of the system for $0 \le t \le 0.5$ s with $G_c(s)$ as given in part (b) and a = 5.50, and 500. Assume zero initial conditions. Record the maximum overshoot of $y(t)$ for each case. Use any available computer simulation program. Comment on the effect of varying the value of α of the controller on the transient response.

(d) Set $r(t) = 0$ and $G_c(s) = 1$. Find the steady-state value of $y(t)$ when $d(t) = u_s(t)$.

(e) Let $G_c(s) = (s + \alpha)/s$. Find the steady-state value of $y(t)$ when $d(t) = u_s(t)$.

(f) Obtain the output response for $0 \le t \le 0.5$ s, with $G_c(s)$ as given in part (e) when $r(t) = 0$ and $d(t) = u_s(t)$; α = 5.50, and 500. Use zero initial conditions.

(g) Comment on the effect of varying the value of α of the controller on the transient response of $y(t)$ and $d(t)$.

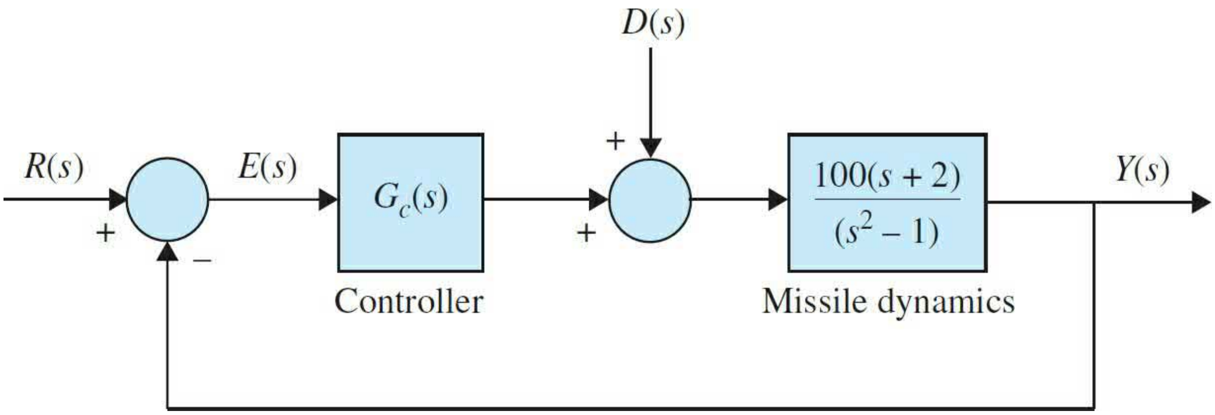


Figure 7P-41

7-42. The block diagram shown in Fig. 7P-42 represents the liquid-level control system described in Prob. 7-19. The liquid level is represented by $h(t)$, and N denotes the number of inlets.

(a) Because one of the poles of the open-loop transfer function is relatively far to the left on the real axis of the s-plane at $s = –10$, it is suggested that this pole can be neglected. Approximate the system by a second-order system be neglecting the pole of $G(s)$ at $s = –10$. The approximation should be valid for both the transient and the steady-state responses. Apply the formulas for the maximum overshoot and the peak time t_{max} to the second-order model for $N = 1$ and $N = 10$.

(b) Obtain the unit-step response (with zero initial conditions) of the original third-order system with $N = 1$ and then with $N = 10$. Compare the responses of the original system with those of the second-order approximating system. Comment on the accuracy of the approximation as a function of N.

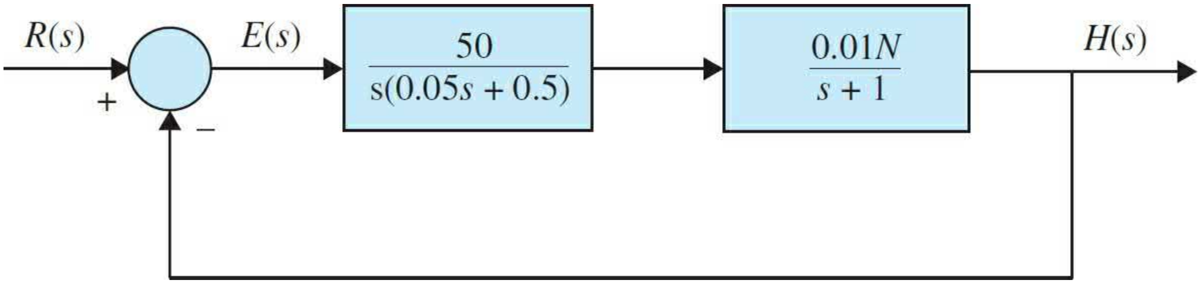


Figure 7P-42

7-43. The forward-path transfer function of a unity-feedback control system is

$$G(s)=\frac{1+T_z s}{s(s+1)^2}$$

Compute and plot the unit-step responses of the closed-loop system for T_z = 0, 0.5, 1.0, 10.0, and 50.0. Assume zero initial conditions. Use any computer simulation program that is available. Comment on the effects of the various values of T_z on the step response.

7-44. The forward-path transfer function of a unity-feedback control system is

$$G(s)=\frac{1}{s(s+1)^2(1+T_p s)}$$

Compute and plot the unit-step responses of the closed-loop system for T_p = 0, 0.5, and 0.707. Assume zero initial conditions. Use any computer simulation program. Find the critical value of T_p so that the closed-loop system is marginally stable. Comment on the effects of the pole at $s = -1/T_p$ in $G(s)$.

7-45. Compare and plot the unit-step responses of the unity-feedback closed-loop systems with the forward-path transfer functions given. Assume zero initial conditions. Use the timetool program.

(a) $$G(s)=\frac{1+T_z s}{s(s+0.55)(s+1.5)} \quad \text{For} \quad T_z=0,1,5,20$$

(b) $$G(s)=\frac{1+T_z s}{(s^2+2s+2)} \quad \text{For} \quad T_z=0,1,5,20$$

(c) $$G(s)=\frac{2}{(s^2+2s+2)(1+T_p s)} \quad \text{For} \quad T_p=0,0.5,1.0$$

(d) $$G(s)=\frac{10}{s(s+5)(1+T_p s)} \quad \text{For} \quad T_p=0,0.5,1.0$$

(e) $$G(s)=\frac{K}{s(s+1.25)(s^2+2.5s+10)}$$

(i) For $K = 5$

(ii) For $K = 10$

(iii) For $K = 30$

(f) $$G(s)=\frac{K(s+2.5)}{s(s+1.25)(s^2+2.5s+10)}$$

(i) For $K = 5$

(ii) For $K = 10$

(iii) For $K = 30$

7-46. Figure 7P-46 shows the block diagram of a servomotor with tachometer feedback.

(a) Find the error signal $E(s)$ in the presence of the reference input $X(s)$ and disturbance input $D(s)$.

(b) Calculate the steady-state error of the system when $X(s)$ is a unit ramp and D(s) is a unit-step.

(c) Use MATLAB to plot the response of the system for part (b).

(d) Use MATLAB to plot the response of the system when $X(s)$ is a unit-step input and $D(s)$ is a unit impulse input.

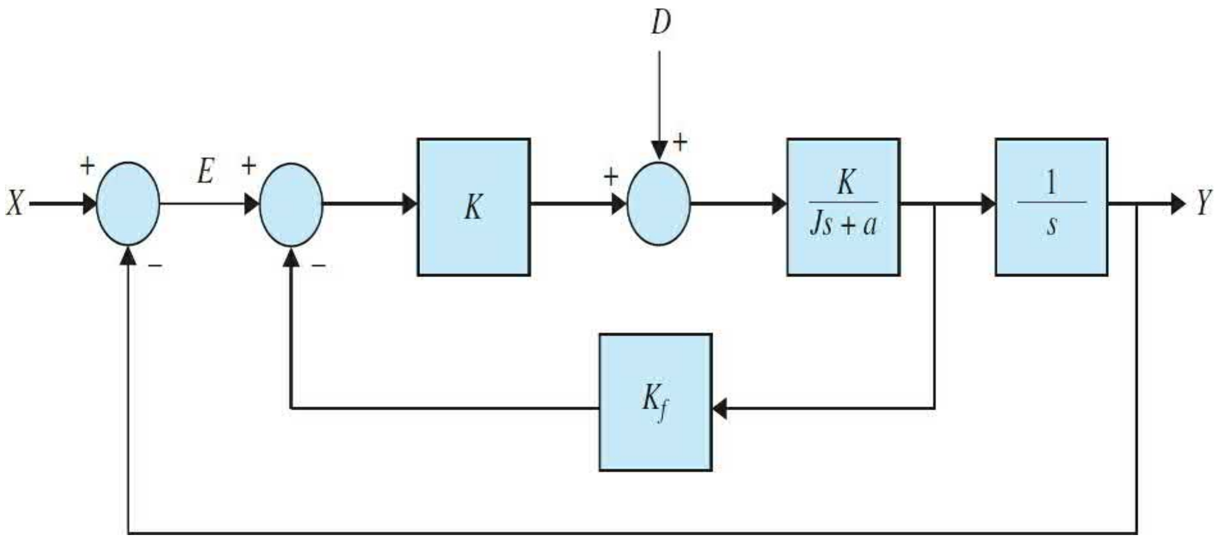


Figure 7P-46

7-47. The feedforward transfer function of a stable unity-feedback system is $G(s)$. If the closed-loop transfer function can be rewritten as

$$\frac{Y(s)}{X(s)}=\frac{G(s)}{1+G(s)}=\frac{(A_1s+1)(A_2s+1)\ldots(A_ns+1)}{(B_1s+1)(B_2s+1)\ldots(B_ms+1)}$$

(a) Find the $\int_0^\infty e(s)$ when $e(t)$ is the error in the unit-step response.

(b) Calculate $\frac{1}{K}=\frac{1}{\lim_{s\to 0}sG(s)}$

7-48. If the maximum overshoot and 1-percent settling time of the unit-step response of the closed-loop system shown in Fig. 7P-48 are no more that 25 percent and 0.1 s, find the gain K and pole location P of the compensator. Also, Use MATLAB to plot the unit-step input response of the system and verify your controller design.

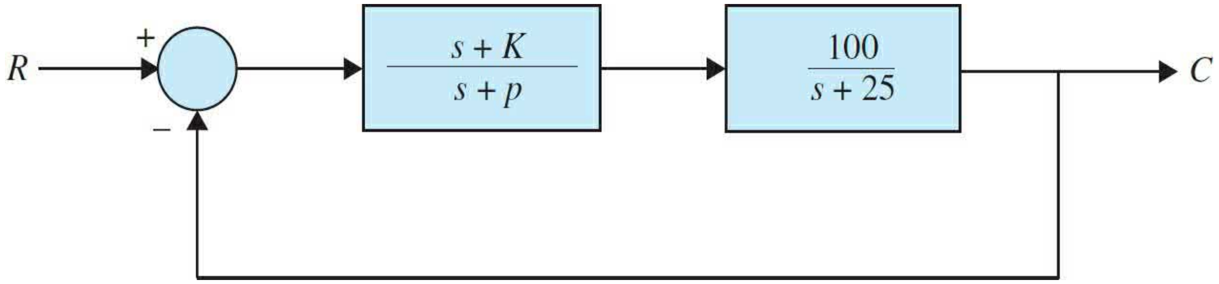


Figure 7P-48

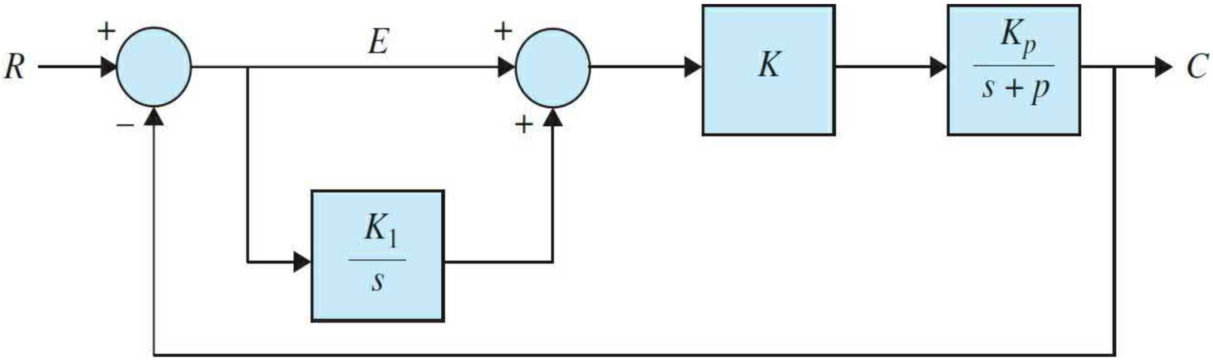
7-49. If a given second-order system is required to have a peak time less than *t*, find the region in the *s*-plane corresponding to the poles that meet this specification.

7-50. A unity-feedback control system shown in Fig. 7P-50*a* is designed so that its closed-loop poles lie within the region shown in Fig. 7P-50*b*.

(a) Find the values for ω_n and ζ.

(b) If $K_p = 2$ and $P = 2$, then find the values for K and KI.

(c) Show that, regardless of values K_p and P, the controller can be designed to place the poles anywhere in the left side of the *s*-plane.



(a)

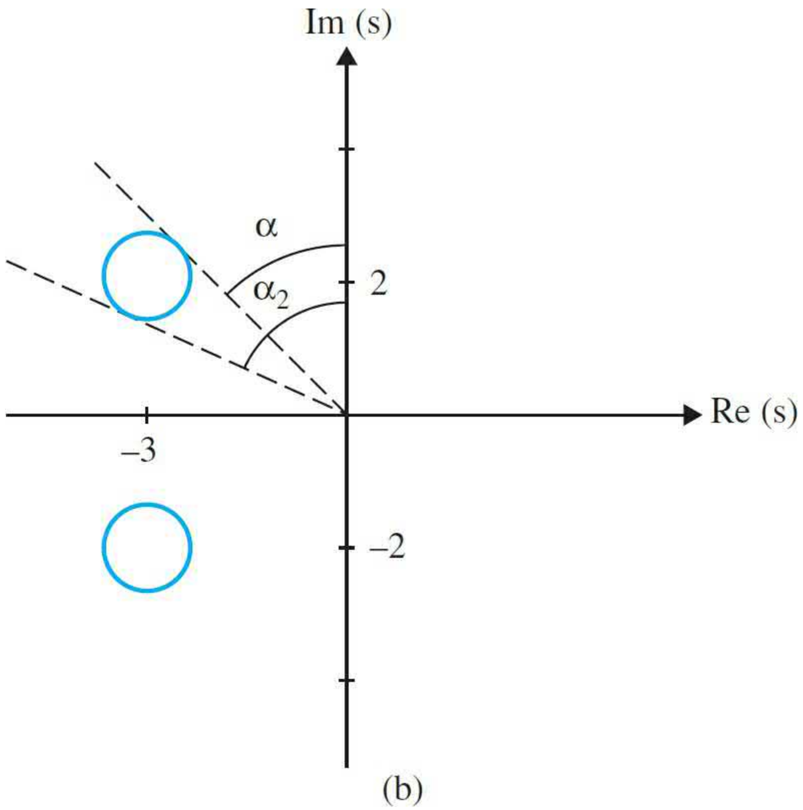


Figure 7P-50

7-51. The motion equation of a dc motor is given by

$$J_m \ddot{\theta}_m + \left(B + \frac{K_1 K_2}{R} \right) \ddot{\theta}_m = \frac{K_1}{R} v$$

Assuming $J_m = 0.02$ kg · m², $B = 0.002$ N · m · s, $K_1 = 0.04$ N · m/A, $K_2 = 0.04$ V · s, and $R = 20$ Ω.

(a) Find the transfer function between the applied voltage and the motor speed.

(b) Calculate the steady-state speed of the motor after applying a voltage of 10 V.

(c) Determine the transfer function between the applied voltage and the shaft angle θ_m.

(d) Including a closed-loop feedback to part (c) such that $v = K(\theta_p - \theta_m)$, where K is the feedback gain, obtain the transfer function between θ_p and θ_m.

(e) If the maximum overshoot is less than 25 percent, determine K.

(f) If the rise time is less than 3 s, determine *K*.

(g) Use MATLAB to plot the step response of the position servo system for *K* = 0.5, 1.0, and 2.0. Find the rise time and overshoot.

7-52. In the unity-feedback closed-loop system in a configuration similar to that in Fig. 7P-48, the plant transfer function is $G(s)=\frac{1}{s(s+3)}$

and the controller transfer function is

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

Design the controller parameters so that the closed-loop system has a 10 percent overshoot far a unit step input and a 1-percent settling time of 1.5 s.

7-53. An autopilot is designed to maintain the pitch attitude α of an airplane. The transfer function between pitch angle α and elevator angle β are given by

$$\frac{\alpha(s)}{\beta(s)}=\frac{60(s+1)(s+2)}{(s^2+6s+40)(s^2+0.04s+0.07)}$$

The autopilot pitch controller uses the pitch error *e* to adjust the elevator as

$$\frac{\beta_e(s)}{E(s)}=\frac{K(s+3)}{s+10}$$

Use MATLAB to find *K* with an overshoot of less than 10 percent and a rise time faster than 0.5 s for a unit-step input. Explain controller design difficulties for complex systems.

7-54. The block diagram of a control system with a series controller is shown in Fig. 7P-54. Find the transfer function of the controller $G_c(s)$ so that the following specifications are satisfied:

(a) The ramp-error constant K_v is 5.

(b) The closed-loop transfer function is of the form

$$M(s)=\frac{Y(s)}{R(s)}=\frac{K}{(s^2+20s+200)(s+a)}$$

where *K* and *a* are real constants. Use MATLAB to find the values of *K* and *a*.

The design strategy is to place the closed-loop poles at –10 + j10 and –10 – j10, and then adjust the values of K and a to satisfy the steady-state requirement. The value of a is large so that it will not affect the transient response appreciably. Find the maximum overshoot of the designed system.

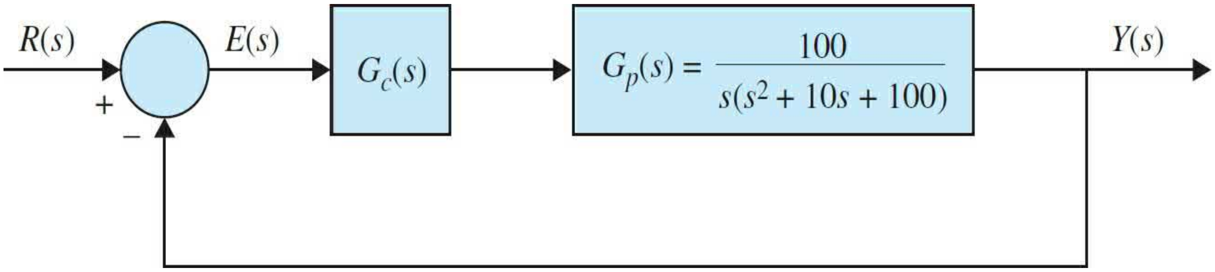


Figure 7P-54

7-55. Repeat Prob. 7-54 if the ramp-error constant is to be 9. What is the maximum value of K_v that can be realized? Comment on the difficulties that may arise in attempting to realize a very large K_v.

7-56. A control system with a PD controller is shown in Fig. 7P-56. Use MATLAB to

(a) Find the values of K_P and K_D so that the ramp-error constant K_v is 1000 and the damping ratio is 0:5.

(b) Find the values of K_P and K_D so that the ramp-error constant K_v is 1000 and the damping ratio is 0.707.

(c) Find the values of K_P and K_D so that the ramp-error constant K_v is 1000 and the damping ratio is 1.0.

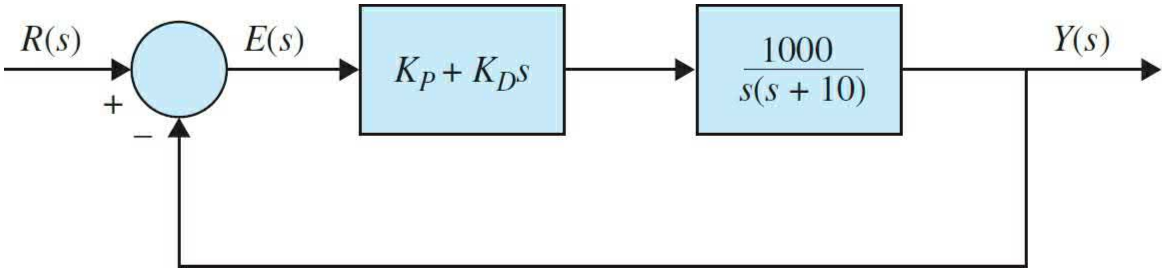


Figure 7P-56

7-57. For the control system shown in Fig. 7P-56, set the value of K_P so that the ramp-error constant is 1000. Use MATLAB to

(a) Vary the value of K_D from 0.2 to 1.0 in increments of 0.2 and determine the values of rise time and maximum overshoot of the system.

(b) Vary the value of K_D from 0.2 to 1.0 in increments of 0.2 and find the value of K_D so that the maximum overshoot is minimum.

7-58. Consider the second-order model of the aircraft attitude control system shown in Fig. 7-29. The transfer function of the process is $G_p(s) = \frac{4500K}{s(s+361.2)}$. Use MATLAB to design a series PD controller with the transfer function $G_c(s) = K_D + K_p s$ so that the following performance specifications are satisfied:

Steady-state error due to a unit-ramp input ≤ 0.001

Maximum overshoot ≤ 5%

Rise time $t_r \le 0.005$ sec

Setting time $t_s \le 0.005$ sec

7-59. Figure 7P-59 shows the block diagram of the liquid-level control system described in Prob. 7-42. The number of inlets is denoted by N. Set N = 20. Use MATLAB to design the PD controller so that with a unit-step input the tank is filled to within 5 percent of the reference level in less than 3 s without overshoot.

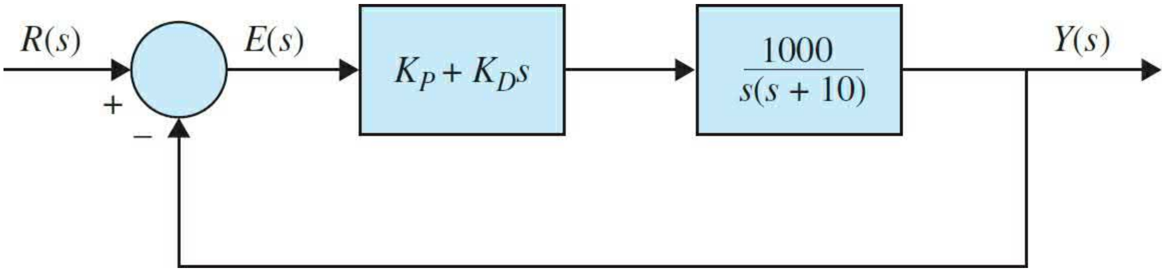


Figure 7P-59

7-60. For the liquid-level control system described in Prob. 7-59, set K_P so that the ramp-error constant is 1. Use MATLAB to vary K_D from 0 to 0.5 and determine the values of rise time and maximum overshoot of the system.

7-61. A control system with a type 0 process $G_p(s)$ and a PI controller is shown in Fig. 7P-61. Use MATLAB to

(a) Find the value of K_I so that the ramp-error constant K_v is 10.

(b) Find the value of K_P so that the magnitude of the imaginary parts of the complex roots of the characteristic equation of the system is 15 rad/s. Find the roots of the characteristic equation.

(c) Sketch the root contours of the characteristic equation with the value of K_I as determined in part (a) and for $0 \le K_p < \infty$.

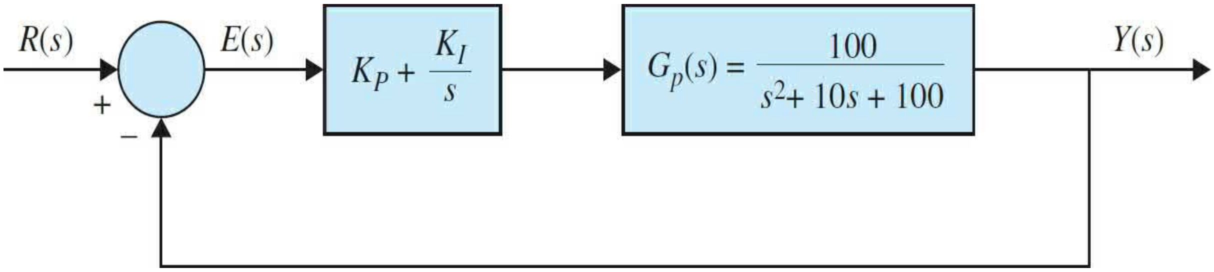


Figure 7P-61

7-62. For the control system described in Prob. 7-61, set K_I so that the ramp-error constant is 10. Use MATLAB to vary K_P and determine the values of rise time and maximum overshoot of the system.

7-63. For the control system shown in Fig. 7P-61, use MATLAB to perform the following:

(a) Find the value of K_I so that the ramp-error constant K_v is 100.

(b) With the value of K_I found in part (a), find the critical value of K_P so that the system is stable. Sketch the root contours of the characteristic equation for $0 \le K_p < \infty$.

(c) Show that the maximum overshoot is high for both large and small values of K_P. Use the value of K_I found in part (a). Find the value of K_P when the maximum overshoot is a minimum. What is the value of this maximum overshoot?

7-64. Repeat Prob. 7-63 for $K_p = 10$.

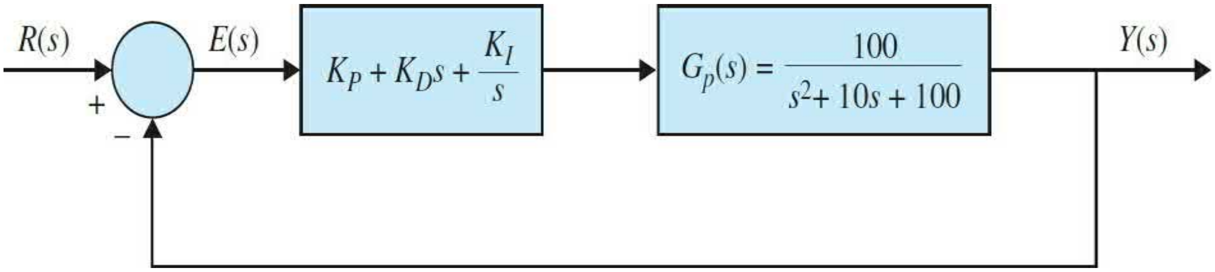
7-65. A control system with a type 0 process and a PID controller is shown in Fig. 7P-65. Use MATLAB to design the controller parameters so that the following specifications are satisfied:

Ramp-error constant $K_v = 100$

Rise time $t_r \leq 0.01$ sec

Maximum overshoot ≤ 2%

Plot the unit-step response of the designed system.



$R(s)$
$E(s)$
+
−
$K_P + K_D s + \frac{K_I}{s}$
$G_p(s) = \frac{100}{s^2 + 10s + 100}$
$Y(s)$

Figure 7P-65

7-66. Consider the quarter-car model of vehicle suspension systems in Prob. 2-8, for the following system parameters:

Effective ¼ car mass	10 kg
Effective stiffness	2.7135 N/m
Effective damping	$0.9135\ \text{N} \cdot \text{m/s}^{-1}$
Absolute displacement of the mass *m*	In meters
Absolute displacement of the base	In meters
Relative displacement $(x(t) - y(t))$	In meters

the equation of motion of the system is defined as follows:

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = c\dot{y}(t) + ky(t)$$

which can be simplified by substituting the relation $z(t) = x(t) - y(t)$ and nondimensionalizing the coefficients to the form

$$\ddot{z}(t) + 2\zeta\omega_n\dot{z}(t) + \omega_n{}^2 z(t) = -\ddot{y}(t)$$

The Laplace transform between the base acceleration and displacement is given by

$$\frac{Z(s)}{\ddot{Y}}(s) = \frac{-1}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

(a) It is desired to design a proportional controller. Use MATLAB to design the controller parameters where the rise time is no more than 0.05 s and the overshoot is no more than 3 percent. Plot the unit-step response of the designed system.

(b) It is desired to design a PD controller. Use MATLAB to design the controller parameters where the rise time is no more than 0.05 s and the overshoot is no more than 3 percent. Plot the unit-step response of the designed system.

(c) It is desired to design a PI controller. Use MATLAB to design the controller parameters where the rise time is no more than 0.05 s and the overshoot is no more than 3 percent. Plot the unit-step response of the designed system.

(d) It is desired to design a PID controller. Use MATLAB to design the controller parameters where the rise time is no more than 0.05 s and the overshoot is no more than 3 percent. Plot the unit-step response of the designed system.

7-67. Consider the spring-mass system shown in Fig. 7P-67.

Its transfer function is given by $\ddot{z}(t) + 2\zeta\omega_n \dot{z}(t) + \omega_n{}^2 z(t) = -\ddot{y}(t)$.

Repeat Prob. 7-66, where M = 1 kg, B = 10 N · s/m, K = 20 N/m.

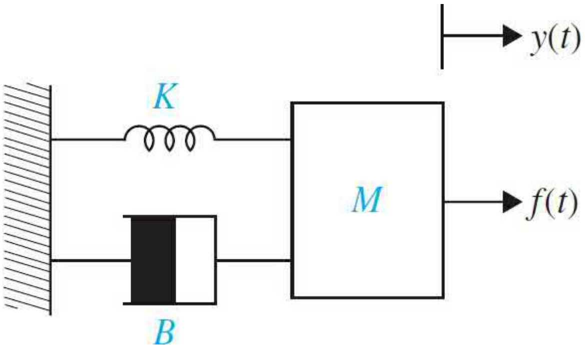


Figure 7P-67

7-68. Consider the vehicle suspension system hitting a bump described in Prob. 4-3. Use MATLAB to design a proportional controller where the 1-percent settling time is less than 0.1 s and the overshoot is no more than 2-

percent. Assume m = 25 kg, J = 5 kg·m^2, K = 100 N/m, and r = 0.35 m. Plot the impulse response of the system.

7-69. Consider the train system described in Prob. 4-6. Use MATLAB to design a proportional controller where the peak time is less than 0.05 s and the overshoot is no more than 4 percent. Assume M = 1 kg, m = 0.5 kg, k = 1 N/m, μ = 0.002 s/m, and g = 9.8 m/s^2.

7-70. Consider the inverted pendulum described in Prob. 4-9, where M = 1 kg, m = 0.2 kg, μ = 0.1 N/m/s (friction of the cart), I = 0.006 kg·m^2, g = 9.8 m/s^2, and 1 = 0.3 m.

Use MATLAB to design a PD controller where the rise time is less than 0.2 s and the overshoot is no more than 10 percent.

CHAPTER 8

State-Space Analysis and Controller Design

8-1 STATE-VARIABLE ANALYSIS

In Chaps. 2 and 3, we presented the concept and definition of state variables and state equations for linear continuous-data dynamic systems. In Chap. 4, we used block diagrams and signal-flow graphs (SFGs) to obtain the transfer function of linear systems. We further extended the SFG concept to the modeling of the state equations, and the result was the **state diagram**. In contrast to the transfer-function approach to the analysis and design of linear control systems, the state-variable method is regarded as modern, and it is the basis for optimal control design. The basic characteristic of the state-variable formulation is that linear and nonlinear systems, time-invariant and time-varying systems, and single-variable and multivariable systems can all be modeled in a unified manner. Transfer functions, on the other hand, are defined only for linear time-invariant systems.

Learning Outcomes

After successful completion of this chapter, you will be able to

1. Gain a working knowledge of the state-space approach.
2. Use transformations that are used to facilitate the analysis and design of linear control systems in the state-variable domain.
3. Establish relationships between the conventional transfer functions and the state variables.
4. Utilize controllability and observability of linear systems and their applications.
5. Gain a practical sense of real life control problems, through the use

of **LEGO MINDSTORMS,** and **MATLAB** tools.

The objective of this chapter is to introduce the basic methods of state variables and state equations so that the reader can gain a working knowledge of the subject for further studies when the state-space approach is used for modern and optimal control design. Specifically, the closed-form solutions of linear time-invariant state equations are presented. Various transformations that may be used to facilitate the analysis and design of linear control systems in the state-variable domain are introduced. The relationship between the conventional transfer-function approach and the state-variable approach is established so that the analyst will be able to investigate a system problem with various alternative methods. The controllability and observability of linear systems are defined and their applications investigated. In the end, we provide **state-space controller design problems**, followed by a case study involving the **LEGO MINDSTORMS NXT** set that was earlier studied in Chaps. 2 and 7. At the end of the chapter, we also present our MATLAB Automatic Control Systems (ACSYS) State Tool that can help you solve most state-space problems.

8-2 BLOCK DIAGRAMS, TRANSFER FUNCTIONS, AND STATE DIAGRAMS

8-2-1 Transfer Functions (Multivariable Systems)

The definition of a transfer function is easily extended to a system with multiple inputs and outputs. A system of this type is often referred to as a multivariable system. As discussed in Chap. 3, in a multivariable system, a differential equation of the form of Eq. (8-1) may be used to describe the relationship between a pair of input and output variables, when all other inputs are set to zero.

$$\frac{d^n y(t)}{dt^n}+a_{n-1}\frac{d^{n-1}y(t)}{dt^{n-1}}+\cdots+a_1\frac{dy(t)}{dt}+a_0 y(t)$$
$$=b_m\frac{d^m u(t)}{dt^m}+b_{m-1}\frac{d^{m-1}u(t)}{dt^{m-1}}+\cdots+b_1\frac{du(t)}{dt}+b_0 u(t) \qquad (8\text{-}1)$$

The coefficients a_0, a_1, …, a_{n-1} and b_0, b_1, …, b_m are real constants. Because the principle of superposition is valid for linear systems, the total effect on any output due to all the inputs acting simultaneously is obtained by adding up the outputs due to each input acting alone.

In general, if a linear system has p inputs and q outputs, the transfer function between the jth input and the ith output is defined as

$$G_{ij}(s)=\frac{Y_i(s)}{R_j(s)} \qquad (8\text{-}2)$$

with $R_k(s)=0, k=1,2,\ldots,p, k\neq j$. Note that Eq. (8-2) is defined with only the jth input in effect, whereas the other inputs are set to zero. When all the p inputs are in action, the ith output transform is written

$$Y_i(s)=G_{i1}(s)R_1(s)+G_{i2}(s)R_2(s)+\cdots+G_{ip}(s)R_p(s) \qquad (8\text{-}3)$$

It is convenient to express Eq. (8-3) in matrix-vector form:

$$\mathbf{Y}(s)=\mathbf{G}(s)\mathbf{R}(s) \qquad (8\text{-}4)$$

where

$$\mathbf{Y}(s)=\begin{bmatrix} Y_1(s) \\ Y_2(s) \\ \vdots \\ Y_q(s) \end{bmatrix} \qquad (8\text{-}5)$$

is the q × 1 transformed output vector,

$$\mathbf{R}(s)=\begin{bmatrix} R_1(s) \\ R_2(s) \\ \vdots \\ R_p(s) \end{bmatrix} \tag{8-6}$$

is the $p \times 1$ transformed input vector, and

$$\mathbf{G}(s)=\begin{bmatrix} G_{11}(s) & G_{12}(s) & \cdots & G_{1p}(s) \\ G_{21}(s) & G_{22}(s) & \cdots & G_{2p}(s) \\ \cdot & \cdot & \cdots & \cdot \\ G_{q1}(s) & G_{q2}(s) & \cdots & G_{qp}(s) \end{bmatrix} \tag{8-7}$$

is the $q \times p$ transfer-function matrix.

8-2-2 Block Diagrams and Transfer Functions of Multivariable Systems

In this section, we illustrate the block diagram and matrix representations of multivariable systems. Two block-diagram representations of a multivariable system with p inputs and q outputs are shown in Fig. 8-1*a* and *b*. In Fig. 8-1*a*, the individual input and output signals are designated, whereas in the block diagram of Fig. 8-1*b*, the multiplicity of the inputs and outputs is denoted by vectors. The case of Fig. 8-1*b* is preferable in practice because of its simplicity.

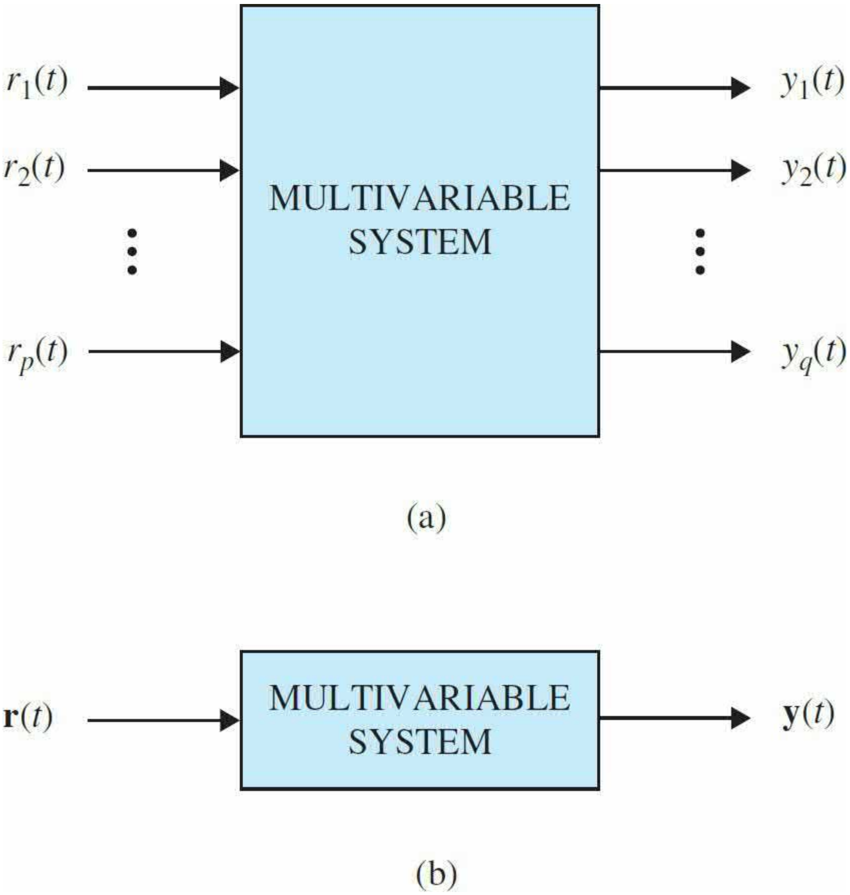


Figure 8-1 Block diagram representations of a multivariable system.

Figure 8-2 shows the block diagram of a multivariable feedback control system. The transfer function relationships of the system are expressed in vector-matrix form (see Sec. 8-3 for more detail):

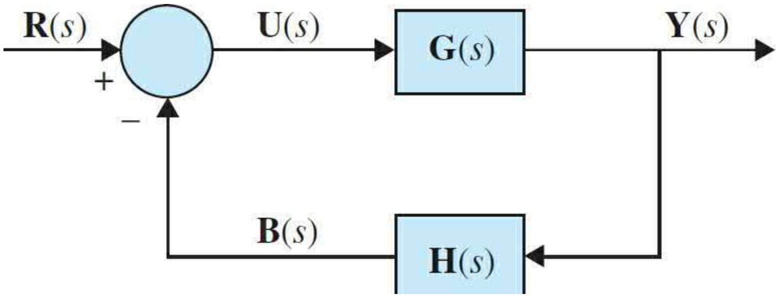


Figure 8-2 Block diagram of a multivariable feedback control system.

$$\mathbf{Y}(s) = \mathbf{G}(s)\mathbf{U}(s) \tag{8-8}$$

$$\mathbf{U}(s) = \mathbf{R}(s) - \mathbf{B}(s) \tag{8-9}$$

$$\mathbf{B}(s)=\mathbf{H}(s)\mathbf{Y}(s) \tag{8-10}$$

where **Y**(*s*) is the *q* × 1 output vector; **U**(*s*), **R**(*s*), and **B**(*s*) are all *p* × 1 vectors; and **G**(*s*) and **H**(*s*) are *q* × *p* and *p* × *q* transfer-function matrices, respectively. Substituting Eq. (8-9) into Eq. (8-8) and then from Eq. (8-8) to Eq. (8-10), we get

$$\mathbf{Y}(s)=\mathbf{G}(s)\mathbf{R}(s)-\mathbf{G}(s)\mathbf{H}(s)\mathbf{Y}(s) \tag{8-11}$$

Solving for **Y**(*s*) from Eq. (8-11) gives

$$\mathbf{Y}(s)=[\mathbf{I}+\mathbf{G}(s)\mathbf{H}(s)]^{-1}\mathbf{G}(s)\mathbf{R}(s) \tag{8-12}$$

provided that **I+G(s)H(s)** is nonsingular. The closed-loop transfer matrix is defined as

$$\mathbf{M}(s)=[\mathbf{I}+\mathbf{G}(s)\mathbf{H}(s)]^{-1}\mathbf{G}(s) \tag{8-13}$$

Then Eq. (8-12) is written

$$\mathbf{Y}(s)=\mathbf{M}(s)\mathbf{R}(s) \tag{8-14}$$

EXAMPLE 8-2-1 Consider that the forward-path transfer function matrix and the feedback-path transfer function matrix of the system shown in Fig. 8-2 are

$$\mathbf{G}(s)=\begin{bmatrix} \dfrac{1}{s+1} & -\dfrac{1}{s} \\ 2 & \dfrac{1}{s+2} \end{bmatrix} \quad \mathbf{H}(s)=\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{8-15}$$

respectively. The closed-loop transfer function matrix of the system is given by Eq. (8-14) and is evaluated as follows:

$$\mathbf{I}+\mathbf{G}(s)\mathbf{H}(s)=\begin{bmatrix} 1+\dfrac{1}{s+1} & -\dfrac{1}{s} \\ 2 & 1+\dfrac{1}{s+2} \end{bmatrix}=\begin{bmatrix} \dfrac{s+2}{s+1} & -\dfrac{1}{s} \\ 2 & \dfrac{s+3}{s+2} \end{bmatrix} \tag{8-16}$$

The closed-loop transfer function matrix is

$$\mathbf{M}(s)=[\mathbf{I}+\mathbf{G}(s)\mathbf{H}(s)]^{-1}\mathbf{G}(s)=\frac{1}{\Delta}\begin{bmatrix} \dfrac{s+3}{s+2} & \dfrac{1}{s} \\ -2 & \dfrac{s+2}{s+1} \end{bmatrix}\begin{bmatrix} \dfrac{1}{s+1} & -\dfrac{1}{s} \\ 2 & \dfrac{1}{s+2} \end{bmatrix} \tag{8-17}$$

where

$$\Delta=\frac{s+2}{s+1}\frac{s+3}{s+2}+\frac{2}{s}=\frac{s^2+5s+2}{s(s+1)} \tag{8-18}$$

Thus,

$$\mathbf{M}(s)=\frac{s(s+1)}{s^2+5s+2}\begin{bmatrix} \dfrac{3s^2+9s+4}{s(s+1)(s+2)} & -\dfrac{1}{s} \\ 2 & \dfrac{3s+2}{s(s+1)} \end{bmatrix} \tag{8-19}$$

8-3 SYSTEMS OF FIRST-ORDER DIFFERENTIAL EQUATIONS: STATE EQUATIONS

As discussed in Chap. 3, state equations provide an alternative to the transfer function approach, discussed earlier, to study differential equations. This technique particularly provides a powerful means to treat and analyze higher-order differential equations, and is highly utilized in modern control theory and more advanced topics in control systems, such as optimal control design.

In general, an *n*th-order differential equation can be decomposed into *n* first-order differential equations. Because, in principle, first-order differential equations are simpler to solve than higher-order ones, first-order differential equations are used in the analytical studies of control systems.

For Eq. (8-1), if we define

$$
\begin{aligned}
x_1(t) &= y(t) \\
x_2(t) &= \frac{dy(t)}{dt} \\
&\vdots \\
x_n(t) &= \frac{d^{n-1}y(t)}{dt^{n-1}}
\end{aligned}
\tag{8-20}
$$

then the *n*th-order differential equation is decomposed into *n* first-order differential equations:

$$
\begin{aligned}
\frac{dx_1(t)}{dt} &= x_2(t) \\
\frac{dx_2(t)}{dt} &= x_3(t) \\
&\vdots \\
\frac{dx_n(t)}{dt} &= -a_0x_1(t) - a_1x_2(t) - \cdots - a_{n-2}x_{n-1}(t) - a_{n-1}x_n(t) + f(t)
\end{aligned}
\tag{8-21}
$$

Notice that the last equation is obtained by equating the highest-ordered derivative term in Eq. (8-1) to the rest of the terms. In control systems theory, the set of first-order differential equations in Eq. (8-21) is called the **state equations**, and x_1, x_2,…, x_n are called the **state variables**. Finally, *the minimum number of state variables needed is usually the same as the order n of the differential equation of the system*.

8-3-1 Definition of State Variables

The state of a system refers to the past, present, and future conditions of the system. From a mathematical perspective, it is convenient to define a set of state variables and state equations to model dynamic systems. As stated earlier, the variables $x_1(t)$, $x_2(t)$,…, $x_n(t)$ defined in Eq. (8-20) are the **state variables** of the *n*th-order system described by Eq. (8-1), and the *n* first-order differential equations, in Eq. (8-21), are the **state equations**. In general, there are some basic rules regarding the definition of a state variable and what constitutes a state equation. The state variables must satisfy the following conditions:

The minimum number of state variables needed to represent a differential equation, is usually the same as the order of the differential equation of the system.

- At any initial time $t = t_0$, the state variables $x_1(t_0)$, $x_2(t_0)$,…, $x_n(t_0)$ define the **initial states** of the system.
- Once the inputs of the system for $t \geq t_0$ and the initial states just defined are specified, the state variables should completely define the future behavior of the system.

The state variables of a system are defined as a **minimal set** of variables, $x_1(t)$, $x_2(t)$,…, $x_n(t)$, such that knowledge of these variables at any time t_0 and information on the applied input at time t_0 are sufficient to determine the state of the system at any time $t \geq t_0$. Hence, the **state-space form** for *n* state variables is

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{8-22}$$

where **x(t)** is the state vector having *n* rows,

$$\mathbf{x}(t)=\begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix} \tag{8-23}$$

and **u(t)** is the input vector with *p* rows,

$$\mathbf{u}(t)=\begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_p(t) \end{bmatrix} \tag{8-24}$$

The coefficient matrices **A** and **B** are defined as

$$\mathbf{A}=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}(n\times n) \tag{8-25}$$

$$\mathbf{B}=\begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix}(n\times p) \tag{8-26}$$

8-3-2 The Output Equation

One should not confuse the state variables with the outputs of a system. An **output** of a system is a variable that *can be measured*, but a state variable does not always need to satisfy this requirement. For instance, in an electric motor, such state variables as the winding current, rotor velocity, and displacement can be measured physically, and these variables all qualify as output variables. On the other hand, magnetic flux can also be regarded as a state variable in an electric motor because it represents the past, present, and future states of the motor, but it cannot be measured directly during operation and therefore does not ordinarily qualify as an output variable. In general, an output variable can be expressed as an algebraic combination of the state variables. For the system described by Eq. (8-1), if $y(t)$ is designated as the output, then the output equation is simply $y(t) = x_1(t)$. In general,

$$\mathbf{y}(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_q(t) \end{bmatrix} = \mathbf{Cx}(t) + \mathbf{Du}(t) \tag{8-27}$$

$$\mathbf{C} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qn} \end{bmatrix} \tag{8-28}$$

$$\mathbf{D} = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1p} \\ d_{21} & d_{22} & \cdots & d_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ d_{q1} & d_{q2} & \cdots & d_{qp} \end{bmatrix} \tag{8-29}$$

We will utilize these concepts in the modeling of various dynamical systems next.

EXAMPLE 8-3-1 Consider the second-order differential equation, which was also studied in Example 3-4-1,

$$\frac{d^2y(t)}{dt^2}+3\frac{dy(t)}{dt}+2y(t)=2u(t) \tag{8-30}$$

If we let

$$\begin{aligned} x_1(t)&=y(t) \\ x_2(t)&=\frac{dx_1(t)}{dt}=\frac{dy(t)}{dt} \end{aligned} \tag{8-31}$$

then Eq. (8-30) is decomposed into the following two first-order differential equations:

$$\frac{dx_1(t)}{dt}=x_2(t) \tag{8-32}$$

$$\frac{dx_2(t)}{dt}=-2x_1(t)-3x_2(t)+2u(t) \tag{8-33}$$

where $x_1(t)$, $x_2(t)$ are the state variables, and $u(t)$ is the input, we can—at this point arbitrarily—define $y(t)$ as the output represented by

$$y(t)=x_1(t) \tag{8-34}$$

In this case we are simply interested in state variable $x_1(t)$ to be our output. As a result,

$$\begin{gathered} \mathbf{x}(t)=\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}; \quad \mathbf{u}(t)=u(t) \\ \mathbf{A}=\begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}; \quad \mathbf{B}=\begin{bmatrix} 0 \\ 2 \end{bmatrix}; \quad \mathbf{C}=\begin{bmatrix} 1 & 0 \end{bmatrix}; \quad \mathbf{D}=0 \end{gathered} \tag{8-35}$$

8-4 VECTOR-MATRIX REPRESENTATION OF STATE EQUATIONS

Let the *n* state equations of an *n*th-order dynamic system be represented as

$$\frac{dx_1(t)}{dt} = f_i[x_1(t), x_2(t), \ldots, x_n(t), u_1(t), u_2(t), \ldots, u_p(t), w_1(t), w_2(t), \ldots, w_v(t)] \tag{8-36}$$

where i =1,2,....,n. The *i*th state variable is represented by $x_i(t)$; $u_j(t)$ denotes the *j*th input for *j* = 1, 2,...,*p*; and w_k(37) denotes the *k*th disturbance input, with k = 1, 2,...,v.

Let the variables y_1(t),y_2 (t),....,$y_q$be the *q* output variables of the system. In general, the output variables are functions of the state variables and the input variables. The **output equations** can be expressed as

$$y_j(t) = g_j[x_1(t), x_2(t), \ldots, x_n(t), u_1(t), u_2(t), \ldots, u_p(t), w_1(t), w_2(t), \ldots, w_v(t)] \tag{8-37}$$

where *j* = 1, 2, …, *q*.

The set of *n* state equations in Eq. (8-36) and *q* output equations in Eq. (8-37) together form the **dynamic equations**. For ease of expression and manipulation, it is convenient to represent the dynamic equations in vector-matrix form. Let us define the following vectors:

State vector:

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix} (n \times 1) \tag{8-38}$$

Input vector:

$$\mathbf{u}(t)=\begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_p(t) \end{bmatrix}(p\times 1) \tag{8-39}$$

Output vector:

$$\mathbf{y}(t)=\begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_q(t) \end{bmatrix}(q\times 1) \tag{8-40}$$

Disturbance vector:

$$\mathbf{w}(t)=\begin{bmatrix} w_1(t) \\ w_2(t) \\ \vdots \\ w_v(t) \end{bmatrix}(v\times 1) \tag{8-41}$$

Note: In most textbooks on this subject, the disturbance vector is considered as—and for simplicity it is absorbed into—the input vector.

By using these vectors, the *n* state equations of Eq. (8-36) can be written

$$\frac{d\mathbf{x}(t)}{dt}=\mathbf{f}[\mathbf{x}(t),\mathbf{u}(t),\mathbf{w}(t)] \tag{8-42}$$

where **f** denotes an *n* × 1 column matrix that contains the functions $f_1, f_2, \ldots. f_n$ as elements. Similarly, the *q* output equations in Eq. (8-37) become

$$\mathbf{y}(t)=\mathbf{g}[\mathbf{x}(t),\mathbf{u}(t),\mathbf{w}(t)] \tag{8-43}$$

where **g** denotes a $q \times 1$ column matrix that contains the functions g_1, g_2, …, g_q as elements.

For a linear time-invariant system, the dynamic equations are written as

State equations:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{E}\mathbf{w}(t) \tag{8-44}$$

Output equations:

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) + \mathbf{H}\mathbf{w}(t) \tag{8-45}$$

where

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} (n \times n) \tag{8-46}$$

$$\mathbf{B} = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} (n \times p) \tag{8-47}$$

$$\mathbf{C} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qn} \end{bmatrix} (q \times n) \tag{8-48}$$

$$\mathbf{D} = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1p} \\ d_{21} & d_{22} & \cdots & d_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ d_{q1} & d_{q2} & \cdots & d_{qp} \end{bmatrix} (q \times p) \tag{8-49}$$

$$\mathbf{E} = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1v} \\ e_{21} & e_{22} & \cdots & e_{2v} \\ \vdots & \vdots & \ddots & \vdots \\ e_{n1} & e_{n2} & \cdots & e_{nv} \end{bmatrix} (n \times v) \tag{8-50}$$

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1v} \\ h_{12} & h_{22} & \cdots & h_{2v} \\ \vdots & \vdots & \ddots & \vdots \\ h_{q1} & h_{q2} & \cdots & h_{qv} \end{bmatrix} (q \times v) \tag{8-51}$$

8-5 STATE-TRANSITION MATRIX

Once the state equations of a linear time-invariant system are expressed in the form of Eq. (8-44), the next step often involves the solutions of these equations given the initial state vector $\mathbf{x}(t_0)$, the input vector $\mathbf{u}(t)$, and the disturbance vector $\mathbf{w}(t)$, for $t \geq t_0$. The first term on the right-hand side of Eq. (8-44) is known as the homogeneous part of the state equation, and the last two terms represent the forcing functions $\mathbf{u}(t)$ and $\mathbf{w}(t)$.

The **state-transition matrix** is defined as a matrix that satisfies the linear homogeneous state equation:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) \tag{8-52}$$

Let ϕ (t) *be the* $n \times n$ matrix that represents the state-transition matrix; then it must satisfy the equation

$$\frac{d\phi(t)}{dt} = \mathbf{A}\phi(t) \tag{8-53}$$

Furthermore, let $\mathbf{x}(0)$ denote the initial state at $t = 0$; then ϕ (t) is also defined by the matrix equation

$$\mathbf{x}(t) = \phi(t)\mathbf{x}(0) \tag{8-54}$$

which is the solution of the homogeneous state equation for $t \geq 0$.

One way of determining ϕ (t) is by taking the Laplace transform on both sides of Eq. (8-52), we have

$$s\mathbf{X}(s) - \mathbf{x}(0) = \mathbf{A}\mathbf{X}(s) \tag{8-55}$$

Solving for $\mathbf{X}(s)$ from Eq. (8-55), we get

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{x}(0) \tag{8-56}$$

where it is assumed that the matrix (s$\mathbf{I}$ − $\mathbf{A}$) is nonsingular. Taking the inverse Laplace transform on both sides of Eq. (8-56) yields

$$\mathbf{x}(t) = \mathcal{L}^{-1}[(s\mathbf{I} - \mathbf{A})^{-1}]\mathbf{x}(0) \quad t \geq 0 \tag{8-57}$$

By comparing Eq. (8-54) with Eq. (8-57), the state-transition matrix is identified to be

$$\phi(t) = {}^{-1}\,[(s\mathbf{I} - \mathbf{A})^{-1}] \tag{8-58}$$

An alternative way of solving the homogeneous state equation is to assume a solution, as in the classical method of solving linear differential equations. We let the solution to Eq. (8-52) be

$$\mathbf{x}(t) = e^{\mathbf{A}t}\mathbf{x}(0) \tag{8-59}$$

for $t \geq 0$, where e^{At} represents the following power series of the matrix $\mathbf{A}t$, and

$$e^{\mathbf{A}t} = \mathbf{I} + \mathbf{A}\mathbf{t} + \frac{1}{2!}\mathbf{A}^2 t^2 + \frac{1}{3!}\mathbf{A}^3 t^3 + \cdots \tag{8-60}$$

It is easy to show that Eq. (8-59) is a solution of the homogeneous state equation, since, from Eq. (8-60),

$$\frac{de^{\mathbf{A}t}}{dt} = \mathbf{A}e^{\mathbf{A}t} \tag{8-61}$$

Therefore, in addition to Eq. (8-58), we have obtained another expression for the state-transition matrix:

$$\phi(t) = \mathbf{e}^{\mathbf{A}t} = \mathbf{I} + \mathbf{A}t + \frac{1}{2!}\mathbf{A}^2 t^2 + \frac{1}{3!}\mathbf{A}^3 t^3 + \cdots \tag{8-62}$$

Equation (8-62) can also be obtained directly from Eq. (8-58). This is left as an exercise for the reader (Prob. 8-5).

8-5-1 Significance of the State-Transition Matrix

Because the state-transition matrix satisfies the homogeneous state equation, it represents the **free response** of the system. In other words, it governs the response that is excited by the initial conditions only. In view of Eqs. (8-58) and (8-62), the state-transition matrix is dependent only upon the matrix **A** and, therefore, is sometimes referred to as the **state-transition matrix of A**. As the name implies, the state-transition matrix ϕ (*t*) completely defines the transition of the states from the initial time *t* = 0 to any time *t* when the inputs are zero.

8-5-2 Properties of the State-Transition Matrix

The state-transition matrix ϕ (*t*) possesses the following properties:

1.

$$\phi(0) = \mathrm{I} \quad \text{(the identity matrix)} \tag{8-63}$$

Proof: Equation (8-63) follows directly from Eq. (8-62) by setting *t* = 0.

2.

$$\phi^{-1}(t) = \phi(-t) \tag{8-64}$$

Proof: Postmultiplying both sides of Eq. (8-65) by $e{-}^{\mathbf{A}t}$, we get

$$\phi(t)e^{-\mathbf{A}t} = \mathbf{e}^{\mathbf{A}t}\mathbf{e}^{-\mathbf{A}t} = \mathrm{I} \tag{8-65}$$

Then, pre-multiplying both sides of Eq. (8-65) by $\phi^{-1}(t)$, we get

$$e^{-\mathbf{A}t} = \phi^{-1}(t) \tag{8-66}$$

Thus,

$$\phi(-t) = \phi^{-1}(t) = e^{-\mathbf{A}t} \tag{8-67}$$

An interesting result from this property of $\phi(t)$ is that Eq. (8-59) can be rearranged to read

$$\mathbf{x}(0) = \phi(-t)\mathbf{x}(t) \tag{8-68}$$

which means that the state-transition process can be considered as bilateral in time. That is, the transition in time can take place in either direction.

3.

$$\phi(t_2 - t_1)\phi(t_1 - t_0) = \phi(t_2 - t_0) \quad \text{for any } t_0, t_1, t_2 \tag{8-69}$$

Proof:

$$\begin{aligned}\phi(t_2 - t_1)\phi(t_1 - t_0) &= e^{\mathbf{A}(t_2-t_1)}e^{\mathbf{A}(t_1-t_0)}\\ &= e^{\mathbf{A}(t_2-t_0)} = \phi(t_2 - t_0)\end{aligned} \tag{8-70}$$

This property of the state-transition matrix is important because it implies that a state-transition process can be divided into a number of sequential transitions. Figure 8-3 illustrates that the transition from $t = t_0$ to $t = t_2$ is equal to the transition from t_0 to t_1 and then from t_1 to t_2. In general, of course, the state-transition process can be divided into any number of parts.

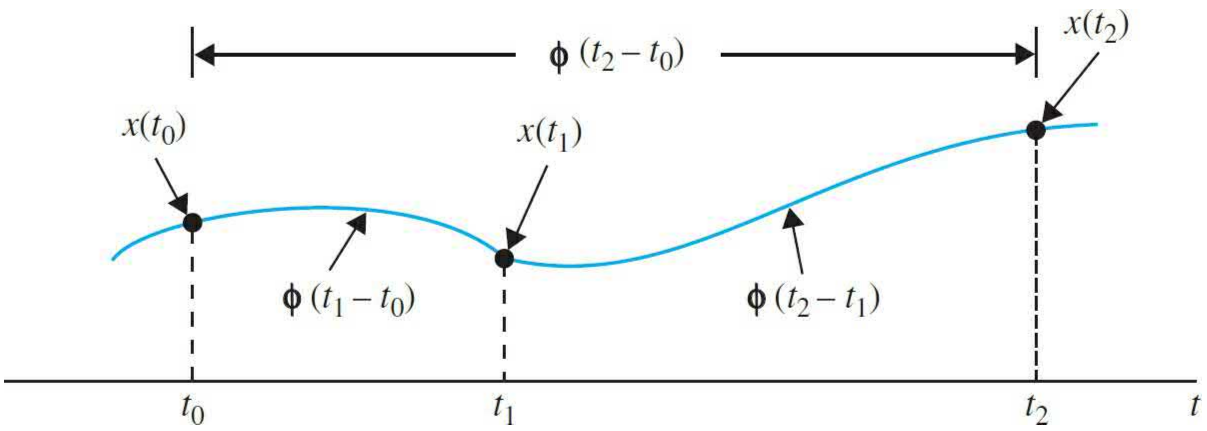


Figure 8-3 Property of the state-transition matrix.

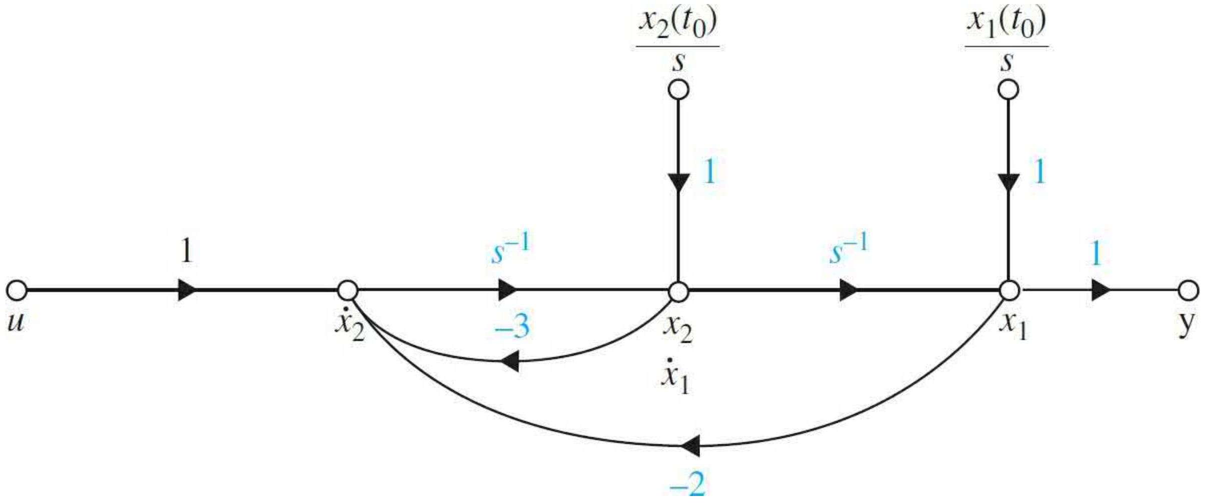


Figure 8-4 State diagram for Eq. (8-81).

4.

$$[\phi(t)]^k = \phi(kt) \quad \text{for } k = \text{positive integer} \tag{8-71}$$

Proof:

$$\begin{aligned}[\phi(t)]^k &= e^{\mathbf{A}t}e^{\mathbf{A}t}\dots e^{\mathbf{A}t} \\ &= e^{k\mathbf{A}t} = \phi(kt)\end{aligned} \tag{8-72}$$

8-6 STATE-TRANSITION EQUATION

The **state-transition equation** *is defined as the solution of the linear* *homogeneous state equation*. The linear time-invariant state equation

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{E}\mathbf{w}(t) \tag{8-73}$$

can be solved using either the classical method of solving linear differential equations or the Laplace transform method. The Laplace transform solution is presented in the following equations.

Taking the Laplace transform on both sides of Eq. (8-73), we have

$$s\mathbf{X}(s) - \mathbf{x}(0) = \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) + \mathbf{E}\mathbf{W}(s) \tag{8-74}$$

where **x**(0) denotes the initial-state vector evaluated at $t = 0$. Solving for **X**(s) in Eq. (8-74) yields

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{x}(0) + (s\mathbf{I} - \mathbf{A})^{-1}[\mathbf{B}\mathbf{U}(s) + \mathbf{E}\mathbf{W}(s)] \tag{8-75}$$

The state-transition equation of Eq. (8-73) is obtained by taking the inverse Laplace transform on both sides of Eq. (8-75):

$$\begin{aligned}\mathbf{x}(t) &= \mathcal{L}^{-1}[(s\mathbf{I} - \mathbf{A})^{-1}]\mathbf{x}(0) + \mathcal{L}^{-1}\{(s\mathbf{I} - \mathbf{A})^{-1}[\mathbf{B}\mathbf{U}(s) + \mathbf{E}\mathbf{W}(s)]\} \\ &= \phi(t)\mathbf{x}(0) + \int_0^t \phi(t-\tau)[\mathbf{B}\mathbf{u}(\tau) + \mathbf{E}\mathbf{w}(\tau)]d\tau \quad t \geq 0\end{aligned} \tag{8-76}$$

The state-transition equation in Eq. (8-76) is useful only when the initial time is defined to be at $t = 0$. In the study of control systems, especially discrete-data control systems, it is often desirable to break up a state-transition process into a sequence of transitions, so a more flexible initial time must be chosen. Let the initial time be represented by t_0 and the corresponding initial state by $\mathbf{x}(t_0)$, and assume that the input **u**(t) and the disturbance **w**(t) are applied at $t \geq 0$. We start with Eq. (8-76) by setting $t = t_0$, and solving for **x**(0), we get

$$\mathbf{x}(0) = \phi(-t_0)\mathbf{x}(t_0) - \phi(-t_0)\int_0^{t_0} \phi(t_0-\tau)[\mathbf{B}\mathbf{u}(\tau) + \mathbf{E}\mathbf{w}(\tau)]d\tau \tag{8-77}$$

where the property on $\phi(t)$ of Eq. (8-64) has been applied.

Substituting Eq. (8-77) into Eq. (8-76) yields

$$\mathbf{x}(t)=\phi(t)\phi(-t_0)\mathbf{x}(t_0)-\phi(t)\phi(-t_0)\int_0^{t_0}\phi(t_0-\tau)[\mathbf{Bu}(\tau)+\mathbf{Ew}(\tau)]d\tau$$
$$+\int_0^t\phi(t-\tau)[\mathbf{Bu}(\tau)+\mathbf{Ew}(\tau)]d\tau \qquad (8\text{-}78)$$

Now by using the property of Eq. (8-69) and combining the last two integrals, Eq. (8-78) becomes

$$\mathbf{x}(t)=\phi(t-t_0)\mathbf{x}(t_0)+\int_0^t\phi(t-\tau)[\mathbf{Bu}(\tau)+\mathbf{Ew}(\tau)]d\tau \quad t\ge t_0 \qquad (8\text{-}79)$$

It is apparent that Eq. (8-79) reverts to Eq. (8-77) when $t_0 = 0$.

Once the state-transition equation is determined, the output vector can be expressed as a function of the initial state and the input vector simply by substituting **x**(*t*) from Eq. (8-79) into Eq. (8-45). Thus, the output vector is

$$y(t)=\mathbf{C}\phi(t-t_0)\mathbf{x}(t_0)+\int_{t_0}^t\mathbf{C}\phi(t-\tau)[\mathbf{Bu}(\tau)+\mathbf{Ew}(\tau)]d\tau$$
$$+\mathbf{Du}(t)+\mathbf{Hw}(t) \quad t\ge t_0 \qquad (8\text{-}80)$$

The following example illustrates the determination of the state-transition matrix and equation.

EXAMPLE 8-6-1 Consider the state equation

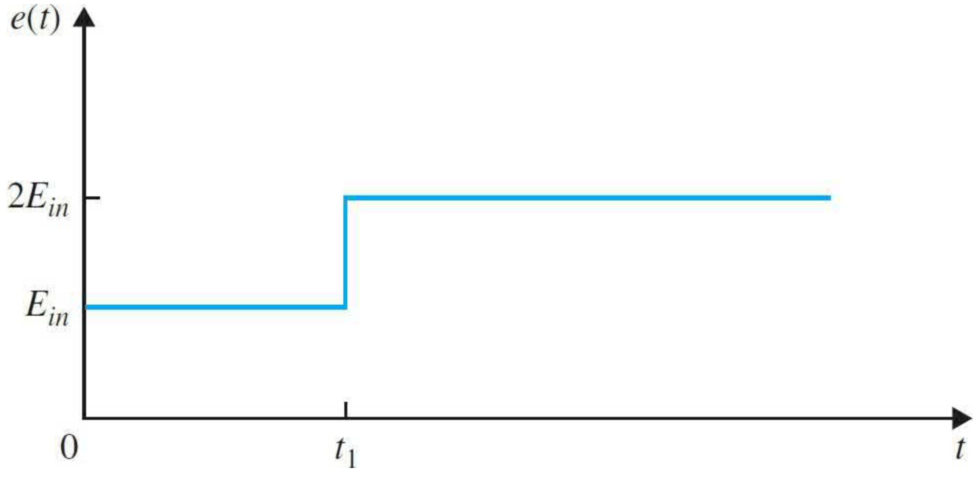


Figure 8-6 Input voltage waveform for the network in Fig. 8-5.

$$\begin{bmatrix} \dfrac{dx_1(t)}{dt} \\ \dfrac{dx_2(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \tag{8-81}$$

The problem is to determine the state-transition matrix ϕ (t) and the state vector x(t) for $t \geq 0$ when the input is $u(t) = 1$ for $t \geq 0$. The coefficient matrices are identified to be

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \mathbf{E} = 0 \tag{8-82}$$

Therefore,

$$s\mathbf{I} - \mathbf{A} = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} = \begin{bmatrix} s & -1 \\ 2 & s+3 \end{bmatrix} \tag{8-83}$$

The inverse matrix of (s**I** – **A**) is

$$(s\mathbf{I} - \mathbf{A})^{-1} = \frac{1}{s^2 + 3s + 2} \begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix} \tag{8-84}$$

The state-transition matrix of **A** is found by taking the inverse Laplace transform of Eq. (8-84). Thus,

$$\phi(t) = \mathcal{L}^{-1}[(s\mathbf{I} - \mathbf{A})^{-1}] = \begin{bmatrix} 2e^{-t} - e^{-2t} & e^{-t} - e^{-2t} \\ -2e^{-t} + 2e^{-2t} & -e^{-t} + 2e^{-2t} \end{bmatrix} \tag{8-85}$$

The state-transition equation for $t \geq 0$ is obtained by substituting Eq. (8-85), **B**, and $u(t)$ into Eq. (8-76). We have

$$\mathbf{x}(t)=\begin{bmatrix} 2e^{-t}-e^{-2t} & e^{-t}-e^{-2t} \\ -2e^{-t}+2e^{-2t} & -e^{-t}+2e^{-2t} \end{bmatrix} x(0)$$

$$+\int_0^t \begin{bmatrix} 2e^{-(t-\tau)}-e^{-2(t-\tau)} & e^{-(t-\tau)}-e^{-2(t-\tau)} \\ -2e^{-(t-\tau)}+e^{-2(t-\tau)} & -e^{-(t-\tau)}+2e^{-2(t-\tau)} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} d\tau \qquad (8\text{-}86)$$

or

$$\mathbf{x}(t)=\begin{bmatrix} 2e^{-t}-e^{-2t} & e^{-t}-e^{-2t} \\ -2e^{-t}+2e^{-2t} & -e^{-t}+2e^{-2t} \end{bmatrix} \mathbf{x}(0)+\begin{bmatrix} 0.5-e^{-t}+0.5e^{-2t} \\ e^{-t}-e^{-2t} \end{bmatrix} \quad t\geq 0 \qquad (8\text{-}87)$$

As an alternative, the second term of the state-transition equation can be obtained by taking the inverse Laplace transform of ($s\mathbf{I}$ − $\mathbf{A}$)$^{-1}$$\mathbf{BU}$($s$). Thus, we have

$$\mathcal{L}^{-1}[(s\mathbf{I}-\mathbf{A})^{-1}]\mathbf{BU}(s)={}^{-1}\left(\frac{1}{s^2+3s+2}\begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix}\frac{1}{s}\right)$$

$$={}^{-1}\left(\frac{1}{s^2+3s+2}\begin{bmatrix} \frac{1}{s} \\ 1 \end{bmatrix}\right)=\begin{bmatrix} 0.5-e^{-t}+0.5e^{-2t} \\ e^{-t}-e^{-2t} \end{bmatrix} \quad t\geq 0 \qquad (8\text{-}88)$$

8-6-1 State-Transition Equation Determined from the State Diagram

Equations (8-75) and (8-76) show that the Laplace transform method of solving the state equations requires obtaining the inverse of matrix ($s\mathbf{I}$ − $\mathbf{A}$). We shall now show that the state diagram and the SFG gain formula (Chap. 4) can be used to solve for the state-transition equation in the Laplace domain of Eq. (8-75). Let the initial time be t_0; then Eq. (8-75) is rewritten as

$$\mathbf{X}(s)=(s\mathbf{I}-\mathbf{A})^{-1}\mathbf{x}(t_0)+(s\mathbf{I}-\mathbf{A})^{-1}[\mathbf{BU}(s)+\mathbf{EW}(s)] \quad t\geq t_0 \qquad (8\text{-}89)$$

The last equation can be written directly from the state diagram using the gain formula, with $X_i(s)$, i = 1, 2,...,n, *as the output nodes*. The following example illustrates the state-diagram method of finding the state-transition equations for the system described in Example 8-2-1.

EXAMPLE 8-6-2 The state diagram for the system described by Eq. (8-81) is shown in Fig. 8-4 with t_0 as the initial time. The outputs of the integrators are assigned as state variables. Applying the gain formula to the state diagram in Fig. 8-4, with $X_1(s)$ and $X_2(s)$ as output nodes and $x_1(t_0)$, $x_2(t_0)$, and $u(t)$, or $U(s)$ in the s domain, as input nodes, we have

$$X_1(s) = \frac{s^{-1}(1+3s^{-1})}{\Delta}x_1(t_0) + \frac{s^{-2}}{\Delta}x_2(t_0) + \frac{s^{-2}}{\Delta}U(s) \tag{8-90}$$

$$X_2(s) = \frac{-2s^{-2}}{\Delta}x_1(t_0) + \frac{s^{-1}}{\Delta}x_2(t_0) + \frac{s^{-1}}{\Delta}U(s) \tag{8-91}$$

where

$$\Delta = 1+3s^{-1}+2s^{-2} \tag{8-92}$$

After simplification, Eqs. (8-90) and (8-91) are presented in vector-matrix form:

$$\begin{bmatrix} X_1(s) \\ X_2(s) \end{bmatrix} = \frac{1}{(s+1)(s+2)}\begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix}\begin{bmatrix} x_1(t_0) \\ x_2(t_0) \end{bmatrix} + \frac{1}{(s+1)(s+2)}\begin{bmatrix} 1 \\ s \end{bmatrix}U(s) \tag{8-93}$$

The state-transition equation for $t \geq t_0$ is obtained by taking the inverse Laplace transform on both sides of Eq. (8-93).

Consider that the input $u(t)$ is a unit-step function applied at $t = t_0$. Then the following inverse Laplace-transform relationships are identified:

$$\mathcal{L}^{-1}\left(\frac{1}{s}\right)=u_s(t-t_0) \quad t\geq t_0 \tag{8-94}$$

$$\mathcal{L}^{-1}\left(\frac{1}{s+a}\right)=e^{-a(t-t_0)}u_s(t-t_0) \quad t\geq t_0 \tag{8-95}$$

Because the initial time is defined to be t_0, the Laplace transform expressions here do not have the delay factor *e*-$^{-t_05}$. The inverse Laplace transform of Eq. (8-93) is

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}=\begin{bmatrix} 2e^{-(t-t_0)}-e^{-2(t-t_0)} & e^{-(t-t_0)}-e^{-2(t-t_0)} \\ -2e^{-(t-t_0)}+2e^{-2(t-t_0)} & -e^{(t-t_0)}+2e^{-2(t-t_0)} \end{bmatrix}\begin{bmatrix} x_1(t_0) \\ x_2(t_0) \end{bmatrix}$$
$$+\begin{bmatrix} 0.5u_s(t-t_0)-e^{-(t-t_0)}+0.5e^{-2(t-t_0)} \\ e^{-(t-t_0)}-e^{-2(t-t_0)} \end{bmatrix} t\geq t_0 \tag{8-96}$$

The reader should compare this result with that in Eq. (8-87), which is obtained for $t \geq 0$. ▲

EXAMPLE 8-6-3 In this example, we illustrate the utilization of the state-transition method to a system with input discontinuity. An *RL* network is shown in Fig. 8-5. The history of the network is completely specified by the initial current of the inductance, $i(0)$ at $t = 0$. At time $t = 0$, the voltage $e_{in}(t)$ with the profile shown in Fig. 8-6 is applied to the network. The state equation of the network for $t \geq 0$ is

$$\frac{di(t)}{dt}=-\frac{R}{L}i(t)+\frac{1}{L}e_{in}(t) \tag{8-97}$$

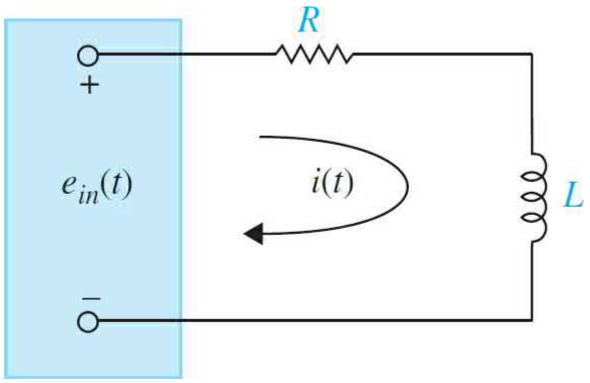


Figure 8-5 *RL* network.

Comparing the last equation with Eq. (8-44), the scalar coefficients of the state equation are identified to be

$$A = -\frac{R}{L} \quad B = \frac{1}{L} \quad E = 0 \tag{8-98}$$

The state-transition matrix is

$$\phi(t) = e^{-At} = e^{-Rt/L} \tag{8-99}$$

The conventional approach of solving for $i(t)$ for $t \geq 0$ is to express the input voltage as

$$e(t) = E_{in}u_s(t) + E_{in}u_s(t - t_1) \tag{8-100}$$

where $u_s(t)$ is the unit-step function. The Laplace transform of e(t) is

$$E_{in}(s) = \frac{E_{in}}{s}(1 + e^{-t_1 s}) \tag{8-101}$$

Then

$$(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{BU}(s) = \frac{E_{in}}{Ls(s + R/L)}(1 + e^{-t_1 s}) \tag{8-102}$$

By substituting Eq. (8-102) into Eq. (8-76), the state-transition equation, the current for $t \geq 0$ is obtained:

$$i(t) = e^{-Rt/L} i(0) u_s(t) + \frac{E_{in}}{R}(1 - e^{-Rt/L}) u_s(t) + \frac{E_{in}}{R}(1 - e^{-R(t-t_1)/L}) u_s(t - t_1) \tag{8-103}$$

Using the state-transition approach, we can divide the transition period into two parts: $t = 0$ to $t = t_1$, and $t = t_1$ to $t = \infty$. First, for the time interval $0 \le t \le t_1$, the input is

$$e(t) = E_{in} u_s(t) \quad 0 \le t < t_1 \tag{8-104}$$

Then

$$(s\mathbf{I} - \mathbf{A})^{-1} \mathbf{BU}(s) = \frac{E_{in}}{Ls(s + R/L)} = \frac{E_{in}}{Rs[1 + (L/R)s]} \tag{8-105}$$

Thus, the state-transition equation for the time interval $0 \le t \le t_1$ is

$$i(t) = \left[e^{-Rt/L} i(0) + \frac{E_{in}}{R}(1 - e^{-Rt/L}) \right] u_s(t) \tag{8-106}$$

Substituting $t = t_1$ into Eq. (8-106), we get

$$i(t_1) = e^{-Rt_1/L} i(0) + \frac{E_{in}}{R}(1 - e^{-Rt_1/L}) \tag{8-107}$$

The value of $i(t)$ at $t = t_1$ is now used as the initial state for the next transition period of $t_1 \le t < \infty$. The amplitude of the input for the interval is $2E_{in}$. The state-transition equation for the second transition period is

$$i(t) = e^{-R(t-t_1)/L} i(t_1) + \frac{2E_{in}}{R}(1 - e^{-R(t-t_1)/L}) \quad t \ge t_1 \tag{8-108}$$

where $i(t_1)$ is given by Eq. (8-107).

This example illustrates two possible ways of solving a state-transition problem. In the first approach, the transition is treated as one continuous process, whereas in the second, the transition period is divided into parts over which the input can be more easily presented. Although the first approach requires only one operation, the second method yields relatively simple results to the state-transition equation, and it often presents computational advantages. Notice that, in the second method, the state at $t = t_1$ is used as the initial state for the next transition period, which begins at t_1.

8-7 RELATIONSHIP BETWEEN STATE EQUATIONS AND HIGH-ORDER DIFFERENTIAL EQUATIONS

In the preceding sections, we defined the state equations and their solutions for linear time-invariant systems. Although it is usually possible to write the state equations directly from the schematic diagram of a system, in practice the system may have been described by a high-order differential equation or transfer function. It becomes necessary to investigate how state equations can be written directly from the high-order differential equation or the transfer function. In Chap. 2, we illustrated how the state variables of an *n*th-order differential equation in Eq. (2-97) are intuitively defined, as shown in Eq. (2-105). The results are the *n* state equations in Eq. (2-106).

The state equations are written in vector-matrix form:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{8-109}$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix} (n \times n) \tag{8-110}$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} (n \times 1) \tag{8-111}$$

Notice that the last row of **A** contains the negative values of the coefficients of the homogeneous part of the differential equation in ascending order, except for the coefficient of the highest-order term, which is unity. **B** is a column matrix with the last row equal to one, and the rest of the elements are all zeros. The state equations in Eq. (8-109) with **A** and **B** given in Eqs. (8-110) and (8-111) are known as the **phase-variable canonical form** (**PVCF**), or the **controllability canonical form** (**CCF**).

The output equation of the system is written

$$y(t) = \mathbf{C}\mathbf{x}(t) = x_1(t) \tag{8-112}$$

where

$$\mathbf{C} = [1 \quad 0 \quad 0 \quad \cdots \quad 0] \tag{8-113}$$

We have shown earlier that the state variables of a given system are not unique. In general, we seek the most convenient way of assigning the state variables as long as the definition of state variables is satisfied. In Sec. 8-11, we shall show that, by first writing the transfer function and then drawing the state diagram of the system by decomposition of the transfer function, the state variables and state equations of any system can be found very easily.

EXAMPLE 8-7-1 Consider the differential equation

$$\frac{d^3y(t)}{dt^3} + 5\frac{d^2y(t)}{dt^2} + \frac{dy(t)}{dt} + 2y(t) = u(t) \tag{8-114}$$

Rearranging the last equation so that the highest-order derivative term is set equal to the rest of the terms, we have

$$\frac{d^3y(t)}{dt^3} = -5\frac{d^2y(t)}{dt^2} - \frac{dy(t)}{dt} - 2y(t) + u(t) \tag{8-115}$$

$$x_1(t) = y(t)$$

$$x_2(t) = \frac{dy(t)}{dt}$$

$$x_3(t) = \frac{d^2y(t)}{dt^2} \tag{8-116}$$

Then the state equations are represented by the vector-matrix equation

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \tag{8-117}$$

where x(*t*) is the 2 × 1 state vector, *u*(*t*) is the scalar input, and

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -2 & -1 & -5 \end{bmatrix} \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{8-118}$$

The output equation is

$$y(t) = x_1(t) = [1 \quad 0]\mathbf{x}(t) \tag{8-119}$$

8-8 RELATIONSHIP BETWEEN STATE EQUATIONS AND TRANSFER FUNCTIONS

We have presented the methods of modeling a linear time-invariant system by transfer functions and dynamic equations. We now investigate the relationship between these two representations.

Consider a linear time-invariant system described by the following dynamic equations:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{Bu}(t) + \mathbf{Ew}(t) \tag{8-120}$$

$$\mathbf{y}(t) = \mathbf{Cx}(t) + \mathbf{Du}(t) + \mathbf{Hw}(t) \tag{8-121}$$

where

$\mathbf{x}(t) = n \times 1$ state vector
$\mathbf{u}(t) = p \times 1$ input vector
$\mathbf{y}(t) = q \times 1$ output vector
$\mathbf{w}(t) = v \times 1$ disturbance vector

and **A**, **B**, **C**, **D**, **E**, and **H** are coefficient matrices of appropriate dimensions.

Taking the Laplace transform on both sides of Eq. (8-120) and solving for $\mathbf{X}(s)$, we have

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{x}(0) + (s\mathbf{I} - \mathbf{A})^{-1}[\mathbf{BU}(s) + \mathbf{EW}(s)] \tag{8-122}$$

The Laplace transform of Eq. (8-121) is

$$\mathbf{Y}(s) = \mathbf{CX}(s) + \mathbf{DU}(s) + \mathbf{HW}(s) \tag{8-123}$$

Substituting Eq. (8-122) into Eq. (8-123), we have

$$\mathbf{Y}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{x}(0) + \mathbf{C}(s\mathbf{I} - \mathbf{A})[\mathbf{BU}(s) + \mathbf{EW}(s)] + \mathbf{DU}(s) + \mathbf{HW}(s) \tag{8-124}$$

Because the definition of a transfer function requires that the initial conditions be set to zero, $\mathbf{x}(0) = \mathbf{0}$; thus, Eq. (8-124) becomes

$$\mathbf{Y}(s) = [\mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}]\mathbf{U}(s) + [\mathbf{C}(s\mathbf{I} - \mathbf{A})\mathbf{E} + \mathbf{H}]\mathbf{W}(s) \tag{8-125}$$

Let us define

$$\mathbf{G}_u(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \tag{8-126}$$

$$\mathbf{G}_w(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{E} + \mathbf{H} \tag{8-127}$$

where $\mathbf{G}_u(s)$ is a $q \times p$ transfer-function matrix between $\mathbf{u}(t)$ and $\mathbf{y}(t)$ when

w(*t*) = **0**, and $\mathbf{G}_w(s)$ is a *q* × *v* transfer-function matrix between **w**(*t*) and **y**(*t*) when **u**(*t*) = 0.

Then, Eq. (8-125) becomes

$$\mathbf{Y}(s)=\mathbf{G_u}(\mathbf{s})\mathbf{U}(s)+\mathbf{G_w}(\mathbf{s})\mathbf{W}(s) \tag{8-128}$$

EXAMPLE 8-8-1 Consider that a multivariable system is described by the differential equations

$$\frac{d^2y_1(t)}{dt^2}+4\frac{dy_1(t)}{dt}-3y_2(t)=u_1(t)+2w(t) \tag{8-129}$$

$$\frac{dy_1(t)}{dt}+\frac{dy_2(t)}{dt}+y_1(t)+2y_2(t)=u_2(t) \tag{8-130}$$

The state variables of the system are assigned as

$$\begin{aligned} x_1(t)&=y_1(t)\\ x_2(t)&=\frac{dy_1(t)}{dt}\\ x_3(t)&=y_2(t) \end{aligned} \tag{8-131}$$

These state variables are defined by mere inspection of the two differential equations because no particular reasons for the definitions are given other than that these are the most convenient. Now equating the first term of each of the equations of Eqs. (8-129) and (8-130) to the rest of the terms and using the state-variable relations of Eq. (8-131), we arrive at the following state equations and output equations in vector-matrix form:

$$\begin{bmatrix} \dfrac{dx_1(t)}{dt} \\ \dfrac{dx_2(t)}{dt} \\ \dfrac{dx_3(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -4 & 3 \\ -1 & -1 & -2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 2 \\ 0 \end{bmatrix} w(t) \tag{8-132}$$

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} = \mathbf{C}\mathbf{x}(t) \tag{8-133}$$

To determine the transfer-function matrix of the system using the state-variable formulation, we substitute the **A**, **B**, **C**, **D**, and **E** matrices into Eq. (8-125). First, we form the matrix (s**I** – **A**):

$$(s\mathbf{I}-\mathbf{A}) = \begin{bmatrix} s & -1 & 0 \\ 0 & s+4 & -3 \\ 1 & 1 & s+2 \end{bmatrix} \tag{8-134}$$

The determinant of (s**I** – **A**) is

$$\left| s\mathbf{I}-\mathbf{A} \right| = s^3 + 6s^2 + 11s + 3 \tag{8-135}$$

Thus,

$$(s\mathbf{I}-\mathbf{A})^{-1} = \frac{1}{\left| s\mathbf{I}-\mathbf{A} \right|} \begin{bmatrix} s^2+6s+11 & s+2 & 3 \\ -3 & s(s+2) & 3s \\ -(s+4) & -(s+1) & s(s+4) \end{bmatrix} \tag{8-136}$$

The transfer-function matrix between **u**(*t*) and **y**(*t*) is

$$\mathbf{G_u}(\mathbf{s}) = \mathbf{C}(s\mathbf{I}-\mathbf{A})^{-1}\mathbf{B} = \frac{1}{s^3+6s^2+11s+3}\begin{bmatrix} s+2 & 3 \\ -(s+1) & s(s+4) \end{bmatrix} \tag{8-137}$$

and that between w(t) and y(t) is

$$\mathbf{G_w}(\mathbf{s}) = \mathbf{C}(s\mathbf{I}-\mathbf{A})^{-1}\mathbf{E} = \frac{1}{s^3+6s+11s+3}\begin{bmatrix} 2(s+2) \\ -2(s+1) \end{bmatrix} \tag{8-138}$$

Using the conventional approach, we take the Laplace transform on both sides of Eqs. (8-129) and (8-130) and assume zero initial conditions. The resulting transformed equations are written in vector-matrix form as

$$\begin{bmatrix} s(s+4) & -3 \\ s+1 & s+2 \end{bmatrix}\begin{bmatrix} Y_1(s) \\ Y_2(s) \end{bmatrix} = \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} + \begin{bmatrix} 2 \\ 0 \end{bmatrix} W(s) \tag{8-139}$$

Solving for $\mathbf{Y}(s)$ from Eq. (8-139), we obtain

$$\mathbf{Y}(s) = \mathbf{G}_u(s)\mathbf{U}(s) + \mathbf{G}_w(s)\mathbf{W}(s) \tag{8-140}$$

where

$$\mathbf{G}_u(s) = \left[\begin{pmatrix} s(s+4) & -3 \\ s+1 & s+2 \end{pmatrix}\right]^{-1} \tag{8-141}$$

$$\mathbf{G}_w(s) = \begin{bmatrix} s(s+4) & -3 \\ s+1 & s+2 \end{bmatrix}^{-1}\begin{bmatrix} 2 \\ 0 \end{bmatrix} \tag{8-142}$$

which will give the same results as in Eqs. (8-137) and (8-138), respectively, when the matrix inverses are carried out. ▲

8-9 CHARACTERISTIC EQUATIONS,

EIGENVALUES, AND EIGENVECTORS

Characteristic equations play an important role in the study of linear systems. They can be defined with respect to differential equations, transfer functions, or state equations.

8-9-1 Characteristic Equation from a Differential Equation

Consider that a linear time-invariant system is described by the differential equation

$$\frac{d^n y(t)}{dt^n}+a_{n-1}\frac{d^{n-1}y(t)}{dt^{n-1}}+\cdots+a_1\frac{dy(t)}{dt}+a_0y(t)$$
$$=b_m\frac{d^m u(t)}{dt^m}+b_{m-1}\frac{d^{m-1}u(t)}{dt^{m-1}}+\cdots+b_1\frac{du(t)}{dt}+b_0u(t) \qquad (8\text{-}143)$$

where $n > m$. By defining the operator s as

$$s^k=\frac{d^k}{dt^k} \quad k=1,2,\ldots,n \qquad (8\text{-}144)$$

Equation (8-143) is written

$$\left(s^n+a_{n-1}s^{n-1}+\cdots+a_1s+a_0\right)y(t)=\left(b_ms^m+b_{m-1}s^{m-1}+\cdots+b_1s+b_0\right)u(t) \qquad (8\text{-}145)$$

The **characteristic equation** of the system is defined as

$$s^n+a_{n-1}s^{n-1}+\cdots+a_1s+a_0=0 \qquad (8\text{-}146)$$

which is obtained by setting the homogeneous part of Eq. (8-145) to zero.

EXAMPLE 8-9-1 Consider the differential equation in Eq. (8-114). The characteristic equation is obtained by inspection,

$$s^3+5s^2+s+2=0 \qquad (8\text{-}147)$$

8-9-2 Characteristic Equation from a Transfer Function

The transfer function of the system described by Eq. (8-143) is

$$G(s)=\frac{b_m s^m + b_{m-1} s^{m-1} + \cdots + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0} \tag{8-148}$$

The characteristic equation is obtained by equating the denominator polynomial of the transfer function to zero.

EXAMPLE 8-9-2 The transfer function of the system described by the differential equation in Eq. (8-114) is

$$\frac{Y(s)}{U(s)}=\frac{1}{s^3+5s^2+s+2} \tag{8-149}$$

The same characteristic equation as in Eq. (8-147) is obtained by setting the denominator polynomial of Eq. (8-149) to zero. ▲

8-9-3 Characteristic Equation from State Equations

From the state-variable approach, we can write Eq. (8-126) as

$$\begin{aligned}\mathbf{G}_u(s) &= \mathbf{C}\frac{\operatorname{adj}(s\mathbf{I}-\mathbf{A})}{(s\mathbf{I}-\mathbf{A})}\mathbf{B}+\mathbf{D} \\ &= \frac{\mathbf{C}[\operatorname{adj}(s\mathbf{I}-\mathbf{A})]\mathbf{B}+|s\mathbf{I}-\mathbf{A}|\mathbf{D}}{|s\mathbf{I}-\mathbf{A}|}\end{aligned} \tag{8-150}$$

Setting the denominator of the transfer-function matrix $\mathbf{G}_u(s)$ to zero, we get the characteristic equation

$$|s\mathbf{I}-\mathbf{A}|=0 \tag{8-151}$$

which is an alternative form of the characteristic equation but should lead to the same equation as in Eq. (8-146). *An important property of the characteristic equation is that, if the coefficients of* **A** *are real, then the* *coefficients of* |s**I**–A| *are also real*.

EXAMPLE 8-9-3 The matrix **A** for the state equations of the differential equation in Eq. (8-114) is given in Eq. (8-114). The characteristic equation of **A** is

$$|s\mathbf{I}-\mathbf{A}|=\begin{vmatrix} s & -1 & 0 \\ 0 & s & -1 \\ 2 & 1 & s+5 \end{vmatrix}=s^3+5s^2+s+2=0 \qquad (8\text{-}152)$$

▲

8-9-4 Eigenvalues

The roots of the characteristic equation are often referred to as the eigenvalues of the matrix **A**.

Some of the important properties of eigenvalues are given as follows.

1. If the coefficients of **A** are all real, then its eigenvalues are either real or in complex-conjugate pairs.
2. If $\lambda_1, \lambda_2, \ldots, \lambda_n$ are the eigenvalues of **A**, then

$$\operatorname{tr}(\mathbf{A})=\sum_{i=1}^{n}\lambda_i \qquad (8\text{-}153)$$

That is, the trace of **A** is the sum of all the eigenvalues of **A**.

3. If λ_i, $i = 1, 2, \ldots, n$, is an eigenvalue of **A**, then it is an eigenvalue of **A**′.
4. If **A** is nonsingular, with eigenvalues λ_i, $i = 1, 2, \ldots, n$, then $1/\lambda_i$, $i = 1, 2, \ldots, n$, are the eigenvalues of $\mathbf{A}^{-1}$.

EXAMPLE 8-9-4 The eigenvalues or the roots of the characteristic equation of the matrix A in Eq. (8-118) are obtained by solving for the roots of Eq. (8-152). The results are

$$s=-0.06047+j0.63738 \quad s=-0.06047-j0.63738 \quad s=-4.87906 \qquad (8\text{-}154)$$

▲

8-9-5 Eigenvectors

Eigenvectors are useful in modern control methods, one of which is the similarity transformation, which will be discussed in a later section.

Any nonzero vector $\mathbf{p}_i$ *that satisfies the matrix equation*

$$(\lambda_i \mathbf{I} - \mathbf{A})\mathbf{p}_i = 0 \qquad (8\text{-}155)$$

where λ_i, i = *1, 2, …,* n, *denotes the* i*th eigenvalue of* **A**, *called the eigenvector of* **A** *associated with the eigenvalue* λ_i. If **A** has distinct eigenvalues, the eigenvectors can be solved directly from Eq. (8-155).

EXAMPLE 8-9-5 Consider that the state equation of Eq. (8-44) has the coefficient matrices

$$\mathbf{A} = \begin{bmatrix} 1 & -1 \\ 0 & -1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \mathbf{E} = 0 \qquad (8\text{-}156)$$

The characteristic equation of A is

$$|s\mathbf{I} - \mathbf{A}| = s^2 - 1 \qquad (8\text{-}157)$$

The eigenvalues are $\lambda_1 = 1$ and $\lambda_2 = -1$. Let the eigenvectors be written as

$$\mathbf{p}_1 = \begin{bmatrix} p_{11} \\ p_{21} \end{bmatrix} \quad \mathbf{p}_2 = \begin{bmatrix} p_{12} \\ p_{22} \end{bmatrix} \qquad (8\text{-}158)$$

Substituting $\lambda_1 = 1$ and $\mathbf{p}_1$ into Eq. (8-155), we get

$$\begin{bmatrix} 0 & 1 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} p_{11} \\ p_{21} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad (8\text{-}159)$$

Thus, $p_{21} = 0$, and p_{11} is arbitrary, which in this case can be set equal to 1.

Similarly, for $\lambda_2 = -1$, Eq. (8-155) becomes

$$\begin{bmatrix} -2 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} p_{12} \\ p_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{8-160}$$

which leads to

$$-2p_{12} + p_{22} = 0 \tag{8-161}$$

The last equation has two unknowns, which means that one can be set arbitrarily. Let $p_{12} = 1$, then $p_{12} = 2$. The eigenvectors are

$$\mathbf{p}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \mathbf{p}_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \tag{8-162}$$

8-9-6 Generalized Eigenvectors

It should be pointed out that if **A** has multiple-order eigenvalues and is nonsymmetric, not all the **eigenvectors** can be found using Eq. (8-155). Let us assume that there are $q(<n)$distinct eigenvalues among the n eigenvalues of **A**. The eigenvectors that correspond to the q distinct eigenvalues can be determined in the usual manner from

$$(\lambda_i \mathbf{I} - \mathbf{A})\mathbf{p}_i = 0 \tag{8-163}$$

where λ_i denotes the ith distinct eigenvalue, $\mathbf{i} = 1, 2, \ldots, q$. Among the remaining high-order eigenvalues, let λ_j be of the mth order ($m \le n - q$). The corresponding eigenvectors are called the **generalized eigenvectors** and can be determined from the following m vector equations:

$$
\begin{aligned}
&(\mathbf{l}_j\mathbf{I}-\mathbf{A})\mathbf{p}_{n-q+1}=0\\
&(\lambda_j\mathbf{I}-\mathbf{A})\mathbf{p}_{n-q+2}=-\mathbf{p}_{n-q+1}\\
&(\lambda_j\mathbf{I}-\mathbf{A})\mathbf{p}_{n-q+3}=-\mathbf{p}_{n-q+2}\\
&\qquad\vdots\\
&(\lambda_j\mathbf{I}-\mathbf{A})\mathbf{p}_{n-q+m}=-\mathbf{p}_{n-q+m-1}
\end{aligned}
\tag{8-164}
$$

EXAMPLE 8-9-6 Given the matrix

$$
\mathbf{A}=\begin{bmatrix} 0 & 6 & -5 \\ 1 & 0 & 2 \\ 3 & 2 & 4 \end{bmatrix}
\tag{8-165}
$$

The eigenvalues of **A** are $\lambda_1 = 2$, $\lambda_2 = \lambda_3 = 1$. Thus, **A** is a second-order eigenvalue at 1. The eigenvector that is associated with $\lambda_1 = 2$ is determined using Eq. (8-163). Thus,

$$
(\lambda_1\mathbf{I}-\mathbf{A})\mathbf{p}_1=\begin{bmatrix} 2 & -6 & 5 \\ -1 & 2 & -2 \\ -3 & -2 & -2 \end{bmatrix}\begin{bmatrix} p_{11} \\ p_{21} \\ p_{31} \end{bmatrix}=0
\tag{8-166}
$$

Because there are only two independent equations in Eq. (8-166), we arbitrarily set $p_{11} = 2$, and we have $p_{21} = -1$ and $p_{31} = -2$. Thus,

$$
\mathbf{p}_1=\begin{bmatrix} 2 \\ -1 \\ -2 \end{bmatrix}
\tag{8-167}
$$

For the generalized eigenvectors that are associated with the second-order eigenvalues, we substitute $\lambda_2 = 1$ into the first equation of Eq. (8-164). We have

$$(\lambda_2\mathbf{I}-\mathbf{A})\mathbf{p}_2 = \begin{bmatrix} 1 & -6 & 5 \\ -1 & 1 & -2 \\ -3 & -2 & -3 \end{bmatrix}\begin{bmatrix} p_{12} \\ p_{22} \\ p_{32} \end{bmatrix} = 0 \tag{8-168}$$

Setting p_{12} = 1 arbitrarily, we have $p_{22} = -\frac{3}{7}$ and $p_{32} = -\frac{5}{7}$. Thus,

$$\mathbf{p}_2 = \begin{bmatrix} 1 \\ -\frac{3}{7} \\ -\frac{5}{7} \end{bmatrix} \tag{8-169}$$

Substituting λ_3 = 1 into the second equation of Eq. (8-164), we have

$$(\lambda_3\mathbf{I}-\mathbf{A})\mathbf{p}_3 = \begin{bmatrix} 1 & -6 & -5 \\ -1 & 1 & -2 \\ -3 & -2 & -3 \end{bmatrix}\begin{bmatrix} p_{13} \\ p_{23} \\ p_{33} \end{bmatrix} = -\mathbf{p}_2 = \begin{bmatrix} -1 \\ \frac{3}{7} \\ \frac{5}{7} \end{bmatrix} \tag{8-170}$$

Setting p_{13} arbitrarily to 1, we have the generalized eigenvector

$$\mathbf{p}_3 = \begin{bmatrix} 1 \\ -\frac{22}{49} \\ -\frac{46}{49} \end{bmatrix} \tag{8-171}$$

8-10 SIMILARITY TRANSFORMATION

The dynamic equations of a single-input single-output (SISO) system are

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \tag{8-172}$$

$$\mathbf{y}(\mathbf{t}) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}u(t) \tag{8-173}$$

where **x**(*t*) is the *n* × 1 state vector, and *u*(*t*) and *y*(*t*) are the scalar input and output, respectively. When carrying out analysis and design in the state domain, it is often advantageous to transform these equations into particular forms. For example, as we will show later, the controllability canonical form (CCF) has many interesting properties that make it convenient for controllability tests and state-feedback design.

Let us consider that the dynamic equations of Eqs. (8-172) and (8-173) are transformed into another set of equations of the same dimension by the following transformation:

$$\mathbf{x}(t) = \mathbf{P}\overline{\mathbf{x}}(t) \tag{8-174}$$

where **P** is an *n* × *n* nonsingular matrix, so

$$\overline{\mathbf{x}}(t) = \mathbf{P}^{-1}\mathbf{x}(t) \tag{8-175}$$

The transformed dynamic equations are written

$$\frac{d\overline{\mathbf{x}}(t)}{dt} = \overline{\mathbf{A}}\overline{\mathbf{x}}(t) + \overline{\mathbf{B}}u(t) \tag{8-176}$$

$$\overline{y}(t) = \overline{\mathbf{C}}\overline{\mathbf{x}}(t) + \overline{\mathbf{D}}u(t) \tag{8-177}$$

Taking the derivative on both sides of Eq. (8-175) with respect to *t*, we have

$$\frac{d\bar{\mathbf{x}}(t)}{dt} = \mathbf{P}^{-1}\frac{d\mathbf{x}(t)}{dt} = \mathbf{P}^{-1}\mathbf{A}\mathbf{x}(t) + \mathbf{P}^{-1}\mathbf{B}u(t)$$

$$= \mathbf{P}^{-1}\mathbf{A}\mathbf{P}\bar{\mathbf{x}} + P^{-1}\mathbf{B}u(t) \tag{8-178}$$

Comparing Eq. (8-178) with Eq. (8-176), we get

$$\bar{\mathbf{A}} = \mathbf{P}^{-1}\mathbf{A}\mathbf{P} \tag{8-179}$$

and

$$\bar{\mathbf{B}} = \mathbf{P}^{-1}\mathbf{B} \tag{8-180}$$

Using Eq. (8-174), Eq. (8-177) is written

$$\bar{y}(t) = \mathbf{C}\mathbf{P}\mathbf{x}(t) + \bar{\mathbf{D}}u(t) \tag{8-181}$$

Comparing Eq. (8-181) with Eq. (8-173), we see that

$$\bar{\mathbf{C}} = \mathbf{C}\mathbf{P} \quad \bar{\mathbf{D}} = \mathbf{D} \tag{8-182}$$

The transformation just described is called a **similarity transformation**, because in the transformed system such properties as the characteristic equation, eigenvectors, eigenvalues, and transfer function are all preserved by the transformation. We shall describe the controllability canonical form (CCF), the observability canonical form (OCF), and the diagonal canonical form (DCF) transformations in the following sections. The transformation equations are given without proofs.

8-10-1 Invariance Properties of the Similarity Transformations

One of the important properties of the similarity transformations is that the characteristic equation, eigenvalues, eigenvectors, and transfer functions are invariant under the transformations.

8-10-2 Characteristic Equations, Eigenvalues, and Eigenvectors

The characteristic equation of the system described by Eq. (8-176) is ||*s***I–A**| = 0 and is written

$$|s\mathbf{I}-\mathbf{A}|=|s\mathbf{I}-\mathbf{P}^{-1}\mathbf{AP}|=|s\mathbf{P}^{-1}\mathbf{P}-\mathbf{P}^{-1}\mathbf{AP}| \tag{8-183}$$

Because the determinant of a product matrix is equal to the product of the determinants of the matrices, the last equation becomes

$$|s\mathbf{I}-\overline{\mathbf{A}}|-|\mathbf{P}^{-1}||s\mathbf{I}-\mathbf{A}||\mathbf{P}|=|s\mathbf{I}-\mathbf{A}| \tag{8-184}$$

Thus, the characteristic equation is preserved, which naturally leads to the same eigenvalues and eigenvectors.

8-10-3 Transfer-Function Matrix

From Eq. (8-126), the transfer-function matrix of the system of Eqs. (8-176) and (8-177) is

$$\begin{aligned}\overline{\mathbf{G}}(s)&=\overline{\mathbf{C}}(s\mathbf{I}-\overline{\mathbf{A}})\overline{\mathbf{B}}+\overline{\mathbf{D}}\\&=\mathbf{CP}(s\mathbf{I}-\mathbf{P}^{-1}\mathbf{AP})\mathbf{P}^{-1}\mathbf{B}+\mathbf{D}\end{aligned} \tag{8-185}$$

which is simplified to

$$\overline{\mathbf{G}}(s)=\mathbf{C}(s\mathbf{I}-\mathbf{A})\mathbf{B}+\mathbf{D}=\mathbf{G}(s) \tag{8-186}$$

8-10-4 Controllability Canonical Form

Consider the dynamic equations given in Eqs. (8-172) and (8-173). The characteristic equation of **A** is

$$|s\mathbf{I}-\mathbf{A}|=s^n+a_{n-1}s^{n-1}+\cdots+a_1s+a_0=0 \tag{8-187}$$

The dynamic equations in Eqs. (8-172) and (8-173) are transformed into CCF of the form of Eqs. (8-176) and (8-177) by the transformation of Eq. (8-174), with

$$\mathbf{P}=\mathbf{SM} \tag{8-188}$$

where

$$\mathbf{S} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B}\ldots\mathbf{A}^{n-1}\mathbf{B} \end{bmatrix} \tag{8-189}$$

and

$$\mathbf{M} = \begin{bmatrix} a_1 & a_2 & & a_{n-1} & 1 \\ a_2 & a_3 & & 1 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{n-1} & 1 & & 0 & 0 \\ 1 & 0 & & 0 & 0 \end{bmatrix} \tag{8-190}$$

Then,

$$\overline{\mathbf{A}} = \mathbf{P}^{-1}\mathbf{AP} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix} \tag{8-191}$$

$$\overline{\mathbf{B}} = \mathbf{P}^{-1}\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \tag{8-192}$$

The matrices **C** and **D** are given by Eq. (8-182) and do not follow any particular pattern. The CCF transformation requires that $\mathbf{P}^{-1}$ exists, which implies that the matrix **S** must have an inverse because the inverse of **M** always exists because its determinant is $(-1)^{n-1}$, which is nonzero. The $n \times n$ matrix **S** in Eq. (8-189) is later defined as the **controllability matrix**.

EXAMPLE 8-10-1 Consider the coefficient matrices of the state equations in Eq. (8-172):

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 1 & 1 & 1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \tag{8-193}$$

The state equations are to be transformed to CCF.

The characteristic equation of **A** is

$$|s\mathbf{I} - \mathbf{A}| = \begin{vmatrix} s-1 & -2 & -1 \\ 0 & s-1 & -3 \\ -1 & -1 & s-1 \end{vmatrix} = s^3 - 3s^2 - s - 3 = 0 \tag{8-194}$$

Thus, the coefficients of the characteristic equation are identified as $a_0 = -3$, $a_1 = -1$, and $a_2 = -3$. From Eq. (8-190),

$$\mathbf{M} = \begin{bmatrix} a_1 & a_2 & 1 \\ a_2 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} -1 & -3 & 1 \\ -3 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{8-195}$$

The controllability matrix is

$$\mathbf{S} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 10 \\ 0 & 3 & 9 \\ 1 & 2 & 7 \end{bmatrix} \tag{8-196}$$

We can show that **S** is nonsingular, so the system can be transformed into the CCF. Substituting **S** and **M** into Eq. (8-188), we get

$$\mathbf{P} = \mathbf{SM} = \begin{bmatrix} 3 & -1 & 1 \\ 0 & 3 & 0 \\ 0 & -1 & 1 \end{bmatrix} \tag{8-197}$$

Thus, from Eqs. (8-191) and (8-192), the CCF model is given by

$$\bar{\mathbf{A}} = \mathbf{P}^{-1}\mathbf{AP} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 3 & 1 & 3 \end{bmatrix} \bar{\mathbf{B}} = \mathbf{P}^{-1}\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{8-198}$$

which could have been determined once the coefficients of the characteristic equation are known; however, the exercise is to show how the CCF transformation matrix **P** is obtained. ▲

8-10-5 Observability Canonical Form

A dual form of transformation of the CCF is the **observability canonical form** (OCF). The system described by Eqs. (8-172) and (8-173) is transformed to the OCF by the transformation

$$\mathbf{x}(t) = \mathbf{Q}\bar{\mathbf{x}}(t) \tag{8-199}$$

The transformed equations are as given in Eqs. (8-176) and (8-177). Thus,

$$\bar{\mathbf{A}} = \mathbf{Q}^{-1}\mathbf{AQ} \quad \bar{\mathbf{B}} = \mathbf{Q}^{-1}\mathbf{B} \quad \bar{\mathbf{C}} = \mathbf{CQ} \quad \bar{\mathbf{D}} = \mathbf{D} \tag{8-200}$$

where

$$\bar{\mathbf{A}} = \mathbf{Q}^{-1}\mathbf{AQ} = \begin{bmatrix} 0 & 0 & \cdots & 0 & -a_0 \\ 1 & 0 & \cdots & 0 & -a_1 \\ 0 & 1 & \cdots & 0 & -a_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -a_{n-1} \end{bmatrix} \tag{8-201}$$

$$\bar{\mathbf{C}} = \mathbf{CQ} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \tag{8-202}$$

The elements of the matrices $\bar{\mathbf{B}}$ and $\bar{\mathbf{D}}$ are not restricted to any form. Notice that $\bar{\mathbf{A}}$ and $\bar{\mathbf{C}}$ are the transpose of the $\bar{\mathbf{A}}$ and $\bar{\mathbf{B}}$ in Eqs. (8-191) and (8-192), respectively.

The OCF transformation matrix **Q** is given by

$$\mathbf{Q} = (\mathbf{MV})^{-1} \tag{8-203}$$

where **M** is as given in Eq. (8-190), and

$$\mathbf{V} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix} (n \times n) \tag{8-204}$$

The matrix **V** is often defined as the **observability matrix**, and $\mathbf{V}^{-1}$ must exist in order for the OCF transformation to be possible.

EXAMPLE 8-10-2 Consider that the coefficient matrices of the system described by Eqs. (8-172) and (8-138) are

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 1 & 1 & 1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix} \quad \mathbf{D} = 0 \tag{8-205}$$

Because the matrix A is identical to that of the system in Example 8-8-1, the matrix M is the same as that in Eq. (8-195). The observability matrix is

$$\mathbf{V} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 3 & 4 \\ 5 & 9 & 14 \end{bmatrix} \tag{8-206}$$

We can show that **V** is nonsingular, so the system can be transformed into the OCF. Substituting **V** and **M** into Eq. (8-203), we have the OCF transformation matrix,

$$\mathbf{Q} = (\mathbf{MV})^{-1} = \begin{bmatrix} 0.3333 & -0.1667 & 0.3333 \\ -0.3333 & 0.1667 & 0.6667 \\ 0.1667 & 0.1667 & 0.1667 \end{bmatrix} \qquad (8\text{-}207)$$

From Eq. (8-191), the OCF model of the system is described by

$$\bar{\mathbf{A}} = \mathbf{Q}^{-1}\mathbf{AQ} = \begin{bmatrix} 0 & 0 & 3 \\ 1 & 0 & 1 \\ 0 & 1 & 3 \end{bmatrix} \quad \bar{\mathbf{C}} = \mathbf{CQ} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \quad \bar{\mathbf{B}} = \mathbf{Q}^{-1}\mathbf{B} = \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix} \qquad (8\text{-}208)$$

Thus, $\bar{\mathbf{A}}$ and $\bar{\mathbf{C}}$ are of the OCF form given in Eqs. (8-201) and (8-202), respectively, and **B** does not conform to any particular form. ▲

8-10-6 Diagonal Canonical Form

Given the dynamic equations in Eqs. (8-172) and (8-173), if **A** has distinct eigenvalues, there is a nonsingular transformation

$$\mathbf{x}(t) = \mathbf{T}\bar{\mathbf{x}}(t) \qquad (8\text{-}209)$$

which transforms these equations to the dynamic equations of Eqs. (8-176) and (8-177), where

$$\bar{\mathbf{A}} = \mathbf{T}^{-1}\mathbf{AT} \quad \bar{\mathbf{B}} = \mathbf{T}^{-1}\mathbf{B} \quad \bar{\mathbf{C}} = \mathbf{CT} \quad \bar{\mathbf{D}} = \mathbf{D} \qquad (8\text{-}210)$$

The matrix $\bar{\mathbf{A}}$ is a diagonal matrix,

$$\bar{\mathbf{A}} = \begin{bmatrix} \lambda_1 & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & 0 & \cdots & 0 \\ 0 & 0 & \lambda_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_n \end{bmatrix} (n \times n) \tag{8-211}$$

where $\lambda_1, \lambda_2, \ldots, \lambda_n$ are the n distinct eigenvalues of **A**. The coefficient matrices **B**, and **C** are given in Eq. (8-210) and do not follow any particular form.

It is apparent that one of the advantages of the **diagonal canonical form** (DCF) is that the transformed state equations are *decoupled* from each other and, therefore, can be solved individually.

We show in the following that the DCF transformation matrix **T** can be formed by use of the eigenvectors of **A** as its columns; that is,

$$\mathbf{T} = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 & \cdots & \mathbf{p}_n \end{bmatrix} \tag{8-212}$$

where $\mathbf{p}_i$, $i = 1, 2, \ldots, n$, denotes the eigenvector associated with the eigenvalue λ_i. This is proved by use of Eq. (8-155), which is written as

$$\lambda_i \mathbf{p}_i = \mathbf{A}\mathbf{p}_i \quad i = 1, 2, \ldots, n \tag{8-213}$$

Now, forming the $n \times n$ matrix,

$$\begin{bmatrix} \lambda_1\mathbf{p}_1 & \lambda_2\mathbf{p}_2 & \cdots & \lambda_n\mathbf{p}_n \end{bmatrix} = \begin{bmatrix} \mathbf{A}\mathbf{p}_1 & \mathbf{A}\mathbf{p}_2 & \cdots & \mathbf{A}\mathbf{p}_n \end{bmatrix} = \mathbf{A}\begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \cdots & \mathbf{p}_n \end{bmatrix} \tag{8-214}$$

The last equation is written

$$\begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \cdots & \mathbf{p}_n \end{bmatrix} \bar{\mathbf{A}} = \mathbf{A}\begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \cdots & \mathbf{p}_n \end{bmatrix} \tag{8-215}$$

where $\bar{\mathbf{A}}$ is as given in Eq. (8-211). Thus, if we let

$$\mathbf{T} = [\mathbf{p}_1 \quad \mathbf{p}_2 \quad) \quad \mathbf{p}_n] \tag{8-216}$$

Equation (8-215) is written

$$\bar{\mathbf{A}} = \mathbf{T}^{21}\mathbf{AT} \tag{8-217}$$

If the matrix **A** is of the CCF and **A** has distinct eigenvalues, then the DCF transformation matrix is the Vandermonde matrix,

$$\mathbf{T} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ \lambda_1 & \lambda_2 & \lambda_3 & \cdots & \lambda_n \\ \lambda_1^2 & \lambda_2^2 & \lambda_3^2 & \cdots & \lambda_n^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \lambda_1^{n-1} & \lambda_2^{n-1} & \lambda_3^{n-1} & \cdots & \lambda_n^{n-1} \end{bmatrix} \tag{8-218}$$

where λ_1, λ_2, …, λ_n are the eigenvalues of **A**. This can be proven by substituting the CCF of **A** in Eq. (8-110) into Eq. (8-155). The result is that the *i*th eigenvector $\mathbf{p}_i$ is equal to the *i*th column of **T** in Eq. (8-218).

EXAMPLE 8-10-3 Consider the matrix

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -11 & -6 \end{bmatrix} \tag{8-219}$$

which has eigenvalues $\lambda_1 = -1$, $\lambda_2 = -2$, and $\lambda_3 = -3$. Because A is CCF, to transform it into DCF, the transformation matrix can be the Vandermonde matrix in Eq. (8-218). Thus,

$$\mathbf{T} = \begin{bmatrix} 1 & 1 & 1 \\ \lambda_1 & \lambda_2 & \lambda_3 \\ \lambda_1^2 & \lambda_2^2 & \lambda_3^2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ -1 & -2 & -3 \\ 1 & 4 & 9 \end{bmatrix} \tag{8-220}$$

Thus, the DCF of A is written

$$\bar{\mathbf{A}} = \mathbf{T}^{-1}\mathbf{A}\mathbf{T} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{bmatrix} \tag{8-221}$$

8-10-7 Jordan Canonical Form

In general, when the matrix **A** has multiple-order eigenvalues, unless the matrix is symmetric with real elements, it cannot be transformed into a diagonal matrix. However, there exists a similarity transformation in the form of Eq. (8-217) such that the matrix $\bar{\mathbf{A}}$ is almost diagonal. The matrix $\bar{\mathbf{A}}$ is called the **Jordan canonical form** (JCF). A typical JCF is shown below.

$$\bar{\mathbf{A}} = \begin{bmatrix} \lambda_1 & 1 & 0 & 0 & 0 \\ 0 & \lambda_1 & 1 & 0 & 0 \\ 0 & 0 & \lambda_1 & 0 & 0 \\ 0 & 0 & 0 & \lambda_2 & 0 \\ 0 & 0 & 0 & 0 & \lambda_3 \end{bmatrix} \tag{8-222}$$

where it is assumed that **A** has a third-order eigenvalue λ_1 and distinct eigenvalues λ_2 and λ_3.

The JCF generally has the following properties:

1. The elements on the main diagonal are the eigenvalues.
2. All the elements below the main diagonal are zero.
3. Some of the elements immediately above the multiple-order eigenvalues on the main diagonal are 1s, as shown in Eq. (8-222).
4. The 1s together with the eigenvalues form typical blocks called the **Jordan blocks**. As shown in Eq. (8-222), the Jordan blocks are enclosed by dashed lines.
5. When the nonsymmetrical matrix **A** has multiple-order eigenvalues, its eigenvectors are not linearly independent. For an **A** that is $n \times n$,

there are only r (where r is an integer that is less than n and is dependent on the number of multiple-order eigenvalues) linearly independent eigenvectors.

6. The number of Jordan blocks is equal to the number of independent eigenvectors r. There is one and only one linearly independent eigenvector associated with each Jordan block.
7. The number of 1s above the main diagonal is equal to $n - r$.

To perform the JCF transformation, the transformation matrix **T** *is again formed by using the eigenvectors and generalized eigenvectors as its columns.*

EXAMPLE 8-10-4 Consider the matrix given in Eq. (8-165). We have shown that the matrix has eigenvalues 2, 1, and 1. Thus, the DCF transformation matrix can be formed by using the eigenvector and generalized eigenvector given in Eqs. (8-167), (8-169), and (8-171), respectively. That is,

$$\mathbf{T} = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 \\ -1 & -\dfrac{3}{7} & -\dfrac{22}{49} \\ -2 & -\dfrac{5}{7} & -\dfrac{46}{49} \end{bmatrix} \tag{8-223}$$

Thus, the DCF is

$$\bar{\mathbf{A}} = \mathbf{T}^{-1}\mathbf{A}\mathbf{T} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \tag{8-224}$$

Note that in this case there are two Jordan blocks, and there is one element of 1 above the main diagonal. ▲

8-11 DECOMPOSITIONS OF TRANSFER

FUNCTIONS

Up to this point, various methods of characterizing linear systems have been presented. To summarize, it has been shown that the starting point of modeling a linear system may be the system’s differential equation, transfer function, or dynamic equations; all these methods are closely related. Furthermore, the state diagram is also a useful tool that can not only lead to the solutions of state equations but also serve as a vehicle of transformation from one form of description to the others. The block diagram of Fig. 8-7 shows the relationships among the various ways of describing a linear system. For example, the block diagram shows that, starting with the differential equation of a system, one can find the solution by the transfer-function or state-equation method. The block diagram also shows that the majority of the relationships are bilateral, so a great deal of flexibility exists between the methods.

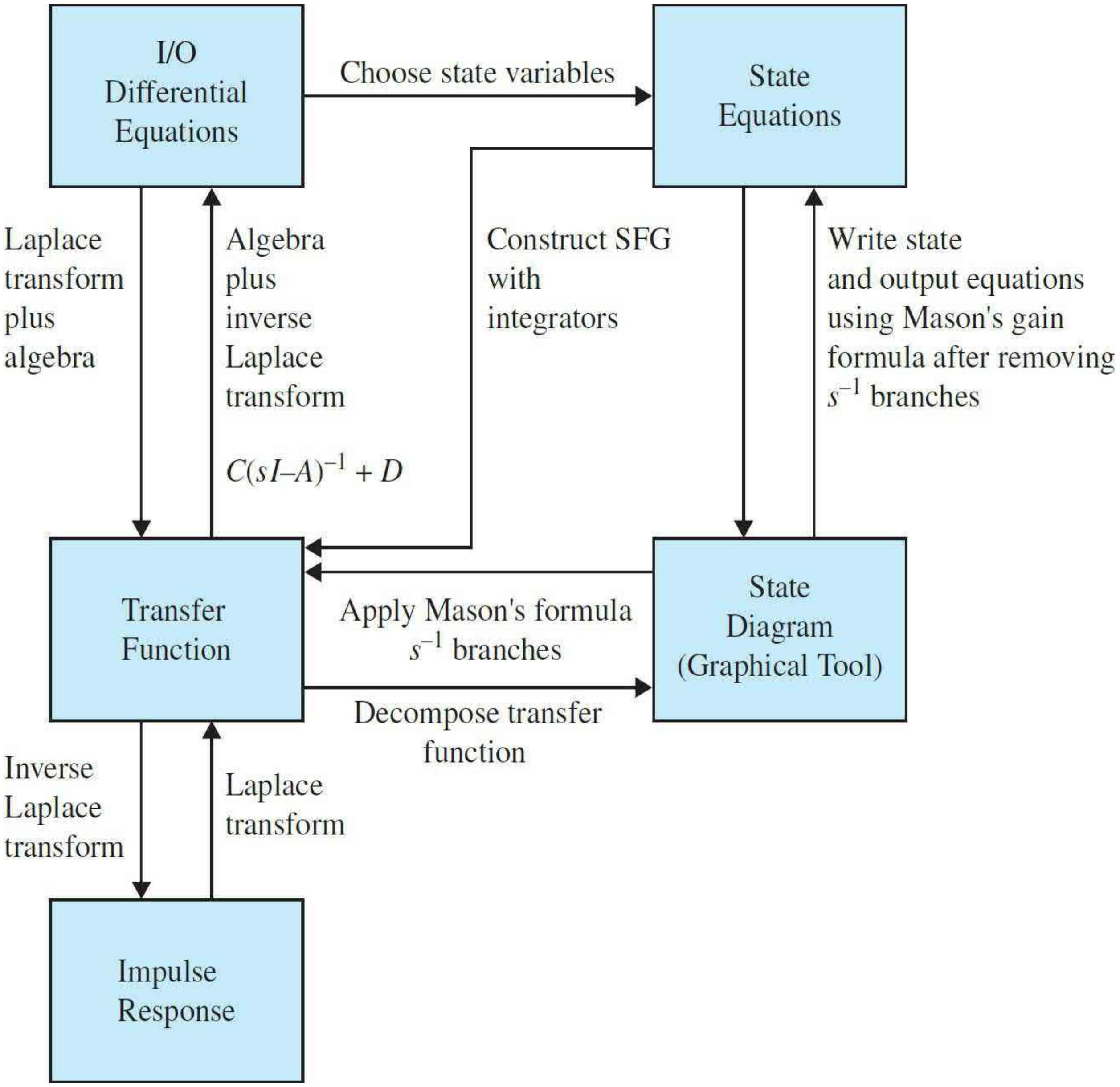

I/O Differential Equations
Choose state variables
State Equations
Laplace transform plus algebra
Algebra plus inverse Laplace transform
$C(sI–A)^{-1} + D$
Construct SFG with integrators
Write state and output equations using Mason's gain formula after removing s^{-1} branches
Transfer Function
Apply Mason's formula s^{-1} branches
Decompose transfer function
State Diagram (Graphical Tool)
Inverse Laplace transform
Laplace transform
Impulse Response

Figure 8-7 Block diagram showing the relationships among various methods of describing linear systems.

One subject remains to be discussed, which involves the construction of the state diagram from the transfer function between the input and the output. The process of going from the transfer function to the state diagram is called **decomposition**. In general, there are three basic ways to decompose transfer functions. These are **direct decomposition**, **cascade decomposition**, and **parallel decomposition**. Each of these three schemes of decomposition has its own merits and is best suited for a particular purpose.

8-11-1 Direct Decomposition

Direct decomposition is applied to an input-output transfer function that is not in factored form. Consider the transfer function of an *n*th-order SISO system between the input *U*(*s*) and output *Y*(*s*):

$$\frac{Y(s)}{U(s)}=\frac{b_{n-1}s^{n-1}+b_{b-2}s^{n-2}+\cdots+b_1s+b_0}{s^n+a_{n-1}s^{n-1}+\cdots+a_1s+a_0} \tag{8-225}$$

where we have assumed that the order of the denominator is at least 1 degree higher than that of the numerator.

We next show that the direct decomposition can be conducted in at least two ways, one leading to a state diagram that corresponds to the CCF and the other to the OCF.

8-11-2 Direct Decomposition to CCF

The objective is to construct a state diagram from the transfer function of Eq. (8-225). The following steps are outlined:

1. Express the transfer function in negative powers of *s*. This is done by multiplying the numerator and the denominator of the transfer function by $s^{-}{}_{n}$.
2. Multiply the numerator and the denominator of the transfer function by a dummy variable *X*(*s*). By implementing the last two steps, Eq. (8-225) becomes

$$\frac{Y(s)}{U(s)}=\frac{b_{n-1}s^{-1}+b_{n-2}s^{-2}+\cdots+b_1s^{-n+1}+b_0s^{-n}}{1+a_{n-1}s^{-1}+\cdots+a_1s^{-n+1}+a_0s^{-n}}\,\frac{X(s)}{X(s)} \tag{8-226}$$

3. The numerators and the denominators on both sides of Eq. (8-226) are equated to each other, respectively. The results are

$$Y(s)=\left(b_{n-1}s^{-1}+b_{n-2}s^{-2}+\cdots+b_1s^{-n+1}+b_0s^{-n}\right)X(s) \tag{8-227}$$

$$U(s)=\left(1+a_{n-1}s^{-1}+\cdots+a_1s^{-n+1}+a_0s^{-n}\right)X(s) \tag{8-228}$$

4. To construct a state diagram using the two equations in Eqs. (8-227) and (8-228), they must first be in the proper cause-and-effect relation. It is apparent that Eq. (8-227) already satisfies this prerequisite. However, Eq. (8-228) has the input on the left-hand side of the equation and must be rearranged. Equation (8-228) is rearranged as

$$X(s)=U(s)-\left(a_{n-1}s^{-1}+a_{n-2}s^{-2}+\cdots+a_1s^{-n+1}+a_0s^{-n}\right)X(s) \tag{8-229}$$

The state diagram is drawn as shown in Fig. 8-8 using Eqs. (8-227) and (8-228). For simplicity, the initial states are not drawn on the diagram. The state variables $x_1(t)$, $x_2(t)$, …, $x_n(t)$ are defined as the outputs of the integrators and are arranged in order from the right to the left on the state diagram. The state equations are obtained by applying the SFG gain formula to Fig. 8-8 with the derivatives of the state variables as the outputs and the state variables and $u(t)$ as the inputs, and overlooking the integrator branches. The output equation is determined by applying the gain formula among the state variables, the input, and the output $y(t)$. The dynamic equations are written

$$\frac{d\mathbf{x}(t)}{dt}=\mathbf{A}\mathbf{x}(t)+\mathbf{B}u(t) \tag{8-230}$$

$$y(t)=\mathbf{C}\mathbf{x}(t)+Du(t) \tag{8-231}$$

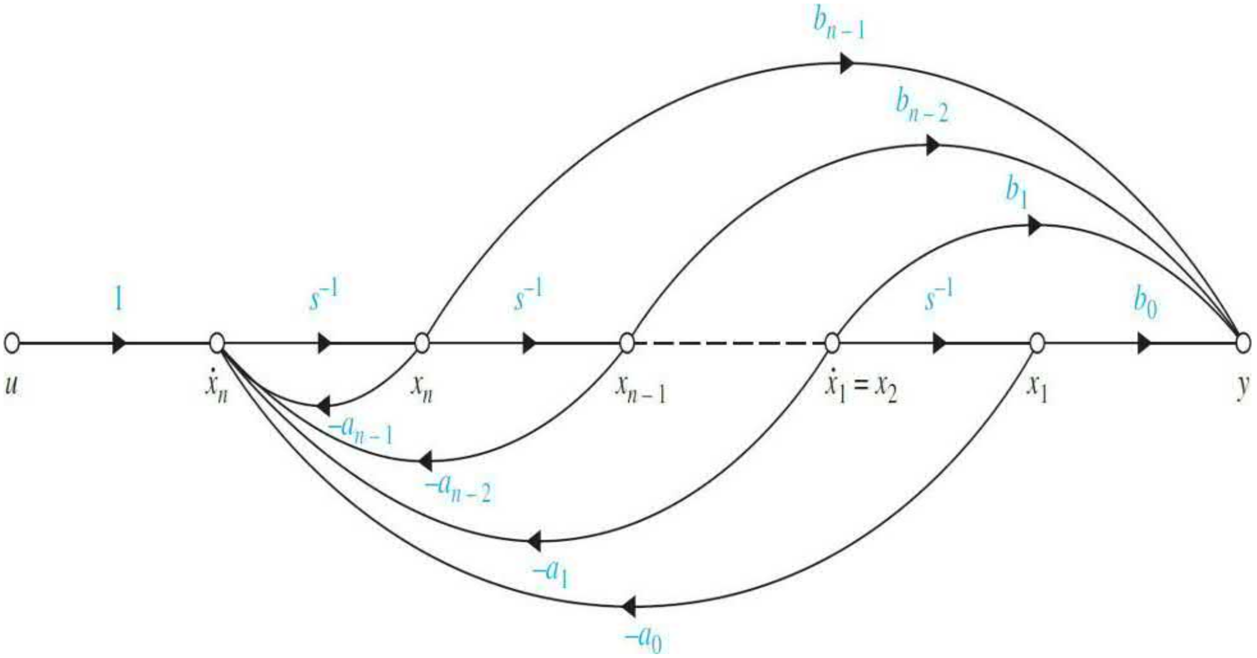


Figure 8-8 CCF state diagram of the transfer function in Eq. (8-225) by direct decomposition.

where

$$\mathbf{A}=\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix} \quad \mathbf{B}=\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \tag{8-232}$$

$$\mathbf{C}=\begin{bmatrix} b_0 & b_1 & \cdots & b_{n-2} & b_{n-1} \end{bmatrix} \quad \mathbf{D}=0 \tag{8-233}$$

Apparently, **A** and **B** in Eq. (8-232) are of the CCF.

8-11-3 Direct Decomposition to OCF

Multiplying the numerator and the denominator of Eq. (8-225) by s^{-n}, the equation is expanded as

$$\begin{aligned}&\left(1+a_{n-1}s^{-1}+\cdots+a_1 s^{-n+1}+a_0 s^{-n}\right)Y(s)\\ &\quad=\left(b_{n-1}s^{-1}+b_{n-2}s^{-2}+\cdots+b_1 s^{-n+1}+b_0 s^{-n}\right)U(s)\end{aligned} \tag{8-234}$$

or

$$\begin{aligned}Y(s)=&-\left(a_{n-1}s^{-1}+\cdots+a_1 s^{-n+1}+a_0 s^{-n}\right)Y(s)\\ &+\left(b_{n-1}s^{-1}+b_{n-2}s^{-2}+\cdots+b_1 s^{-n+1}+b_0 s^{-n}\right)U(s)\end{aligned} \tag{8-235}$$

Figure 8-9 shows the state diagram that results from using Eq. (8-235). The outputs of the integrators are designated as the state variables. However, unlike the usual convention, the state variables are assigned in *descending order* from right to left. Applying the SFG gain formula to the state diagram, the dynamic equations are written as in Eqs. (8-230) and (8-231), with

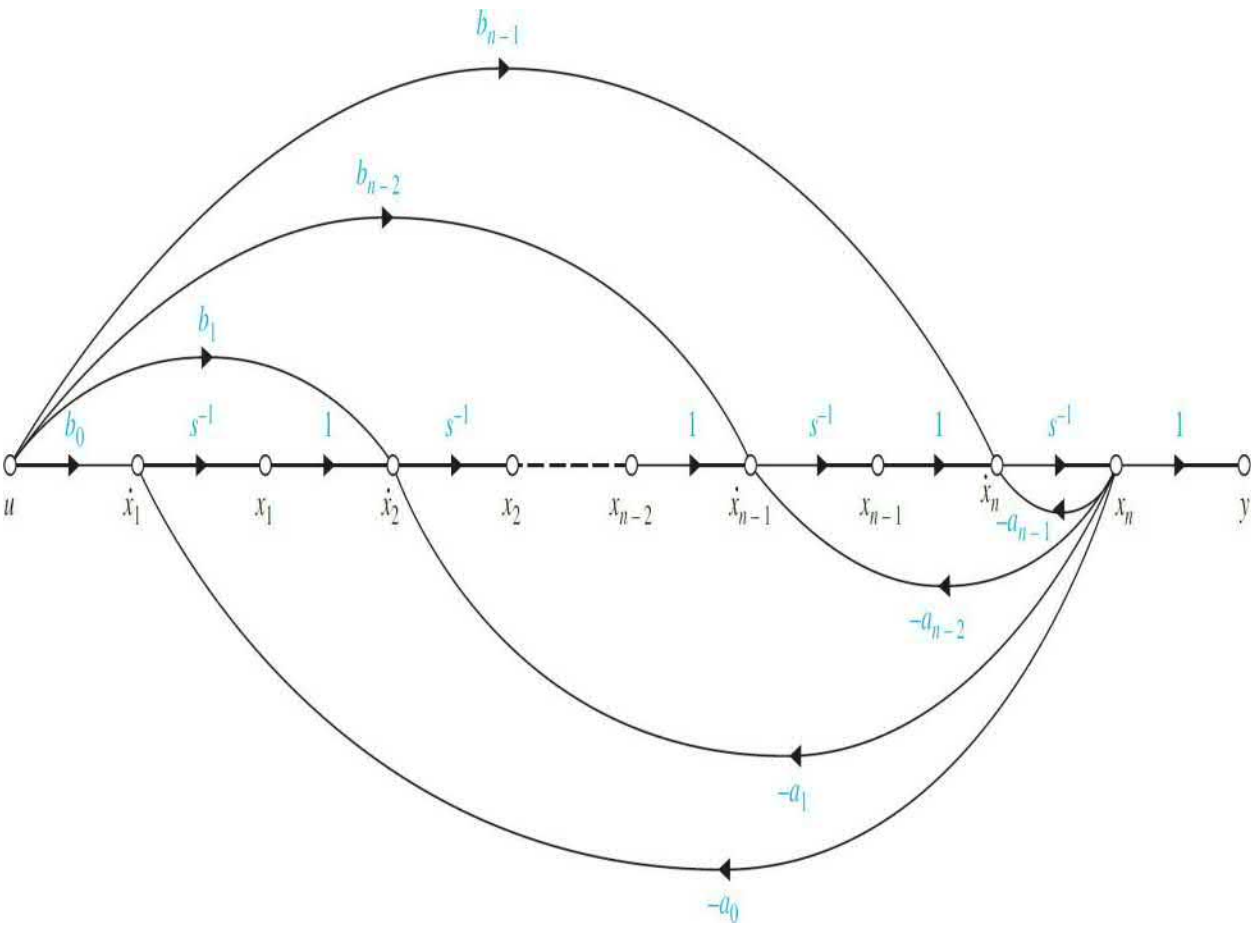


Figure 8-9 CCF state diagram of the transfer function in Eq. (8-225) by direct decomposition.

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & \cdots & 0 & -a_0 \\ 1 & 0 & \cdots & 0 & -a_1 \\ 0 & 1 & \cdots & 0 & -a_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -a_{n-1} \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_{n-1} \end{bmatrix} \tag{8-236}$$

and

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \quad \mathbf{D} = 0 \tag{8-237}$$

The matrices **A** and **C** are in OCF.

It should be pointed out that, given the dynamic equations of a system, the input-output transfer function is unique. However, given the transfer function, the state model is not unique, as shown by the CCF, OCF, and DCF, and many other possibilities. In fact, even for any one of these canonical forms (e.g., CCF), while matrices **A** and **B** are defined, the elements of **C** and **D** could still be different depending on how the state diagram is drawn, that is, how the transfer function is decomposed. In other words, referring to Fig. 8-8, whereas the feedback branches are fixed, the feedforward branches that contain the coefficients of the numerator of the transfer function can still be manipulated to change the contents of **C**.

EXAMPLE 8-11-1 Consider the following input-output transfer function:

$$\frac{Y(s)}{U(s)}=\frac{2s^2+s+5}{s^3+6s^2+11s+4} \tag{8-238}$$

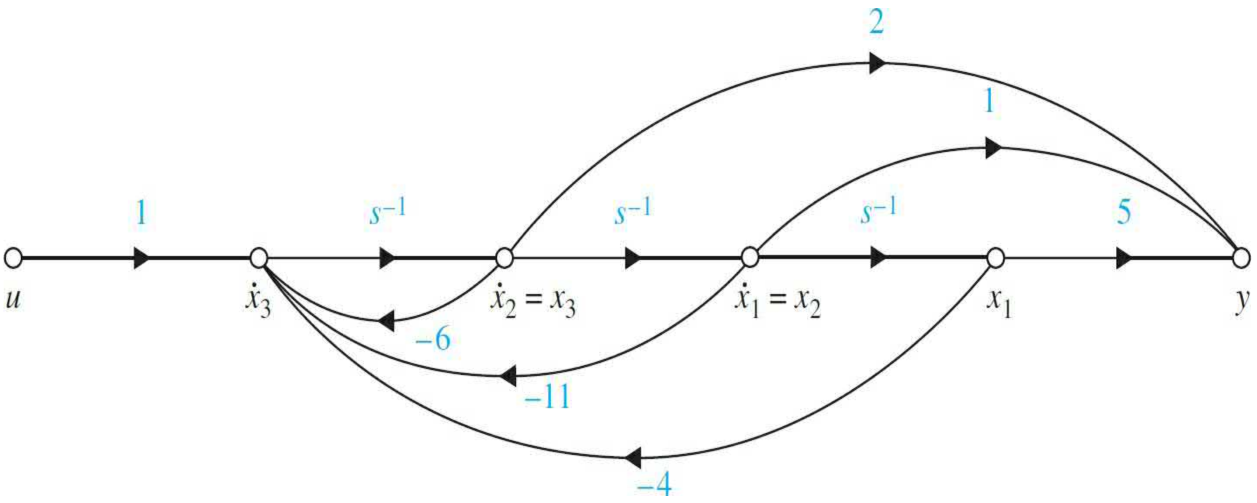


Figure 8-10 CCF state diagram of the transfer function in Eq. (8-238).

The CCF state diagram of the system is shown in Fig. 8-10, which is drawn from the following equations:

$$Y(s)=(2s^{-1}+s^{-2}+5s^{-3})X(s) \tag{8-239}$$

$$X(s)=U(s)-(6s^{-1}+11s^{-2}+4s^{-3})X(s) \tag{8-240}$$

The dynamic equations of the system in CCF are

$$\begin{bmatrix} \dfrac{dx_1(t)}{dt} \\ \dfrac{dx_2(t)}{dt} \\ \dfrac{dx_3(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -4 & -11 & -6 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(t) \tag{8-241}$$

$$y(t) = [5 \quad 1 \quad 2] x(t) \tag{8-242}$$

For the OCF, Eq. (8-238) is expanded to

$$Y(s) = (2s^{-1} + s^{-2} + 5s^{-3})U(s) - (6s^{-1} + 11s^{-2} + 4s^{-3})Y(s) \tag{8-243}$$

which leads to the OCF state diagram shown in Fig. 8-11. The OCF dynamic equations are written

$$\begin{bmatrix} \dfrac{dx_1(t)}{dt} \\ \dfrac{dx_2(t)}{dt} \\ \dfrac{dx_3(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 0 & -4 \\ 1 & 0 & -11 \\ 0 & 1 & -6 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 5 \\ 1 \\ 2 \end{bmatrix} u(t) \tag{8-244}$$

$$y(t) = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{x}(t) \tag{8-245}$$

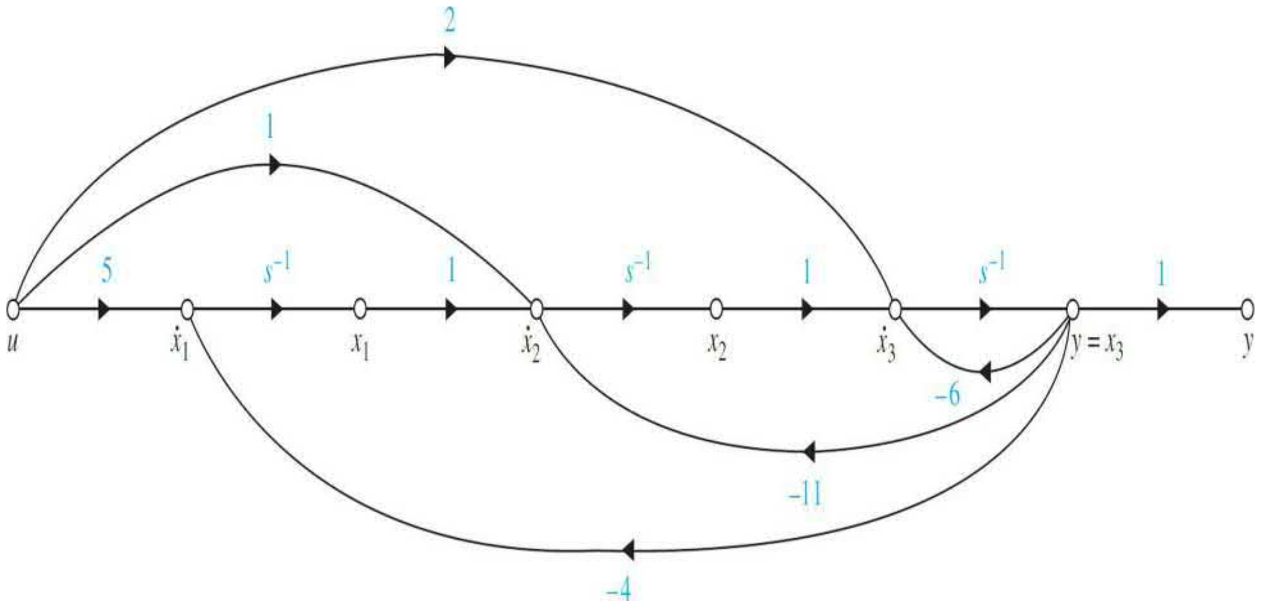


Figure 8-11 OCF state diagram of the transfer function in Eq. (8-238).

8-11-4 Cascade Decomposition

Cascade compensation refers to transfer functions that are written as products of simple first-order or second-order components. Consider the following transfer function, which is the product of two first-order transfer functions.

$$\frac{Y(s)}{U(s)} = K\left(\frac{s+b_1}{s+a_1}\right)\left(\frac{s+b_2}{s+a_2}\right) \tag{8-246}$$

where a_1, a_2, b_1, and b_2 are real constants. Each of the first-order transfer functions is decomposed by the direct decomposition, and the two state diagrams are connected in cascade, as shown in Fig. 8-12. The state equations are obtained by regarding the derivatives of the state variables as outputs and the state variables and $u(t)$ as inputs and then applying the SFG gain formula to the state diagram in Fig. 8-12. The integrator branches are neglected when applying the gain formula. The results are

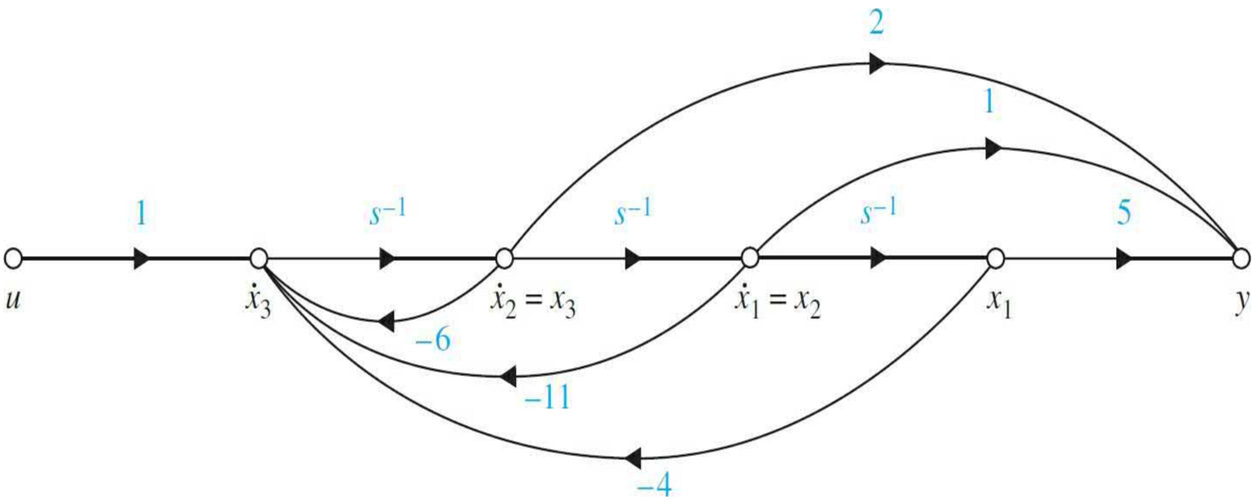


Figure 8-12 State diagram of the transfer function in Eq. (8-246) by cascade decomposition.

$$\begin{bmatrix} \dfrac{dx_1(t)}{dt} \\ \dfrac{dx_2(t)}{dt} \end{bmatrix} = \begin{bmatrix} -a_1 & b_2 - a_2 \\ 0 & -a_2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} K \\ K \end{bmatrix} u(t) \qquad (8\text{-}247)$$

The output equation is obtained by regarding the state variables and $u(t)$ as inputs and $y(t)$ as the output and applying the gain formula to Fig. 8-18. Thus,

$$y(t) = [b_1 - a_1 \quad b_2 - a_2]\, \mathbf{x}(t) + Ku(t) \qquad (8\text{-}248)$$

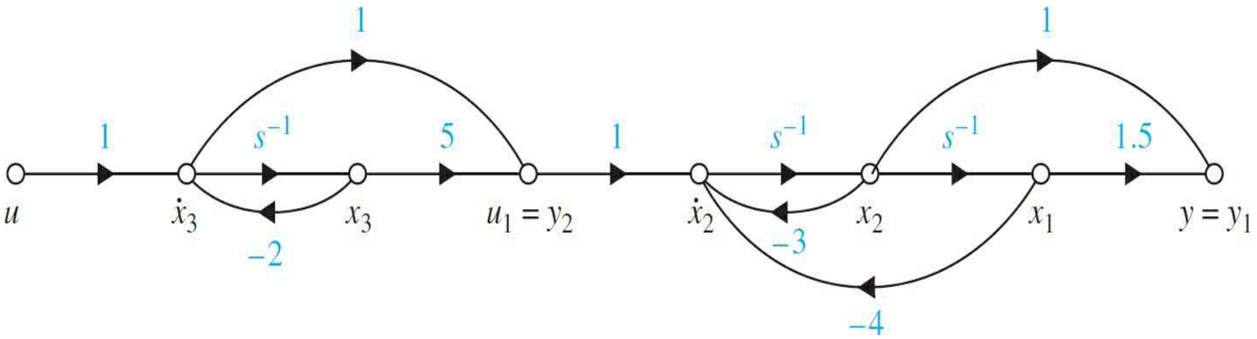


Figure 8-13 State diagram of the transfer function in Eq. (8-249) by cascade decomposition.

When the overall transfer function has complex poles or zeros, the individual factors related to these poles or zeros should be in second-order form. As an example, consider the following transfer function:

$$\frac{Y(s)}{U(s)}=\left(\frac{s+5}{s+2}\right)\left(\frac{s+1.5}{s^2+3s+4}\right) \tag{8-249}$$

where the poles of the second term are complex. The state diagram of the system with the two subsystems connected in cascade is shown in Fig. 8-13. The dynamic equations of the system are

$$\begin{bmatrix} \frac{dx_1(t)}{dt} \\ \frac{dx_2(t)}{dt} \\ \frac{dx_3(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -4 & -3 & 3 \\ 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} u(t) \tag{8-250}$$

$$y(t)=\begin{bmatrix} 1.5 & 1 & 0 \end{bmatrix}\mathbf{x}(t) \tag{8-251}$$

8-11-5 Parallel Decomposition

When the denominator of the transfer function is in factored form, the transfer function may be expanded by partial-fraction expansion. The resulting state diagram will consist of simple first- or second-order systems connected in parallel, which leads to the state equations in DCF or JCF, the latter in the case of multiple-order eigenvalues.

Consider that a second-order system is represented by the transfer function

$$\frac{Y(s)}{U(s)}=\frac{Q(s)}{(s+a_1)(s+a_2)} \tag{8-252}$$

where $Q(s)$ is a polynomial of order less than 2, and a_1 and a_2 are real and distinct. Although, analytically, a_1 and a_2 may be complex, in practice, complex numbers are difficult to implement on a computer. Equation (8-253) is expansion by partial fractions:

$$\frac{Y(s)}{U(s)} = \frac{K_1}{s+a_1} + \frac{K_2}{s+a_2} \tag{8-253}$$

where K_1 and K_2 are real constants.

The state diagram of the system is drawn by the parallel combination of the state diagrams of each of the first-order terms in Eq. (8-253), as shown in Fig. 8-14. The dynamic equations of the system are

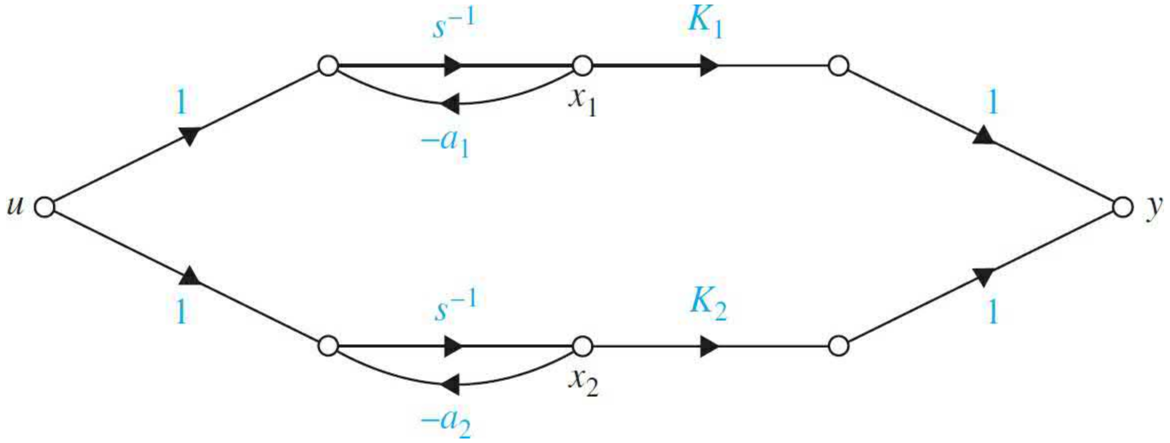


Figure 8-14 State diagram of the transfer function of Eq. (8-252) by parallel decomposition.

$$\begin{bmatrix} \dfrac{dx_1(t)}{dt} \\ \dfrac{dx_2(t)}{dt} \end{bmatrix} = \begin{bmatrix} -a_1 & 0 \\ 0 & -a_2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u(t) \tag{8-254}$$

$$y(t) = [K_1 \quad K_2]\mathbf{x}(t) \tag{8-255}$$

Thus, the state equations are of the DCF.

The conclusion is that, for transfer functions with distinct poles, parallel decomposition will lead to the DCF for the state equations. For transfer functions with multiple-order eigenvalues, parallel decomposition to a state diagram with a minimum number of integrators will lead to the JCF state equations. The following example will clarify this point.

EXAMPLE 8-11-2 Consider the following transfer function and its partial-fraction expansion:

$$\frac{Y(s)}{U(s)}=\frac{2s^2+6s+5}{(s+1)^2(s+2)}=\frac{1}{(s+1)^2}+\frac{1}{s+1}+\frac{1}{s+2} \tag{8-256}$$

Note that the transfer function is of the third order, and, although the total order of the terms on the right-hand side of Eq. (8-256) is four, only three integrators should be used in the state diagram, which is drawn as shown in Fig. 8-15. The minimum number of three integrators is used, with one integrator being shared by two channels. The state equations of the system are written directly from Fig. 8-15.

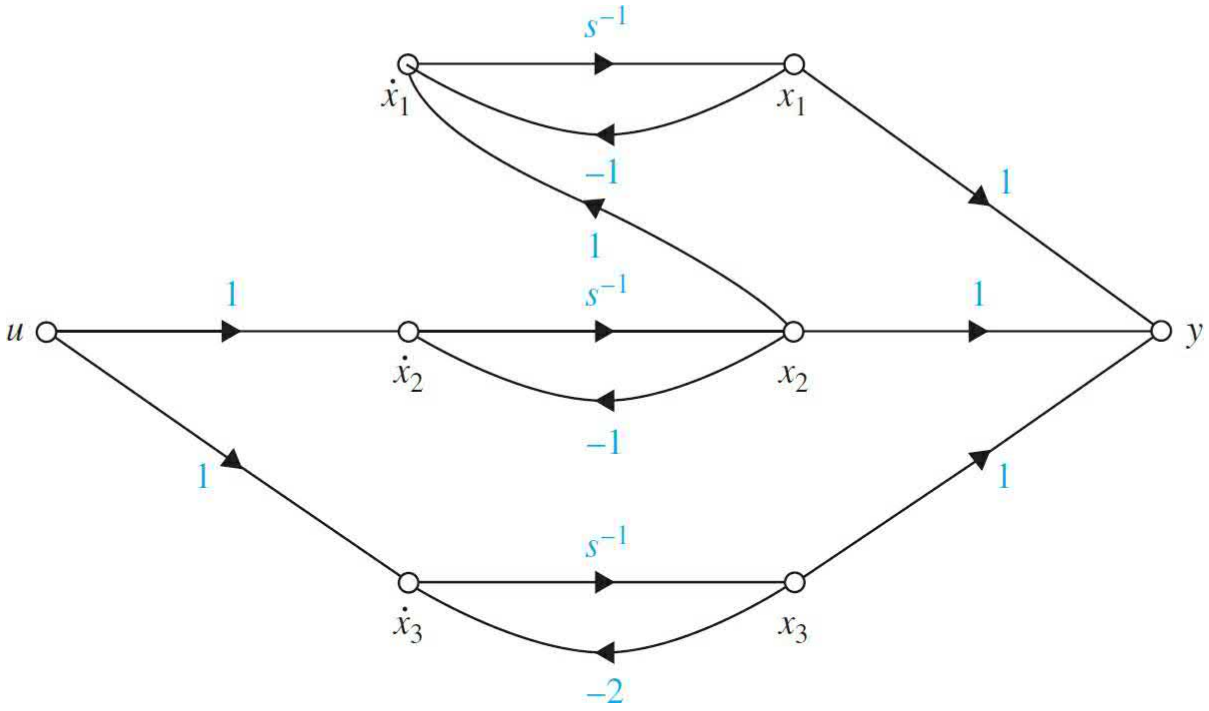


Figure 8-15 State diagram of the transfer function of Eq. (8-256) by parallel decomposition.

$$\begin{bmatrix} \dfrac{dx_1(t)}{dt} \\ \dfrac{dx_2(t)}{dt} \\ \dfrac{dx_3(t)}{dt} \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} u(t) \tag{8-257}$$

which is recognized to be the JCF. ▲

8-12 CONTROLLABILITY OF CONTROL SYSTEMS

The concepts of **controllability** and **observability**, introduced first by Kalman [3], play an important role in both theoretical and practical aspects of modern control. The conditions on controllability and observability essentially govern the existence of a solution to an optimal control problem. This seems to be the basic difference between optimal control theory and classical control theory. In the classical control theory, the design techniques are dominated by trial-and-error methods so that given a set of design specifications the designer at the outset does not know if any solution exists. Optimal control theory, on the other hand, has criteria for determining at the outset if the design solution exists for the system parameters and design objectives.

We shall show that the condition of controllability of a system is closely related to the existence of solutions of state feedback for assigning the values of the eigenvalues of the system arbitrarily. The concept of *observability* relates to the condition of observing or estimating the state variables from the output variables, which are generally measurable.

The block diagram shown in Fig. 8-16 illustrates the motivation behind investigating controllability and observability. Figure 8-16*a* shows a system with the process dynamics described by

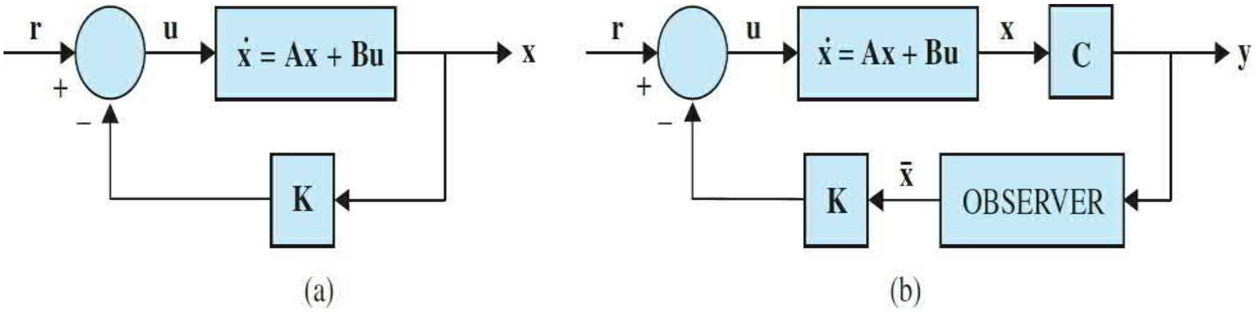


Figure 8-16 (*a*) Control system with state feedback. (*b*) Control system with observer and state feedback.

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{Bu}(t) \tag{8-258}$$

The closed-loop system is formed by feeding back the state variables through a constant feedback gain matrix **K**. Thus, from Fig. 8-16,

$$\mathbf{u}(t) = -\mathbf{Kx}(t) + \mathbf{r}(t) \tag{8-259}$$

where **K** is a *p* × *n* feedback matrix with constant elements. The closed-loop system is thus described by

$$\frac{d\mathbf{x}(t)}{dt} = (\mathbf{A} - \mathbf{BK})\mathbf{x}(t) + \mathbf{Br}(t) \tag{8-260}$$

This problem is also known as the **pole-placement design** through state feedback. The design objective in this case is to find the feedback matrix **K** such that the eigenvalues of (**A** – **BK**), or of the closed-loop system, are of certain prescribed values. The word *pole* refers here to the poles of the closed-loop transfer function, which are the same as the eigenvalues of (**A** – **BK**).

We shall show later that the existence of the solution to the pole-placement design with arbitrarily assigned pole values through state feedback is directly based on the controllability of the states of the system. The result is that *if the system of Eq. (8-225) is controllable, then there exists a constant feedback matrix* **K** *that allows the eigenvalues of* (**A** – **BK**) *to be arbitrarily assigned*.

Once the closed-loop system is designed, the practical problems of implementing the feeding back of the state variables must be considered.

There are two problems with implementing state feedback control: First, the number of state variables may be excessive, which will make the cost of sensing each of these state variables for feedback prohibitive. Second, not all the state variables are physically accessible, and so it may be necessary to design and construct an **observer** that will estimate the state vector from the output vector $\mathbf{y}(t)$. Figure 8-22*b* shows the block diagram of a closed-loop system with an observer. The observed state vector *x(t)* is used to generate the control $\mathbf{u}(t)$ through the feedback matrix $\mathbf{K}$. *The condition that such an observer can be designed for the system is called the observability of the system*.

8-12-1 General Concept of Controllability

The concept of controllability can be stated with reference to the block diagram of Fig. 8-16*a*. *The process is said to be **completely controllable** if every state variable of the process can be controlled to reach a certain objective in finite time by some unconstrained control* $\mathbf{u}(t)$, as shown in Fig. 8-17. Intuitively, it is simple to understand that, if any one of the state variables is independent of the control $\mathbf{u}(t)$, there would be no way of driving this particular state variable to a desired state in finite time by means of a control effort. Therefore, this particular state is said to be uncontrollable, and, as long as there is at least one uncontrollable state, the system is said to be not completely controllable or, simply, uncontrollable.

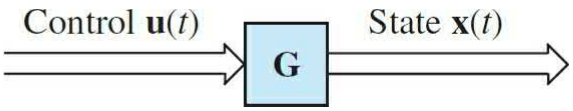


Figure 8-17 Linear time-invariant system.

As a simple example of an uncontrollable system, Fig. 8-18 illustrates the state diagram of a linear system with two state variables. Because the control $\mathbf{U}(t)$ affects only the state $x_1(t)$, the state $x_2(t)$ is uncontrollable. In other words, it would be impossible to drive $x_2(t)$ from an initial state $x_2(t_0)$ to a desired state $x_2(t_f)$ in finite time interval $t_f - t_0$ by the control $\mathbf{U}(t)$. Therefore, the entire system is said to be uncontrollable.

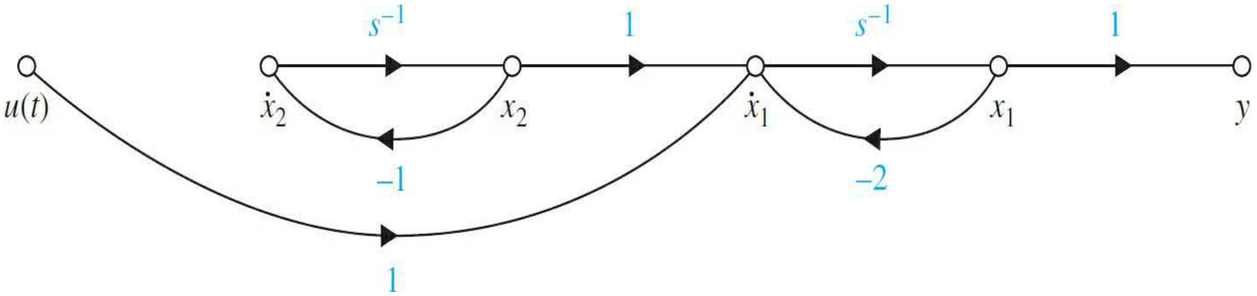


Figure 8-18 State diagram of the system that is not state controllable.

The concept of controllability given here refers to the states and is sometimes referred to as **state controllability**. Controllability can also be defined for the outputs of the system, so there is a difference between state controllability and output controllability.

8-12-2 Definition of State Controllability

Consider that a linear time-invariant system is described by the following dynamic equations:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \qquad (8\text{-}261)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \qquad (8\text{-}262)$$

where $\mathbf{x}(t)$ is the $n \times 1$ state vector, $\mathbf{u}(t)$ is the $r \times 1$ input vector, and $\mathbf{y}(t)$ is the $p \times 1$ output vector. **A**, **B**, **C**, and **D** are coefficients of appropriate dimensions.

The state $\mathbf{x}(t)$ *is said to be controllable at* $t = t_0$ *if there exists a piecewise continuous input* $\mathbf{u}(t)$ *that will drive the state to any final state* $\mathbf{x}(t_f)$ *for a finite time* $(t_f - t_0) \geq 0$. *If every state* $\mathbf{x}(t_0)$ *of the system is controllable in a finite time interval, the system is said to be completely state controllable or, simply, controllable.*

The following theorem shows that the condition of controllability depends on the coefficient matrices **A** and **B** of the system. The theorem also gives one method of testing for state controllability.

▪ **Theorem 8-1.** *For the system described by the state equation of Eq. (8-261) to be* ***completely state controllable****, it is necessary and sufficient that the* *following n × nr* ***controllability matrix*** *has a rank of n:*

$$\mathbf{S} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} & \cdots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix} \tag{8-263}$$

Because the matrices **A** *and* **B** *are involved, sometimes we say that the pair* [**A**, **B**] *is controllable, which implies that* **S** *is of rank n.*

The proof of this theorem is given in any standard textbook on optimal control systems. The idea is to start with the state-transition equation of Eq. (8-79) and then proceed to show that Eq. (8-263) must be satisfied in order that all the states are accessible by the input.

Although the criterion of state controllability given in Theorem 8-1 is quite straightforward, manually it is not very easy to test for high-order systems and/or systems with many inputs. If **S** is nonsquare, we can form the matrix **SS**′, which is *n* × *n*; then, if **SS**′ is nonsingular, **S** has rank *n*.

8-12-3 Alternate Tests on Controllability

There are several alternate methods of testing controllability, and some of these may be more convenient to apply than the condition in Eq. (8-263).

▪ **Theorem 8-2.** *For a single-input, single-output (SISO) system described by the state equation of Eq. (8-261) with r = 1, the pair* [**A**, **B**] *is completely controllable if* **A** *and* **B** *are in CCF or transformable into CCF by a similarity transformation.*

The proof of this theorem is straightforward, since it was established in Sec. 8-10 that the CCF transformation requires that the controllability matrix **S** be nonsingular. Because the CCF transformation in Sec. 8-10 was defined only for SISO systems, the theorem applies only to this type of system.

▪ **Theorem 8-3.** *For a system described by the state equation of Eq. (8-261), if* **A** *is in DCF or JCF, the pair* [**A**, **B**] *is completely controllable if all the elements in the rows of* **B** *that correspond to the last row of each Jordan block are nonzero.*

The proof of this theorem comes directly from the definition of controllability. Let us assume that **A** is diagonal and that it has distinct eigenvalues. Then, the pair [**A**, **B**] is controllable if **B** does not have any row with all zeros. The reason is that, if **A** is diagonal, all the states are decoupled from each other, and, if any row of **B** contains all zero elements, the corresponding state would not be accessed from any of the inputs, and that state would be uncontrollable.

For a system in JCF, such as the **A** and **B** matrices illustrated in Eq. (8-264), for controllability only the elements in the row of **B** that correspond to the last row of the Jordan block cannot all be zeros. The elements in the other rows of **B** need not all be nonzero, since the corresponding states are still coupled through the 1s in the Jordan blocks of **A**.

$$\mathbf{A} = \begin{bmatrix} \lambda_1 & 1 & 0 & 0 \\ 0 & \lambda_1 & 1 & 0 \\ 0 & 0 & \lambda_1 & 0 \\ 0 & 0 & 0 & \lambda_2 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \\ b_{41} & b_{42} \end{bmatrix} \tag{8-264}$$

Thus, the condition of controllability for the **A** and **B** in Eq. (8-264) is $b_{31} \neq 0$, $b_{32} \neq 0$, $b_{41} \neq 0$, and $b_{42} \neq 0$.

Example 8-12-1 The following matrices are for a system with two identical eigenvalues, but the matrix A is diagonal.

$$\mathbf{A} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} b_{11} \\ b_{21} \end{bmatrix} \tag{8-265}$$

The system is uncontrollable, since the two state equations are dependent; that is, it would not be possible to control the states independently by the input. We can easily show that in this case **S** = [**BAB**] is singular. ▲

EXAMPLE 8-12-2 Consider the system shown in Fig. 8-24, which was reasoned earlier to be uncontrollable. Let us investigate the same system using the condition of Eq. (8-263). The state equations of the system are written in the form of Eq. (8-263) with

$$\mathbf{A} = \begin{bmatrix} -2 & 1 \\ 0 & -1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{8-266}$$

Thus, from Eq. (8-263), the controllability matrix is

$$\mathbf{S} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} \end{bmatrix} = \begin{bmatrix} 1 & -2 \\ 0 & 0 \end{bmatrix} \tag{8-267}$$

which is singular, and the system is uncontrollable. ▲

EXAMPLE 8-12-3 Consider that a third-order system has the coefficient matrices

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & -1 \\ 0 & 1 & 0 \\ 1 & -4 & 3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{8-268}$$

The controllability matrix is

$$\mathbf{S} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} \end{bmatrix} = \begin{bmatrix} 0 & -1 & -4 \\ 0 & 0 & 0 \\ 1 & 3 & 8 \end{bmatrix} \tag{8-269}$$

which is singular. Thus, the system is not controllable.

The eigenvalues of A are $\lambda_1 = 2$, $\lambda_2 = 2$, and $\lambda_3 = 1$. The JCF of A and B are obtained with the transformation **x**(t) = **Tx**(*t*), where

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ -1 & 1 & 2 \end{bmatrix} \tag{8-270}$$

Then,

$$\bar{\mathbf{A}} = \mathbf{T}^{-1}\mathbf{A}\mathbf{T} = \begin{bmatrix} 2 & -1 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \bar{\mathbf{B}} = \mathbf{T}^{-1}\mathbf{B} = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} \tag{8-271}$$

Because the last row of **B**, which corresponds to the Jordan block for the eigenvalue λ_3, is zero, the transformed state variable $x_3(t)$ is uncontrollable. From the transformation matrix **T** in Eq. (8-235), $x_2 = x_3$, which means that x_2 is uncontrollable in the original system. It should be noted that the minus sign in front of the 1 in the Jordan block does not alter the basic definition of the block. ▲

8-13 OBSERVABILITY OF LINEAR SYSTEMS

The concept of observability was covered earlier in Sec. 8-11 on controllability and observability. Essentially, *a system is completely observable if every state variable of the system affects some of the outputs*. In other words, it is often desirable to obtain information on the state variables from the measurements of the outputs and the inputs. If any one of the states cannot be observed from the measurements of the outputs, the state is said to be unobservable, and the system is not completely observable or, simply, unobservable. Figure 8-19 shows the state diagram of a linear system in which the state x_2 is not connected to the output $y(t)$ in any way. Once we have measured $y(t)$, we can observe the state $x_1(t)$, since $x_1(t) = y(t)$. However, the state x_2 cannot be observed from the information on $y(t)$. Thus, the system is unobservable.

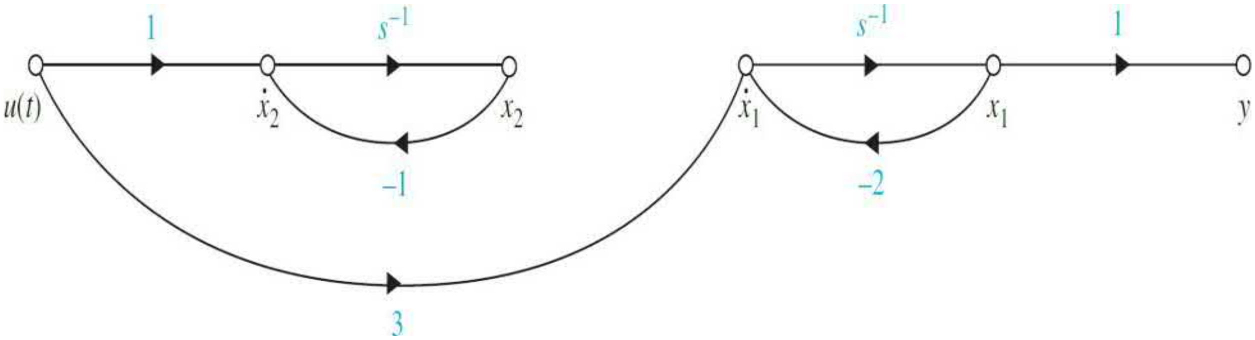


Figure 8-19 State diagram of a system that is not observable.

8-13-1 Definition of Observability

Given a linear time-invariant system that is described by the dynamic equations of Eqs. (8-261) and (8-262), the state $\mathbf{x}(t_0)$ *is said to be observable if given any input* $\mathbf{u}(t)$, *there exists a finite time* $t_f \geq t_0$ *such that the knowledge of* $\mathbf{u}(t)$ *for* $t_0 \leq t < t_f$ *, matrices* **A**, **B**, **C**, *and* **D**; *and the output* $\mathbf{y}(t)$ *for* $t_0 \leq t < t_0$ *are sufficient to determine* $\mathbf{x}(t_0)$. *If every state of the system is observable for a finite* t_0 *, we say that the system is completely observable, or, simply, observable*.

The following theorem shows that the condition of observability depends on the matrices **A** and **C** of the system. The theorem also gives one method of testing observability.

▪ **Theorem 8-4.** *For the system described by Eqs. (8-261) and (8-272) to be completely observable, it is necessary and sufficient that the following n × np* ***observability matrix*** *has a rank of n*

$$\mathbf{V} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix} \tag{8-272}$$

The condition is also referred to as the pair [**A**, **C**] *being observable. In particular, if the system has only one output,* **C** *is a* 1 × *n row matrix*; **V** *is an n* × *n square matrix. Then the system is completely observable if* **V** *is nonsingular*.

The proof of this theorem is not given here. It is based on the principle that Eq. (8-272) must be satisfied so that $\mathbf{x}(t_0)$ can be uniquely determined from the output $\mathbf{y}(t)$.

8-13-2 Alternate Tests on Observability

Just as with controllability, there are several alternate methods of testing observability. These are described in the following theorems.

▪ **Theorem 8-5.** *For a SISO system, described by the dynamic equations of Eqs. (8-261) and (8-262) with $r = 1$ and $p = 1$, the pair* [**A**, **C**] *is completely observable if* **A** *and* **C** *are in OCF or transformable into OCF by a similarity transformation.*

The proof of this theorem is straightforward, since it was established in Sec. 8-10 that the OCF transformation requires that the observability matrix **V** be nonsingular.

▪ **Theorem 8-6.** *For a system described by the dynamic equations of Eqs. (8-261) and (8-262), if* **A** *is in DCF or JCF, the pair* [**A**, **C**] *is completely observable if all the elements in the columns of* **C** *that correspond to the first row of each Jordan block are nonzero.*

Note that this theorem is a dual of the test of controllability given in Theorem 8-3. If the system has distinct eigenvalues, **A** is diagonal, then the condition on observability is that none of the columns of **C** can contain all zeros.

EXAMPLE 8-13-1 Consider the system shown in Fig. 8-19, which was earlier defined to be unobservable. The dynamic equations of the system are expressed in the form of Eqs. (8-261) and (8-262) with

$$\mathbf{A} = \begin{bmatrix} -2 & 0 \\ 0 & -1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 3 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix} \qquad (8\text{-}273)$$

Thus, the observability matrix is

$$\mathbf{V} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -2 & 0 \end{bmatrix} \qquad (8\text{-}274)$$

which is singular. Thus, the pair [**A, C**] is unobservable. In fact, because **A** is of DCF and the second column of **C** is zero, this means that the state $x_2(t)$ is unobservable, as conjectured from Fig. 8-18. ▲

8-14 RELATIONSHIP AMONG CONTROLLABILITY, OBSERVABILITY, AND TRANSFER FUNCTIONS

In the classical analysis of control systems, transfer functions are used for modeling of linear time-invariant systems. Although controllability and observability are concepts and tools of modern control theory, we shall show that they are closely related to the properties of transfer functions.

▪ **Theorem 8-7.** *If the input-output transfer function of a linear system has pole-zero cancellation, the system will be uncontrollable or unobservable, or both, depending on how the state variables are defined. On the other hand, if the input-output transfer function does not have pole-zero cancellation, the system can always be represented by dynamic equations as a completely controllable and observable system.*

The proof of this theorem is not given here. The importance of this theorem is that, if a linear system is modeled by a transfer function with no pole-zero cancellation, then we are assured that it is a controllable and observable system, no matter how the state-variable model is derived. Let us amplify this point further by referring to the following SISO system.

$$\mathbf{A}=\begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 \\ 0 & 0 & -3 & 0 \\ 0 & 0 & 0 & -4 \end{bmatrix} \quad \mathbf{B}=\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{C}=\begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix} \quad \mathbf{D}=0 \qquad (8\text{-}275)$$

Because **A** is a diagonal matrix, the controllability and observability conditions of the four states are determined by inspection. They are as follows:

x_1: Controllable and observable (C and O)

x_2: Controllable but unobservable (C but UO)

x_3: Uncontrollable but observable (UC but O)

x_4: Uncontrollable and unobservable (UC and UO)

The block diagram of the system in Fig. 8-20 shows the DCF decomposition of the system. Clearly, the transfer function of the controllable and observable system should be

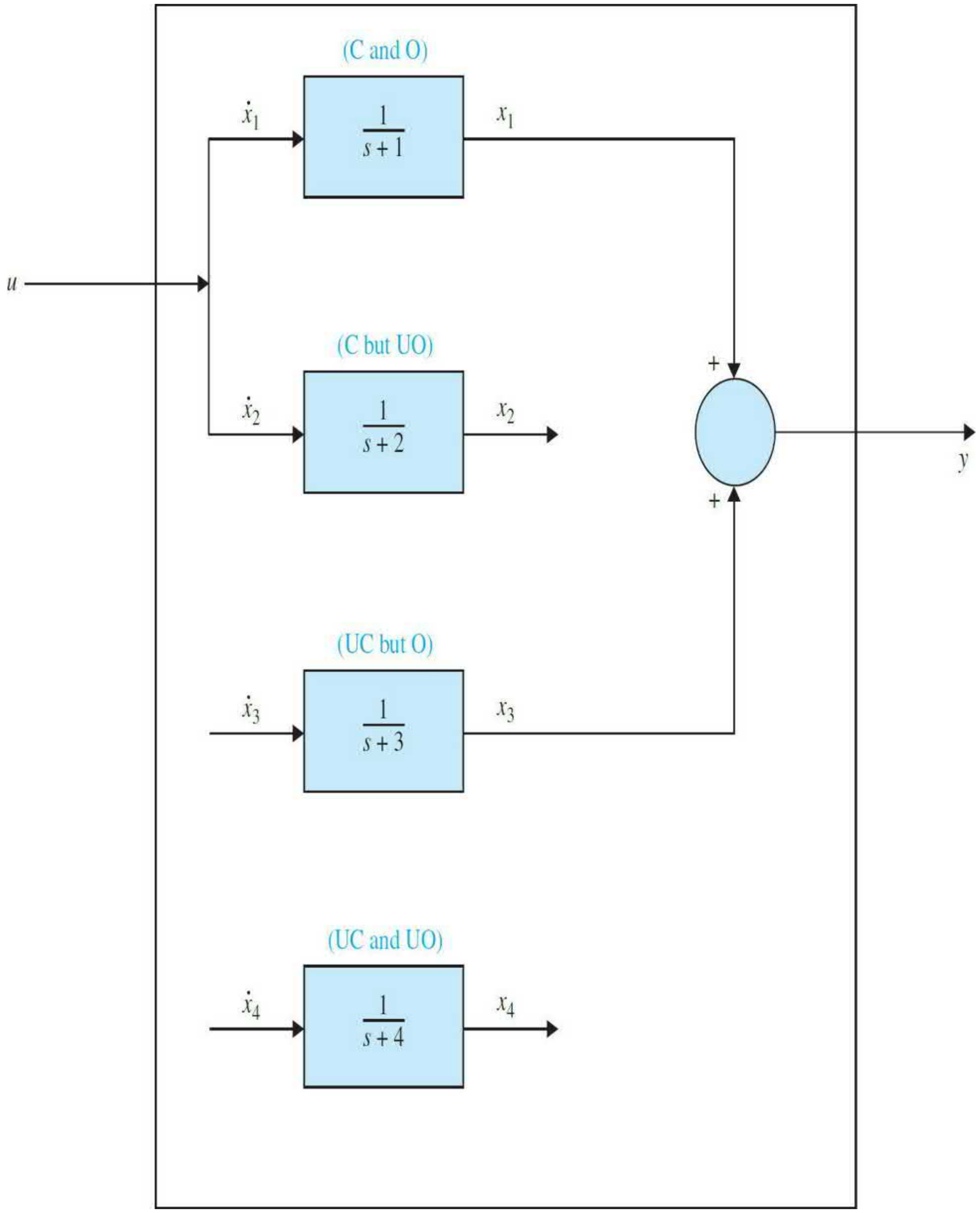


Figure 8-20 Block diagram showing the controllable, uncontrollable, observable, and unobservable components of the system described in Eq. (8-275).

$$\frac{Y(s)}{U(s)} = \frac{1}{s+1} \tag{8-276}$$

whereas the transfer function that corresponds to the dynamics described in Eq. (8-275) is

$$\frac{Y(s)}{U(s)} = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} = \frac{(s+2)(s+3)(s+4)}{(s+1)(s+2)(s+3)(s+4)} \tag{8-277}$$

which has three pole-zero cancellations. This simple-minded example illustrates that a “minimum-order” transfer function without pole-zero cancellation is the only component that corresponds to a system that is controllable and observable.

EXAMPLE 8-14-1 Let us consider the transfer function

$$\frac{Y(s)}{U(s)} = \frac{s+2}{(s+1)(s+2)} \tag{8-278}$$

which is a reduced form of Eq. (8-277). Equation (8-278) is decomposed into CCF and OCF as follows:

CCF:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 1 \end{bmatrix} \tag{8-279}$$

Because the CCF transformation can be made, the pair [**A**, **B**] of the CCF is controllable. The observability matrix is

$$\mathbf{V} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -2 & -2 \end{bmatrix} \tag{8-280}$$

which is singular, and the pair [A, C] of the CCF is unobservable.

OCF:

$$\mathbf{A}=\begin{bmatrix} 0 & -2 \\ 1 & -3 \end{bmatrix} \quad \mathbf{B}=\begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \mathbf{C}=\begin{bmatrix} 0 & 1 \end{bmatrix} \tag{8-281}$$

Because the OCF transformation can be made, the pair [**A**, **C**] of the OCF is observable. However, the controllability matrix is

$$\mathbf{S}=\begin{bmatrix} \mathbf{B} & \mathbf{AB} \end{bmatrix}=\begin{bmatrix} 1 & -2 \\ 1 & -2 \end{bmatrix} \tag{8-282}$$

which is singular, and the pair [A, B] of the OCF is uncontrollable.

The conclusion that can be drawn from this example is that, given a system that is modeled by transfer function, the controllability and observability conditions of the system depend on how the state variables are defined. ▲

8-15 INVARIANT THEOREMS ON CONTROLLABILITY AND OBSERVABILITY

We now investigate the effects of the similarity transformations on controllability and observability. The effects of controllability and observability due to state feedback will be investigated.

▪ **Theorem 8-8.** *Invariant theorem on similarity transformations: Consider that the system described by the dynamic equations of* Eqs. (8-261) *and* (8-262). *The similarity transformation* **x**(t) = **Px**(t), *where* **P** *is nonsingular, transforms the dynamic equations to*

$$\frac{d\overline{\mathbf{x}}(t)}{dt}=\overline{\mathbf{A}}\overline{\mathbf{x}}(t)+\overline{\mathbf{B}}\mathbf{u}(t) \tag{8-283}$$

$$\bar{\mathbf{y}}(t)=\bar{\mathbf{C}}\mathbf{x}(t)+\bar{\mathbf{D}}\mathbf{u}(t) \tag{8-284}$$

where

$$\bar{\mathbf{A}}=\mathbf{P}^{-1}\mathbf{AP} \quad \bar{\mathbf{B}}=\mathbf{P}^{-1}\mathbf{B} \tag{8-285}$$

The controllability of [**Ā, B**] *and the observability of* [**Ā, C**] *are not affected by the transformation.*

In other words, controllability and observability are preserved through similar transformations. The theorem is easily proven by showing that the ranks of **S** and **S** and the ranks of **V** and **V** are identical, where **S** and **V** are the controllability and observability matrices, respectively, of the transformed system.

▪ **Theorem 8-9.** *Theorem on controllability of closed-loop systems with state feedback: If the open-loop system*

$$\frac{d\mathbf{x}(t)}{dt}=\mathbf{Ax}(t)+\mathbf{Bu}(t) \tag{8-286}$$

is completely controllable, then the closed-loop system obtained through state feedback,

$$\mathbf{u}(t)=\mathbf{r}(t)-\mathbf{Kx}(t) \tag{8-287}$$

so that the state equation becomes

$$\frac{d\mathbf{x}(t)}{dt}=(\mathbf{A}-\mathbf{BK})\mathbf{x}(t)+\mathbf{Br}(t) \tag{8-288}$$

is also completely controllable. On the other hand, if [**A**, **B**] *is uncontrollable, then there is no* **K** *that will make the pair* [**A** − **BK**, **B**] *controllable. In other words, if an open-loop system is uncontrollable, it cannot be made controllable through state feedback.*

Proof: The controllability of [**A**, **B**] implies that there exists a control **u**(t) over the time interval [t_0, t_f] such that **the initial state x**(t_0) is driven to the final state **x**(t_f) over the finite time interval $t_f - t_0$. We can write Eq. (8-252) as

$$\mathbf{r}(t) = \mathbf{u}(t) + \mathbf{K}\mathbf{x}(t) \tag{8-289}$$

which is the control of the closed-loop system. Thus, if $\mathbf{u}(t)$ exists that can drive $\mathbf{x}(t_0)$ to any $\mathbf{x}(t_f)$ in finite time, then we cannot find an input $\mathbf{r}(t)$ that will do the same to $\mathbf{x}(t)$, because otherwise we can set $\mathbf{u}(t)$ as in Eq. (8-287) to control the open-loop system.

▪ **Theorem 8-10.** *Theorem on observability of closed-loop systems with state feedback: If an open-loop system is controllable and observable, then state feedback of the form of* Eq. (8-287) *could destroy observability. In other words, the observability of open-loop and closed-loop systems due to state feedback is unrelated*.

The following example illustrates the relation between observability and state feedback.

EXAMPLE 8-15-1 Let the coefficient matrices of a linear system be

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 2 \end{bmatrix} \tag{8-290}$$

We can show that the pair [A, B] is controllable and [A, C] is observable.

Let the state feedback be defined as

$$u(t) = r(t) - \mathbf{K}\mathbf{x}(t) \tag{8-291}$$

where

$$\mathbf{K} = [k_1 \quad k_2] \tag{8-292}$$

Then the closed-loop system is described by the state equation

$$\frac{d\mathbf{x}(t)}{dt} = (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}(t) + \mathbf{B}\mathbf{r}(t) \tag{8-293}$$

$$\mathbf{A} - \mathbf{B}\mathbf{K} = \begin{bmatrix} -k_1 & 1 - k_2 \\ -2 - k_1 & -3 - g_2 \end{bmatrix} \tag{8-294}$$

The observability matrix of the closed-loop system is

$$\mathbf{V} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}(\mathbf{A}-\mathbf{BK}) \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ -k_1-4 & -3k_2-5 \end{bmatrix} \tag{8-295}$$

The determinant of **V** is

$$|\mathbf{V}| = 6k_1 - 3k_2 + 3 \tag{8-296}$$

Thus, if k_1 and k_2 are chosen so that |**V**| = 0, the closed-loop system would be uncontrollable. ▲

8-16 CASE STUDY: MAGNETIC-BALL SUSPENSION SYSTEM

As a case study to illustrate some of the material presented in this chapter, let us consider the magnetic-ball suspension system that was earlier studied in Example 3-9-2 and is shown in Fig. 8-21. The objective of the system is to regulate the current of the electromagnet so that the ball will be suspended at a fixed distance from the end of the magnet. The dynamic equations of the system are

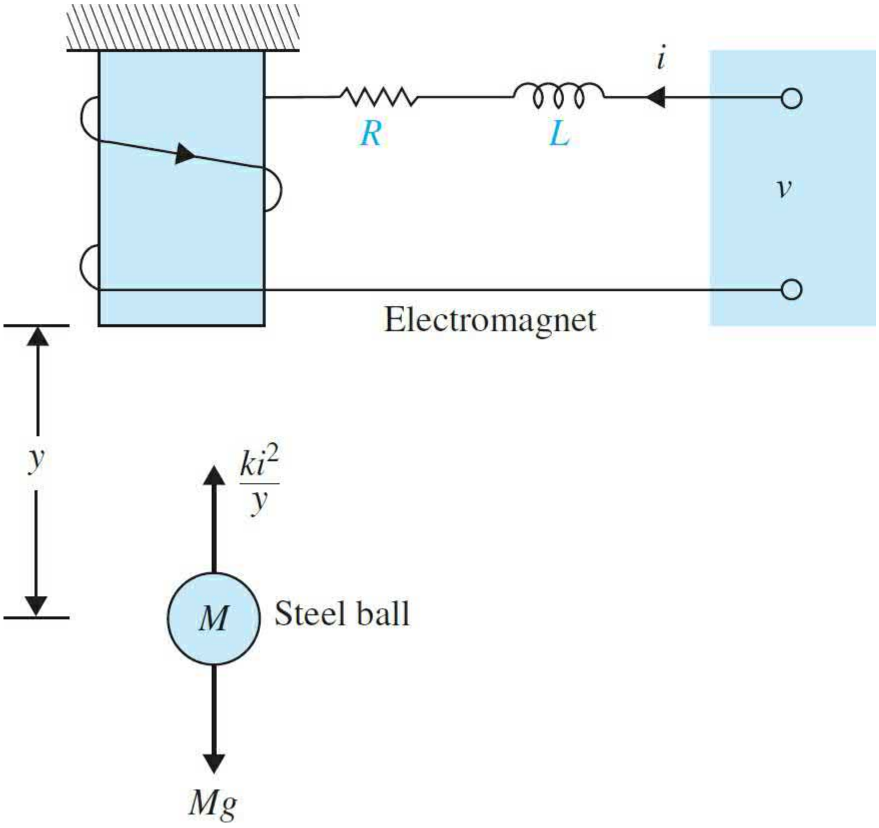


Figure 8-21 Ball-suspension system.

$$M\frac{d^2y(t)}{dt^2} = Mg - \frac{ki^2(t)}{x(t)} \tag{8-297}$$

$$v(t) = Ri(t) + L\frac{di(t)}{dt} \tag{8-298}$$

where Eq. (8-262) is nonlinear. The system variables and parameters are as follows:

$v(t) = \text{input voltage (V)}$

$i(t) = \text{winding current (A)}$

$R = \text{winding resistance} = 1\,\Omega$

$M = \text{ball mass} = 1.0\text{ kg}$

$x(t) = \text{ball position (m)}$

$k = \text{proportional constant} = 1.0$

$L = \text{winding inductance} = 0.01\text{ H}$

$g = \text{gravitational acceleration} = 32.2\text{ m/s}^2$

The state variables are defined as

$$\begin{aligned} x_1(t) &= x(t) \\ x_2(t) &= \frac{dx(t)}{dt} \\ x_3(t) &= i(t) \end{aligned} \qquad (8\text{-}299)$$

The state equations are

$$\frac{dx_1(t)}{dt} = x_2(t) \qquad (8\text{-}300)$$

$$\frac{dx_2(t)}{dt} = g - \frac{k}{M}\frac{x_3^2(t)}{x_1(t)} \qquad (8\text{-}301)$$

$$\frac{dx_3(t)}{dt} = -\frac{R}{L}x_3(t) + \frac{v(t)}{L} \qquad (8\text{-}302)$$

Let us linearize the system, using the method described in Sec. 3-10, about the equilibrium point y_0(t) = x_{01} = 0.5 m. Then, $x_{02}(t) = \frac{dx_{01}(t)}{dt} = 0$ and $\frac{d^2 y_0(t)}{dt^2} = 0.$ After substituting the parameter values, the linearized equations are written

$$\Delta\dot{\mathbf{x}}(t) = \mathbf{A}^* \Delta\mathbf{x}(t) + \mathbf{B}^* \Delta v(t) \qquad (8\text{-}303)$$

where Δ**x**(*t*) denotes the state vector, and Δ*v*(308) is the input voltage of the linearized system. The coefficient matrices are

$$\mathbf{A}^* = \begin{bmatrix} 0 & 1 & 0 \\ 64.4 & 0 & -16 \\ 0 & 0 & -100 \end{bmatrix} \quad \mathbf{B}^* = \begin{bmatrix} 0 \\ 0 \\ 100 \end{bmatrix} \qquad (8\text{-}304)$$

All the computations done in the following section can be carried out with the MATLAB state tool (Sec. 8-20). To show the analytical method, we carry out the steps of the derivations as follows.

8-16-1 The Characteristic Equation

$$|s\mathbf{I}-\mathbf{A}^*| = \begin{bmatrix} s & -1 & 0 \\ -64.4 & s & 16 \\ 0 & 0 & s+100 \end{bmatrix} = s^3+100s^2-64.4s-6440=0 \qquad (8\text{-}305)$$

Eigenvalues

The eigenvalues of **A***, or the roots of the characteristic equation, are

s = –100 *s*–8.025 *s* = 8.025

The State-Transition Matrix

The state-transition matrix of **A*** is

$$\phi(t) =^{-1} [(s\mathbf{I}-\mathbf{A}^*)^{-1}] =^{-1} \left(\begin{bmatrix} s & -1 & 0 \\ -64.4 & s & 16 \\ 0 & 0 & s+100 \end{bmatrix}^{-1} \right) \qquad (8\text{-}306)$$

or

$$\phi(t) =^{-1} \left(\frac{1}{(s+100)(s+8.025)(s-8.025)} \begin{bmatrix} s(s+100) & s+100 & -16 \\ 64.4(s+100) & s(s+100) & -16s \\ 0 & 0 & s^2-64.4 \end{bmatrix} \right) \qquad (8\text{-}307)$$

By performing the partial-fraction expansion and carrying out the inverse Laplace transform, the state-transition matrix is

$$\phi(t)=\begin{bmatrix} 0 & 0 & -0.0016 \\ 0 & 0 & 0.16 \\ 0 & 0 & 1 \end{bmatrix} e^{-100t} + \begin{bmatrix} 0.5 & -0.062 & 0.0108 \\ -4.012 & 0.5 & -0.087 \\ 0 & 0 & 0 \end{bmatrix} e^{-8.025t}$$
$$+\begin{bmatrix} 0.5 & 0.062 & -0.0092 \\ 4.012 & 0.5 & -0.074 \\ 0 & 0 & 0 \end{bmatrix} e^{8.025t} \qquad (8\text{-}308)$$

Because the last term in Eq. (8-308) has a positive exponent, the response of *f*(*t*) increases with time, and the system is unstable. This is expected, since without control, the steel ball would be attracted by the magnet until it hits the bottom of the magnet.

Transfer Function

Let us define the ball position *x*(*t*) as the output *y*(*t*); then, given the input, *v*(*t*), the input-output transfer function of the system is

$$\frac{Y(s)}{V(s)} = \mathbf{C}^*(s\mathbf{I}-\mathbf{A}^*)^{-1}\mathbf{B}^* = [1 \quad 0 \quad 0](s\mathbf{I}-\mathbf{A}^*)^{-1}\mathbf{B}^* = \frac{-1600}{(s+100)(s+8.025)(s-8.025)} \qquad (8\text{-}309)$$

Controllability

The controllability matrix is

$$\mathbf{S} = \begin{bmatrix} \mathbf{B}^* & \mathbf{A}^*\mathbf{B}^* & \mathbf{A}^*2\mathbf{B}^* \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1{,}600 \\ 0 & -1{,}600 & 160{,}000 \\ 100 & -10{,}000 & 1{,}000{,}000 \end{bmatrix} \qquad (8\text{-}310)$$

Because the rank of **S** is 3, the system is completely controllable.

Observability

The observability of the system depends on which variable is defined at the output. For state-feedback control, which will be discussed later in Chap. 10, the full controller requires feeding back all three state variables, x_1, x_2, and x_3. However, for reasons of economy, we may want to feed back only one of the three state variables. To make the problem more general, we may want to investigate which state, if chosen as the output, would render the system unobservable.

1. *y*(*t*) = ball position = *x*(*t*): **C*** = [1 0 0]. The observability matrix is

$$\mathbf{V}=\begin{bmatrix}\mathbf{C}^{*}\\ \mathbf{C}^{*}\mathbf{A}^{*}\\ \mathbf{C}^{*}\mathbf{A}^{*}2\end{bmatrix}=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 64.4 & 0 & -16\end{bmatrix} \qquad (8\text{-}311)$$

   which has a rank of 3. Thus, the system is completely observable.

2. *y*(*t*) = all velocity = *dx*(*t*)/*dt*: **C*** = [0 1 0]. The observability matrix is

$$\mathbf{V}=\begin{bmatrix}\mathbf{C}^{*}\\ \mathbf{C}^{*}\mathbf{A}^{*}\\ \mathbf{C}^{*}\mathbf{A}^{*}2\end{bmatrix}=\begin{bmatrix}0 & 1 & 0\\ 64.4 & 0 & -16\\ 0 & 64.4 & 1600\end{bmatrix} \qquad (8\text{-}312)$$

   which has a rank of 3. Thus, the system is completely observable.

3. *y*(*t*) = winding current = *i*(*t*): **C*** = [0,0,1]

   The observability matrix is

$$\mathbf{V}=\begin{bmatrix}\mathbf{C}^{*}\\ \mathbf{C}^{*}\mathbf{A}^{*}\\ \mathbf{C}^{*}\mathbf{A}^{*}2\end{bmatrix}=\begin{bmatrix}0 & 0 & 1\\ 0 & 0 & -100\\ 0 & 0 & -10{,}000\end{bmatrix} \qquad (8\text{-}313)$$

which has a rank of 1. Thus, the system is unobservable. The physical interpretation of this result is that, if we choose the current *i*(*t*) as the measurable output, we would not be able to reconstruct the state variables from the measured information.

The interested reader can enter the data of this system into any available computer program and verify the results obtained.

8-17 STATE-FEEDBACK CONTROL

A majority of the design techniques in modern control theory is based on the state-feedback configuration. That is, instead of using controllers with fixed configurations in the forward or feedback path, control is achieved by feeding back the state variables through real constant gains. The block diagram of a system with state-feedback control is shown in Fig. 8-22.

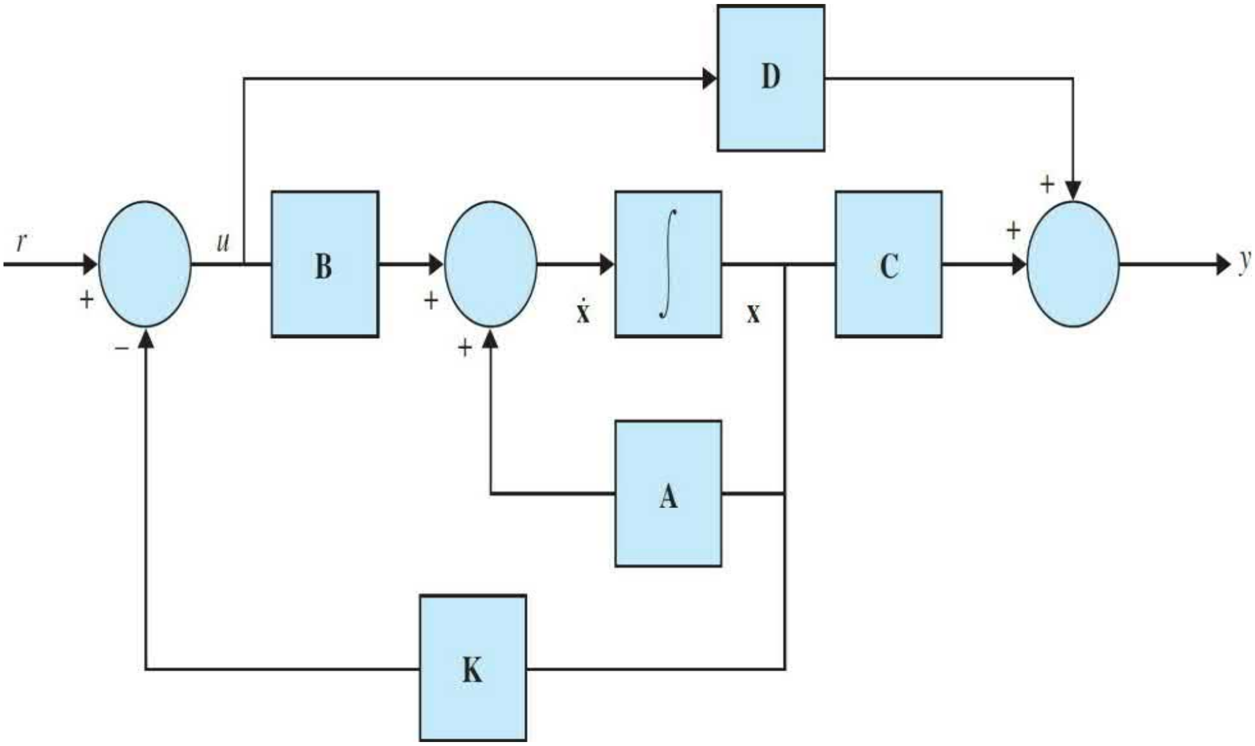


Figure 8-22 Block diagram of a control system with state feedback.

We can show that the PID control and the tachometer-feedback control discussed earlier are all special cases of the state-feedback control scheme. Let us consider a second-order prototype system with tachometer-feedback control. The process is decomposed by direct decomposition and is represented by the state diagram of Fig. 8-23*a*. If the states $x_1(t)$ and $x_2(t)$ are physically accessible, these variables may be fed back through constant real gains $-k_1$ and $-k_2$, respectively, to form the control $u(t)$, as shown in Fig. 8-23*b*. The transfer function of the system with state feedback is

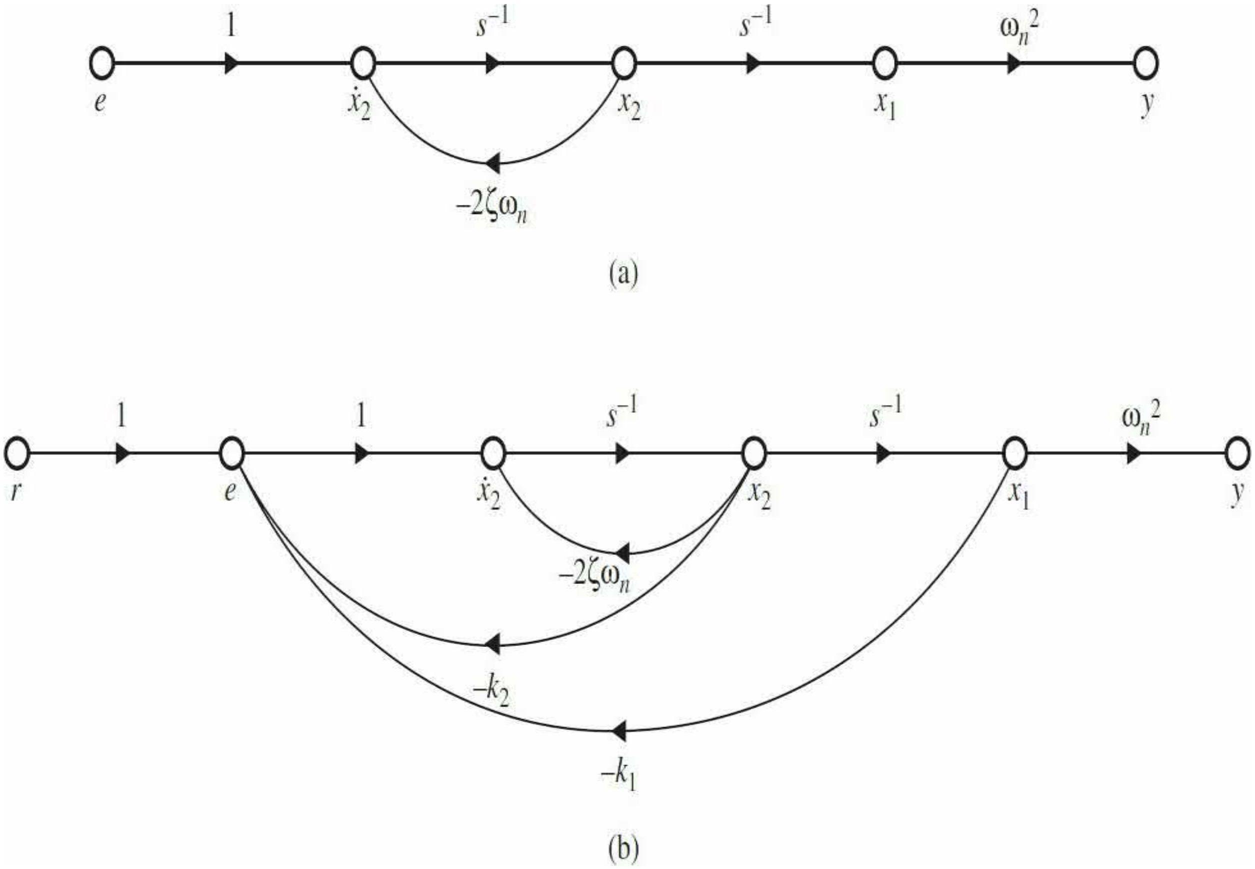


Figure 8-23 Control of a second-order system by state feedback.

$$\frac{Y(s)}{R(s)}=\frac{\omega_n^2}{s^2+(2\zeta\omega_n+K_2)s+K_1} \tag{8-314}$$

For comparison purposes, we display the transfer functions of the systems with tachometer feedback and with PD control as follows:

Tachometer feedback:

$$\frac{Y(s)}{R(s)}=\frac{\omega_n^2}{s^2+\left(2\zeta\omega_n+K_t\omega_n^2\right)s+\omega_n^2} \tag{8-315}$$

PD control:

$$\frac{Y(s)}{R(s)}=\frac{\omega_n^2(K_P+K_Ds)}{s^2+\left(2\zeta\omega+K_D\omega_n^2\right)s+\omega_n^2K_P} \tag{8-316}$$

Thus, tachometer feedback is equivalent to state feedback if $k_1 = \omega^2_n$ and $k_2 = K_t\omega^2_n$. Comparing Eq. (8-314) with Eq. (8-316), we see that the characteristic equation of the system with state feedback would be identical to that of the system with PD control if $k_1 = \omega^2_n K_p$ and $k_2 = \omega^2_n K_D$. However, the numerators of the two transfer functions are different.

The systems with zero reference input, $r(t) = 0$, are commonly known as **regulators**. When $r(t) = 0$, the control objective is to drive any arbitrary initial conditions of the system to zero in some prescribed manner, for example, "as quickly as possible." Then a second-order system with PD control is the same as state-feedback control.

It should be emphasized that the comparisons just made are all for second-order systems. For higher-order systems, the PD control and tachometer-feedback control are equivalent to feeding back only the state variables x_1 and x_2, while state-feedback control feeds back all the state variables.

Because PI control increases the order of the system by one, it cannot be made equivalent to state feedback through constant gains. We show in Sec. 8-18 that if we combine state feedback with integral control we can again realize PI control in the sense of state-feedback control.

8-18 POLE-PLACEMENT DESIGN THROUGH STATE FEEDBACK

When root loci are utilized for the design of control systems, the general approach may be described as that of **pole placement**; the poles here refer to that of the closed-loop transfer function, which are also the roots of the characteristic equation. Knowing the relation between the closed-loop poles and the system performance, we can effectively carry out the design by specifying the location of these poles.

The design methods discussed in the preceding sections are all characterized by the property that the poles are selected based on what can be achieved with the fixed-controller configuration and the physical range of the controller parameters. A natural question would be: *Under what condition can the poles be placed arbitrarily*? This is an entirely new design philosophy and freedom that apparently can be achieved only under certain conditions.

When we have a controlled process of the third order or higher, the PD, PI, single-stage phase-lead, and phase-lag controllers would not be able to control independently all the poles of the system because there are only two free parameters in each of these controllers.

To investigate the condition required for arbitrary pole placement in an *n*th-order system, let us consider that the process is described by the following state equation:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{8-317}$$

where **x**(*t*) is an *n* × 1 state vector, and *u*(*t*) is the scalar control. The state-feedback control is

$$u(t) = -\mathbf{K}\mathbf{x}(t) + r(t) \tag{8-318}$$

where **K** is the 1 × *n* feedback matrix with constant-gain elements. By substituting Eq. (8-318) into Eq. (8-317), the closed-loop system is represented by the state equation

$$\frac{d\mathbf{x}(t)}{dt} = (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}(t) + \mathbf{B}r(t) \tag{8-319}$$

It will be shown in the following that if the pair [**A, B**] is completely controllable, then a matrix **K** exists that can give an arbitrary set of eigenvalues of (**A** – **BK**); that is, the *n* roots of the characteristic equation

$$\left| s\mathbf{I} - \mathbf{A} + \mathbf{B}\mathbf{K} \right| = 0 \tag{8-320}$$

can be arbitrarily placed. To show that this is true, that if a system is completely controllable, it can always be represented in the controllable canonical form (CCF); that is, in Eq. (8-317),

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \ddots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix} \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \tag{8-321}$$

The feedback gain matrix **K** is expressed as

$$\mathbf{K} = [k_1 \quad k_2 \quad \cdots \quad k_n] \tag{8-322}$$

where k_1, k_2, …, k_n are real constants. Then,

$$\mathbf{A} - \mathbf{BK} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 - k_1 & -a_1 - k_2 & -a_2 - k_3 & \cdots & -a_{n-1} - k_n \end{bmatrix} \tag{8-323}$$

The eigenvalues of **A** − **BK** are then found from the characteristic equation

$$\left| s\mathbf{I} - (\mathbf{A} - \mathbf{BK}) \right| = s^n + (a_{n-1} + k_n)s^{n-1} + (a_{n-2} + k_{n-1})s^{n-2} + \cdots + (a_0 + k_1) = 0 \tag{8-324}$$

Clearly, the eigenvalues can be arbitrarily assigned because the feedback gains k_1, k_2, …, k_n are isolated in each coefficient of the characteristic equation. Intuitively, it makes sense that a system must be controllable for the poles to be placed arbitrarily. If one or more state variables are uncontrollable, then the poles associated with these state variables are also uncontrollable and cannot be moved as desired. The following example illustrates the design of a control system with state feedback.

EXAMPLE 8-18-1 Consider the magnetic-ball suspension system analyzed in Sec. 8-16. This is a typical regulator system for which the control problem is to maintain the ball at its equilibrium position. It is shown in Sec. 8-16 that the system without control is unstable.

The linearized state model of the magnetic-ball system is represented by the state equation

$$\frac{d\Delta\mathbf{x}(t)}{dt} = \mathbf{A}^* \, \Delta\mathbf{x}(t) + \mathbf{B}^* \, \Delta v(t) \tag{8-325}$$

where Δx(*t*) denotes the linearized state vector, and Δ*v*(*t*) is the linearized input voltage. The coefficient matrices are

$$\mathbf{A}^* = \begin{bmatrix} 0 & 1 & 0 \\ 64.4 & 0 & -16 \\ 0 & 0 & -100 \end{bmatrix} \quad \mathbf{B}^* = \begin{bmatrix} 0 \\ 0 \\ 100 \end{bmatrix} \tag{8-326}$$

The eigenvalues of A* are *s* = −100, −8.025, and 8.025. Thus, the system without feedback control is unstable.

Let us give the following design specifications:

1. The system must be stable.
2. For any initial disturbance on the position of the ball from its equilibrium position, the ball must return to the equilibrium position with zero steady-state error.
3. The time response should settle to within 5 percent of the initial disturbance in not more than 0.5 s.
4. The control is to be realized by state feedback

$$\Delta v(t) = -\mathbf{K}\Delta\mathbf{x}(t) = -[k_1 \quad k_2 \quad k_3]\Delta\mathbf{x}(t) \tag{8-327}$$

where k_1, k_2, and k_3 are real constants.

A state diagram of the “open-loop” ball-suspension system is shown in Fig. 8-24*a*, and the same of the “closed-loop” system with state feedback is shown in Fig. 8-24*b*.

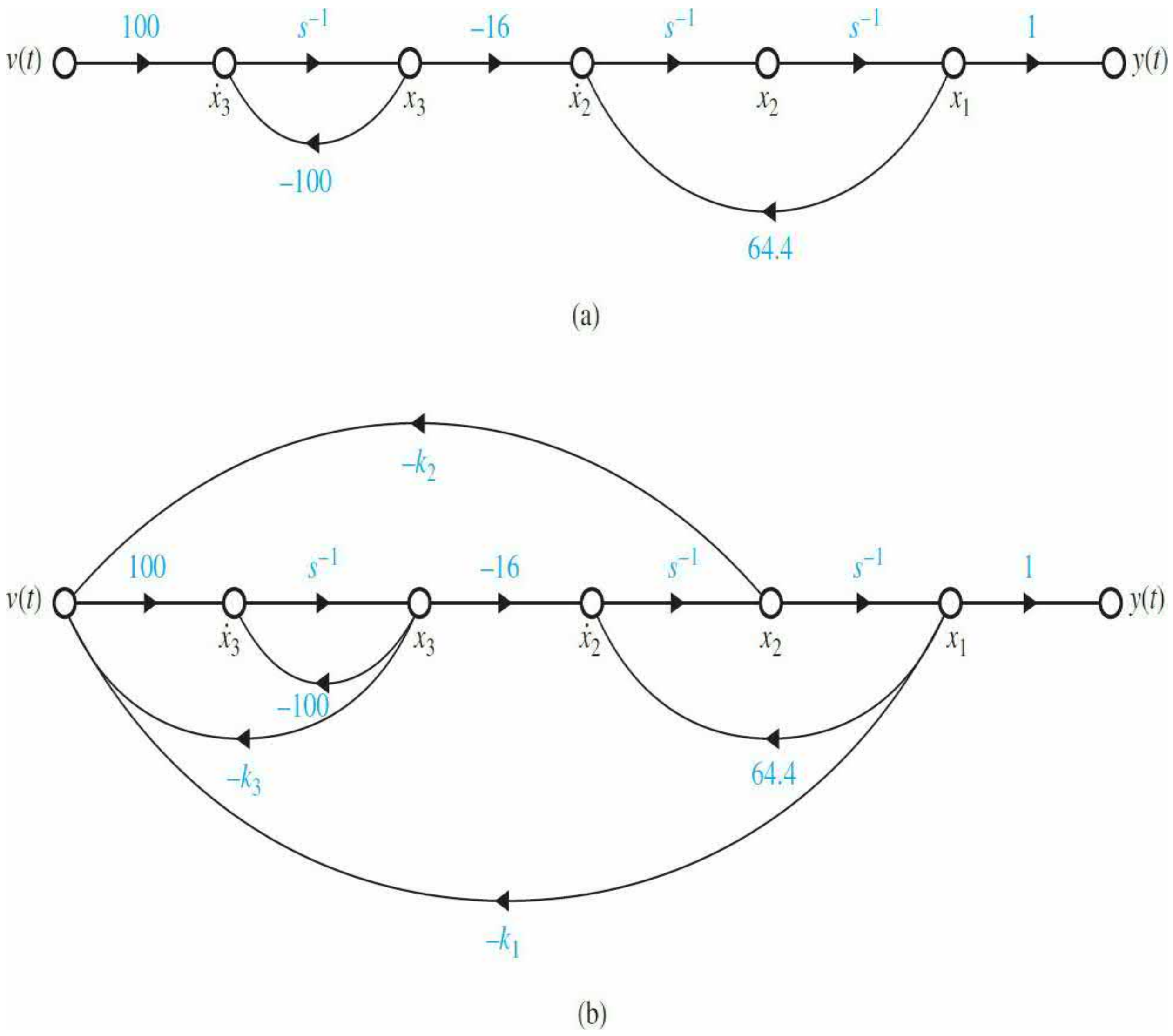


Figure 8-24 (*a*) State diagram of magnetic-ball-suspension system. (*b*) State diagram of magnetic-ball-suspension system with state feedback.

We must select the desired location of the eigenvalues of ($s\mathbf{I} - \mathbf{A}^* + \mathbf{B}^*\mathbf{K}$) so that requirement 3 in the preceding list on the time response is satisfied. Without entirely resorting to trial and error, we can start with the following decisions:

1. The system dynamics should be controlled by two dominant roots.
2. To achieve a relatively fast response, the two dominant roots should be complex.
3. The damping that is controlled by the real parts of the complex roots

should be adequate, and the imaginary parts should be high enough for the transient to die out sufficiently fast.

After a few trial-and-error runs, using the **ACSYS/MATLAB** tool (see Sec. 8-20), we found that the following characteristic equation roots should satisfy the design requirements:

$$s = -20 \quad s = -6 + j4.9 \quad s = -6 - j4.9$$

The corresponding characteristic equation is

$$s^3 + 32s^2 + 300s + 1200 = 0 \tag{8-328}$$

The characteristic equation of the closed-loop system with state feedback is written

$$s\mathbf{I} - \mathbf{A}^* + \mathbf{B}^*\mathbf{K} = \begin{vmatrix} s & -1 & 0 \\ -64.4 & s & 16 \\ 100k_1 & 100k_2 & s + 100 + 100k_3 \end{vmatrix}$$

$$= s^3 + 100(k_3 + 1)s^2 - (64.4 + 1600k_2)s - 1600k_1 - 6440(k_3 + 1) = 0 \tag{8-329}$$

which can also be obtained directly from Fig. 8-24*b* using the SFG gain formula. Equating like coefficients of Eqs. (8-328) and (8-329), we get the following simultaneous equations:

$$\begin{aligned} 100(k_3 + 1) &= 32 \\ -64.4 - 1600k_2 &= 300 \\ -1600k_1 - 6440(k_3 + 1) &= 1200 \end{aligned} \tag{8-330}$$

Solving the last three equations, and being assured that the solutions exist and are unique, we get the feedback-gain matrix

$$\mathbf{K} = \begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix} = \begin{bmatrix} -2.038 & -0.22775 & -0.68 \end{bmatrix} \tag{8-331}$$

Figure 8-25 shows the output response $y(t)$ when the system is subject to the initial condition

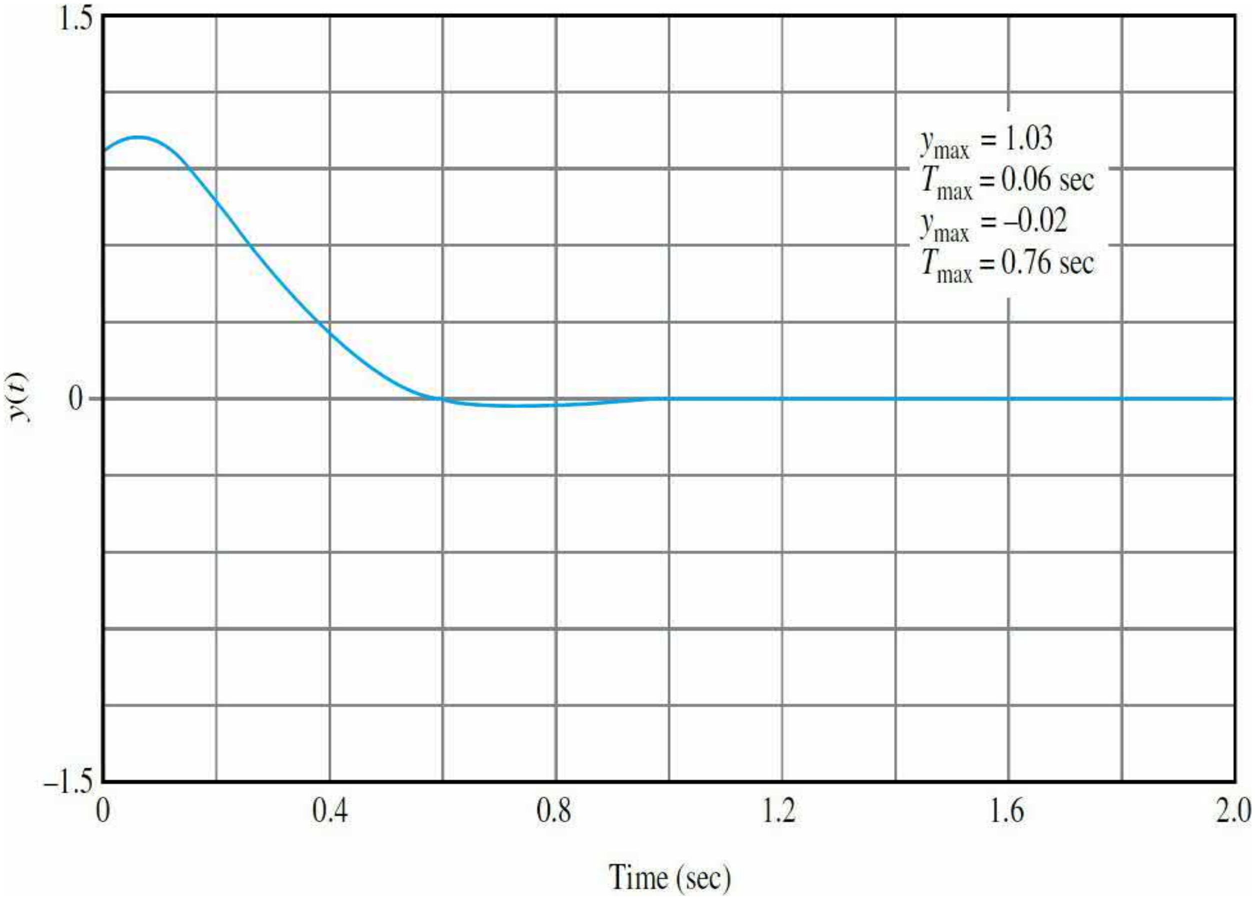


Figure 8-25 Output response of magnetic-ball-suspension system with state feedback, subject to initial condition $y(0) = x_1(0) = 1$.

$$\mathbf{x}(0) = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{8-331}$$

EXAMPLE 8-18-2 In this example, we shall design a state-feedback control for the second-order sun-seeker system treated in Example 6-5-1; also see Chap. 11. The CCF state diagram of the process with $K = 1$ is shown in Fig. 8-26*a*. The problem involves the design of a state-feedback control with

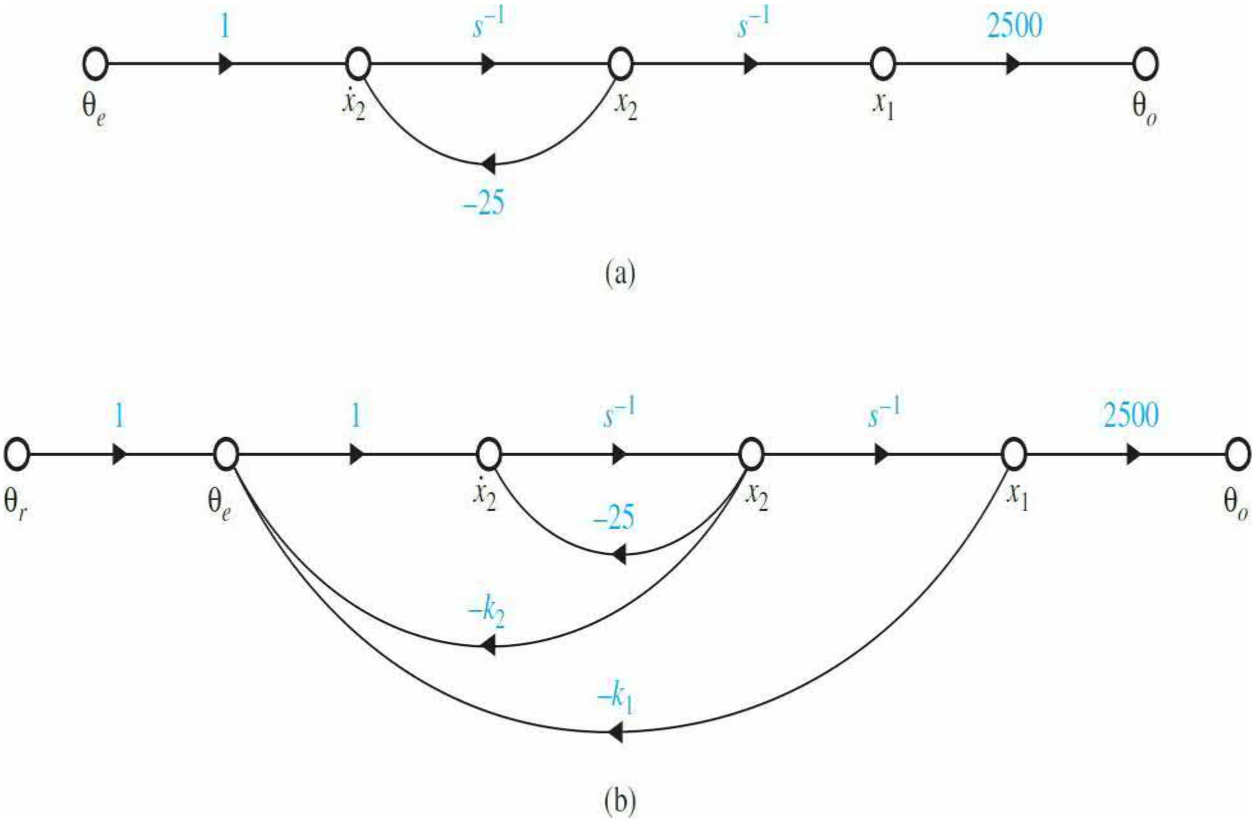


Figure 8-26 (*a*) State diagram of second-order sun-seeker system. (*b*) State diagram of second-order sun-seeker system with state feedback.

$$\theta_e(t) = -\mathbf{K}\mathbf{x}(t) = -[k_1 \quad k_2]\mathbf{x}(t) \tag{8-332}$$

The state equations are represented in vector-matrix form as

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\theta_e(t) \tag{8-333}$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & -25 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{8-334}$$

The output equation is

$$\theta_o(t) = \mathbf{C}\mathbf{x}(t) \tag{8-335}$$

where

$$\mathbf{C} = [1 \quad 0] \tag{8-336}$$

The design objectives are as follows:

1. The steady-state error due to a step function input should equal 0.
2. With the state-feedback control, the unit-step response should have minimum overshoot, rise time, and settling time.

The transfer function of the system with state feedback is written

$$\frac{\Theta_o(s)}{\Theta_r(s)} = \frac{2500}{s^2 + (25 + k_2)s + k_1} \tag{8-337}$$

Thus, for a step input, if the output has zero steady-state error, the constant terms in the numerator and denominator must be equal to each other—that is k_1 = 2500. This means that, while the system is completely controllable, we cannot arbitrarily assign the two roots of the characteristic equation, which is now

$$s^2 + (25 + k_2)s + 2500 = 0 \tag{8-338}$$

In other words, only one of the roots of Eq. (8-339) can be arbitrarily assigned. The problem is solved using **ACSYS** (see Sec. 8-20). After a few trial-and-error runs, we found out that the maximum overshoot, rise time, and settling time are all at a minimum when k_2 = 75. The two roots are s = −50 and −50. The attributes of the unit-step response are

Maximum overshoot = 0% *t*, = 0.06717s t_s = 0.09467s

The state-feedback gain matrix is

$$\mathbf{K} = [2500 \quad 75] \tag{8-339}$$

The lesson that we learned from this illustrative example is that state-feedback control generally produces a system that is type 0. For the system to track a step input without steady-state error, which requires a type 1 or higher-type system, the feedback gain k_1 of the system in the CCF state diagram cannot be assigned arbitrarily. This means that, for an *n*th-order system, only $n - 1$ roots of the characteristic equation can be placed arbitrarily.

8-19 STATE FEEDBACK WITH INTEGRAL CONTROL

The state-feedback control structured in the preceding section has one deficiency in that it does not improve the type of the system. As a result, the state-feedback control with constant-gain feedback is generally useful only for regulator systems for which the system does not track inputs, if all the roots of the characteristic equation are to be placed at will.

In general, most control systems must track inputs. One solution to this problem is to introduce integral control, just as with PI controller, together with the constant-gain state feedback. The block diagram of a system with constant-gain state feedback and integral control feedback of the output is shown in Fig. 8-27. The system is also subject to a noise input $n(t)$. For a SISO system, the integral control adds one integrator to the system. As shown in Fig. 8-27, the output of the $(n + 1)$st integrator is designated as x_{n+1}. The dynamic equations of the system in Fig. 8-27 are written as

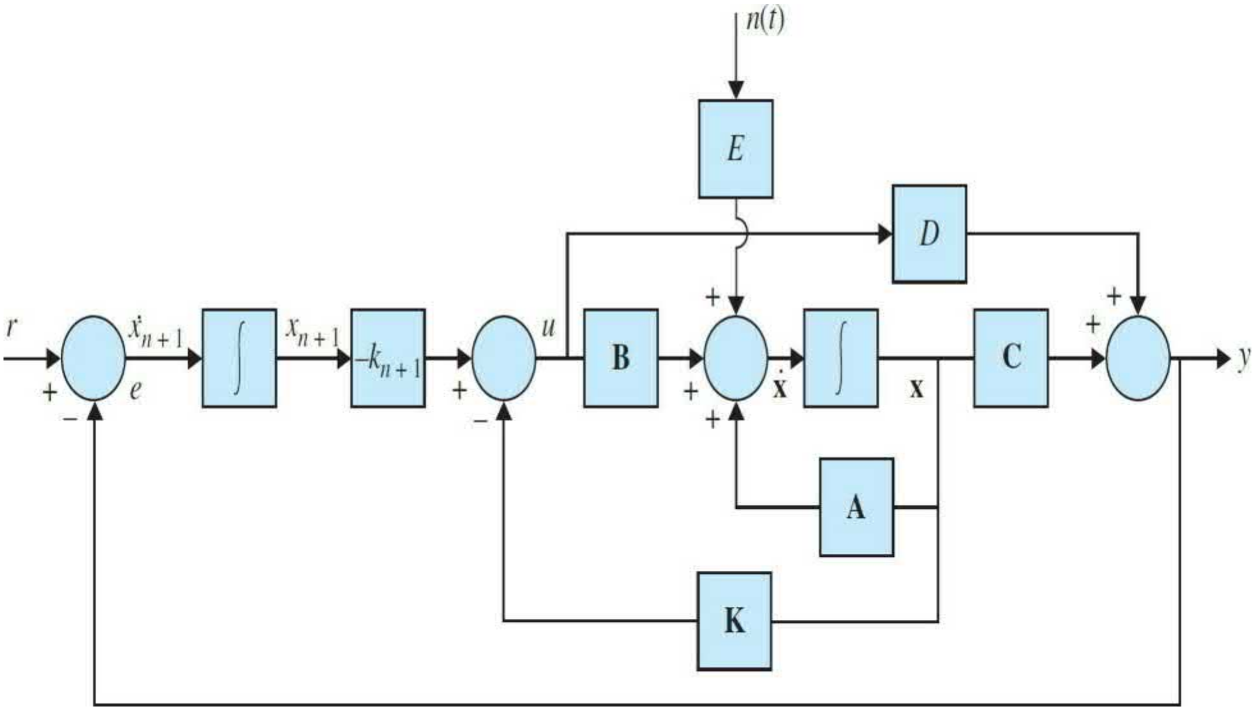


Figure 8-27 Block diagram of a control system with state feedback and integral output feedback.

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) + \mathbf{E}n(t) \tag{8-340}$$

$$\frac{dx_{n+1}(t)}{dt} = r(t) - y(t) \tag{8-341}$$

$$y(t) = \mathbf{C}\mathbf{x}(t) + Du(t) \tag{8-342}$$

where $\mathbf{x}(t)$ is the $n \times 1$ state vector; $u(t)$ and $y(t)$ are the scalar actuating signal and output, respectively; $r(t)$ is the scalar reference input; and $n(t)$ is the scalar disturbance input. The coefficient matrices are represented by **A, B, C**, D, and **E**, with appropriate dimensions. The actuating signal $u(t)$ is related to the state variables through constant-state and integral feedback,

$$u(t) = -\mathbf{K}\mathbf{x}(t) - k_{n+1}x_{n+1}(t) \tag{8-343}$$

where

$$\mathbf{K} = [k_1 \quad k_2 \quad k_3 \quad \cdots \quad k_n] \tag{8-344}$$

with constant real gain elements, and k_{n+1} is the scalar integral-feedback gain.

Substituting Eq. (8-343) into Eq. (8-340) and combining with Eq. (8-341), the $n + 1$ state equations of the overall system with constant-gain and integral feedback are written

$$\frac{d\overline{\mathbf{x}}(t)}{dt} = (\overline{\mathbf{A}} - \overline{\mathbf{B}}\overline{\mathbf{K}})\overline{\mathbf{x}}(t) + \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} r(t) + \overline{\mathbf{E}}n(t) \tag{8-345}$$

where

$$\bar{\mathbf{x}}(t) = \begin{bmatrix} \dfrac{d\mathbf{x}(t)}{dt} \\ \dfrac{dx_{n+1}(t)}{dt} \end{bmatrix} (n+1)\times 1 \tag{8-346}$$

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & 0 \\ -\mathbf{C} & \mathbf{0} \end{bmatrix} \quad (n+1)\times(n+1) \quad \bar{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ D \end{bmatrix} (n+1)\times 1 \tag{8-347}$$

$$\bar{\mathbf{K}} = \begin{bmatrix} K & K_{n+1} \end{bmatrix} = \begin{bmatrix} k_1 & k_2 & \cdots & k_n & k_{n+1} \end{bmatrix} 1\times(n+1) \tag{8-348}$$

$$\bar{\mathbf{E}} = \begin{bmatrix} \mathbf{E} \\ 0 \end{bmatrix} [(n+1)\times 1] \tag{8-349}$$

Substituting Eq. (8-343) into Eq. (8-342), the output equation of the overall system is written

$$y(t) = \bar{\mathbf{C}}\bar{\mathbf{x}}(t) \tag{8-350}$$

where

$$\bar{\mathbf{C}} = [\mathbf{C} - D\mathbf{K} \quad D\mathbf{K}][1\times(n+1)] \tag{8-351}$$

The design objectives are as follows:

1. The steady-state value of the output $y(t)$ follows a step-function input with zero error; that is,

$$e_{ss} = \lim_{t\to\infty} e(t) = 0 \tag{8-352}$$

2. The $n + 1$ eigenvalues of $(\bar{\mathbf{A}} - \bar{\mathbf{B}}\bar{\mathbf{K}})$ are placed at desirable locations. For the last condition to be possible, the pair $[\bar{\mathbf{A}}, \bar{\mathbf{B}}]$ must be completely controllable.

The following example illustrates the applications of state-feedback with integral control.

EXAMPLE 8-18-1 We have shown in Example 8-18-1 that, with constant-gain state-feedback control, the second-order sun-seeker system can have only one of its two roots placed at will for the system to track a step input without steady-state error. Now let us consider the same second-order sun-seeker system in Example 8-18-1, except that an integral control is added to the forward path. The state diagram of the overall system is shown in Fig. 8-28. The coefficient matrices are

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & -25 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 2500 & 0 \end{bmatrix} \quad D = 0 \tag{8-353}$$

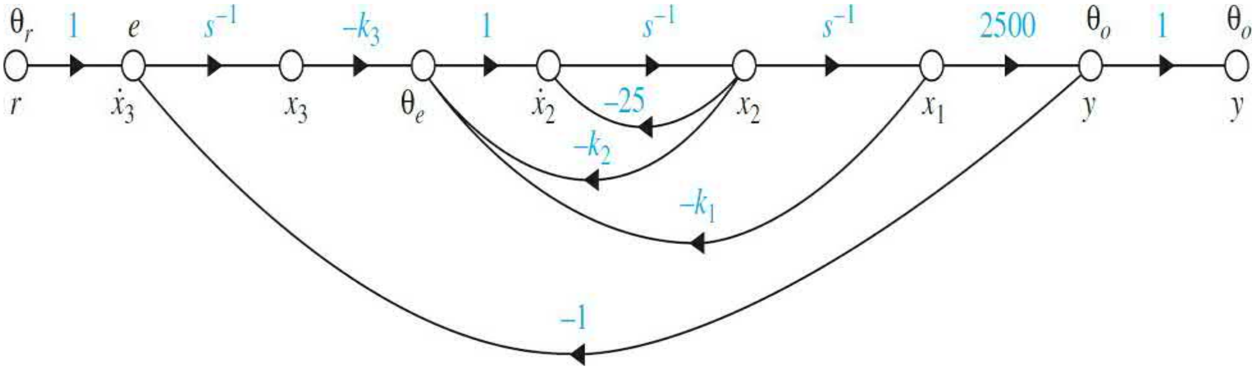


Figure 8-28 Sun-seeker system with state feedback and integral control in Example 8-18-1.

From Eq. (8-347),

$$\overline{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -25 & 0 \\ -2500 & 0 & 0 \end{bmatrix} \quad \overline{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ D \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \tag{8-354}$$

We can show that the pair $[\overline{\mathbf{A}}, \overline{\mathbf{B}}]$ is completely controllable. Thus, the eigenvalues of $(|s\mathbf{I} - \overline{\mathbf{A}} + \overline{\mathbf{B}}\overline{\mathbf{K}}|)$ can be arbitrarily placed. Substituting $\overline{\mathbf{A}}$, $\overline{\mathbf{B}}$, and $\overline{\mathbf{K}}$ in the characteristic equation of the closed-loop system with state and integral feedback, we have

$$|s\mathbf{I}-\bar{\mathbf{A}}+\bar{\mathbf{B}}\bar{\mathbf{K}}|=\begin{vmatrix} s & -1 & 0 \\ k_1 & s+25+k_2 & k_3 \\ -2500 & 0 & s \end{vmatrix}$$

$$=s^3+(25+k_2)s^2+k_1s+2500k_3=0 \qquad (8\text{-}355)$$

which can also be found from Fig. 8-28 using the SFG gain formula.

The design objectives are as follows:

1. The steady-state output must follow a step function input with zero error.
2. The rise time and settling time must be less than 0.05 s.
3. The maximum overshoot of the unit-step response must be less than 5 percent.

Because all three roots of the characteristic equation can be placed arbitrarily, it is not realistic to require minimum rise and settling times, as in Example 8-18-1.

Again, to realize a fast rise time and settling time, the roots of the characteristic equation should be placed far to the left in the *s*-plane, and the natural frequency should be high. Keep in mind that *roots with large magnitudes will lead to high gains for the state-feedback matrix*.

The **ACSYS**/MATLAB software was used to carry out the design. After a few trial-and-error runs, the design specifications can be satisfied by placing the roots at

$$s = -200\ -50+j50 \text{ and } -50-j50$$

The desired characteristic equation is

$$s^3+300s^2+25{,}000s+1{,}000{,}000=0 \qquad (8\text{-}356)$$

Equating like coefficients of Eqs. (8-355) and (8-356), we get

$$k_1 = 25{,}000\ k_2 = 275 \text{ and } k_3 = 400$$

The attributes of the unit-step response are as follows:

Maximumovershoot=4%

$t_r = 0.03247\text{s}$

$t_s = 0.04667\text{s}$

Notice that the high feedback gain of k_1, which is due to the large values of the roots selected, may pose physical problems; if so, the design specifications may have to be revised. ▲

EXAMPLE 8-18-2 In this example we illustrate the application of state-feedback with integral control to a system with a disturbance input.

Consider a dc-motor control system that is described by the following state equations:

$$\frac{d\omega(t)}{dt} = \frac{-B}{J}\omega(t) + \frac{K_i}{J}i_a(t) - \frac{1}{J}T_L \tag{8-357}$$

$$\frac{di_a(t)}{dt} = \frac{-K_b}{L}\omega(t) - \frac{R}{L}i_a(t) + \frac{1}{L}e_a(t) \tag{8-358}$$

where

$i_a(t) = \text{armature current, A}$

$e_a(t) = \text{armature applied voltage, V}$

$\omega(t) = \text{motor velocity, rad/s}$

$B = \text{viscous-friction coefficient of motor and load} = 0$

$J = \text{moment of inertia of motor and load} = 0.02\ \text{N}\cdot\text{m/rad/s}^2$

$K_i = \text{motor torque constant} = 1\ \text{N}\cdot\text{m/A}$

$K_b = \text{motor back-emf constant} = 1\ \text{V/rad/s}$

$T_L = \text{constant load torque (magnitude not known), N}\cdot\text{m}$

$L = \text{armature inductance} = 0.005\ \text{H}$

$R = \text{armature resistance} = 1\ \Omega$

The output equation is

$$y(t) = \mathbf{C}\mathbf{x}(t) = \begin{bmatrix} 1 & 0 \end{bmatrix}\mathbf{x}(t) \tag{8-359}$$

The design problem is to find the control $u(t) = e_a(t)$ through state feedback and integral control such that

1. $$\lim_{t\to\infty} i_a(t) = 0 \quad \text{and} \quad \lim_{t\to\infty}\frac{d\omega(t)}{dt} = 0 \tag{8-360}$$
2. $$\lim_{t\to\infty} \omega(t) = \text{step input } r(t) = u_s(t) \tag{8-361}$$
3. The eigenvalues of the closed-loop system with state feedback and integral control are at $s = -300$, $-10 + j\,10$, and $-10 - j10$.

Let the state variables be defined as $x_1(t) = \omega\,(t)$ and $x_2(t) = i_a(t)$. The state equations in Eqs. (8-357) and (8-358) are written in vector-matrix form:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) + \mathbf{E}n(t) \tag{8-362}$$

where $n(t) = T_L u_s(t)$.

$$\mathbf{A} = \begin{bmatrix} -\dfrac{B}{J} & \dfrac{K_i}{J} \\ -\dfrac{K_b}{L} & -\dfrac{R}{L} \end{bmatrix} = \begin{bmatrix} 0 & 50 \\ -200 & -200 \end{bmatrix} \tag{8-363}$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ \dfrac{1}{L} \end{bmatrix} = \begin{bmatrix} 0 \\ 200 \end{bmatrix} \tag{8-364}$$

$$\mathbf{E} = \begin{bmatrix} -\dfrac{1}{J} \\ 0 \end{bmatrix} = \begin{bmatrix} -50 \\ 0 \end{bmatrix} \tag{8-365}$$

From Eq. (8-347),

$$\overline{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & 0 \\ -\mathbf{C} & 0 \end{bmatrix} = \begin{bmatrix} 0 & 50 & 0 \\ -200 & -200 & 0 \\ -1 & 0 & 0 \end{bmatrix} \quad \overline{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 200 \\ 0 \end{bmatrix} \tag{8-366}$$

$$\overline{\mathbf{C}} = \begin{bmatrix} \mathbf{C} & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \quad \overline{\mathbf{E}} = \begin{bmatrix} \mathbf{E} \\ 0 \end{bmatrix} = \begin{bmatrix} -50 \\ 0 \\ 0 \end{bmatrix} \tag{8-367}$$

The control is given by

$$u(t) = -\mathbf{K}\mathbf{x}(t) - k_{n+1}x_{n+1}(t) = \overline{\mathbf{K}}\,\overline{\mathbf{x}}(t) \tag{8-368}$$

where

$$\overline{\mathbf{K}} = \begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix} \tag{8-369}$$

Figure 8-29 shows the state diagram of the overall designed system.

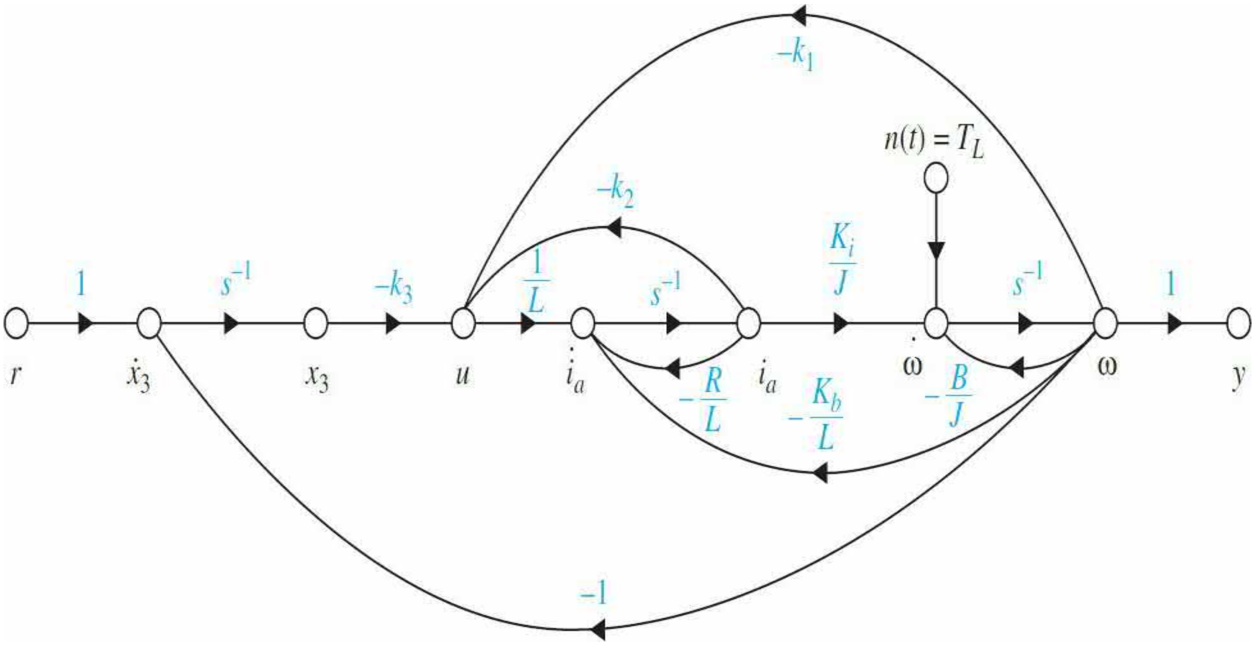


Figure 8-29 Dc-motor control system with state feedback and integral control and disturbance torque in Example 8-18-2.

The coefficient matrix of the closed-loop system is

$$\bar{\mathbf{A}} - \bar{\mathbf{B}}\bar{\mathbf{K}} = \begin{bmatrix} 0 & 50 & 0 \\ -200-200k_1 & -200-200k_2 & -200k_3 \\ -1 & 0 & 0 \end{bmatrix} \tag{8-370}$$

The characteristic equation is

$$\left|s\mathbf{I} - \bar{\mathbf{A}} + \bar{\mathbf{B}}\bar{\mathbf{K}}\right| = s^3 + 200(1+k_2)s^2 + 10{,}000(1+k_1)s - 10{,}000k_3 = 0 \tag{8-371}$$

which is more easily determined by applying the gain formula of SFG to Fig. 8-29.

For the three roots assigned, the last equation must equal

$$s^3 + 320s^2 + 6{,}200s + 60{,}000 = 0 \tag{8-372}$$

Equating the like coefficients of Eqs. (8-371) and (8-372), we get

$$k_1 = -0.38 \quad k_2 = 0.6 \quad k = -6.0$$

Applying the SFG gain formula to Fig. 8-29 between the inputs $r(t)$ and $n(t)$ and the states $\omega(t)$ and $i_a(t)$, we have

$$\begin{bmatrix} \Omega(s) \\ I_a(s) \end{bmatrix} = \frac{1}{\Delta_c(s)} \begin{bmatrix} -\frac{1}{J}\left(s^2 + \frac{R}{L}s + \frac{k_2}{L}s\right) & -\frac{k_3K_i}{JL} \\ -\frac{1}{J}\left(-\frac{K_b}{L}s - \frac{k_1}{L}s + \frac{k_3}{L}\right) & -\frac{k_3}{L}\left(s + \frac{B}{J}\right) \end{bmatrix} \begin{bmatrix} \frac{T_L}{s} \\ \frac{1}{s} \end{bmatrix} \tag{8-373}$$

where $\Delta_c(s)$ is the characteristic polynomial given in Eq. (8-372).

Applying the final-value theorem to the last equation, the steady-state values of the state variables are found to be

$$\lim_{t\to\infty} \begin{bmatrix} \omega(t) \\ i_a(t) \end{bmatrix} = \lim_{s\to 0} s \begin{bmatrix} \Omega(t) \\ I_a(t) \end{bmatrix} = \begin{bmatrix} 0 & K_i \\ 1 & B \end{bmatrix} \begin{bmatrix} T_L \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ T_L \end{bmatrix} \tag{8-374}$$

Thus, the motor velocity ω (t) will approach the constant reference input step function $r(t) = u_s(t)$ as t approaches infinity, independent of the disturbance torque T_L. Substituting the system parameters into Eq. (8-373), we get

$$\begin{bmatrix} \Omega(s) \\ I_a(s) \end{bmatrix} = \frac{1}{\Delta_c(s)} \begin{bmatrix} -50(s+320)s & 60,000 \\ 6200s+60,000 & 1,200s \end{bmatrix} \begin{bmatrix} \dfrac{T_L}{s} \\ \dfrac{1}{s} \end{bmatrix} \qquad (8\text{-}375)$$

Figure 8-30 shows the time responses of ω (t) and $i_a(t)$ when $T_L = 1$ and $T_L = 0$. The reference input is a unit-step function.

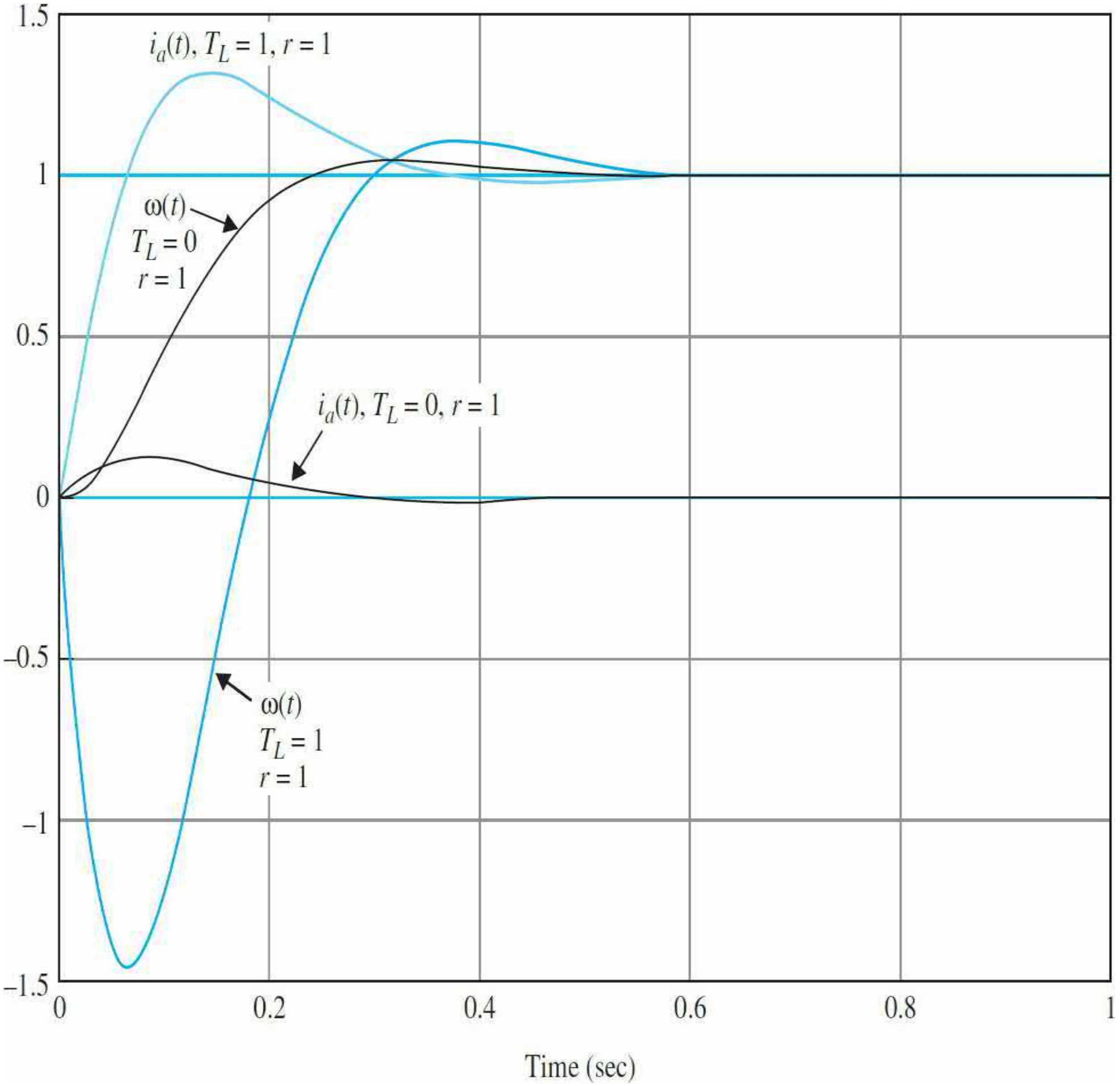


Figure 8-30 Time responses of dc-motor control system with state feedback and integral control and disturbance torque in Example 8-18-2.

8-20 MATLAB TOOLS AND CASE STUDIES

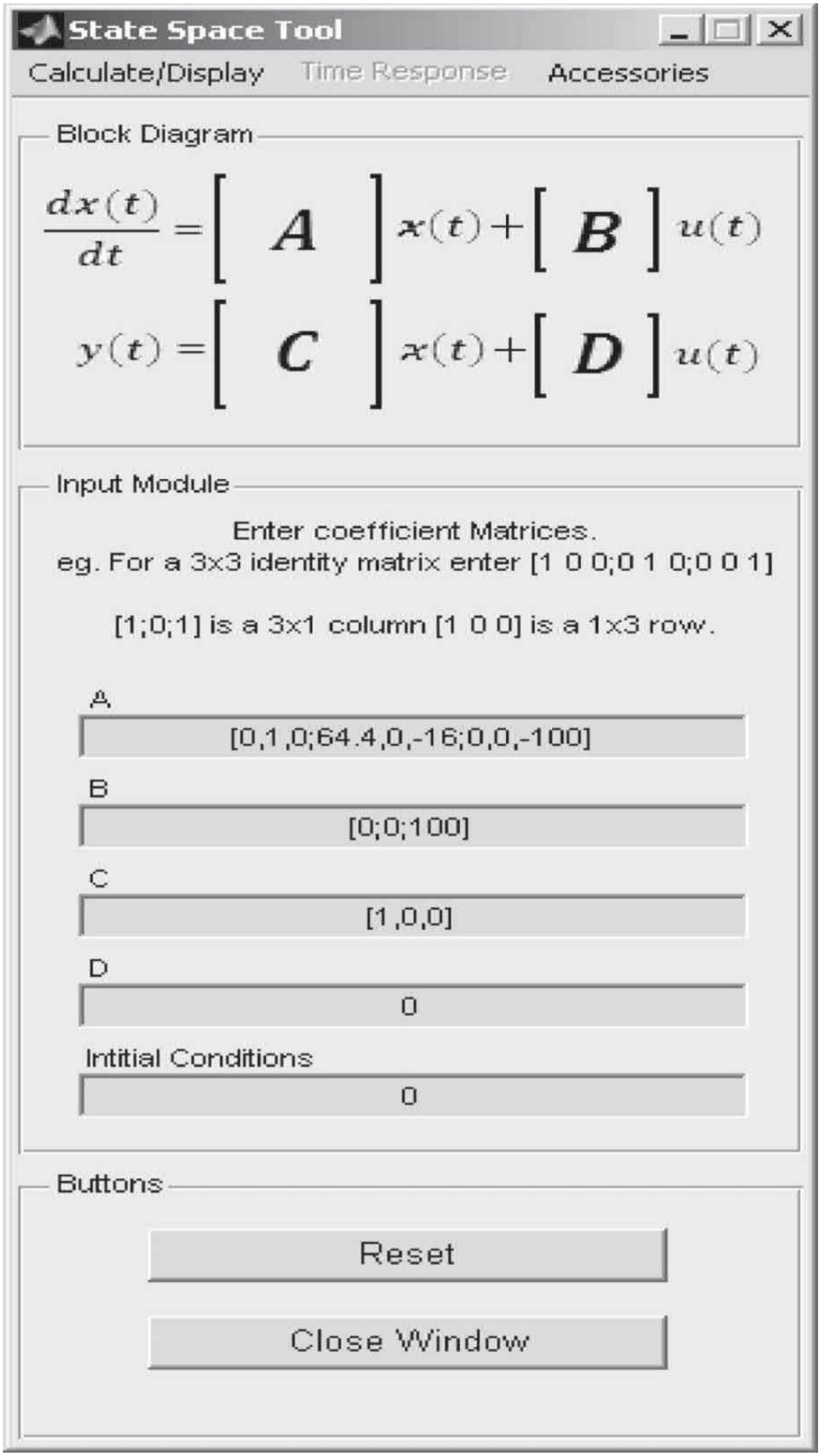
In this section, we present a MATLAB tool to solve most problems addressed in this chapter. The reader is encouraged to apply this tool to all the problems identified by a MATLAB toolbox in the left margin of the text throughout this chapter. As in Chap. 2, we use MATLAB's symbolic tool to solve some of the initial problems in this chapter involving inverse Laplace transformations. Finally, using the **tfcal** tool, discussed in Chap. 3, we can convert from transfer functions to state-space representation. These programs allow the user to conduct the following tasks:

- Enter the state matrices.
- Find the system's characteristic polynomial, eigenvalues, and eigenvectors.
- Find the similarity transformation matrices.
- Examine the system controllability and observability properties.
- Obtain the step, impulse, and natural (response to initial conditions) responses, as well as the time response to any function of time.
- Use the MATLAB symbolic tool to find the state-transition matrix using the inverse Laplace command.
- Convert a transfer function to state-space form or vice versa.

To better illustrate how to use the software, let us go through some of the steps involved in solving earlier examples in this chapter.

8-20-1 Description and Use of the State-Space Analysis Tool

The State-Space Analysis Tool (state tool) consists of a number of m-files and GUIs for the analysis of state-space systems. The state tool can be invoked from the **A**utomatic **C**ontrol **Sys**tems launch applet by clicking on the appropriate button. You will then see the window pictured in Fig. 8-31. We first consider the example in Sec. 8-16 to describe the functionality of the state tool.



$$\frac{dx(t)}{dt} = \left[\quad A \quad\right] x(t) + \left[\ B\ \right] u(t)$$

$$y(t) = \left[\quad C \quad\right] x(t) + \left[\ D\ \right] u(t)$$

Enter coefficient Matrices.
eg. For a 3x3 identity matrix enter [1 0 0;0 1 0;0 0 1]

[1;0;1] is a 3x1 column [1 0 0] is a 1x3 row.

A
[0,1,0;64.4,0,-16;0,0,-100]

B
[0;0;100]

C
[1,0,0]

D
0

Intitial Conditions
0

Figure 8-31 The State-Space Analysis window.

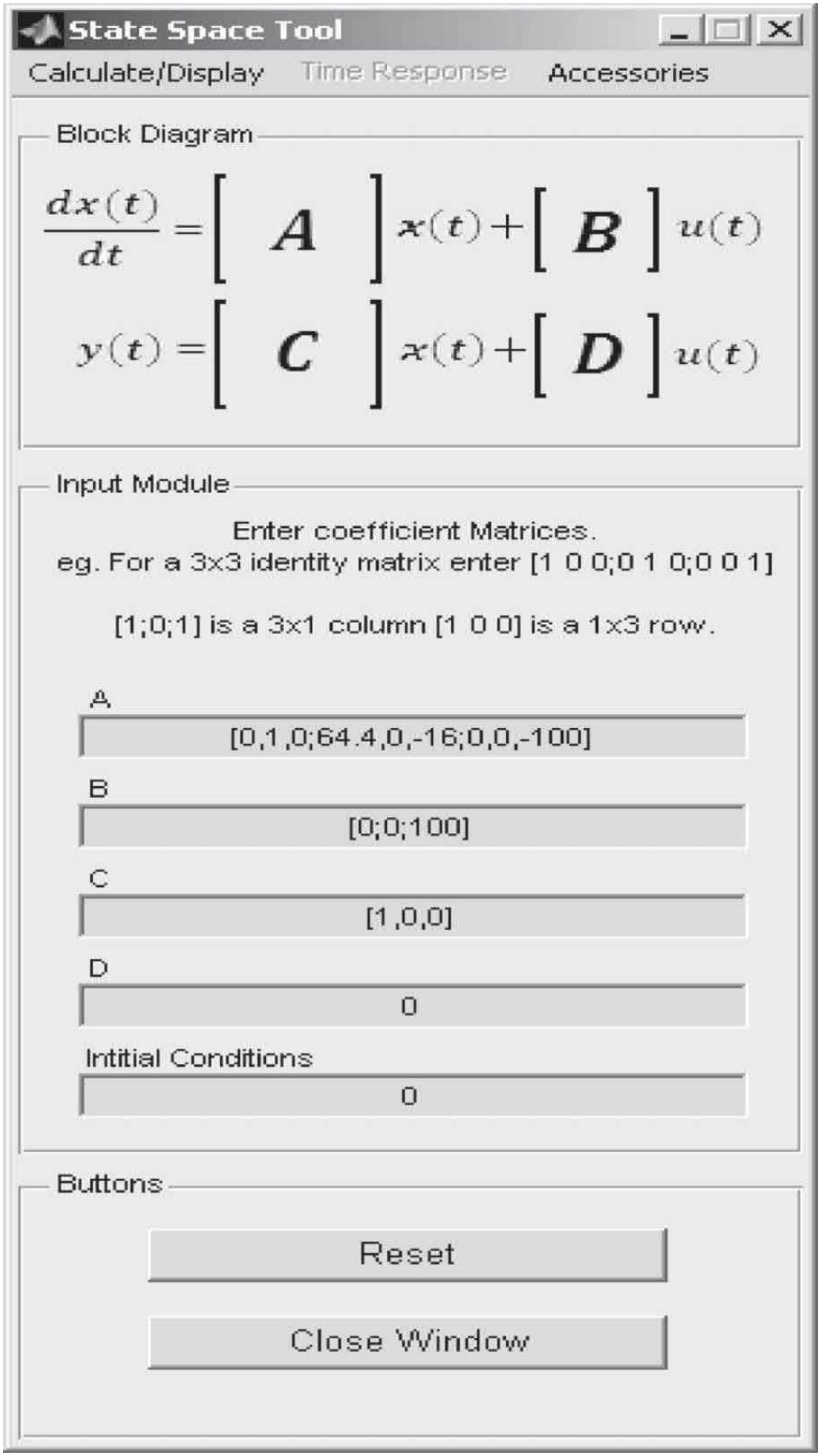
To enter the following coefficient matrices,

$$\mathbf{A}^* = \begin{bmatrix} 0 & 1 & 0 \\ 64.4 & 0 & -16 \\ 0 & 0 & -100 \end{bmatrix} \tag{8-376}$$

$$\mathbf{B}^* = \begin{bmatrix} 0 \\ 0 \\ 100 \end{bmatrix} \tag{8-377}$$

$$\mathbf{C}^* = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \tag{8-378}$$

enter the values in the appropriate edit boxes. Note that the default value of initial conditions is set to zero and you do not have to adjust it for this example. Follow the instructions on the screen very carefully. The elements in the row of a matrix may be separated by a space or a comma, while the rows themselves must be separated by a semicolon. For example, to enter matrix **A**, enter [0,1,0;64.4,0,-16;0,0,-100] in the **A** edit box, and to enter matrix **B**, enter [0;0;100] in the **B** edit box, as shown in Fig. 8-32. In this case, the **D** matrix is set to zero (default value). To find the characteristic Eq. (8-270), eigenvalues, and eigenvectors, choose the “Eigenvals & vects of A” option from the Calculate/Display menu. The detailed solution will be displayed on the MATLAB command window. The **A** matrix, eigenvalues of **A**, and eigenvectors of **A** are shown in Fig. 8-33. Note that the matrix representation of the eigenvalues corresponds to the diagonal canonical form (DCF) of **A**, while matrix **T**, representing the eigenvectors, is the DCF transformation matrix discussed in Sec. 8-11-4. To find the state-transition matrix $\phi(t)$, you must use the tfsym tool, which will be discussed in Sec. 8-20-2.



$$\frac{dx(t)}{dt} = \left[\quad A \quad\right] x(t) + \left[\ B\ \right] u(t)$$

$$y(t) = \left[\quad C \quad\right] x(t) + \left[\ D\ \right] u(t)$$

Enter coefficient Matrices.
eg. For a 3x3 identity matrix enter [1 0 0;0 1 0;0 0 1]

[1;0;1] is a 3x1 column [1 0 0] is a 1x3 row.

A: [0,1,0;64.4,0,-16;0,0,-100]

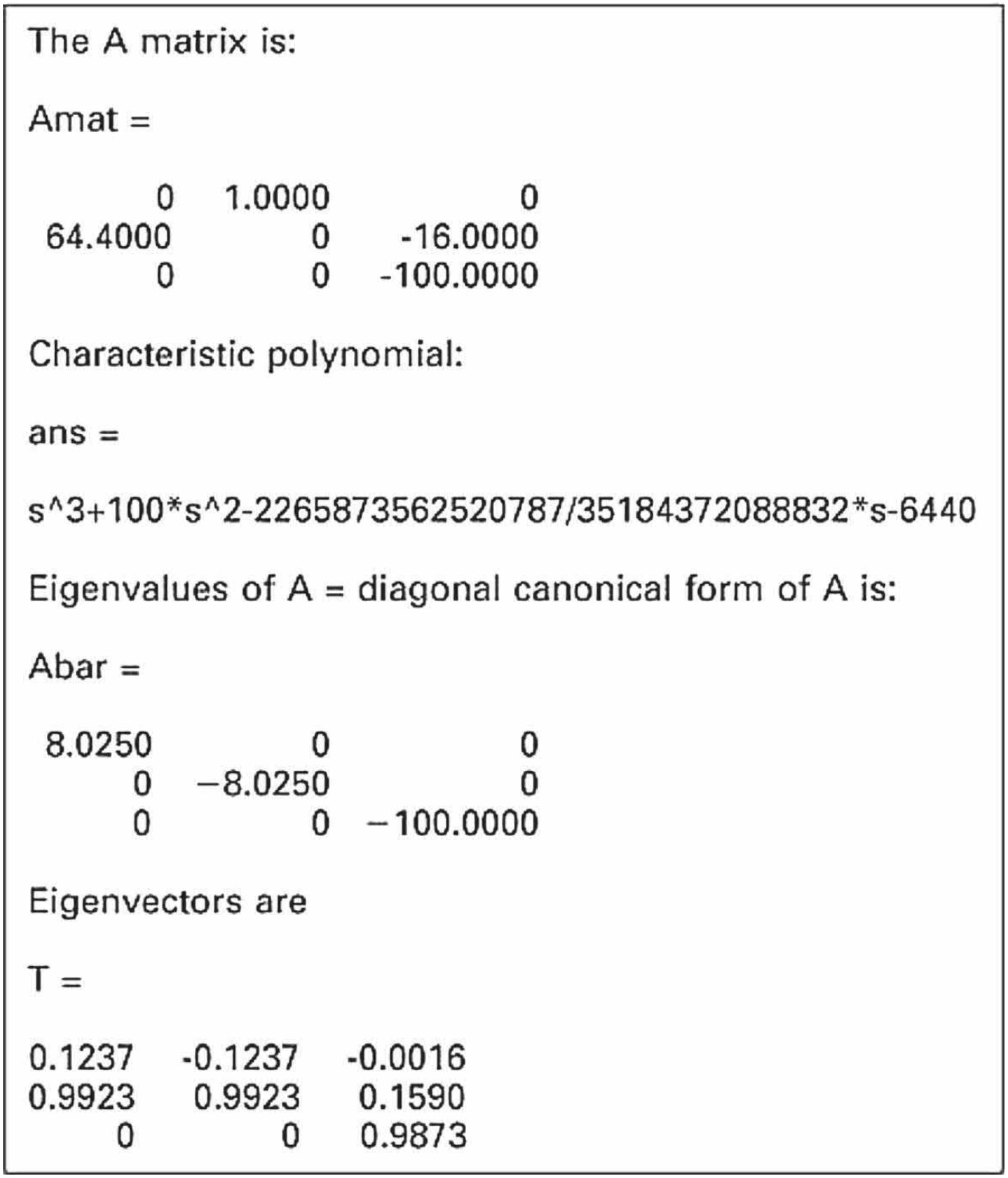
B: [0;0;100]

C: [1,0,0]

D: 0

Intitial Conditions: 0

Figure 8-32 Inputting values in the state-space window.



```
The A matrix is:

Amat =

         0     1.0000            0
   64.4000          0     -16.0000
         0          0    -100.0000

Characteristic polynomial:

ans =

s^3+100*s^2-2265873562520787/35184372088832*s-6440

Eigenvalues of A = diagonal canonical form of A is:

Abar =

  8.0250            0            0
       0      −8.0250            0
       0            0    −100.0000

Eigenvectors are

T =

0.1237    -0.1237    -0.0016
0.9923     0.9923     0.1590
     0          0     0.9873
```

Figure 8-33 The MATLAB command window display after clicking the “Eigenvals & vects of A” button.

The choice of the **C** in Eq. (8-376) makes the ball position the output $y(t)$

for input *v*(*t*). Then the input-output transfer function of the system can be obtained by choosing the “State-Space Calculations” option. The final output appearing in the MATLAB command window is the transfer function in both polynomial and factored forms, as shown in Fig. 8-34. As you can see, there is a small error due to numerical simulation. You may set the small terms to zero in the resulting transfer function to get Eq. (8-309).

```
State-space model is:

a =

        x1    x2      x3
x1       0     1       0
x2    64.4     0     -16
x3       0     0    -100


b =

        ul
x1       0
x2       0
x3     100


c =

      x1    x2    x3
y1     1     0     0


d =

      u1
y1     0

Continuous-time model.
Characteristic polynomial:

ans =

s^3+100*s^2-2265873562520787/35184372088832*s - 6440

Equivalent transfer-function model is:

Transfer function:
4.263e - 014 s^2 + 8.527e-014 s - 1600
--------------------------------------
    s^3 + 100s^2 - 64.4s - 6440

Pole, zero form:

Zero/pole/gain:
4.2633e - 014 (s+1.937e008)(s - 1.937e008)
------------------------------------------
        (s+100)(s+8.025)(s - 8.025)
```

Figure 8-34 The MATLAB command window after clicking the "State-Space Calculations" button.

Click the "Controllability" and "Observability" menu options to determine whether the system is controllable or observable. Note that these options are only enabled after pressing the "State-Space Calculations" button. After clicking the "Controllability" button, you get the MATLAB command window display, shown in Fig. 8-35. The **S** matrix in this case is the same as Eq. (8-310) with the rank of 3. As a result, the system is completely controllable. The program also provides the **M** and **P** matrices and the system controllability canonical form (CCF) representation as defined in Sec. 8-11-2.

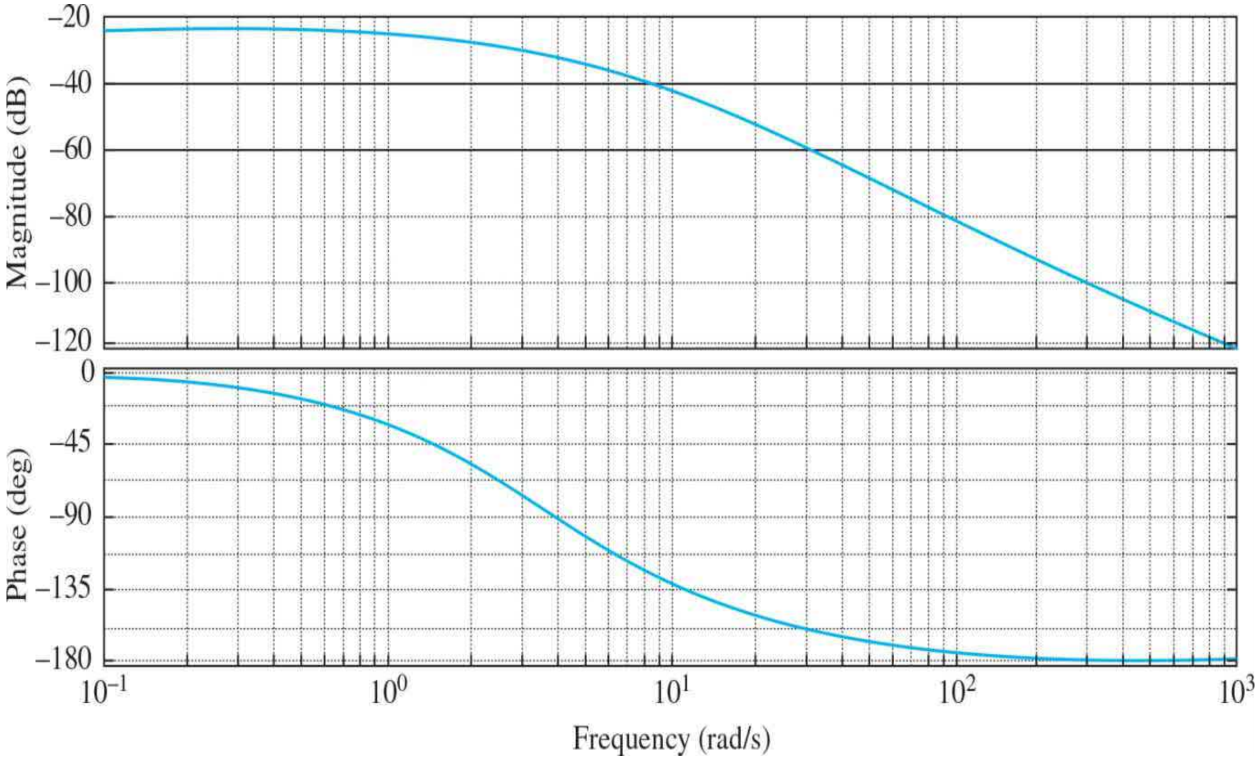


Figure 8-35 The MATLAB command window after clicking the "Controllability" button.

Once you choose the "Observability" option, the system observability is assessed in the MATLAB command window, as shown in Fig. 8-36. The system is completely observable, since the **V** matrix has a rank of 3. Note that the **V** matrix in Fig. 8-42 is the same as in Eq. (8-311). The program also provides the **M** and **Q** matrices and the system observability canonical form (OCF) representation as defined in Sec. 8-11-3. As an exercise, the user is urged to reproduce Eqs. (8-312) and (8-313) using this software.

```
The observability matrix (transpose:[C CA CA^2 ...]) is =

 Vmat =

   1.0000         0          0
        0    1.0000          0
  64.4000         0   -16.0000

The system is therefore observable, rank of V matrix is =

rankV =

  3

Mmat =

 -64.4000    100.0000    1.0000
100.0000      1.0000         0
   1.0000           0         0

The observability canonical form (OCF) transformation matrix is:

Qtran =

        0          0      1.0000
        0     1.0000   -100.0000
  -0.0625     6.2500   -625.0000

The transformed matrices using OCF are:

Abar =

1.0e+003 *

 0.0000    -0.0000    6.4400
 0.0010    -0.0000    0.0644
      0     0.0010   -0.1000

Bbar =

  -1600
    0
    0

Cbar =

 0   0   1

Dbar =

 0
```

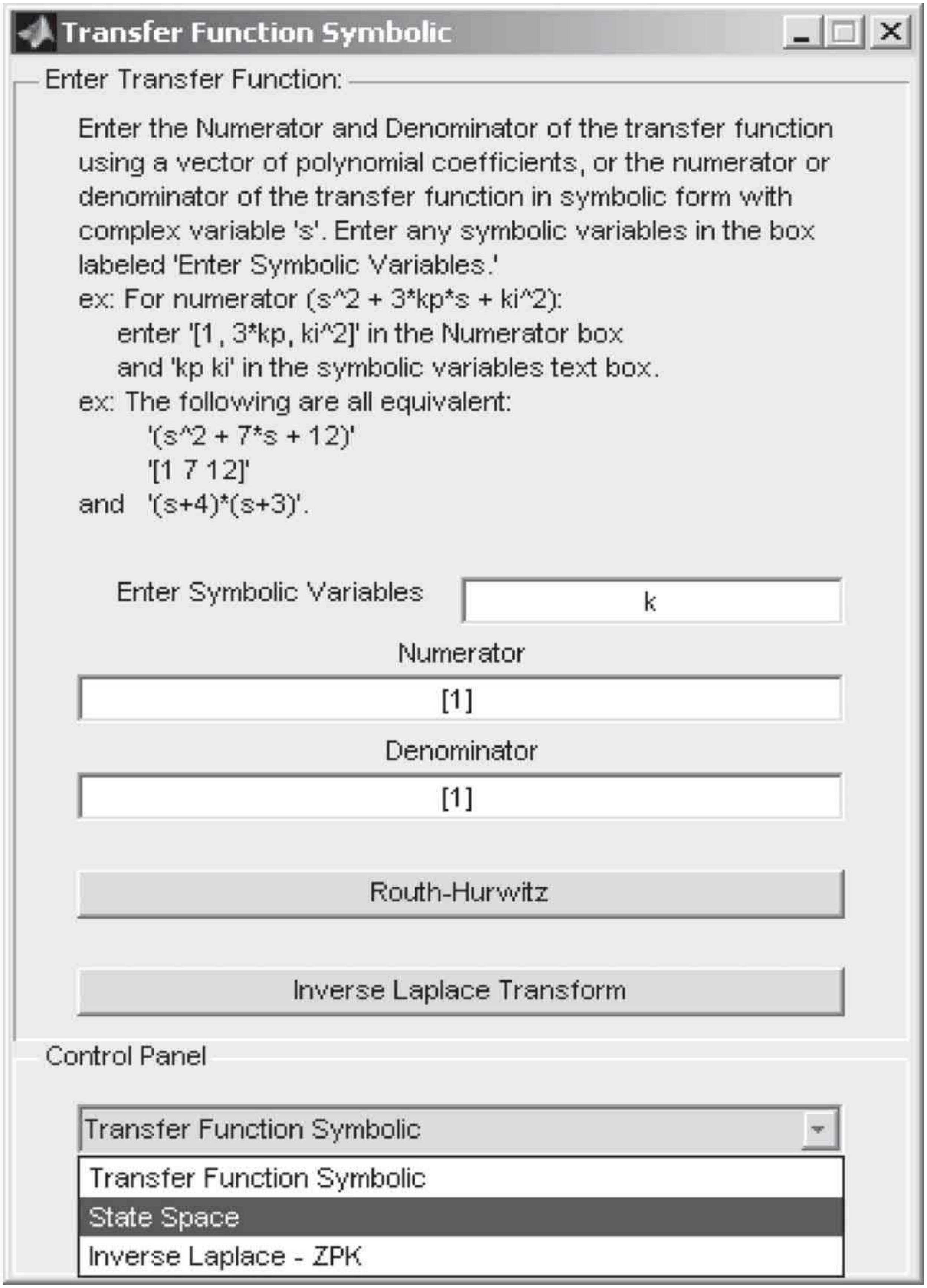
Figure 8-36 The MATLAB command window after clicking the “Observability” button.

You may obtain the output $y(t)$ natural time response (response to initial conditions only), the step response, the impulse response, or the time response to any other input function by choosing the appropriate option from the Time Response menu.

The state tool program may be used on all the examples identified by a MATLAB toolbox in the left margin of the text throughout this chapter, except problems involving inverse Laplace transformations and closed-form solutions. To address the analytical solutions, we need to use the tfsym tool, which requires the symbolic tool of MATLAB.

8-20-2 Description and Use of tfsym for State-Space Applications

You may run the Transfer Function Symbolic Tool by clicking the “Transfer Function Symbolic” button in the **ACSYS** window. You should get the window in Fig. 8-37. For this example, we will use the State-Space mode. Choose the appropriate option from the drop-down menu as shown in Fig. 8-37.



Enter Transfer Function:

Enter the Numerator and Denominator of the transfer function using a vector of polynomial coefficients, or the numerator or denominator of the transfer function in symbolic form with complex variable 's'. Enter any symbolic variables in the box labeled 'Enter Symbolic Variables.'

ex: For numerator (s^2 + 3*kp*s + ki^2):
enter '[1 , 3*kp, ki^2]' in the Numerator box
and 'kp ki' in the symbolic variables text box.
ex: The following are all equivalent:
'(s^2 + 7*s + 12)'
'[1 7 12]'
and '(s+4)*(s+3)'.

Enter Symbolic Variables: k

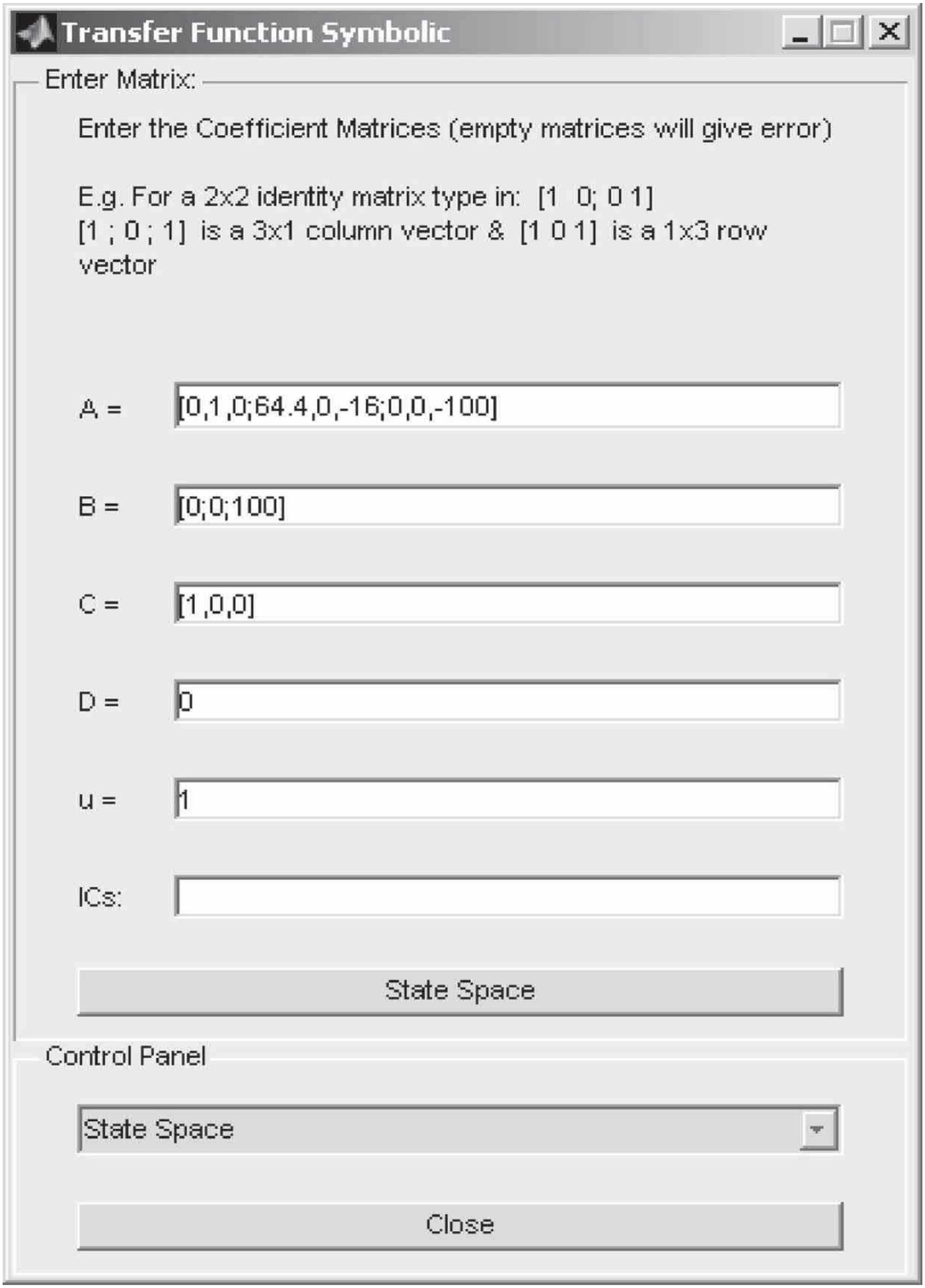
Numerator: [1]

Denominator: [1]

Control Panel: Transfer Function Symbolic

Figure 8-37 The Transfer Function Symbolic window.

Let us continue our solution to the example in previous section. Figure 8-38 shows the input of the matrices for this example into the state-space window. The input and output displays in the MATLAB command window are selectively shown in Fig. 8-39. Note that at first glance the $(s\mathbf{I}-\mathbf{A})^{-1}$ and $\phi(t)$ matrices may appear different from Eqs. (8-366) and (8-367). However, after minor manipulations, you may be able to verify that they are the same. This difference in representation is because of MATLAB symbolic approach. You may further simplify these matrices by using the “simple” command in the MATLAB command window. For example, to simplify $\phi(t)$, type “simple(phi)” in the MATLAB command window. If the desired format has not been achieved, you may have reached the software limit.



Transfer Function Symbolic

Enter Matrix:

Enter the Coefficient Matrices (empty matrices will give error)

E.g. For a 2x2 identity matrix type in: [1 0; 0 1]
[1 ; 0 ; 1] is a 3x1 column vector & [1 0 1] is a 1x3 row vector

A = [0,1,0;64.4,0,-16;0,0,-100]

B = [0;0;100]

C = [1,0,0]

D = 0

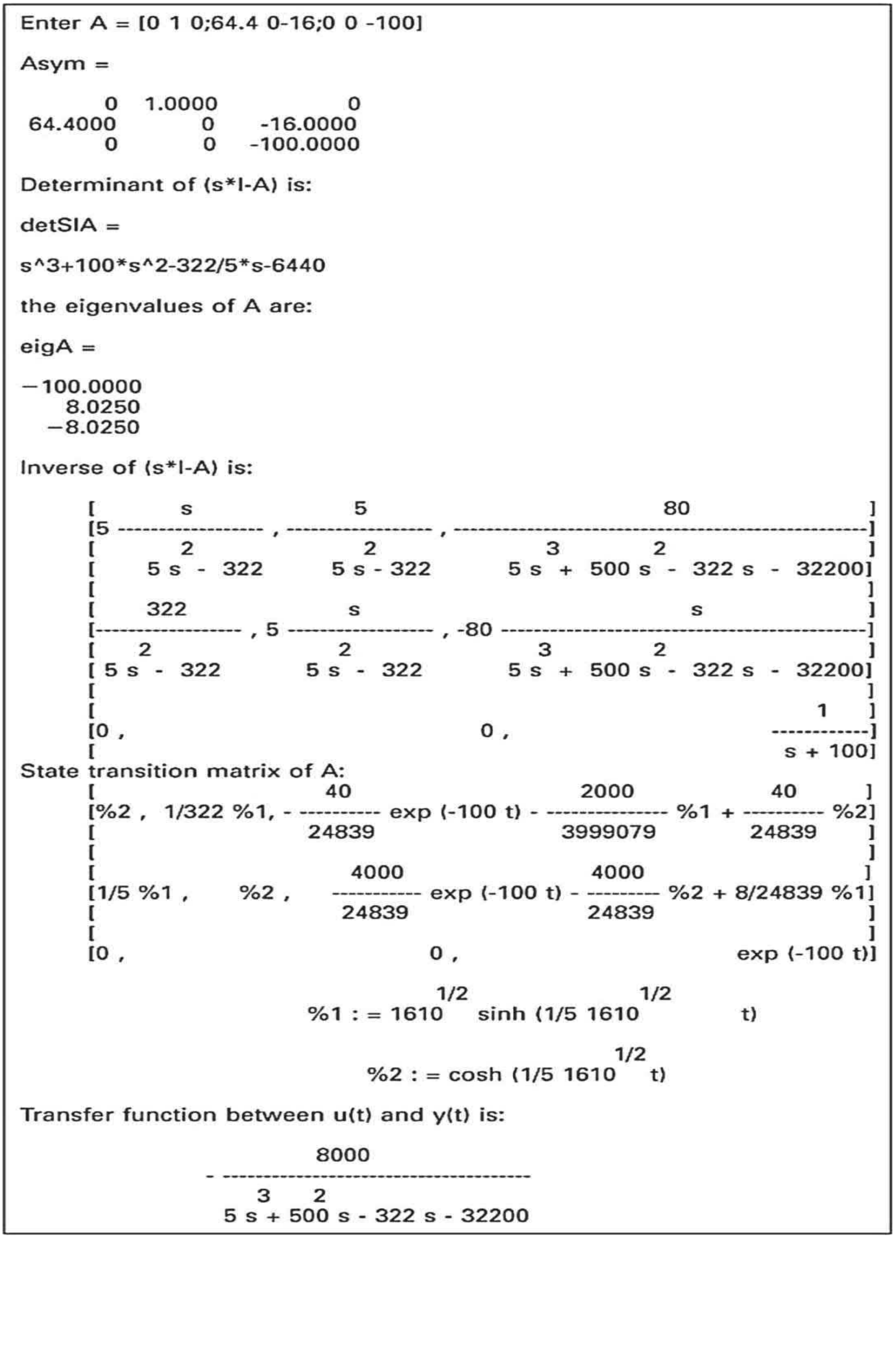
u = 1

ICs:

Control Panel

State Space

Figure 8-38 Inputting values into the Transfer Function Symbolic window.



```
Enter A = [0 1 0;64.4 0-16;0 0 -100]

Asym =

       0   1.0000          0
 64.4000        0    -16.0000
       0        0   -100.0000

Determinant of (s*I-A) is:

detSIA =

s^3+100*s^2-322/5*s-6440

the eigenvalues of A are:

eigA =

−100.0000
   8.0250
  −8.0250

Inverse of (s*I-A) is:

      [         s                  5                                  80                ]
      [5 ----------------- , ----------------- , ------------------------------------------------]
      [       2                  2                      3        2                      ]
      [    5 s  -  322        5 s - 322           5 s  +  500 s  -  322 s  -  32200]
      [                                                                             ]
      [    322                   s                                     s                ]
      [------------------ , 5 ----------------- , -80 ------------------------------------------]
      [    2                   2                        3        2                      ]
      [ 5 s  -  322         5 s  -  322           5 s  +  500 s  -  322 s  -  32200]
      [                                                                             ]
      [                                                                        1     ]
      [0 ,                                    0 ,                         ------------]
      [                                                                    s + 100]
State transition matrix of A:
      [                    40                          2000              40        ]
      [%2 ,  1/322 %1, - ---------- exp (-100 t) - --------------- %1 + ---------- %2]
      [                  24839                        3999079           24839      ]
      [                                                                            ]
      [                   4000                     4000                            ]
      [1/5 %1 ,      %2 ,     ----------- exp (-100 t) - --------- %2 + 8/24839 %1]
      [                       24839                    24839                       ]
      [                                                                            ]
      [0 ,                             0 ,                              exp (-100 t)]

                                      1/2                  1/2
                        %1 : = 1610       sinh (1/5 1610           t)

                                                       1/2
                              %2 : = cosh (1/5 1610       t)

Transfer function between u(t) and y(t) is:

                          8000
                  - ----------------------------------------
                       3      2
                    5 s + 500 s - 322 s - 32200
```

Figure 8-39 Selective display of the MATLAB command window for the tfsym tool.

8-21 CASE STUDY: POSITION CONTROL OF THE LEGO MINDSTORMS ROBOTIC ARM SYSTEM

Let us consider the robotic arm system that was shown in Fig. 2-41, and in App. D, where using a proportional controller, in Sec. D-1-6, we control the position of the robot arm. In this section we use the state-space approach to do the same.

Proportional Control

From Eq. (6-57), the system transfer function is

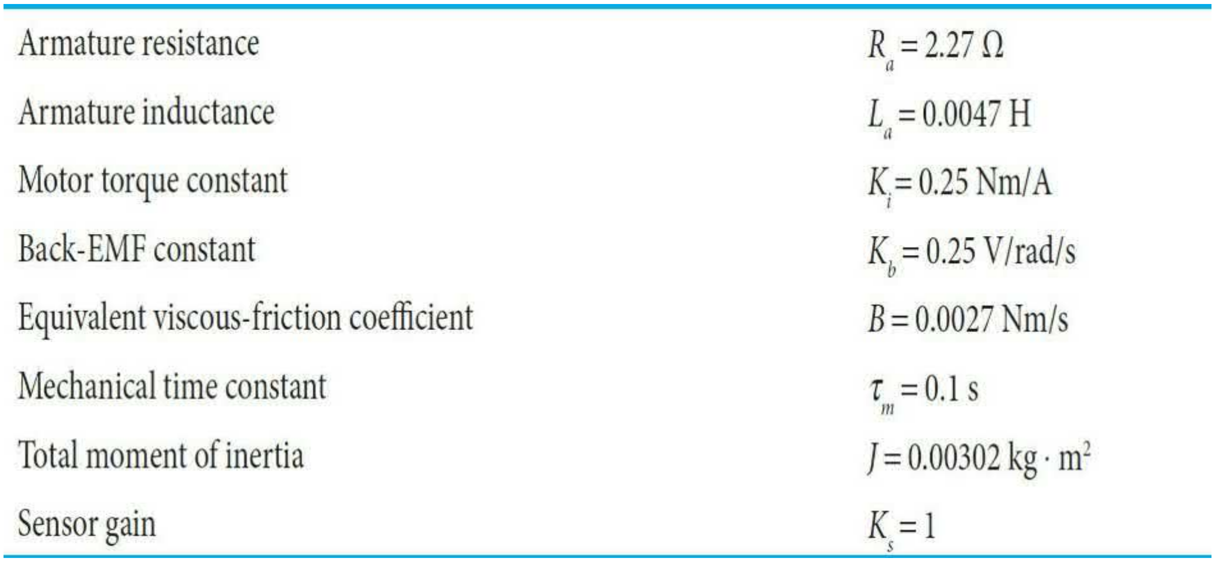
$$\frac{\Theta(s)}{\Theta_{in}(s)}=\frac{\dfrac{K_P K_i K_s}{R_a J}}{\left(\dfrac{L_a}{R_a}s+1\right)\left\{Js^2\left(B+\dfrac{K_b K_i}{R_a}\right)s+\dfrac{K_P K_i K_s}{R_a J}\right\}} \qquad (8\text{-}379)$$

As before, L_a/R_a may be neglected for small L_a. Hence, from Eq. (6-58), the simplified transfer functions of the systems is

$$\frac{\Theta(s)}{\Theta_{in}(s)}=\frac{\dfrac{K_P K_i K_s}{R_a J}}{s^2+\left(\dfrac{R_a B+K_b K_i}{R_a J}\right)s+\dfrac{K_P K_i K_s}{R_a J}} \qquad (8\text{-}380)$$

Table 8-1 restates the parameter values that were shown earlier in Table 8-7 for the LEGO NXT motor.

TABLE 8-1 Robotic Arm and Payload Experimental Parameters



Armature resistance	$R_a = 2.27\ \Omega$
Armature inductance	$L_a = 0.0047$ H
Motor torque constant	$K_i = 0.25$ Nm/A
Back-EMF constant	$K_b = 0.25$ V/rad/s
Equivalent viscous-friction coefficient	$B = 0.0027$ Nm/s
Mechanical time constant	$\tau_m = 0.1$ s
Total moment of inertia	$J = 0.00302\ \text{kg} \cdot \text{m}^2$
Sensor gain	$K_s = 1$

Premultiplying both sides of Eq. (8-380) by the denominator, we get

$$\left(s^2 + \frac{R_a B + K_b K_i}{R_a J} s + \frac{K_P K_i K_s}{R_a J}\right)\Theta(s) = \frac{K_P K_i K_s}{R_a J}\Theta_{in}(s) \tag{8-381}$$

Taking the inverse Laplace transform of Eq. (8-381), while recalling that the transfer function is independent of initial conditions of the system, we get the following differential equation with constant real coefficients:

$$\frac{d^2\theta(t)}{dt^2} + \frac{R_a B + K_b K_i}{R_a J}\frac{d\theta(t)}{dt} + \frac{K_P K_i K_s}{R_a J}\theta(t) = \frac{K_P K_i K_s}{R_a J}\theta_{in}(t) \tag{8-382}$$

Obviously, you can also directly arrive at Eq. (8-383) using the modeling techniques discussed in Chaps. 2 and 6. Defining

$$\begin{aligned} x_1(t) &= \theta(t) \\ x_2(t) &= \frac{d\theta(t)}{dt} \\ u(t) &= \theta_{in}(t) \\ y(t) &= x_1(t) \end{aligned} \tag{8-383}$$

where $x_1(t)$, $x_2(t)$ are the state variables, and $u(t)$ is the input, and $y(t)$ is the output.

The state equations are

$$\begin{bmatrix} \dfrac{dx_1(t)}{dt} \\ \dfrac{dx_2(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\dfrac{K_P K_i K_s}{R_a J} & -\dfrac{R_a B + K_b K_i}{R_a J} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \dfrac{K_P K_i K_s}{R_a J} \end{bmatrix} u(t) \tag{8-384}$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}(t) \tag{8-385}$$

There is a slight, but important, difference between the proportional control system in Eqs. (8-384) and (8-385) and the state feedback control system in Eqs. (8-317) and (8-318). Comparing the current system to that in Example 8-18-2, we can see because K_p in the proportional control appears in the forward path, as opposed to in the feedback loops, it also appears in the **B** matrix as the coefficient of the input *u*(*t*); also see Eq. (8-337) and Fig. 8-26*b* for comparison. As a result, the coefficient matrices become

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -\dfrac{K_P K_i K_s}{R_a J} & -\dfrac{R_a B + K_b K_i}{R_a J} \end{bmatrix}; \qquad \mathbf{B} = \begin{bmatrix} 0 \\ \dfrac{K_P K_i K_s}{R_a J} \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}; \qquad D = 0 \tag{8-386}$$

Using $K_p = 3$ (also see Sec. 8-1-6), and substituting the parameter values from Table 8-1 into the coefficient matrices, we get

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -109.4028 & -10.0109 \end{bmatrix}; \qquad \mathbf{B} = \begin{bmatrix} 0 \\ 109.4028 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}; \qquad D = 0 \tag{8-387}$$

Using the ACSYS state tool we can confirm the following results:

The characteristic equation is

$$|s\mathbf{I}-\mathbf{A}| = \begin{bmatrix} s & 1 \\ 109.4028 & s+10.0109 \end{bmatrix} = s^2 + 10.0109s + 109.4028 = 0 \tag{8-388}$$

The eigenvalues of A, or the roots of the characteristic equation, are

$$s_{1,2} = -5.0054 \pm j9.1841 = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \tag{8-389}$$

Controllability: The controllability matrix is

$$\mathbf{S} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} \end{bmatrix} = \begin{bmatrix} 0 & 109.4028 \\ 109.4028 & -1095.22 \end{bmatrix} \tag{8-390}$$

Because the rank of **S** is 2, the system is completely controllable.

The CCF representation of the system is

$$\overline{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ -109.4028 & -10.0109 \end{bmatrix}; \qquad \overline{\mathbf{B}} = \begin{bmatrix} 0 \\ 109.4028 \end{bmatrix}$$
$$\overline{\mathbf{C}} = \begin{bmatrix} 1 & 0 \end{bmatrix}; \qquad \overline{D} = 0 \tag{8-391}$$

Observability: The observability matrix is

$$\mathbf{V} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{8-392}$$

which has a rank of 2. Thus, the system is completely observable.

The OCF representation of the system is

$$\overline{\mathbf{A}} = \begin{bmatrix} 0 & -109.4028 \\ 1 & -10.0109 \end{bmatrix}; \qquad \overline{\mathbf{B}} = \begin{bmatrix} 109.4028 \\ 0 \end{bmatrix}$$
$$\overline{\mathbf{C}} = \begin{bmatrix} 0 & 1 \end{bmatrix}; \qquad \overline{D} = 0 \tag{8-393}$$

Using the Time Response menu in the State-Space Tool, we can obtain the step response of the system for a unit step input, which is shown in Fig. 8-40. The results are identical to Fig. 8-31, which was obtained for a step input of 160°. For the sake of comparison with the PD controller, we used Toolbox 8-21-2 instead.

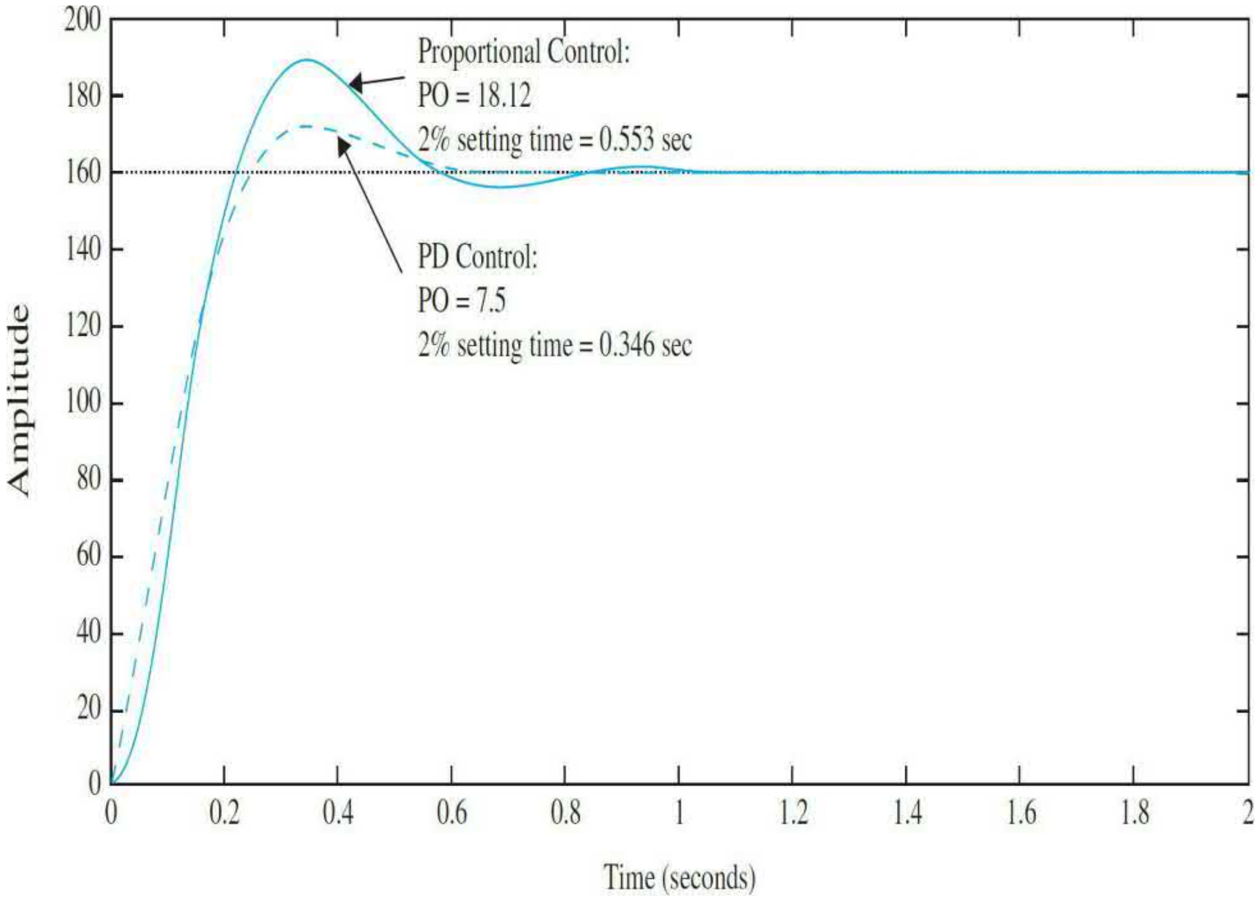


Figure 8-40 Comparison of unit step response of the NXT robot arm for proportional controller with K_P = 3 and PD controller with K_P = 3 and K_D = 0.1.

PD Control

The transfer functions of the systems is

$$\frac{\Theta(s)}{\Theta_{in}(s)} = \frac{\dfrac{K_i K_s}{R_a J}(K_P + K_D s)}{s^2 + \left(\dfrac{R_a B + K_b K_i}{R_a J} + \dfrac{K_D K_i K_s}{R_a J}\right)s + \dfrac{K_P K_i K_s}{R_a J}} \tag{8-394}$$

Substituting the parameter values in Table 8-1, we get

$$\frac{\Theta(s)}{\Theta_{in}(s)} = \frac{36.4676(K_P + K_D s)}{s^2 + (10.0109 + 36.4676 K_D)s + 36.4676 K_P} \tag{8-395}$$

For comparison purposes, with the proportional controller in Fig. 8-40, we retain the earlier value of $K_p = 3$ and vary K_D. For $K_D = 0.1$ the transfer function in Eq. (8-395) becomes

$$\frac{\Theta(s)}{\Theta_{in}(s)} = \frac{3.647s + 109.4028}{s^2 + 13.66s + 109.4028} \tag{8-396}$$

Using Toolbox 8-21-1 we can find the equivalent system state equations.

Toolbox 8-21-1

Conversion of the transfer function in Eq. (8-396) to the state-space form:

```
s=tf('s')
G=36.4676*(0.1*s+3)/(s^2+(10.0109+36.4676*0.1)*s+36.4676*3)
controller_ss=ss(G)
```

The coefficient matrices of PD control become

$$\mathbf{A} = \begin{bmatrix} -13.66 & -13.68 \\ 8 & 0 \end{bmatrix}; \qquad \mathbf{B} = \begin{bmatrix} 4 \\ 0 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 0.9117 & 3.419 \end{bmatrix}; \qquad D = 0 \tag{8-397}$$

Using the ACSYS state tool we can confirm the following results:
The characteristic equation is

$$|s\mathbf{I}-\mathbf{A}| = \begin{bmatrix} s+13.66 & 13.68 \\ -8 & s \end{bmatrix} = s^2 + 13.66s + \ 109.4028 = 0 \qquad (8\text{-}398)$$

The eigenvalues of A, or the roots of the characteristic equation, are

$$s_{1,2} = -6.83 \pm j7.9241 = \omega_n(-\zeta \pm j\sqrt{1-\zeta^2}) \qquad (8\text{-}399)$$

Controllability: The controllability matrix is

$$\mathbf{S} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} \end{bmatrix} = \begin{bmatrix} 4 & -54.64 \\ 0 & 32 \end{bmatrix} \qquad (8\text{-}400)$$

Because the rank of **S** is 2, the system is completely controllable.

The CCF representation of the system is

$$\overline{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ -109.4028 & -13.66 \end{bmatrix}; \qquad \overline{\mathbf{B}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\overline{\mathbf{C}} = \begin{bmatrix} 109.4028 & 3.6468 \end{bmatrix}; \qquad \overline{D} = 0 \qquad (8\text{-}401)$$

Observability: The observability matrix is

$$\mathbf{V} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \end{bmatrix} = \begin{bmatrix} 0.9117 & 3.4190 \\ 14.8982 & -12.4721 \end{bmatrix} \qquad (8\text{-}402)$$

If $K_p \neq 0$, the matrix has a rank of 2. Thus, the system is completely observable.

The OCF representation of the system is

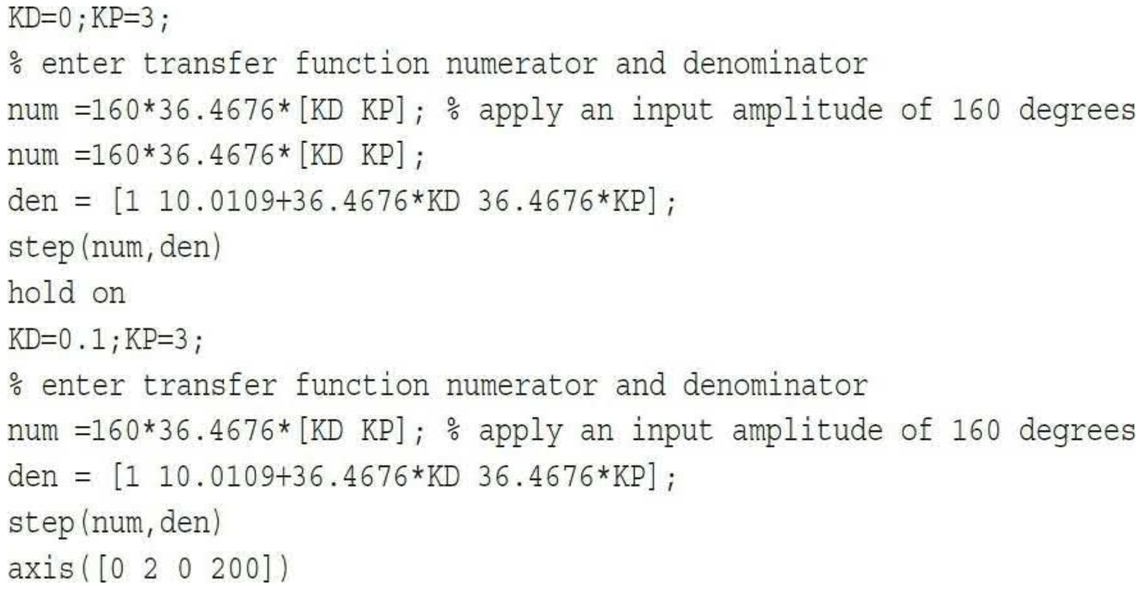
$$\overline{\mathbf{A}} = \begin{bmatrix} 0 & -109.4028 \\ 1 & -13.66 \end{bmatrix}; \qquad \overline{\mathbf{B}} = \begin{bmatrix} 109.4028 \\ 3.6468 \end{bmatrix}$$

$$\overline{\mathbf{C}} = \begin{bmatrix} 0 & 1 \end{bmatrix}; \qquad \overline{D} = 0 \tag{8-403}$$

Using the Time Response menu in the State-Space Tool, we can obtain the step response of the system for a unit step input, which is shown in Fig. 8-40. Comparing the proportional and PD controllers, we can see that the PD controller improves the percent overshoot and settling time by increasing the damping of the system.

Toolbox 8-21-2

Figure 8-40 is obtained by the following sequence of MATLAB functions:



```
KD=0;KP=3;
% enter transfer function numerator and denominator
num =160*36.4676*[KD KP]; % apply an input amplitude of 160 degrees
num =160*36.4676*[KD KP];
den = [1 10.0109+36.4676*KD 36.4676*KP];
step(num,den)
hold on
KD=0.1;KP=3;
% enter transfer function numerator and denominator
num =160*36.4676*[KD KP]; % apply an input amplitude of 160 degrees
den = [1 10.0109+36.4676*KD 36.4676*KP];
step(num,den)
axis([0 2 0 200])
```

8-22 SUMMARY

This chapter was devoted to the state-variable analysis of linear systems. The fundamentals on state variables and state equations were introduced in Chaps. 2 and 3, and formal discussions on these subjects were covered in this chapter. Specifically, the state-transition matrix and state-transition equations were introduced and the relationship between the state equations and transfer functions was established. Given the transfer function of a linear system, the state equations of the system can be obtained by decomposition of the transfer function. Given the state equations and the output equations, the transfer function can be determined either analytically or directly from the state diagram.

Characteristic equations and eigenvalues were defined in terms of the state equations and the transfer function. Eigenvectors of **A** were also defined for distinct and multiple-order eigenvalues. Similarity transformations to controllability canonical form (CCF), observability canonical form (OCF), diagonal canonical form (DCF), and Jordan canonical form (JCF) were discussed. State controllability and observability of linear time-invariant systems were defined and illustrated, and a final example, on the magnetic-ball-suspension system, summarized the important elements of the state-variable analysis of linear systems.

The MATLAB software tools state tool and tfsym were described in Sec. 8-20. The program functionality was discussed with two examples. Together these tools can solve most of the homework problems and examples in this chapter. Finally, the NXT robot example studied in Chaps. 2, 6, and 7 and App. D was extended to state-space form in Sec. 8-21, where the effects of a proportional and a PD controller were studied.

REFERENCES

State Variables and State Equations

1. B. C. Kuo, *Linear Networks and Systems*, McGraw-Hill, New York, 1967.
2. R. A. Gabel and R. A. Roberts, *Signals and Linear Systems*, 3rd Ed., John Wiley & Sons, New York, 1987.

Controllability and Observability

3. R. E. Kalman, “On the General Theory of Control Systems,” *Proc. IFAC*, Vol. 1, pp. 481–492, Butterworths, London, 1961.
4. W. L. Brogan, *Modern Control Theory*, 2nd Ed., Prentice Hall,

Englewood Cliffs, NJ, 1985.

PROBLEMS

8-1. The following differential equations represent linear time-invariant systems. Write the dynamic equations (state equations and output equations) in vector-matrix form.

(a) $$\frac{d^2y(t)}{dt^2}+4\frac{dy(t)}{dt}+y(t)=5r(t)$$

(b) $$2\frac{d^3y(t)}{dt^3}+3\frac{d^2y(t)}{dt^2}+5\frac{dy(t)}{dt}+2y(t)=r(t)$$

(c) $$\frac{d^3y(t)}{dt^3}+5\frac{d^2y(t)}{dt^2}+3\frac{dy(t)}{dt}+y(t)+\int_0^t y(\tau)d\tau=r(\tau)$$

(d) $$\frac{d^4y(t)}{dt^4}+1.5\frac{d^3y(t)}{dt^3}+2.5\frac{dy(t)}{dt}+y(t)=2r(t)$$

8-2. The following transfer functions show linear time-invariant systems. Write the dynamic equations (state equations and output equations) in vector-matrix form.

(a) $$G(s)=\frac{s+3}{s^2+3s+2}$$

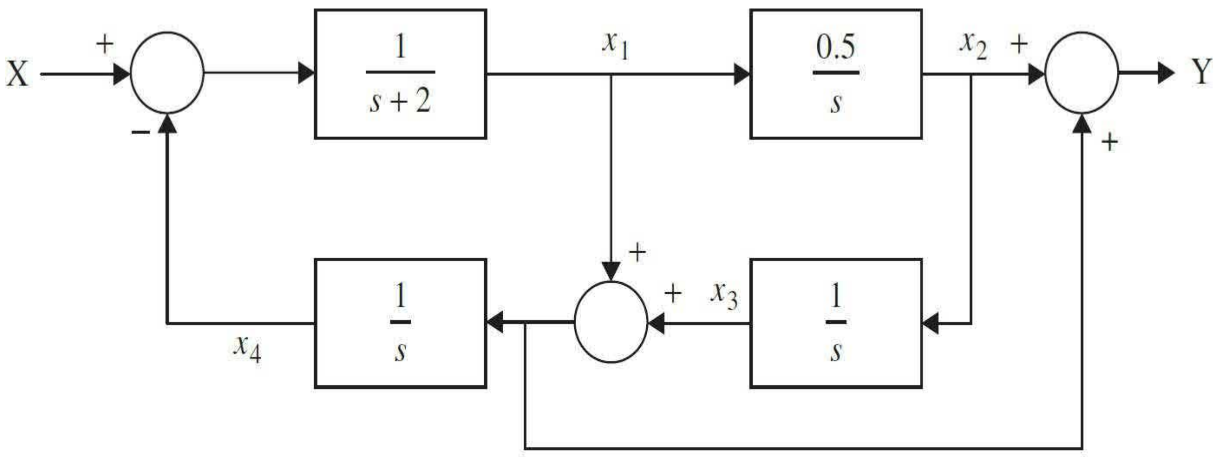
(b) $$G(s)=\frac{6}{s^3+6s^2+11s+6}$$

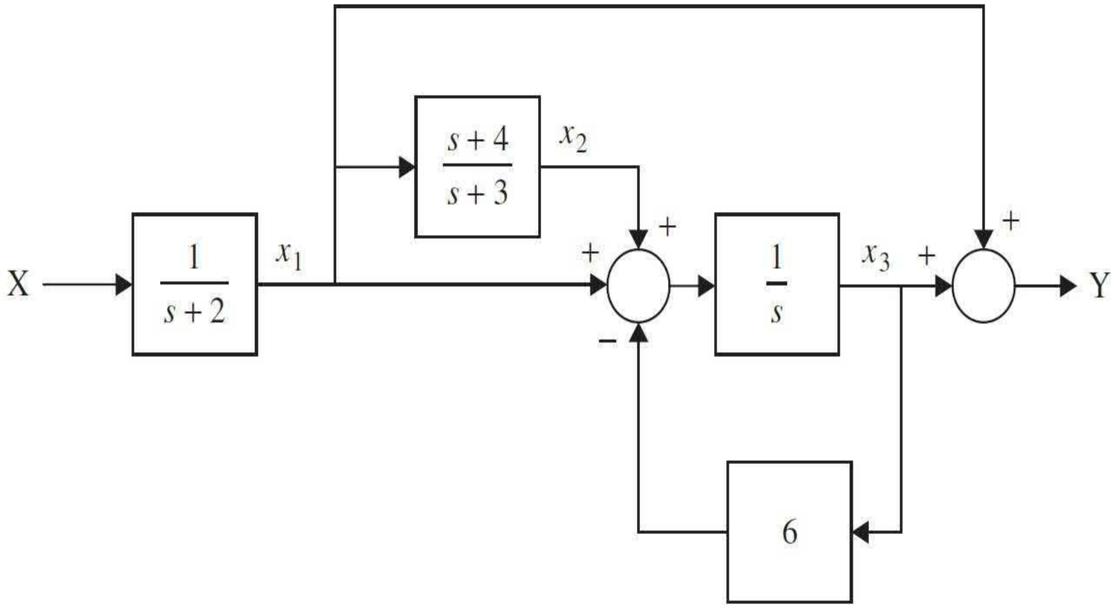
(c) $$G(s)=\frac{s+2}{s^2+7s+12}$$

(d) $$G(s)=\frac{s^3+11s^2+35s+250}{s^2(s^3+4s^2+39s+108)}$$

8-3. Repeat Prob. 8-2 by using MATLAB.

8-4. Write the state equations for the block diagrams of the systems shown in Fig. 8P-4.


X
+
−
1
s + 2
x1
0.5
s
x2
+
+
Y
+
+
+
x3
1
s
1
s
x4


X
1
s + 2
x1
s + 4
s + 3
x2
+
+
−
1
s
x3
+
+
Y
6

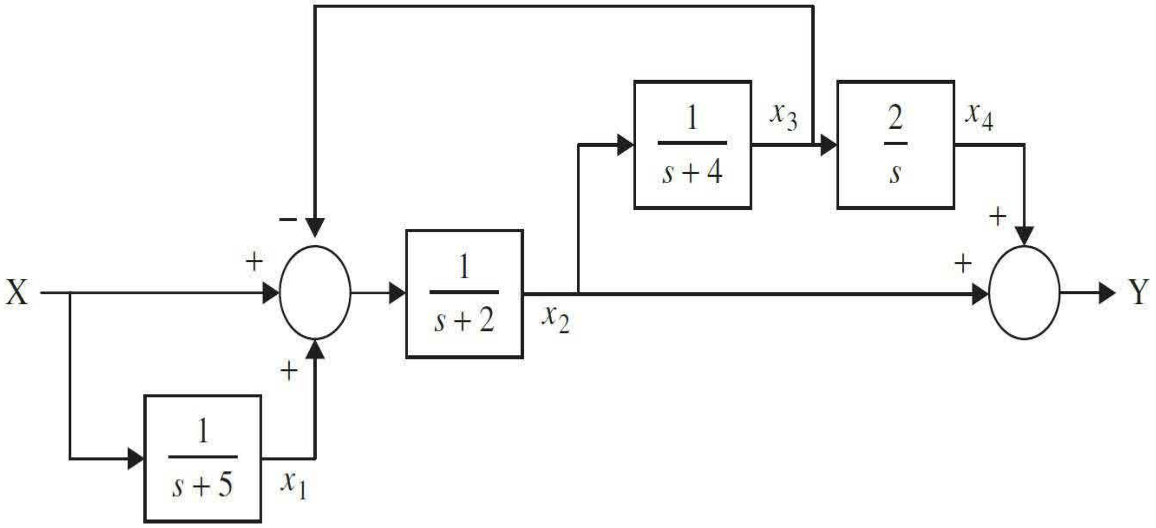

X
+
−
+
1
s + 5
x1
1
s + 2
x2
1
s + 4
x3
2
s
x4
+
+
Y

Figure 8P-4

8-5. By use of Eq. (8-58), show that

$$\phi(t) = \mathbf{I} + \mathbf{A}t + \frac{1}{2!}\mathbf{A}^2t^2 + \frac{1}{3!}\mathbf{A}^3t^3 + \cdots$$

8-6. The state equations of a linear time-invariant system are represented by

$$\frac{dx(t)}{dt} = \mathrm{Ax}(t) + \mathrm{Bu}(t)$$

Find the state-transition matrix $\phi(t)$, the characteristic equation, and the eigenvalues of **A** for the following cases:

(a) $\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -2 & -1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

(b) $\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -4 & -5 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

(c) $\mathbf{A} = \begin{bmatrix} -3 & 0 \\ 0 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

(d) $\mathbf{A} = \begin{bmatrix} 3 & 0 \\ 0 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

(e) $\mathbf{A} = \begin{bmatrix} 0 & 2 \\ -2 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

(f) $\mathbf{A} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -2 & 1 \\ 0 & 0 & -2 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$

(g) $\mathbf{A} = \begin{bmatrix} -5 & 1 & 0 \\ 0 & -5 & 1 \\ 0 & 0 & -5 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

8-7. Find $\phi(t)$ and the characteristic equation of the state variables in Prob. 8-6, using a computer program.

8-8. Find the state-transition equation of each of the systems described in Prob. 8-6 for $t \geq 0$. Assume that $\mathbf{x}(0)$ is the initial state vector, and the components of the input vector $\mathbf{u}(t)$ are all unit-step functions.

8-9. Find out if the matrices given in the following can be state-transition matrices. [Hint: check the properties of $\phi(t)$.]

(a) $\begin{bmatrix} -e^{-t} & 0 \\ 0 & 1-e^{-t} \end{bmatrix}$

(b) $\begin{bmatrix} 1-e^{-t} & 0 \\ 1 & e^{-t} \end{bmatrix}$

(c) $\begin{bmatrix} 1 & 0 \\ 1-e^{-t} & e^{-t} \end{bmatrix}$

(d) $\begin{bmatrix} e^{-2t} & te^{-2t} & t^2e^{-2t}/2 \\ 0 & e^{-2t} & te^{-2t} \\ 0 & 0 & e^{-2t} \end{bmatrix}$

8-10. Find the time response of the following systems:

(a) $\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$

(b) $\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & -0.5 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0.5 \\ 0 \end{bmatrix} uy = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

8-11. Given a system described by the dynamic equations:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{B}u(t) \quad y(t) = \mathbf{Cx}(t)$$

(a) $\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -2 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$

(b) $\mathbf{A} = \begin{bmatrix} -1 & 1 \\ 0 & -1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 1 \end{bmatrix}$

(c) $\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & -2 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$

(1) Find the eigenvalues of **A**. Use the **ACSYS** computer program to check the answers. You may get the characteristic equation and solve for the roots using tfsym or tcal components of **ACSYS**.

(2) Find the transfer-function relation between **X**(s) and $U(s)$.

(3) Find the transfer function $Y(s)/U(s)$.

8-12. Given the dynamic equations of a time-invariant system:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{B}u(t) \quad y(t) = \mathbf{Cx}(t)$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -2 & -3 \end{bmatrix} \quad \mathrm{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \mathrm{C} = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$$

Find the matrices $\mathbf{A}_1$ and $\mathbf{B}_1$ so that the state equations are written as

$$\frac{d\overline{\mathrm{x}}(t)}{dt} = \mathrm{A}_1\overline{\mathbf{x}}(t) + \mathbf{B}_1 u(t)$$

where

$$\overline{\mathbf{x}}(t) = \begin{bmatrix} x_1(t) \\ y(t) \\ \dfrac{dy(t)}{dt} \end{bmatrix}$$

8-13. Given the dynamic equations

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad y(t) = \mathbf{C}\mathbf{x}(t)$$

(a) $\mathbf{A} = \begin{bmatrix} 0 & 2 & 0 \\ 1 & 2 & 0 \\ -1 & 0 & 1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$

(b) $\mathbf{A} = \begin{bmatrix} 0 & 2 & 0 \\ 1 & 2 & 0 \\ -1 & 1 & 1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$

(c) $\mathbf{A} = \begin{bmatrix} -2 & 1 & 0 \\ 0 & -2 & 0 \\ -1 & -2 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$

(d) $\mathbf{A} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & -1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$

(e) $\mathbf{A} = \begin{bmatrix} 1 & 1 \\ -2 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}$

Find the transformation $\mathbf{x}(t) = \mathbf{P}\overline{\mathbf{x}}(t)$ that transforms the state equations into the controllability canonical form (CCF).

8-14. For the systems described in Prob. 8-13, find the transformation $\mathbf{x}(t) = \mathbf{Q}\overline{\mathbf{x}}(t)$ so that the state equations are transformed into the observability canonical form (OCF).

8-15. For the systems described in Prob. 8-13, find the transformation $\mathbf{x}(t) = \mathbf{T}\,\overline{\mathbf{x}}(t)$ so that the state equations are transformed into the diagonal canonical form (DCF) if **A** has distinct eigenvalues and Jordan canonical form (JCF) if **A** has at least one multiple-order eigenvalue.

8-16. Consider the following transfer functions. Transform the state equations into the controllability canonical form (CCF) and observability canonical form (OCF).

(a) $\dfrac{s^2-1}{s^2(s^2-2)}$

(b) $\dfrac{2s+1}{s^2+4s+4}$

8-17. The state equation of a linear system is described by

$$\frac{d\mathrm{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{Bu}(t)$$

The coefficient matrices are given as follows. Explain why the state equations cannot be transformed into the controllability canonical form (CCF).

(a) $\mathbf{A} = \begin{bmatrix} -2 & 0 \\ 0 & -1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

(b) $\mathbf{A} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$

(c)

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 2 \\ \sqrt{2} \end{bmatrix}$$

(d)

$$\mathbf{A} = \begin{bmatrix} -2 & 1 & 0 \\ 0 & -2 & 0 \\ -1 & -2 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

8-18. Check the controllability of the following systems:

(a)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 2 \\ 5 \end{bmatrix} u$$

(b)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 2 \\ 0 \end{bmatrix} u$$

(c)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 4 & 2 \\ 0 & 0 \\ 3 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

(d)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 4 \\ 3 \end{bmatrix} u$$

(e)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{bmatrix} = \begin{bmatrix} -2 & 1 & 0 & 0 & 0 \\ 0 & -2 & 1 & 0 & 0 \\ 0 & 0 & -2 & 0 & 0 \\ 0 & 0 & 0 & -5 & 1 \\ 0 & 0 & 0 & 0 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 3 & 0 \\ 0 & 0 \\ 2 & \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{bmatrix} = \begin{bmatrix} -2 & 1 & 0 & 0 & 0 \\ 0 & -2 & 1 & 0 & 0 \\ 0 & 0 & -2 & 0 & 0 \\ 0 & 0 & 0 & -5 & 1 \\ 0 & 0 & 0 & 0 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} + \begin{bmatrix} 4 \\ 2 \\ 1 \\ 3 \\ 0 \end{bmatrix} u$$

(f)

8-19. Check the observability of the following systems:

(a) $$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad y = \begin{bmatrix} 1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

(b) $$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad y = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

(c) $$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 2 & 1 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 3 \\ 0 & 2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

(d) $$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 2 & 1 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 3 & 0 & 0 \\ 4 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

(e)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & -3 & 1 \\ 0 & 0 & 0 & 0 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}$$

(f)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & -3 & 1 \\ 0 & 0 & 0 & 0 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \quad \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}$$

8-20. The equations that describe the dynamics of a motor control system are

$$\begin{aligned} e_a(t) &= R_a i_a(t) + L_a \frac{di_a(t)}{dt} + K_b \frac{d\theta_m(t)}{dt} \\ T_m(t) &= K_i i_a(t) \\ T_m(t) &= J \frac{d^2\theta_m(t)}{dt^2} + B \frac{d\theta_m(t)}{dt} + K\theta_m(t) \\ e_a(t) &= K_a e(t) \\ e(t) &= K_s[\theta_r(t) - \theta_m(t)] \end{aligned}$$

(a) Assign the state variables as $x_1(t) = \theta_m(t)$, $x_2(t) = d\theta_m(t)/dt$, and $x_3(t) = i_a(t)$. Write the state equations in the form of

$$\frac{d\mathrm{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{B}\theta_r(t)$$

Write the output equation in the form $y(t) = \mathbf{Cx}(t)$, where $y(t) = \theta_m(t)$.

(b) Find the transfer function $G(s) = \Theta_m(s)/E(s)$ when the feedback path from $\Theta_m(s)$ to $E(s)$ is broken. Find the closed-loop transfer function $M(s) = \Theta_m(s)/\Theta_r(s)$.

8-21. Given the matrix **A** of a linear system described by the state equation

$$\frac{d\mathrm{x}(t)}{dt} = \mathrm{Ax}(t) + \mathrm{B}u(t)$$

(a) $\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$

(b) $\mathbf{A} = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix}$

(c) $\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

Find the state-transition matrix $\phi(t)$ using the following methods:

(1) Infinite-series expansion of e^{At}, expressed in closed form

(2) The inverse Laplace transform of $(s\mathbf{I} - \mathbf{A})^{-1}$

8-22. The schematic diagram of a feedback control system using a dc motor is shown in Fig. 8P-22. The torque developed by the motor is $T_m(t) = K_i i_a(t)$, where K_i is the torque constant.

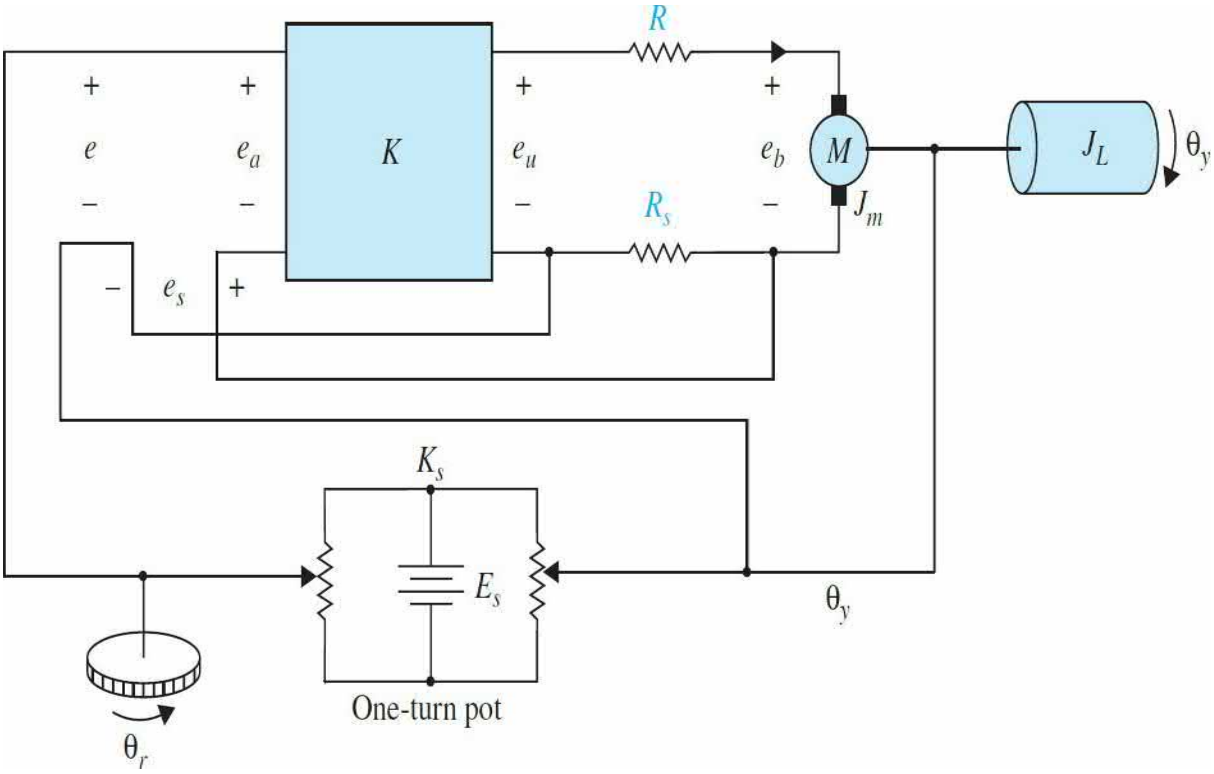


Figure 8P-22

The constants of the system are

$$K_s = 2 \quad R = 2\,\Omega \quad R_s = 0.1\,\Omega$$

$$K_b = 5\,\text{V/rad/s} \quad K_i = 5\,\text{N}\cdot\text{m/A} \quad L_a \cong 0\,\text{H}$$

$$J_m + J_L = 0.1\,\text{N}\cdot\text{m}\cdot\text{s}^2 \quad B_m \cong 0\,\text{N}\cdot\text{m}\cdot\text{s}$$

Assume that all the units are consistent so that no conversion is necessary.

(a) Let the state variables be assigned as $x_1 = \theta_y$ and $x_2 = d\theta_y/dt$. Let the output be $y = \theta_y$. Write the state equations in vector-matrix form. Show that the matrices **A** and **B** are in CCF.

(b) Let $\theta_r(t)$ be a unit-step function. Find $\mathbf{x}(t)$ in terms of $\mathbf{x}(0)$, the initial state. Use the Laplace transform table.

(c) Find the characteristic equation and the eigenvalues of **A**.

(d) Comment on the purpose of the feedback resistor R_s.

8-23. Repeat Prob. 8-22 with the following system parameters:

$$K_s = 1 \quad K = 9 \quad R_a = 0.1\,\Omega$$

$$R_s = 0.1\,\Omega \quad K_b = 1\,\text{V/rad/s} \quad K_i = 1\,\text{N}\cdot\text{m/A}$$

$$L_a \cong 0\,\text{H} \quad J_m + J_L = 0.01\,\text{N}\cdot\text{m}\cdot\text{s}^2 \quad B_m \cong 0\,\text{N}\cdot\text{m}\cdot\text{s}$$

8-24. Consider that matrix A can be diagonalized. Show that $e^{\mathbf{A}t} = \mathbf{P}e^{\mathbf{D}t}\mathbf{P}^{-1}$, where **P** transforms **A** into a diagonal matrix, and $\mathbf{P}^{-1}\mathbf{A}\mathbf{P} = \mathbf{D}$, where **D** is a diagonal matrix.

8-25. Consider that matrix **A** can be transformed to the Jordan canonical form, then $e^{\mathbf{A}t} = \mathbf{S}e^{\mathbf{J}t}\mathbf{S}^{-1}$, where **S** transforms **A** into a Jordan canonical form and **J** is in a Jordan canonical form.

8-26. The block diagram of a feedback control system is shown in Fig. 8P-26.

(a) Find the forward-path transfer function $Y(s)/E(s)$ and the closed-loop transfer function $Y(s)/R(s)$.

(b) Write the dynamic equations in the form of

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}r(t) \quad y(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}r(t)$$

Find **A, B, C,** and **D** in terms of the system parameters.

(c) Apply the final-value theorem to find the steady-state value of the output $y(t)$ when the input $r(t)$ is a unit-step function. Assume that the closed-loop system is stable.

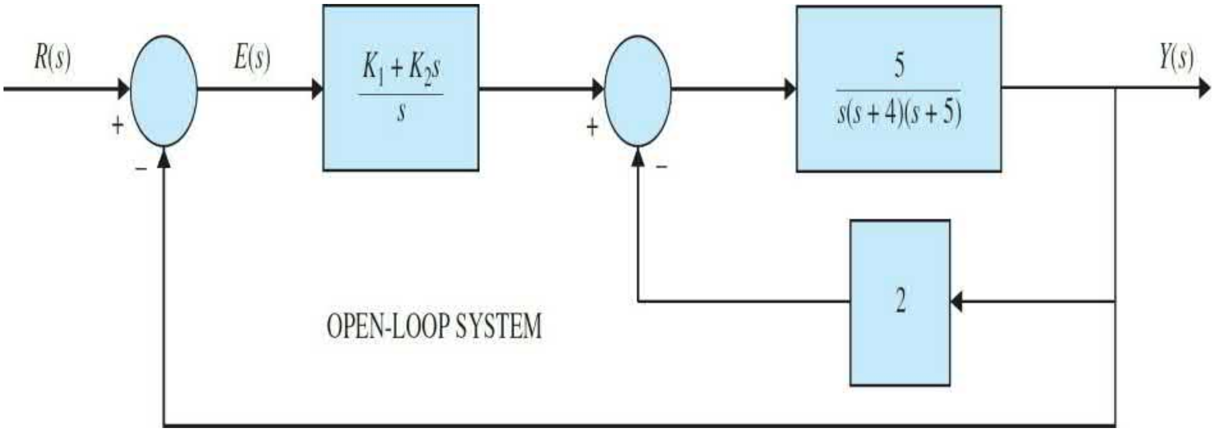


Figure 8P-26

8-27. For the linear time-invariant system whose state equations have the coefficient matrices given by Eqs. (8-191) and (8-192) (CCF), show that

$$\operatorname{adj}(s\mathbf{I} - \mathbf{A})\mathbf{B} = \begin{bmatrix} 1 \\ s \\ s^2 \\ \vdots \\ s^{n-1} \end{bmatrix}$$

and the characteristic equation of **A** is

$$s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 = 0$$

8-28. A linear time-invariant system is described by the differential equation

$$\frac{d^3y(t)}{dt^3}+3\frac{d^2y(t)}{dt^2}+3\frac{dy(t)}{dt}+y(t)=r(t)$$

(a) Let the state variables be defined as $x_1 = y$, $x_2 = dy/dt$, and $x_3 = d^2y/dt^2$. Write the state equations of the system in vector-matrix form.

(b) Find the state-transition matrix $\phi(t)$ of A.

(c) Let $y(0) = 1$, $dy(0)/dt = 0$, $d^2y(0)/dt^2 = 0$, and $r(t) = u_s(t)$. Find the state-transition equation of the system.

(d) Find the characteristic equation and the eigenvalues of **A**.

8-29. A spring-mass-friction system is described by the following differential equation:

$$\frac{d^2y(t)}{dt^2}+2\frac{dy(t)}{dt}+y(t)=r(t)$$

(a) Define the state variables as $x_1(t) = y(t)$ and $x_2(t) = dy(t)/dt$. Write the state equations in vector-matrix form. Find the state-transition matrix $\phi(t)$ of **A**.

(b) Define the state variables as $x_1(t) = y(t)$ and $x_2(t) = y(t) + dy(t)/dt$. Write the state equations in vector-matrix form. Find the state-transition matrix $\phi(t)$ of **A**.

(c) Show that the characteristic equations, $|s\mathbf{I} - \mathbf{A}| = 0$, for parts (a) and (b) are identical.

8-30. Given the state equations $d\mathbf{x}(t)/dt = \mathbf{A}\mathbf{x}(t)$, where σ and ω are real numbers:

(a) Find the state transition matrix of **A**.

(b) Find the eigenvalues of **A**.

8-31. **(a)** Show that the input-output transfer functions of the two systems shown in Fig. 8P-31 are the same.

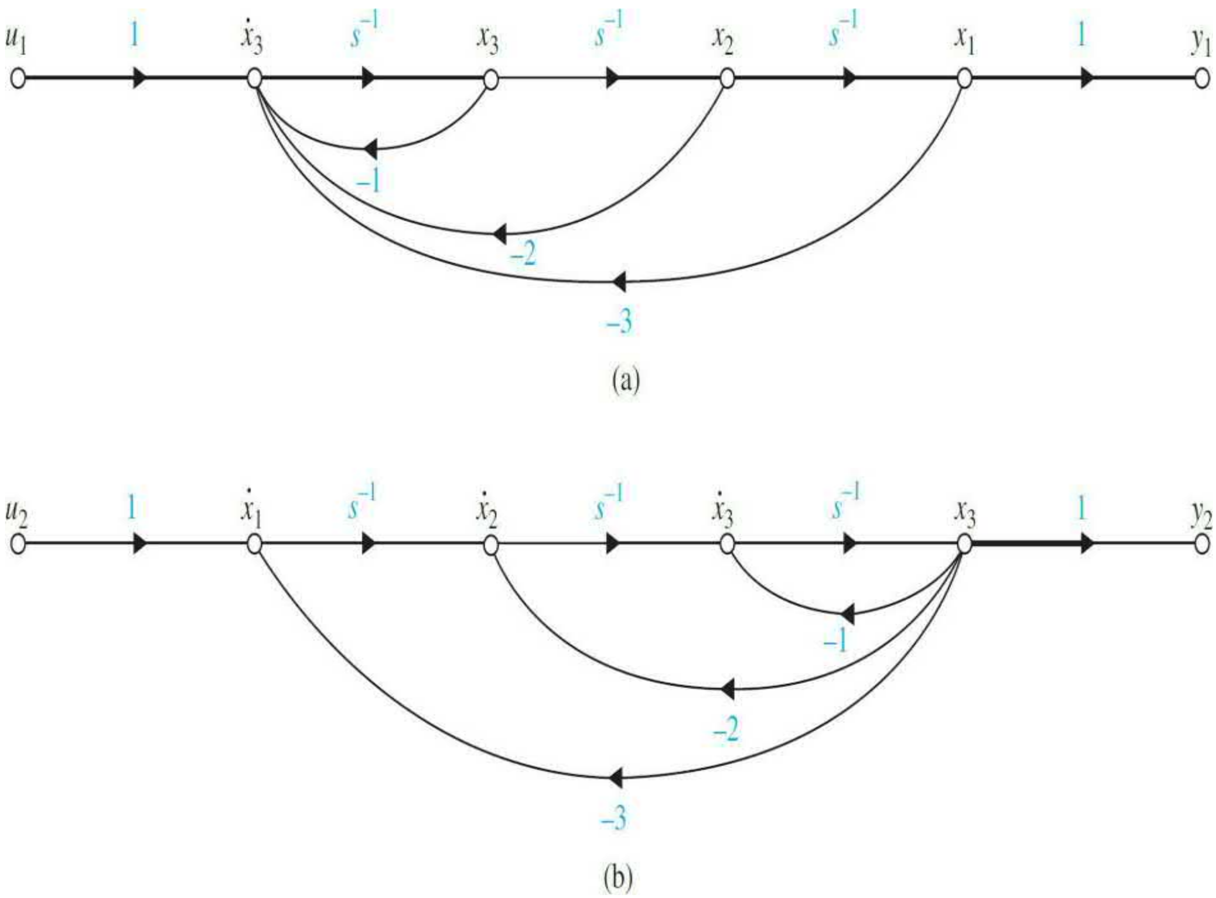


Figure 8P-31

(b) Write the dynamic equations of the system in Fig. 8P-31*a* as

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}_1\mathbf{x}(t) + \mathbf{B}_1 u_1(t) \quad y_1(t) = \mathbf{C}_1\mathbf{x}(t)$$

and those of the system in Fig. 8-31*b* as

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}_2\mathbf{x}(t) + \mathbf{B}_2 u_2(t) \quad y_2(t) = \mathbf{C}_2\mathbf{x}(t)$$

8-32. Draw the state diagrams for the following systems.

$$\frac{d\mathrm{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{Bu}(t)$$

(a) $$\mathbf{A} = \begin{bmatrix} -3 & 2 & 0 \\ -1 & 0 & 1 \\ -2 & -3 & -4 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

(b) Same **A** as in part (a), but with

$$\mathbf{B} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix}$$

8-33. Draw state diagrams for the following transfer functions by direct decomposition. Assign the state variables from right to left for x_1, x_2, …. Write the state equations from the state diagram and show that the equations are in CCF.

(a) $$G(s) = \frac{10}{s^3 + 8.5s^2 + 20.5s + 15}$$

(b) $$G(s) = \frac{10(s+2)}{s^2(s+1)(s+3.5)}$$

(c) $$G(s) = \frac{5(s+1)}{s(s+2)(s+10)}$$

(d) $$G(s) = \frac{1}{s(s+5)(s^2+2s+2)}$$

8-34. Draw state diagrams for the systems described in Prob. 8-33 by parallel decomposition. Make certain that the state diagrams contain a minimum number of integrators. The constant branch gains must be real. Write the state equations from the state diagram.

8-35. Draw the state diagrams for the systems described in Prob. 8-33 by using cascade decomposition. Assign the state variables in ascending order from right to left. Write the state equations from the state diagram.

8-36. The block diagram of a feedback control system is shown in Fig. 8P-36.

(a) Draw a state diagram for the system by first decomposing *G*(*s*) by direct decomposition. Assign the state variables in ascending order, x_1, x_2,…, from right to left. In addition to the state-variable-related nodes, the state diagram should contain nodes for $R(s)$, $E(s)$, and $C(s)$.

(b) Write the dynamic equations of the system in vector-matrix form.

(c) Find the state-transition equations of the system using the state equations found in part (b). The initial state vector is **x**(0), and $r(t) = us(t)$.

(d) Find the output $y(t)$ for $t \geq 0$ with the initial state **x**(0), and $r(t) = u_s(t)$.

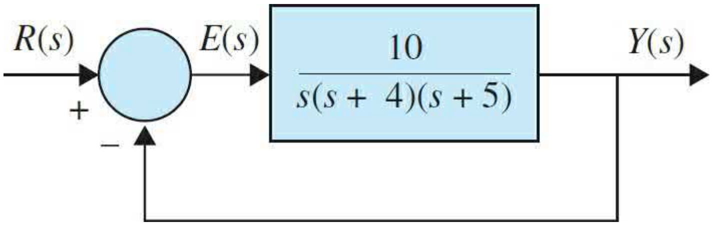


Figure 8P-36

8-37.

(a) Find the closed-loop transfer function $Y(s)/R(s)$, and draw the state diagram.

(b) Perform a direct decomposition to $Y(s)/R(s)$, and draw the state diagram.

(c) Assign the state variables from right to left in ascending order, and write the state equations in vector-matrix form.

(d) Find the state-transition equations of the system using the state equations found in part (c). The initial state vector is **x**(0), and $r(t) = u_s(t)$.

(e) Find the output $y(t)$ for $t \geq 0$ with the initial state **x**(0), and $r(t) = u_s(t)$.

8-38. The block diagram of a linearized idle-speed engine-control system of an automobile is shown in Fig. 8P-38. (For a discussion on linearization of nonlinear systems, refer to Sect. 4-9.) The system is linearized about a nominal operating point, so all the variables represent linear-perturbed quantities. The following variables are defined: $T_m(t)$ is the engine torque; T_D, the constant load-disturbance

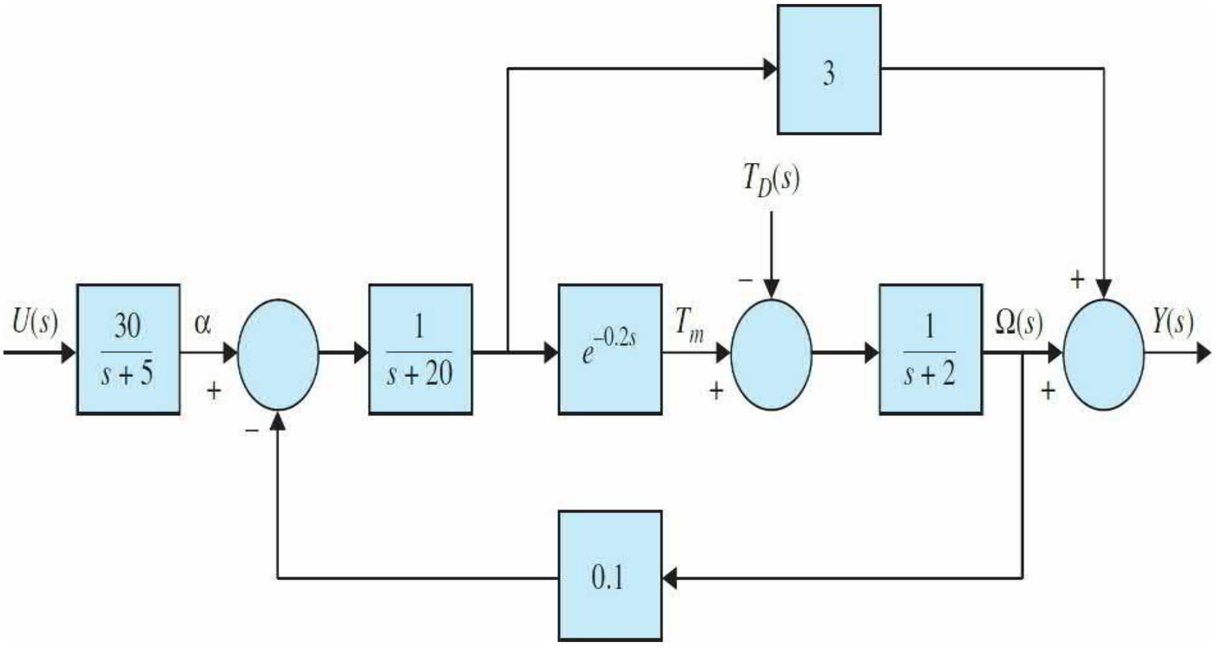


Figure 8P-38

torque; ω (t), the engine speed; $u(t)$, the input-voltage to the throttle actuator; and α, the throttle angle. The time delay in the engine model can be approximated by

$$e^{-0.2s} \cong \frac{1-0.1s}{1+0.1s}$$

(a) Draw a state diagram for the system by decomposing each block individually. Assign the state variables from right to left in ascending order.

(b) Write the state equations from the state diagram obtained in part (a), in the form of

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\begin{bmatrix} u(t) \\ T_D(t) \end{bmatrix}$$

(c) Write $Y(s)$ as a function of $U(s)$ and $T_D(s)$. Write $\Omega(s)$ as a function of $U(s)$ and $T_D(s)$.

8-39. The state diagram of a linear system is shown in Fig. 8P-39.

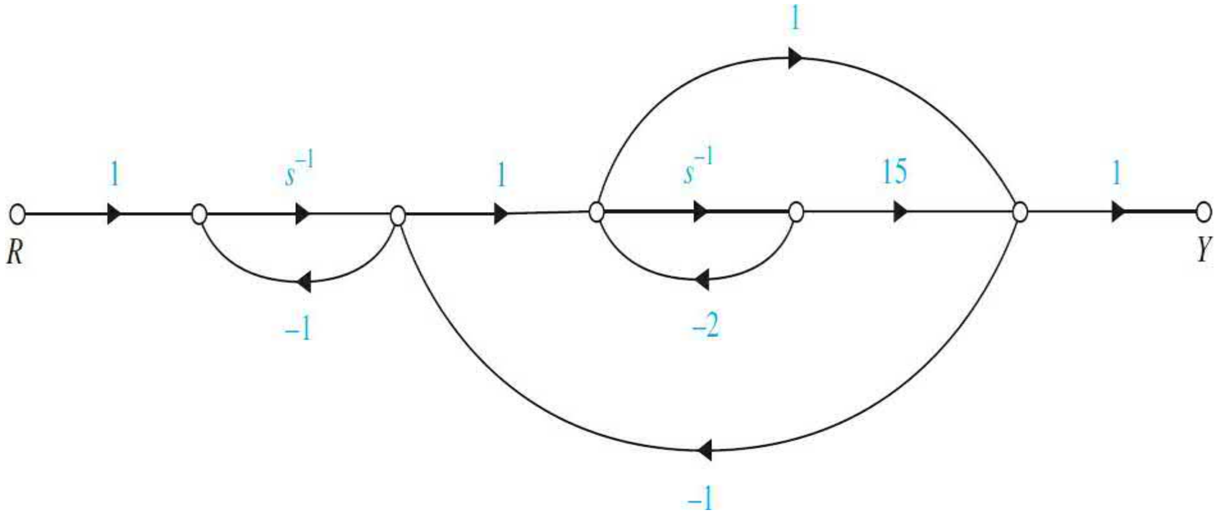


Figure 8P-39

(a) Assign state variables on the state diagram from right to left in ascending order. Create additional artificial nodes if necessary so that the state-variable nodes satisfy as "input nodes" after the integrator branches are deleted.

(b) Write the dynamic equations of the system from the state diagram in part (a).

8-40. The block diagram of a linear spacecraft-control system is shown in Fig. 8P-40.

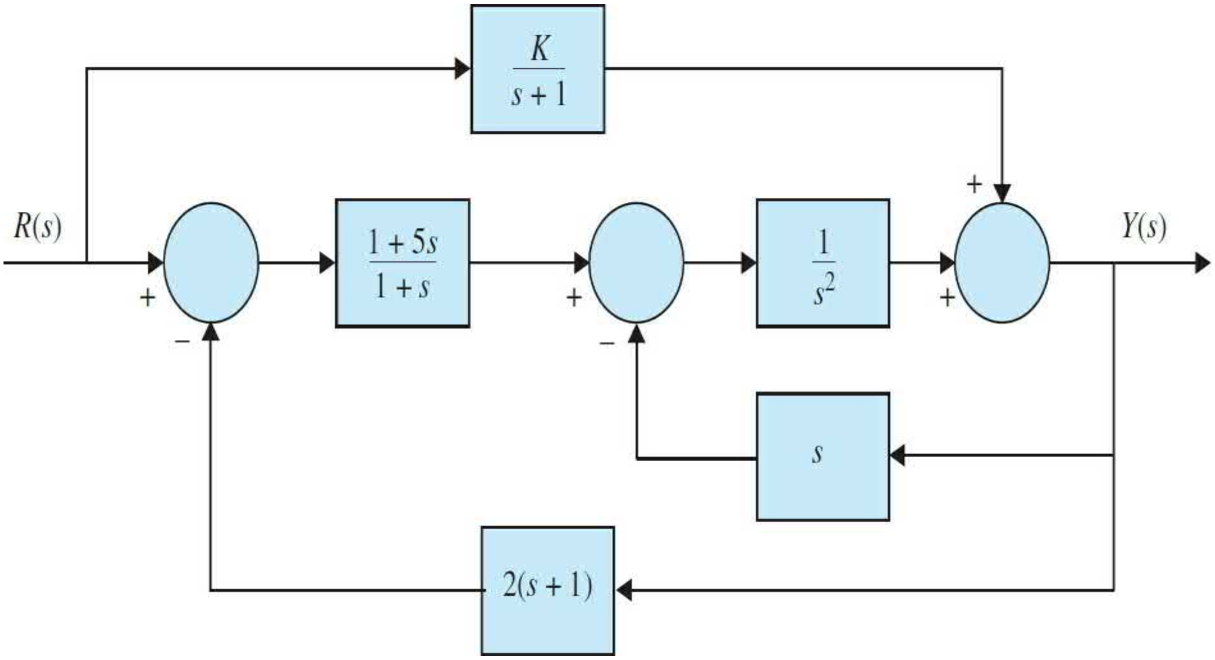


Figure 8P-40

(a) Determine the transfer function *Y*(*s*)/*R*(*s*).

(b) Find the characteristic equation and its roots of the system. Show that the roots of the characteristic equation are not dependent on *K*.

(c) When $K = 1$, draw a state diagram for the system by decomposing *Y*(*s*)/*R*(*s*), using a minimum number of integrators.

(d) Repeat part (c) when $K = 4$.

(e) Determine the values of *K* that must be avoided if the system is to be both state controllable and observable.

8-41. A considerable amount of effort is being spent by automobile manufacturers to meet the exhaust-emission-performance standards set by the government. Modern automobile-power-plant systems consist of an internal combustion engine that has an internal cleanup device called a catalytic converter. Such a system requires control of such variables as the engine air–fuel (A/F) ratio, ignition-spark timing, exhaust-gas recirculation, and injection air. The control-system problem considered in this problem deals with the control of the A/F ratio. In general, depending on fuel composition and other factors, a typical stoichiometric A/F is 14.7:1, that is, 14.7 grams of air to each gram of fuel. An A/F greater or less than stoichiometry will cause high hydrocarbons, carbon monoxide, and nitrous oxides in the tailpipe emission. The control system shown in Fig. 8P-41 is devised to control the air–fuel ratio so that a desired output is achieved for a given input command.

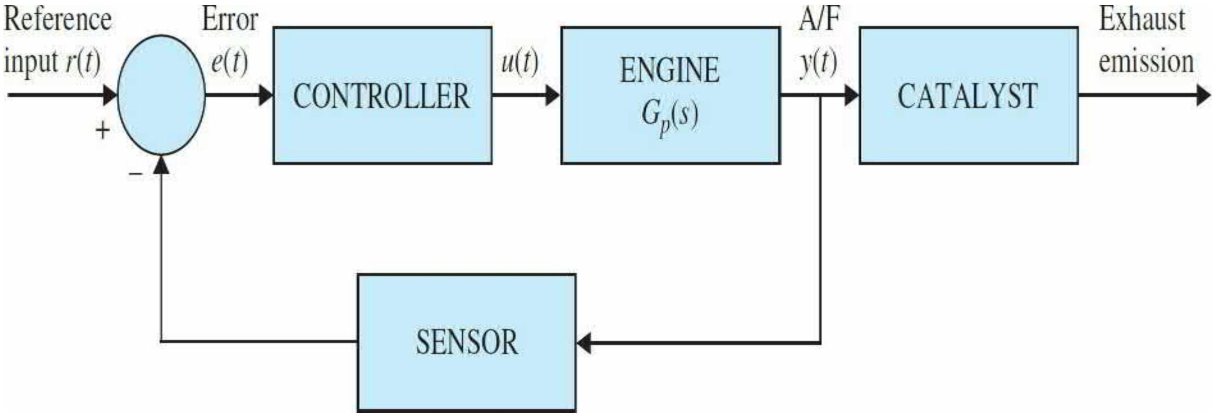


Figure 8P-41

The sensor senses the composition of the exhaust-gas mixture entering the catalytic converter. The electronic controller detects the difference or the error between the command and the error and computes the control signal necessary to achieve the desired exhaust-gas composition. The output $y(t)$ denotes the effective air–fuel ratio. The transfer function of the engine is given by

$$G_p(s) = \frac{Y(s)}{U(s)} = \frac{e^{-T_d s}}{1+0.5s}$$

where T_d = 0.2 s is the time delay and is approximated by

$$e^{-T_d s} = \frac{1}{e^{T_d s}} = \frac{1}{1+T_d s+T_d^2 s^2/2!+\cdots} \cong \frac{1}{1+T_d s+T_d^2 s^2/2!}$$

The gain of the sensor is 1.0.

(a) Using the approximation for $e^{-T_d s}$ given, find the expression for $G_p(s)$. Decompose $G_p(s)$ by direct decomposition, and draw the state diagram with $u(t)$ as the input and $y(t)$ as the output. Assign state variables from right to left in ascending order, and write the state equations in vector-matrix form.

(b) Assuming that the controller is a simple amplifier with a gain of l, i.e., $u(t) = e(t)$, find the characteristic equation and its roots of the closed-loop system.

8-42. Repeat Prob. 8-41 when the time delay of the automobile engine is approximated as

$$e^{-T_d s} \cong \frac{1 - T_d s/3}{1 + \frac{2}{3}T_d s + \frac{1}{6}T_d^2 s^2} \qquad T_d = 0.2 \text{ s}$$

8-43. The schematic diagram in Fig. 8P-43 shows a permanent-magnet dc-motor-control system with a viscous-inertia damper. The system can be used for the control of the printwheel of an electronic word processor. A mechanical damper such as the viscous-inertia type is sometimes used in practice as a simple and economical way of stabilizing a control system. The damping effect is achieved by a rotor suspended in a viscous fluid. The differential and algebraic equations that describe the dynamics of the system are as follows:

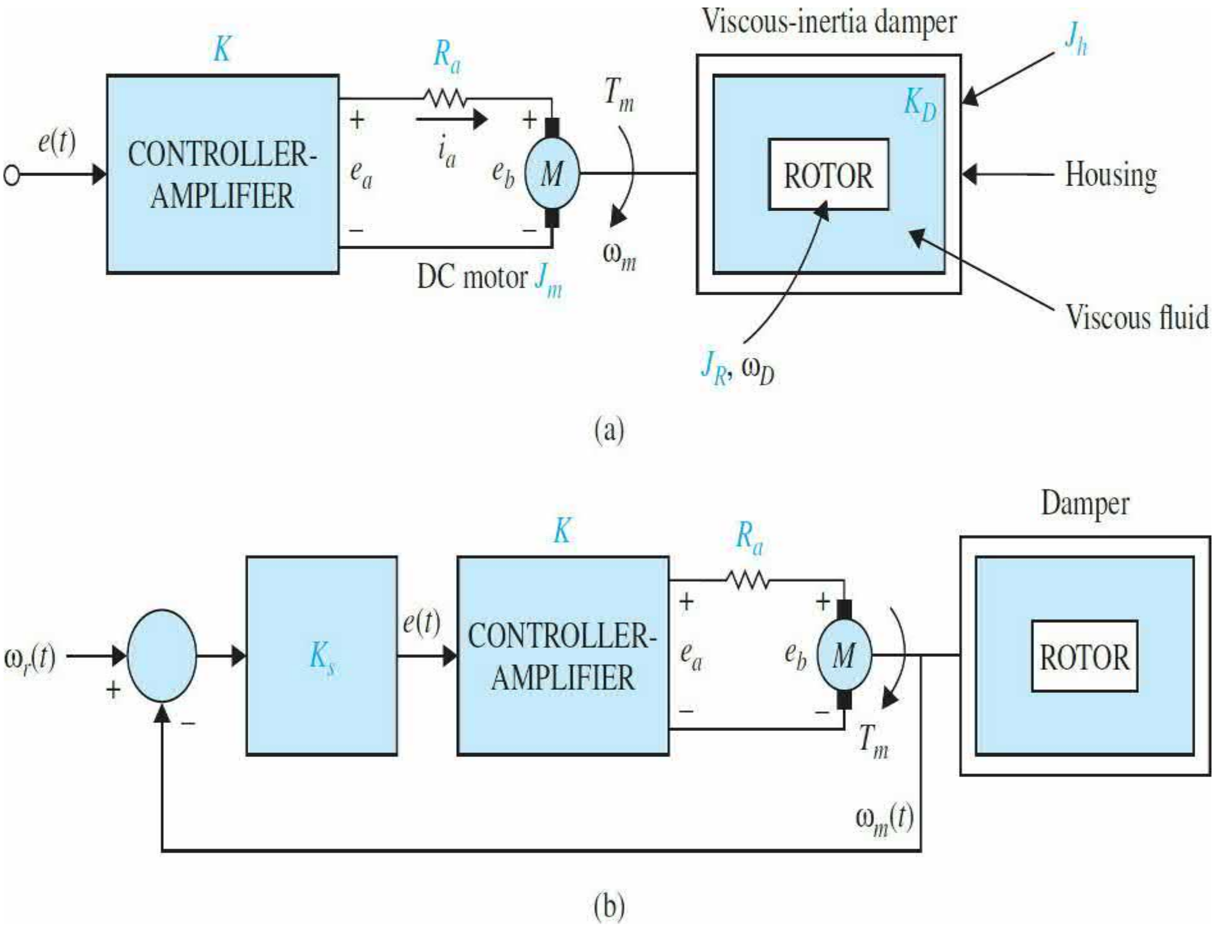


Figure 8P-43

$$\begin{aligned}
&e(t) = K_s[\omega_r(t) - \omega_m(t)] && K_s = 1\,V/\text{rad/s} \\
&e_a(t) = Ke(t) = R_a i_a(t) + e_b(t) && K = 10 \\
&e_b(t) = K_b \omega_m(t) && K_b = 0.0706\ \text{V/rad/s} \\
&T_m(t) = J\frac{d\omega_m(t)}{dt} + K_D[\omega_m(t) - \omega_D(t)] && J = J_h + J_m = 0.1\ \text{oz}\cdot\text{in}\cdot\text{s}^2 \\
&T_m(t) = K_i i_a(t) && K_i = 10\ \text{oz}\cdot\text{in/A} \\
&K_D[\omega_m(t) - \omega_D(t)] = J_R\frac{d\omega_D(t)}{dt} && J_R = 0.05\ \text{oz}\cdot\text{in}\cdot\text{s}^2 \\
&R_a = 1\,\Omega && K_D = 1\ \text{oz}\cdot\text{in}\cdot\text{s}
\end{aligned}$$

(a) Let the state variables be defined as $x_1(t) = \omega_m(t)$ and $x_2(t) = \omega_D(t)$. Write the state equations for the open-loop system with $e(t)$ as the input. (*Open-loop* refers to the feedback path from ω_m to e being open.)

(b) Draw the state diagram for the overall system using the state equations found in part (a) and $e(t) = K_s[\omega_r(t) - \omega_m(t)]$.

(c) Derive the open-loop transfer function $\Omega_m(s)/E(s)$ and the closed-loop transfer function $\Omega_m(s)/\Omega_r(s)$.

8-44. Determine the state controllability of the system shown in Fig. 8P-44.

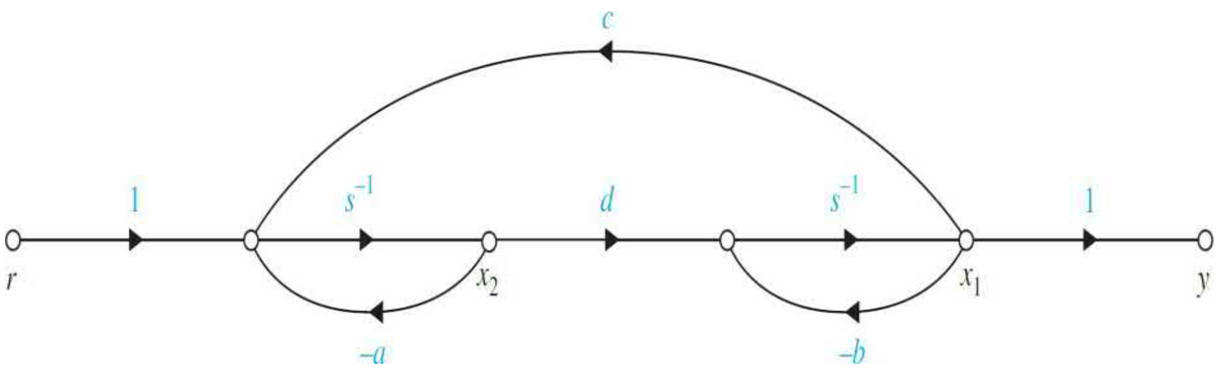


Figure 8P-44

(a) $a = 1$, $b = 2$, $c = 2$, and $d = 1$.

(b) Are there any nonzero values for *a*, *b*, *c*, and *d* such that the system is uncontrollable?

8-45. Determine the controllability of the following systems:

(a)
$$\mathrm{A} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

(b)
$$\mathrm{A} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

8-46. Determine the controllability and observability of the system shown in Fig. 8P-46 by the following methods:

(a) Conditions on the **A, B, C,** and **D** matrices

(b) Conditions on the pole-zero cancellation of the transfer functions

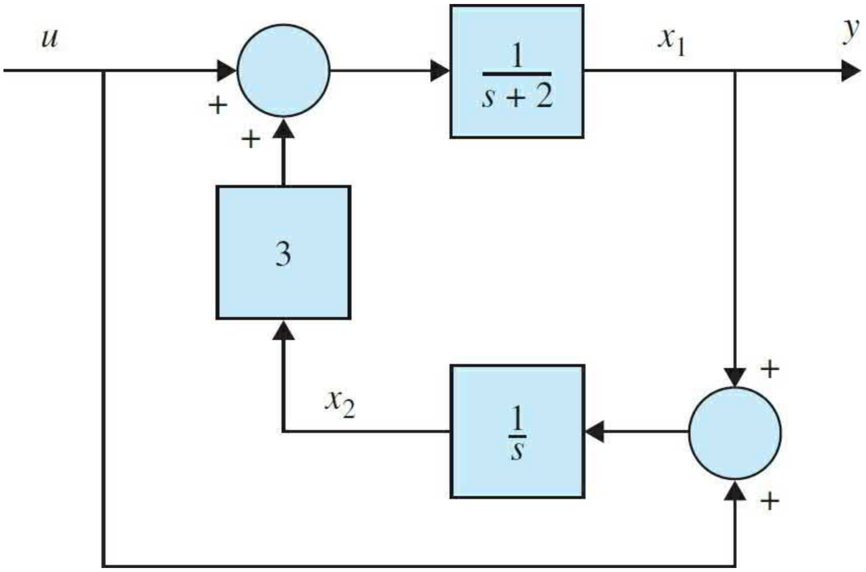


Figure 8P-46

8-47. The transfer function of a linear control system is

$$\frac{Y(s)}{R(s)} = \frac{s+\alpha}{s^3+7s^2+14s+8}$$

(a) Determine the value(s) of *α* so that the system is either uncontrollable or unobservable.

(b) With the value(s) of *α* found in part (a), define the state variables so that one of them is uncontrollable.

(c) With the value(s) of *α* found in part (a), define the state variables so that one of them is unobservable.

8-48. Consider the system described by the state equation

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t)$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -1 & a \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ b \end{bmatrix}$$

Find the region in the *a*-*b* plane such that the system is completely controllable.

8-49. Determine the condition on b_1, b_2, c_1, and c_2 so that the following system is completely controllable and observable.

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \quad y(t) = \mathbf{C}\mathbf{x}(t)$$

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} d_1 & d_2 \end{bmatrix}$$

8-50. The schematic diagram of Fig. 8P-50 represents a control system whose purpose is to hold the level of the liquid in the tank at a desired level. The liquid level is controlled by a float whose position *h*(*t*) is monitored. The input signal of the open-loop system is *e*(*t*). The system parameters and equations are as follows:

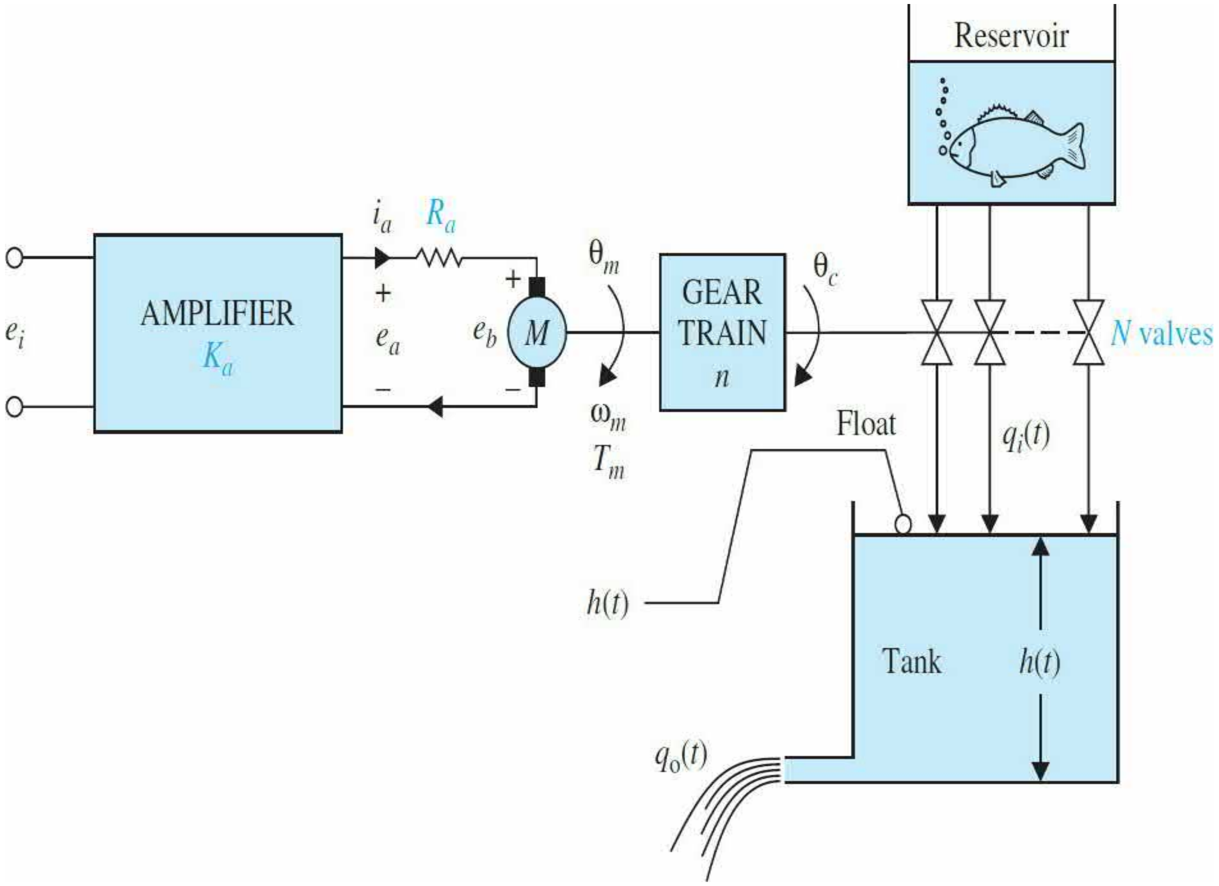


Figure 8P-50

Motor resistance R_a	$= 10\ \Omega$	Motor inductance L_a	$= 0$ H
Torque constant K_i	$= 10$ oz · in/A	Rotor inertia J_m	$= 0.005$ oz · in · s^2
Back-emf constant K_b	$= 0.0706$ V/rad/s	Gear ratio n	$= N_1/N_2 = 1/100$
Load inertia J_L	$= 10$ oz · in · s^2	Load and motor friction	= negligible
Amplifier gain K_a	$= 50$	Area of tank A	$= 50$ ft^2

$$e_a(t) = R_a i_a(t) + K_b \omega_m(t) \quad \omega_m(t) = \frac{d\theta_m(t)}{dt}$$

$$T_m(t) = K_i i_a(t) = (J_m + n^2 J_L)\frac{d\omega_m(t)}{dt} \quad \theta_y(t) = n\theta_m(t)$$

The number of valves connected to the tank from the reservoir is $N = 10$. All the valves have the same characteristics and are controlled simultaneously by θ_y. The equations that govern the volume of flow are as follows:

$$q_i(t) = K_I N \theta_y(t) \quad K_I = 10\,\text{ft}^3/\text{s}\cdot\text{rad}$$

$$q_o(t) = K_o h(t) \qquad K_o = 50\,\text{ft}^2/\text{s}$$

$$h(t) = \frac{\text{volume of tank}}{\text{area of tank}} = \frac{1}{A}\int[q_i(t) - q_o(t)]dt$$

(a) Define the state variables as $x_1(t) = h(t)$, $x_2(t) = \theta_m(t)$, and $x_3(t) = d\theta_m(t)/dt$. Write the state equations of the system in the form of $d\mathbf{x}(t)/dt = \mathbf{A}\mathbf{x}(t) + \mathbf{B}e_i(t)$. Draw a state diagram for the system.

(b) Find the characteristic equation and the eigenvalues of the **A** matrix found in part (a).

(c) Show that the open-loop system is completely controllable; that is, the pair [**A, B**] is controllable.

(d) For reasons of economy, only one of the three state variables is measured and fed back for control purposes. The output equation is $y = \mathbf{C}\mathbf{x}$, where **C** can be one of the following forms:

1. $\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$

2. $\mathbf{C} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$

3. $\mathbf{C} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$

Determine which case (or cases) corresponds to a completely observable system.

8-51. The “broom-balancing” control system described in Prob. 6-21 has the following parameters:

$$M_b = 1\,\text{kg} \quad M_c = 10\,\text{kg} \quad L = 1\,\text{m} \quad g = 32.2\,\text{ft/s}^2$$

The small-signal linearized state equation model of the system is

$$\Delta\dot{\mathbf{x}}(t) = \mathbf{A}^* \, \Delta\mathbf{x}(t) + \mathbf{B}^* \, \Delta r(t)$$

where

$$\mathbf{A}^* = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 25.92 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -2.36 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{B}^* = \begin{bmatrix} 0 \\ -0.0732 \\ 0 \\ 0.0976 \end{bmatrix}$$

(a) Find the characteristic equation of **A*** and its roots.

(b) Determine the controllability of [**A***, **B***].

(c) For reason of economy, only one of the state variables is to be measured for feedback.

The output equation is written

$$\Delta y(t) = C^* \, \Delta\mathbf{x}(t)$$

where

1. $C^* = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$

2. $C^* = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}$

3. $C^* = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}$

4. $C^* = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}$

Determine which **C*** corresponds to an observable system.

8-52. The double-inverted pendulum shown in Fig. 8P-52 is approximately modeled by the following linear state equation:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t)$$

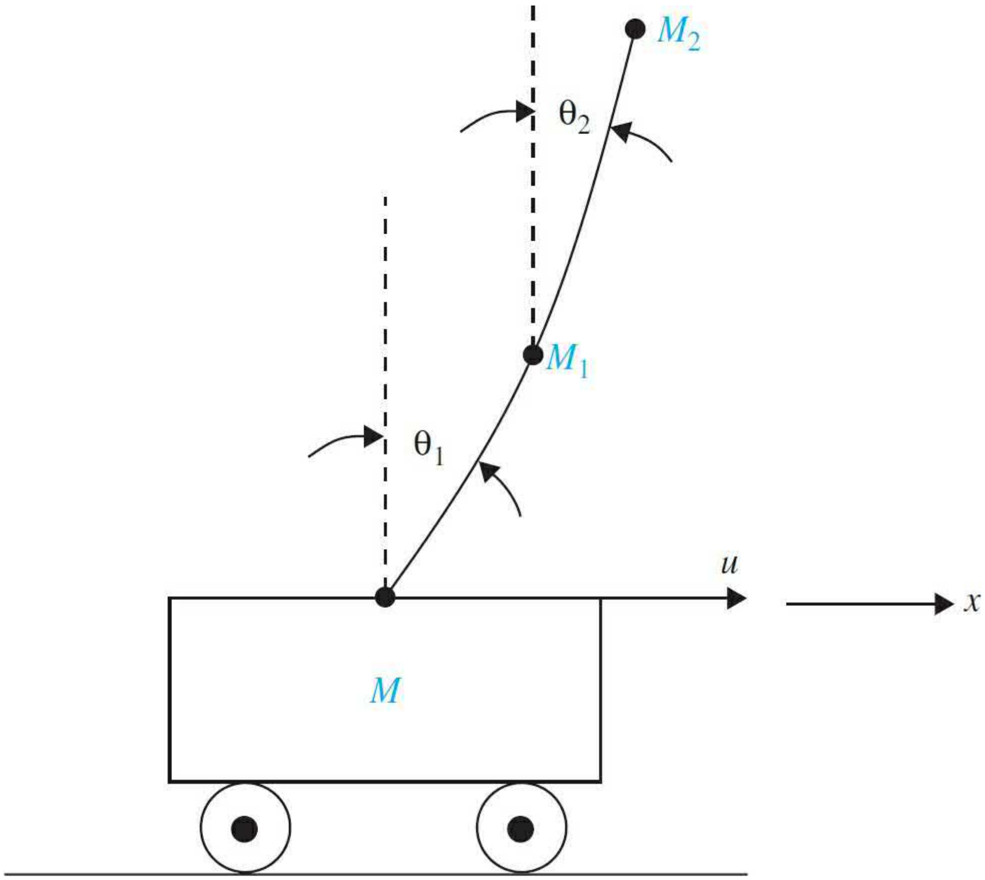


Figure 8P-52

where

$$\mathbf{x}(t)=\begin{bmatrix} \theta_1(t) \\ \dot{\theta}_1(t) \\ \theta_2(t) \\ \dot{\theta}_1(t) \\ x(t) \\ \dot{x}(t) \end{bmatrix}$$

$$\mathbf{A}=\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 16 & 0 & -8 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ -16 & 0 & 16 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{B}=\begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Determine the controllability of the states.

8-53. The block diagram of a simplified control system for the large space telescope (LST) is shown in Fig. 8P-53. For simulation and control purposes, model the system by state equations and by a state diagram.

(a) Draw a state diagram for the system and write the state equations in vector-matrix form. The state diagram should contain a minimum number of state variables, so it would be helpful if the transfer function of the system is written first.

(b) Find the characteristic equation of the system.

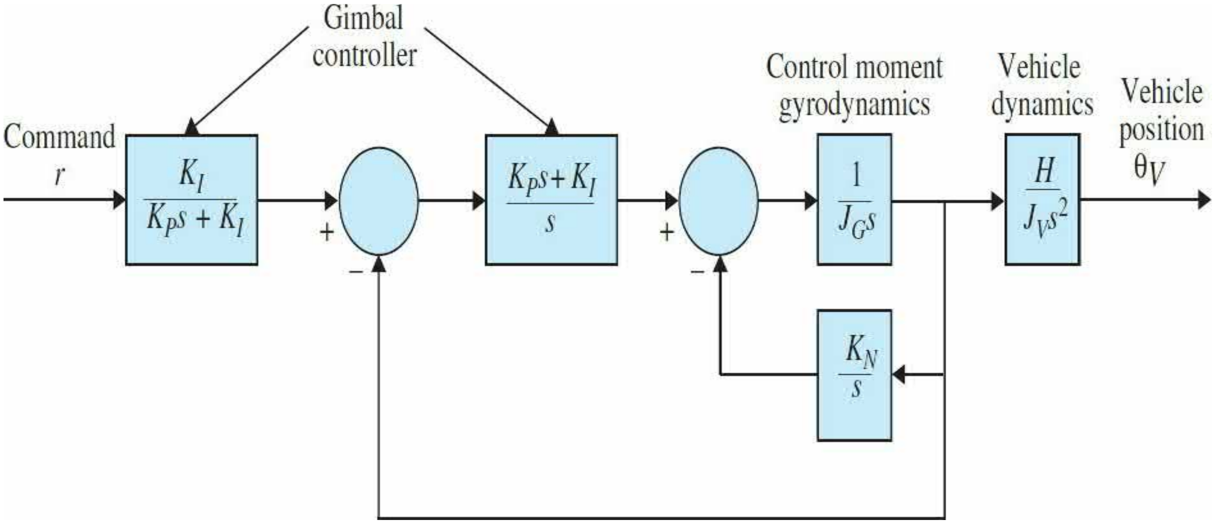


Figure 8P-53

8-54. The state diagram shown in Fig. 8P-54 represents two subsystems connected in cascade.

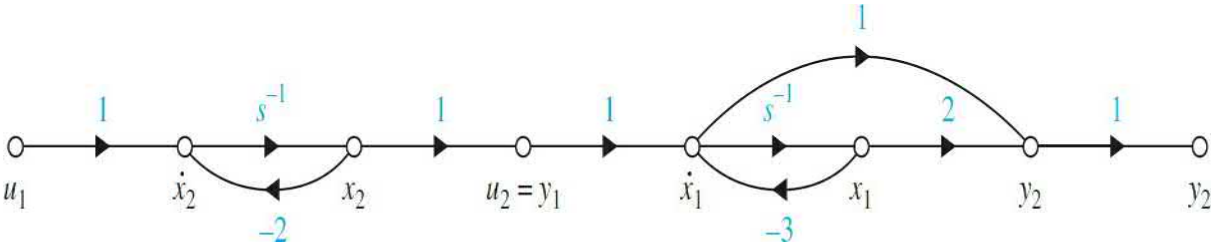


Figure 8P-54

(a) Determine the controllability and observability of the system.

(b) Consider that output feedback is applied by feeding back y_2 to u_2; that is, $u_2 = -ky_2$, where k is a real constant. Determine how the value of k affects the controllability and observability of the system.

8-55. Given the system

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \quad y(t) = \mathbf{C}\mathbf{x}(t)$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -1 & -3 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 1 \end{bmatrix}$$

(a) Determine the state controllability and observability of the system.

(b) Let $u(t) = -\mathbf{Kx}(t)$, where $\mathbf{K} = [k_1 k_2]$, and k_1 and k_2 are real constants. Determine if and how controllability and observability of the closed-loop system are affected by the elements of **K**.

8-56. The torque equation for a system is given by

$$J\frac{d^2\theta(t)}{dt^2} = K_F d_1 \theta(t) + T_s d_2 \delta(t)$$

where $K_F d_1 = 1$ and $J = 1$. Define the state variables as $x_1 = \theta$ and $x_2 = d\theta/dt$. Find the state-transition matrix $\phi(t)$ using ACSYS/MATLAB.

8-57. Starting with the state equation $d\mathbf{x}(t)/dt = \mathbf{Ax}(t) + \mathbf{B}\theta_r$ obtained in Prob. 8-22, use **ACSYS**/MATLAB to do the following:

(a) Find the state-transition matrix of **A**, $\phi(t)$.

(b) Find the characteristic equation of **A**.

(c) Find the eigenvalues of **A**.

(d) Compute and plot the unit-step response of $y(t) = \theta_y(t)$ for 3 s. Set all the initial conditions to zero.

8-58. The block diagram of a control system with state feedback is shown in Fig. 8P-58. Find the real feedback gains k_1, k_2, and k_3 so that

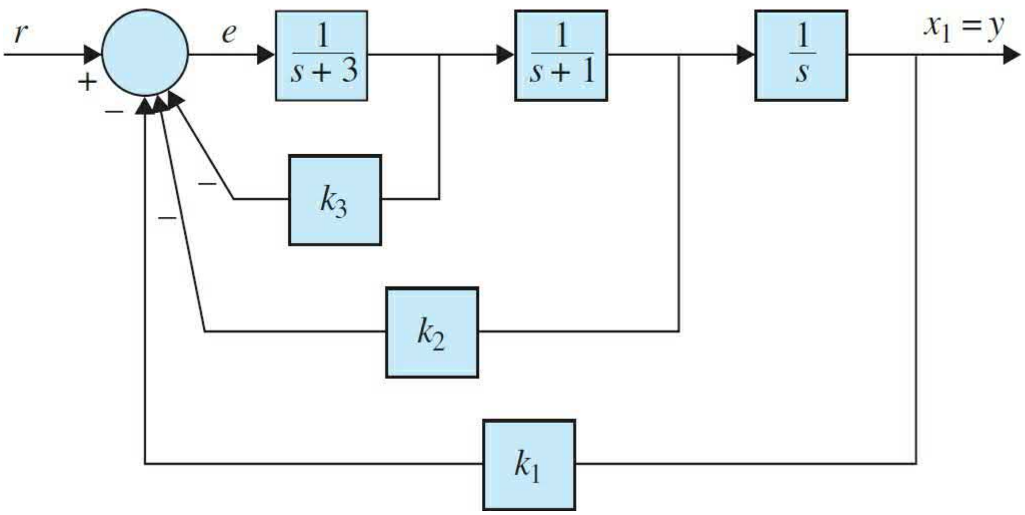


Figure 8P-58

- The steady-state error e_{ss} [$e(t)$ is the error signal] due to a step input is zero.
- The complex roots of the characteristic equation are at −1 + j and −1 −j.
- Find the third root. Can all three roots be arbitrarily assigned while still meeting the steady-state requirement?

8-59. The block diagram of a control system with state feedback is shown in Fig. 8P-59*a*. The feedback gains k_1, k_2, and k_3 are real constants.

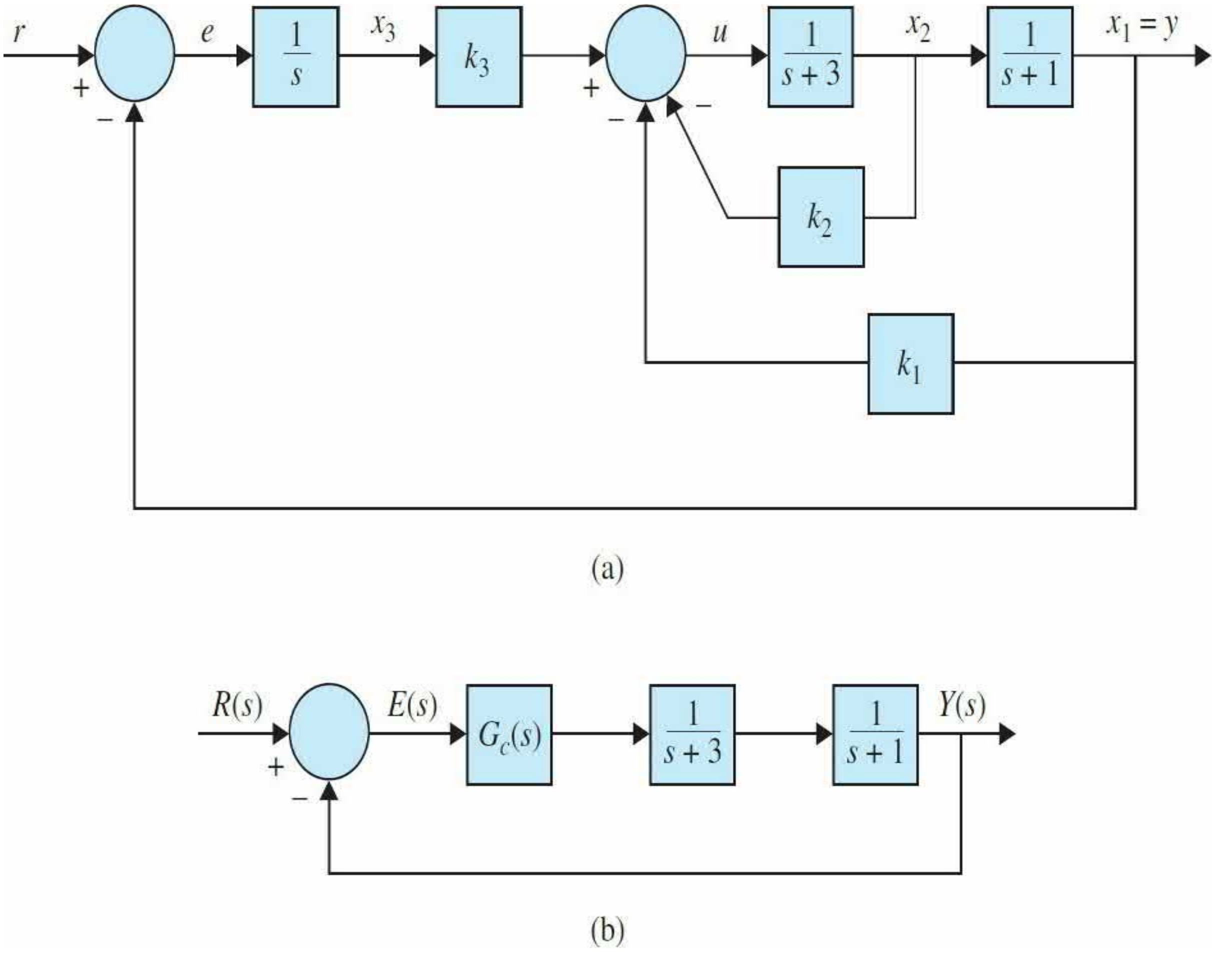


Figure 8P-59

(a) Find the values of the feedback gains so that:

- The steady-state error e_{ss}[$e(t)$ is the error signal] due to a step input is zero.
- The characteristic equation roots are at $-1 + j$, $-1 - j$, and -10.

(b) Instead of using state feedback, a series controller is implemented, as shown in Fig. 8P-59*b*. Find the transfer function of the controller $G_c(s)$ in terms of k_1, k_2, and k_3 found in part (a) and the other system parameters.

8-60. Problem 8-39 has revealed that it is impossible to stabilize the broom-balancing control system described in Probs. 4-21 and 8-51 with a series PD controller. Consider that the system is now controlled by state feedback with $\Delta r(t) = -\mathbf{Kx}(t)$, where

$$\mathbf{K} = \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \end{bmatrix}$$

(a) Find the feedback gains k_1, k_2, k_3, and k_4 so that the eigenvalues of $\mathbf{A}^* - \mathbf{B}^*\mathbf{K}$ are at $-1 + j$, $-1 - j$, -10, and -10. Compute and plot the responses of $\Delta x_1(t)$, $\Delta x_2(t)$, $\Delta x_3(t)$, and $\Delta x_4(t)$ for the initial condition, $\Delta x_1(0) = 0.1$, $\Delta\theta(0) = 0.1$, and all other initial conditions are zero.

(b) Repeat part (a) for the eigenvalues at $-2 + j2$, $-2 - j2$, -20, and -20. Comment on the difference between the two systems.

8-61. The linearized state equations of the ball-suspension control system described in Prob. 4-57 are expressed as

$$\Delta\dot{\mathbf{x}}(t) = \mathbf{A}^*\,\Delta\mathbf{x}(t) + \mathbf{B}^*\,\Delta i(t)$$

where

$$\mathbf{A}^* = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 115.2 & -0.05 & -18.6 & 0 \\ 0 & 0 & 0 & 1 \\ -37.2 & 0 & 37.2 & -0.1 \end{bmatrix} \quad \mathbf{B}^* = \begin{bmatrix} 0 \\ -6.55 \\ 0 \\ -6.55 \end{bmatrix}$$

Let the control current $\Delta i(t)$ be derived from the state feedback $\Delta i(t) = -\mathbf{K}\Delta\mathbf{x}(t)$, where

$$\mathbf{K} = \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \end{bmatrix}$$

(a) Find the elements of $\mathbf{K}$ so that the eigenvalues of $\mathbf{A}^* - \mathbf{B}^*\mathbf{K}$ are at $-1 + j$, $-1 - j$, -10, and -10.

(b) Plot the responses of $\Delta x_1(t) = \Delta y_1(t)$ (magnet displacement) and $\Delta x_3(t) = \Delta y_2(t)$ (ball displacement) with the initial condition

$$\Delta\mathbf{x}(0) = \begin{bmatrix} 0.1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

(c) Repeat part (b) with the initial condition

$$\Delta\mathbf{x}(0)=\begin{bmatrix}0\\0\\0.1\\0\end{bmatrix}$$

Comment on the responses of the closed-loop system with the two sets of initial conditions used in (b) and (c).

8-62. The temperature $x(t)$ in the electric furnace shown in Fig. 8P-62 is described by the differential equation

$$\frac{dx(t)}{dt}=-2x(t)+u(t)+n(t)$$

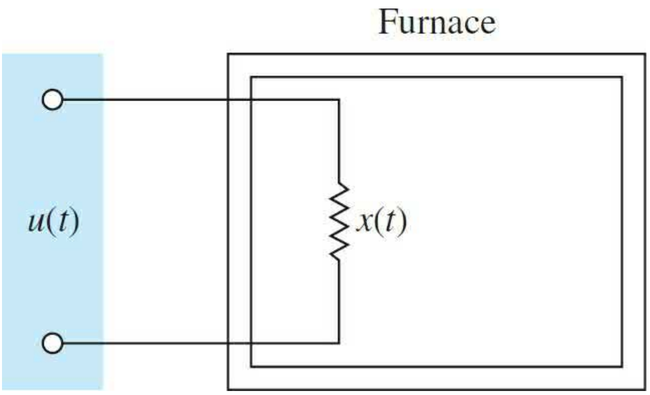


Figure 8P-62

where $u(t)$ is the control signal, and $n(t)$ the constant disturbance of unknown magnitude due to heat loss. It is desired that the temperature $x(t)$ follows a reference input r that is a constant.

(a) Design a control system with state and integral control so that the following specifications are satisfied:

- $\lim_{t\to\infty} x(t)=r=\text{constant}$
- The eigenvalues of the closed-loop system are at −10 and −10.
- Plot the responses of $x(t)$ for $t \geq 0$ with $r = 1$ and $n(t) = -1$, and then

with $r = 1$ and $n(t) = 0$, all with $x(0) = 0$.

(b) Design a PI controller so that

$$G_c(s) = \frac{U(s)}{E(s)} = K_P + \frac{K_I}{s}$$
$$E(s) = R(s) - X(s)$$

where $R(s) = R/s$.

Find K_P and K_I so that the characteristic equation roots are at −10 and −10. Plot the responses of $x(t)$ for $t \geq 0$ with $r = 1$ and $n(t) = -1$, and then with $r = 1$ and $n(t) = 0$, all $x(0) = 0$.

8-63. The transfer function of a system is given by

$$G(s) = \frac{10}{(s+1)(s+2)(s+3)}$$

Find the state-space model of the system if

$$x_1 = y$$
$$x_2 = \dot{x}_1$$
$$x_3 = \dot{x}_2$$

Design a state control feedback $u = -Kx$ so that the closed-loop poles are located at $s = -2 + j2\sqrt{3}$, $s = -2 - j2\sqrt{3}$, and $s = -10$

8-64. Figure 8P-64 shows an inverted pendulum on a moving platform.

Assuming $M = 2$ kg, $m = 0.5$ kg, and $l = 1$ m.

(a) Find the state-space model of the system if $x_1 = \theta, x_2 = \dot{\theta}, x_3 = x, x_4 = \dot{x}, y_1 = x_1 = \theta,$ and $y_2 = x_3 = x$

(b) Design a state feedback control with gain $-K$ so that the closed-loop poles are located at $s = -4 + 4j, s = -4 - 4j, s = -210,$ and $s = -210$

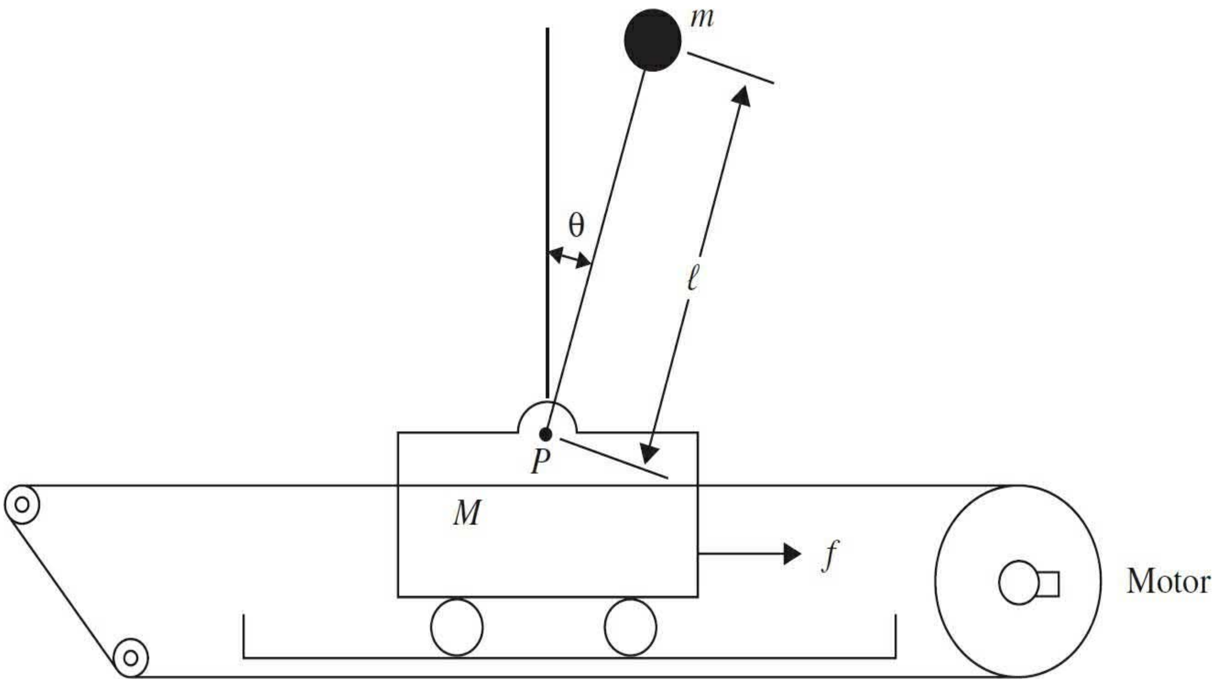


Figure 8P-64

8-65. Consider the following state-space equation of a system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -6 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \mathrm{u}$$

(a) Design a state feedback controller so that

(i) The damping ratio is $\zeta = 0.707$.

(ii) Peak time of the unit-step response is 3 s.

(b) Use MATLAB to plot the step response of the system and show how your design meets the specification in part (a).

8-66. Consider the following state-space equation of a system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -1 & -2 & -2 \\ 0 & -1 & 1 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix} \mathrm{u}$$

(a) Design a state feedback controller so that

(i) Settling time is less than 5 s (1 percent settling time).

(ii) Overshoot is less than 10 percent.

(b) Use MATLAB to verify your design.

8-67. Figure 8P-67 shows an RLC circuit.

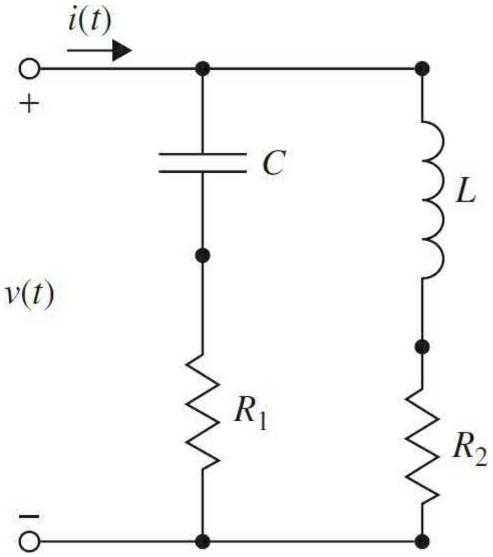


Figure 8P-67

(a) Find the state equation for the circuit when $v(t)$ is an input, $i(t)$ is an output, and capacitor voltage and the inductor current are the state variables.

(b) Find the condition that the system is controllable.

(c) Find the condition that the system is observable.

(d) Repeat parts (a), (b), and (c) when $v(t)$ is an input, the voltage of the R_2 is output, and capacitor voltage and the inductor current are the state variables.

CHAPTER 9

Root-Locus Analysis

Before starting this chapter, the reader is encouraged to refer to App. B to review the theoretical background related to complex variables.

In the preceding chapters, we have demonstrated the importance of the poles and zeros of the closed-loop transfer function of a linear control system on the dynamic performance of the system. The roots of the characteristic equation, which are the poles of the closed-loop transfer function, determine the absolute and the relative stability of linear SISO systems. Keep in mind that the transient properties of the system also depend on the zeros of the closed-loop transfer function.

Learning Outcomes

After successful completion of this chapter, you will be able to

1. Formulate or interpret a root locus to determine the effects of a parameter variation on the closed loop poles of a system.
2. Based on the properties of root loci, manually construct them.
3. Construct the root contours of a system for multi-parameter variation studies (e.g., a PD or a PI controller).
4. Use MATLAB to construct root loci and root contours.
5. Use root loci or root contours to design control systems.

An important study in linear control systems is the investigation of the trajectories of the roots of the characteristic equation—or, simply, the **root loci**—when a certain system parameter varies. In Chap. 7, several examples already illustrated the usefulness of the root loci of the characteristic equation in the study of linear control systems. The basic properties and the systematic construction of the root loci are first due to W. R. Evans[1,3]. In general, root loci may be sketched by following some simple rules and properties.

As discussed in Chap. 7, for plotting the root loci accurately, the

MATLAB root-locus tool can also be used. As design engineers, it may be sufficient for us to learn how to use computer tools to generate the root loci for design purposes. However, it is important to learn the basics of the root loci and their properties, as well as how to interpret the data provided by the root loci for analysis and design purposes. The material in this text is prepared with these objectives in mind.

The root-locus technique is not confined only to the study of control systems. In general, the method can be applied to study the behavior of roots of any algebraic equation with one or more variable parameters. The general root-locus problem can be formulated by referring to the following algebraic equation of the complex variable, say, *s*:

$$F(s) = P(s) + KQ(s) = 0 \tag{9-1}$$

where *P*(*s*) is an *n*th-order polynomial of *s*,

$$P(s) = s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 \tag{9-2}$$

and *Q*(*s*) is an *m*th-order polynomial of *s*; *n* and *m* are positive integers.

$$Q(s) = s^m + b_{m-1}s^{m-1} + \cdots + b_1 s + b_0 \tag{9-3}$$

For the present, we do not place any limitations on the relative magnitudes between *n* and *m*. *K* is a real constant that can vary from –∞ to +∞.

The coefficients $a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_m$ are considered to be real and fixed.

Root loci of multiple variable parameters can be treated by varying one parameter at a time. The resultant loci are called the **root contours**, and the subject is treated in Sec. 9-5. By replacing *s* with *z* in Eqs. (9-1) through (9-3), the root loci of the characteristic equation of a linear discrete-data system can be constructed in a similar fashion (App. H).

For the purpose of identification in this text, we define the following categories of root loci based on the values of *K*:

1. ***Root loci (RL)***. Refers to the entire root loci for –∞ < *K* < ∞.

2. ***Root contours (RC)***. Contour of roots when more than one parameter varies.

In general, for most control-system applications, the values of *K* are positive. Under unusual conditions, when a system has positive feedback or the loop gain is negative, then we have the situation that *K* is negative. Although we should be aware of this possibility, we need to place the emphasis only on positive values of *K* in developing the root-locus techniques.

9-1 BASIC PROPERTIES OF THE ROOT LOCI

Because our main interest is control systems, let us consider the closed-loop transfer function of a single-loop control system

$$\frac{Y(s)}{R(s)}=\frac{G(s)}{1+G(s)H(s)} \tag{9-4}$$

keeping in mind that the transfer function of multiple-loop SISO systems can also be expressed in a similar form. The characteristic equation of the closed-loop system is obtained by setting the denominator polynomial of $Y(s)/R(s)$ to zero. Thus, the roots of the characteristic equation must satisfy

$$1+G(s)H(s)=0 \tag{9-5}$$

Suppose that $G(s)H(s)$ contains a real variable parameter *K* as a multiplying factor, such that the rational function can be written as

$$G(s)H(s)=\frac{KQ(s)}{P(s)} \tag{9-6}$$

where $P(s)$ and $Q(s)$ are polynomials as defined in Eqs. (9-2) and (9-3), respectively. Equation (9-5) is written as

$$1+\frac{KQ(s)}{P(s)}=\frac{P(s)+KQ(s)}{P(s)}=0 \tag{9-7}$$

The numerator polynomial of Eq. (9-7) is identical to Eq. (9-1). Thus, by considering that the loop transfer function $G(s)H(s)$ can be written in the form of Eq. (9-6), we have identified the RL of a control system with the general root-locus problem.

When the variable parameter *K* does not appear as a multiplying factor of *G*(*s*)*H*(*s*), we can always condition the functions in the form of Eq. (9-1). As an illustrative example, consider that the characteristic equation of a control system is

$$s(s+1)(s+2)+s^2+(3+2K)s+5=0 \tag{9-8}$$

To express the last equation in the form of Eq. (9-7), we divide both sides of the equation by the terms that do not contain *K*, and we get

$$1+\frac{2Ks}{s(s+1)(s+2)+s^2+3s+5}=0 \tag{9-9}$$

Comparing the last equation with Eq. (9-7), we get

$$\frac{Q(s)}{P(s)}=\frac{2s}{s^3+4s^2+5s+5} \tag{9-10}$$

Now *K* is isolated as a multiplying factor to the function *Q*(*s*)/*P*(*s*).

We shall show that the RL of Eq. (9-5) can be constructed based on the properties of *Q*(*s*)/*P*(*s*). In the case where $G(s)H(s)=KQ(s)/P(s)$, the root-locus problem is another example in which the characteristics of the closed-loop system, in this case represented by the roots of the characteristic equation, are determined from the knowledge of the loop transfer function *G*(*s*)*H*(*s*).

Now we are ready to investigate the conditions under which Eq. (9-5) or Eq. (9-7) is satisfied.

Let us express *G*(*s*)*H*(*s*) as

$$G(s)H(s)=KG_1(s)H_1(s) \tag{9-11}$$

where $G_1(s)H_1(s)$ does not contain the variable parameter *K*. Then, Eq. (9-5) is written as

$$G_1(s)H_1(s)=-\frac{1}{K} \tag{9-12}$$

To satisfy Eq. (9-12), the following conditions must be satisfied simultaneously:

Condition on magnitude

$$|G_1(s)H_1(s)| = \frac{1}{|K|} \quad -\infty < K < \infty \tag{9-13}$$

Condition on angles

$$\begin{aligned} \angle G_1(s)H_1(s) &= (2i+1)\pi \quad K \geq 0 \\ &= \text{odd multiples of } \pi \text{ radians or } 180° \end{aligned} \tag{9-14}$$

$$\begin{aligned} \angle G_1(s)H_1(s) &= 2i\pi \quad K \leq 0 \\ &= \text{even multiples of } \pi \text{ radians or } 180° \end{aligned} \tag{9-15}$$

where $i = 0, \pm 1, \pm 2, \ldots$ (any integer).

In practice, the conditions stated in Eqs. (9-13) through (9-15) play different roles in the construction of the root loci.

- The conditions on angles in Eq. (9-14) or Eq. (9-15) are used to determine the trajectories of the root loci in the *s*-plane.
- Once the root loci are drawn, the values of *K* on the loci are determined by using the condition on magnitude in Eq. (9-13).

The construction of the root loci is basically a graphical problem, although some of the properties are derived analytically. The graphical construction of the RL is based on the knowledge of the poles and zeros of the function $G(s)H(s)$. In other words, $G(s)H(s)$ must first be written as

$$G(s)H(s) = KG_1(s)H_1(s) = \frac{K(s+z_1)(s+z_2)\cdots(s+z_m)}{(s+p_1)(s+p_2)\cdots(s+p_n)} \tag{9-16}$$

where the zeros and poles of $G(s)H(s)$ are real or in complex-conjugate pairs.

Applying the conditions in Eqs. (9-13), (9-14), and (9-15) to Eq. (9-16), we have

$$\left|G_1(s)H_1(s)\right| = \frac{\prod_{i=1}^{m}\left|s+z_i\right|}{\prod_{k=1}^{n}\left|s+p_k\right|} = \frac{1}{|K|} - \infty < K < \infty \tag{9-17}$$

For $0 \le K < \infty$:

$$\angle G_1(s)H_1(s) = \sum_{k=1}^{m}\angle(s+z_k) - \sum_{j=1}^{n}\angle(s+p_j) = (2i+1)\times 180^\circ \tag{9-18}$$

For $-\infty < K \le 0$:

$$\angle G_1(s)H_1(s) = \sum_{k=1}^{m}\angle(s+z_k) - \sum_{j=1}^{n}\angle(s+p_j) = 2i\times 180^\circ \tag{9-19}$$

where $i = 0, \pm 1, \pm 2, \ldots$

The graphical interpretation of Eq. (9-18) is that any point s_1 on the RL that corresponds to a positive value of K must satisfy the following condition:

The difference between the sums of the angles of the vectors drawn from the zeros and those from the poles of *G(s)H(s) to* s $_1$ is an odd multiple of 180 degrees.

For negative values of K, any point s_1 on the RL must satisfy the following condition:

The difference between the sums of the angles of the vectors drawn from the zeros and those from the poles of *G(s)H(s)* to s_1 is an even multiple of 180 degrees, including zero degrees.

Once the root loci are constructed, the values of K along the loci can be determined by writing Eq. (9-17) as

$$|K| = \frac{\prod_{j=1}^{n}\left|s+p_j\right|}{\prod_{i=1}^{m}\left|s+z_i\right|} \tag{9-20}$$

The value of K at any point s_1 on the RL is obtained from Eq. (9-20) by substituting the value of s_1 into the equation. Graphically, the numerator of Eq. (9-20) represents the product of the lengths of the vectors drawn from the poles of $G(s)H(s)$ to s_1, and the denominator represents the product of lengths of the vectors drawn from the zeros of $G(s)H(s)$ to s_1.

To illustrate the use of Eqs. (9-18) to (9-20) for the construction of the root loci, let us consider the function

$$G(s)H(s)=\frac{K(s+z_1)}{s(s+p_2)(s+p_3)} \tag{9-21}$$

The location of the poles and zero of $G(s)H(s)$ are arbitrarily assigned, as shown in Fig. 9-1. Let us select an arbitrary trial point s_1 in the s-plane and draw vectors directing from the poles and zeros of $G(s)H(s)$ to the point. If s_1 is indeed a point on the RL for positive K, it must satisfy Eq. (9-18); that is, the angles of the vectors shown in Fig. 9-1 must satisfy

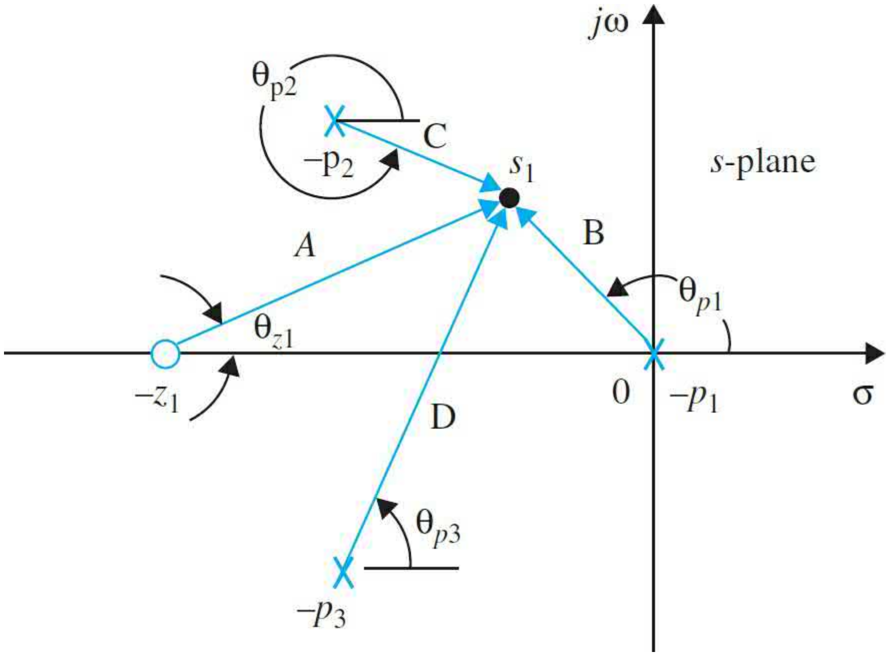


Figure 9-1 Pole-zero configuration of $G(s)H(s)=K(s+z_1)/[s(s+p_2)\times(s+p_3)]$.

$$\angle(s_1+z_1)-\angle s_1-\angle(s_1+p_2)-\angle(s_1+p_3)=\theta_{z1}-\theta_{p1}-\theta_{p2}-\theta_{p3}=(2i+1)\times 180^\circ \tag{9-22}$$

where i = 0, ±1, ±2, … As shown in Fig. 9-1, the angles of the vectors are measured with the positive real axis as reference. Similarly, if s_1 is a point on the RL for negative values of K, it must satisfy Eq. (9-19), that is,

$$\angle(s_1+z_1)-\angle s_1-\angle(s_1+p_2)-\angle(s_1+p_3)=\theta_{z1}-\theta_{p1}-\theta_{p2}-\theta_{p3}=2i\times 180^\circ \tag{9-23}$$

where $i = 0, \pm1, \pm2, \ldots$

If s_1 is found to satisfy either Eq. (9-22) or Eq. (9-23), Eq. (9-20) is used to find the magnitude of K at the point. As shown in Fig. 9-1, the lengths of the vectors are represented by A, B, C, and D. The magnitude of K is

$$|K|=\frac{|s_1||s_1+p_2||s_1+p_3|}{|s_1+z_1|}=\frac{BCD}{A} \tag{9-24}$$

The sign of K depends on whether s_1 satisfies Eq. (9-22) ($K \geq 0$) or Eq. (9-23) ($K \leq 0$). Thus, given the function $G(s)H(s)$ with K as a multiplying factor and the poles and zeros are known, the construction of the RL of the zeros of $1 + G(s)H(s)$ involves the following two steps:

1. A search for all the s_1 points in the s-plane that satisfy Eq. (9-18) for positive K. If the RL for negative values of K are desired, then Eq. (9-19) must be satisfied.

2. Use Eq. (9-20) to find the magnitude of K on the RL.

We have established the basic conditions on the construction of the root-locus diagram. However, if we were to use the trial-and-error method just described, the search for all the root-locus points in the s-plane that satisfy Eq. (9-18) or Eq. (9-19) and Eq. (9-20) would be a very tedious task.

With MATLAB tools, as introduced in Chap. 7, the trial-and-error method have long become obsolete. Nevertheless, even with a high-speed computer and an effective root-locus program, you should still have an understanding of the properties of the root loci to be able to manually sketch the root loci of simple and moderately complex systems, if necessary, and interpret the computer results correctly, when applying the root loci for analysis and design of control systems.

9-2 PROPERTIES OF THE ROOT LOCI

The following properties of the root loci are useful for the purpose of constructing the root loci manually and for the understanding of the root loci. The properties are developed based on the relation between the poles and zeros of $G(s)H(s)$ and the zeros of $1 + G(s)H(s)$, which are the roots of the characteristic equation.

9-2-1 $K = 0$ and $K = \pm\infty$ Points

***The K* = 0 points on the root loci are at the poles of *G*(*s*)*H*(*s*).**

The *K* = ±∞ points on the root loci are at the zeros of *G*(*s*)*H*(*s*).

The poles and zeros referred to here, include those at infinity, if any. The reason for these properties are seen from the condition of the root loci given by Eq. (9-12), which is

$$G_1(s)H_1(s) = -\frac{1}{K} \tag{9-25}$$

As the magnitude of K approaches zero, $G_1(s)H_1(s)$ approaches infinity, so s must approach the poles of $G_1(s)H_1(s)$ or of $G(s)H(s)$. Similarly, as the magnitude of K approaches infinity, s must approach the zeros of $G(s)H(s)$.

EXAMPLE 9-2-1 Consider the equation

$$s(s+2)(s+3)+K(s+1)=0 \tag{9-26}$$

When $K = 0$, the three roots of the equation are at $s = 0, –2$, and $–3$. When the magnitude of K is infinite, the three roots of the equation are at $s = –1, \infty$, and ∞. It is useful to consider that infinity in the s-plane is a point concept. We can visualize that the finite s-plane is only a small portion of a sphere with an infinite radius. Then, infinity in the s-plane is a point on the opposite side of the sphere that we face.

Dividing both sides of Eq. (9-26) by the terms that do not contain K, we get

$$1+G(s)H(s)=1+\frac{K(s+1)}{s(s+2)(s+3)}=0 \tag{9-27}$$

which gives

$$G(s)H(s)=\frac{K(s+1)}{s(s+2)(s+3)} \tag{9-28}$$

Thus, the three roots of Eq. (9-26) when $K = 0$ are the same as the poles of the function $G(s)H(s)$. The three roots of Eq. (9-26) when $K = \pm\infty$ are at the three zeros of $G(s)H(s)$, including those at infinity. In this case, one finite zero is at $s = -1$, but there are two zeros at infinity. The three points on the root loci at which $K = 0$ and those at which $K = \pm\infty$ are shown in Fig. 9-2. △

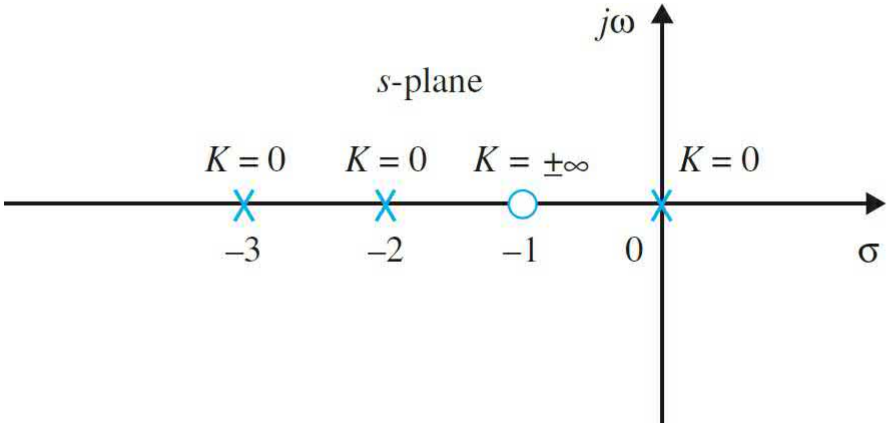


Figure 9-2 Points at which $K = 0$ and $K = \pm\infty$ on the *RL of s(s+2)(s+3)+K(s+1)=0.*

9-2-2 Number of Branches on the Root Loci

It is important to keep track of the total number of branches of the root loci.

A branch of the RL is the locus of one root when K varies between –∞ and ∞. The following property of the RL results, since the number of branches of the RL must equal the number of roots of the equation.

The number of branches of the RL of Eq. (9-1) or Eq. (9-5) is equal to the order of the polynomial.

For example, the number of branches of the root loci of Eq. (9-26) when *K* varies from –∞ to ∞ is three, since the equation has three roots.

Keeping track of the individual branches and the total number of branches of the root-locus diagram is important in making certain that the plot is done correctly. This is particularly true when the root-locus plot is done by a computer because unless each root-locus branch is coded by a different color, it is up to the user to make the distinctions.

EXAMPLE 9-2-2 The number of branches of the root loci of

$$s(s+2)(s+3)+K(s+1)=0 \tag{9-29}$$

is three, since the equation is of the third order. In other words, the equation has three roots, and thus, there should be three root loci. ▲

9-2-3 Symmetry of the RL

The RL are symmetrical with respect to the real axis of the *s*-plane. In general, the RL are symmetrical with respect to the axes of symmetry of the pole-zero configuration of *G*(*s*)*H*(*s*).

It is important to pay attention to the symmetry of the root loci.

The reason behind this property is because for a polynomial with real coefficients the roots must be real or in complex-conjugate pairs. In general, if the poles and zeros of *G*(*s*)*H*(*s*) are symmetrical to an axis in addition to the real axis in the *s*-plane, we can regard this axis of symmetry as if it were the real axis of a new complex plane obtained through a linear transformation.

EXAMPLE 9-2-3 Consider the equation

$$s(s+1)(s+2)+K=0 \tag{9-30}$$

Dividing both sides of the equation by the terms that do not contain *K*, we get

$$G(s)H(s)=\frac{K}{s(s+1)(s+2)} \tag{9-31}$$

The root loci of Eq. (9-30) are shown in Fig. 9-3 for $K = -\infty$ to $K = \infty$. Since the pole-zero configuration of $G(s)H(s)$ is symmetrical with respect to the real axis as well as the $s = -1$ axis, the root-locus plot is symmetrical to the two axes.

As a review of all the properties of the root loci presented thus far, we conduct the following exercise with regard to the root loci in Fig. 9-3.

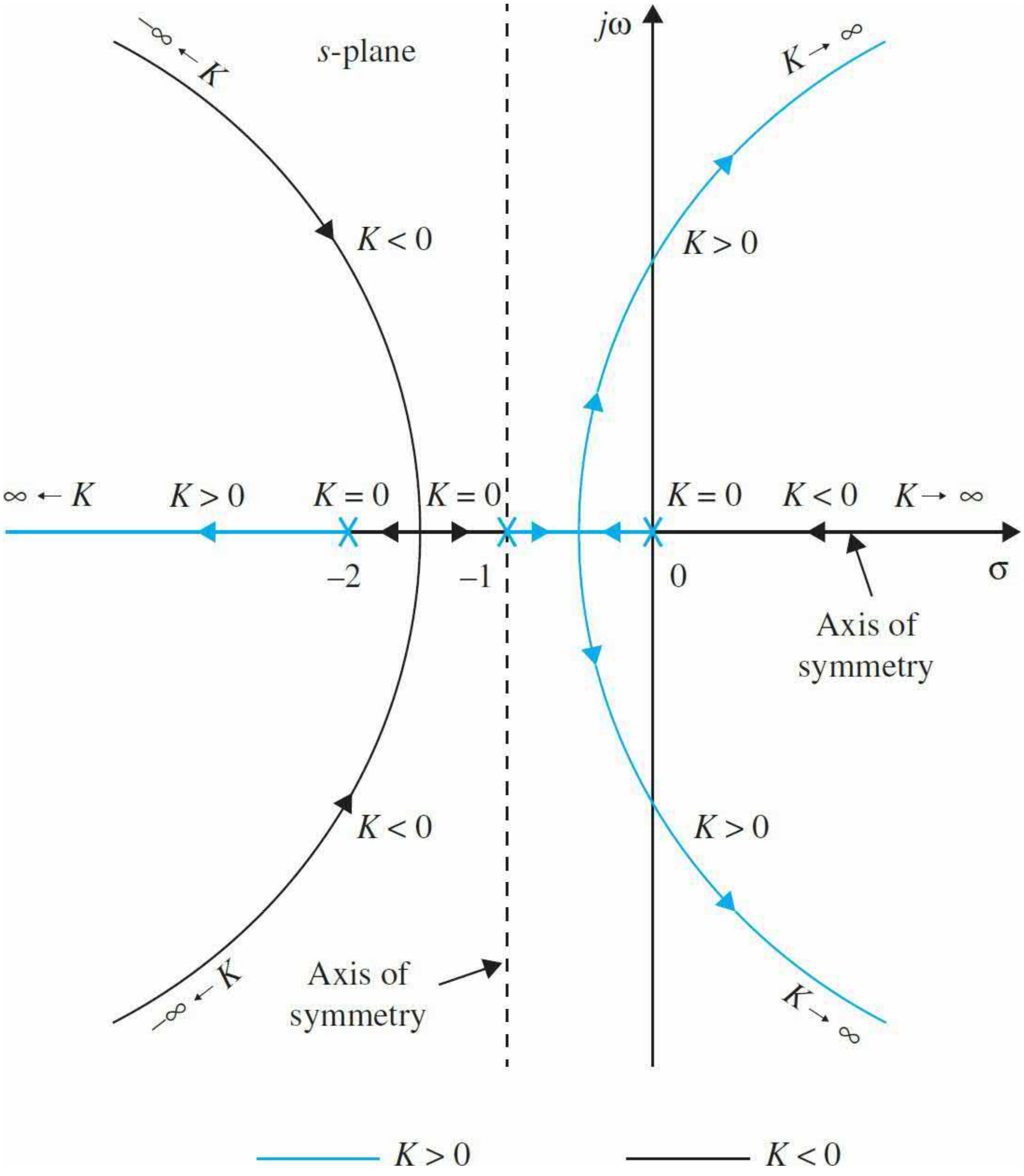


Figure 9-3 Root loci of $s(s + 2)(s + 3) + K(s + 1) = 0$, showing the properties of symmetry.

The points at which $K = 0$ are at the poles of $G(s)H(s)$, $s = 0, -1$, and -2. The function $G(s)H(s)$ has three zeros at $s = \infty$ at which $K = \pm\infty$. The reader should try to trace out the three separate branches of the root loci by starting from one of the $K = -\infty$ points, through the $K = 0$ point on the same branch, and ending at $K = \infty$ at $s = \infty$. ▲

EXAMPLE 9-2-4 When the pole-zero configuration of $G(s)H(s)$ is symmetrical with respect to a point in the s-plane, the root loci will also be symmetrical to that point. This is illustrated by the root-locus plot of

$$s(s+1)(s+1+j)(s+1-j)+K=0 \tag{9-32}$$

shown in Fig. 9-4. ▲

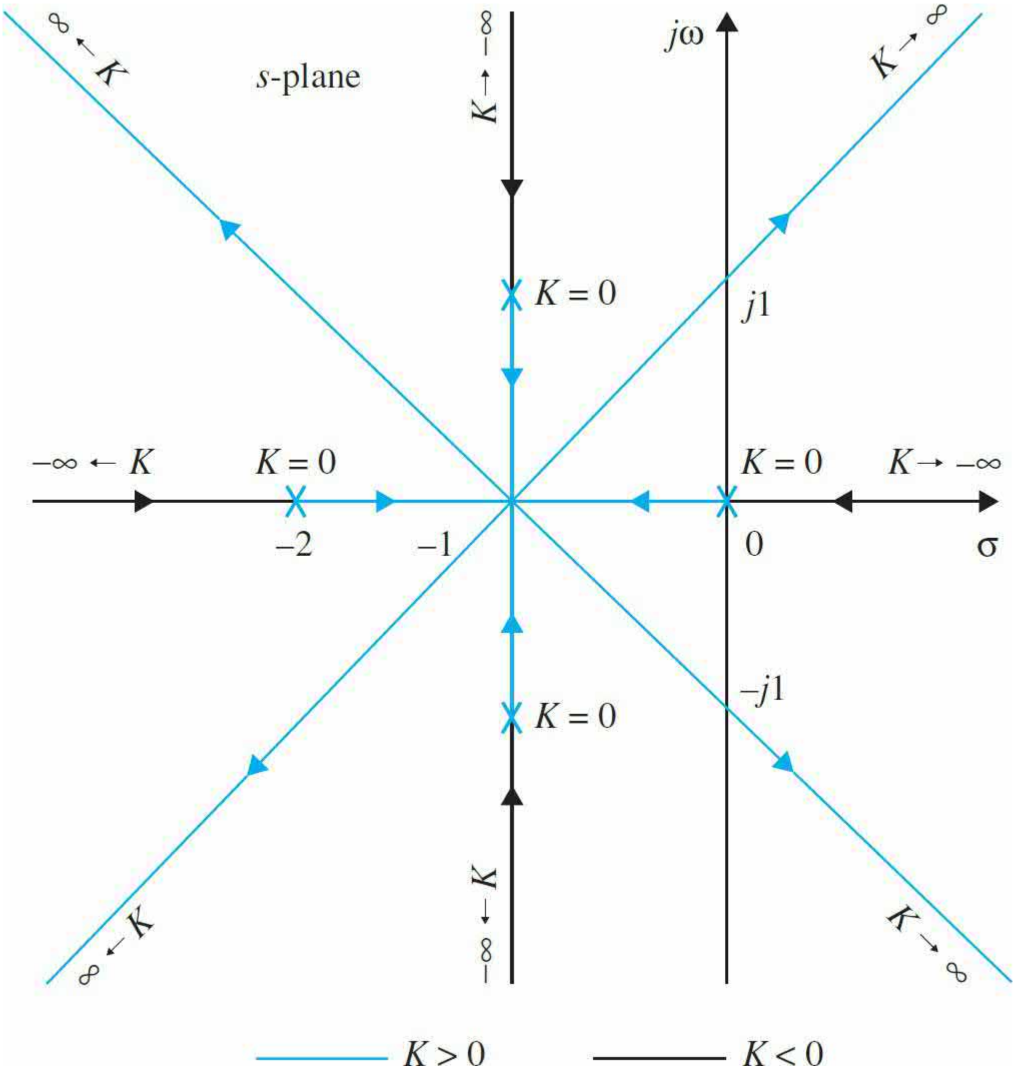


Figure 9-4 Root loci of $s(s + 2)(s^2 + 2s + 2) + K = 0$, showing the properties of symmetry.

9-2-4 Angles of Asymptotes of the RL: Behavior of the RL at $|s| = \infty$

Asymptotes of root loci refers to behavior of root loci at $|s| = \infty$.

When n, the order of $P(s)$, is not equal to m, the order of $Q(s)$, some of the loci will approach infinity in the s-plane. The properties of the RL near infinity in the s-plane are described by the **asymptotes** of the loci when $|s|$

→∞ In general when $n \neq m$, there will be $2|n-m|$ asymptotes that describe the behavior of the RL at $|s|=\infty$ The angles of the asymptotes and their intersect with the real axis of the *s*-plane are described as follows.

For large values of *s*, the RL for $K \geq 0$ are asymptotic to asymptotes with angles given by

$$\theta_i = \frac{(2i+1)}{|n-m|} \times 180^\circ \quad n \neq m \tag{9-33}$$

where $i=0,1,2,\ldots,|n-m|-1$ *n and m* are the number of finite poles and zeros of $G(s)H(s)$, respectively.

For $K \leq 0$ (RL), the angles of the asymptotes are

$$\theta_i = \frac{2i}{|n-m|} \times 180^\circ \quad n \neq m \tag{9-34}$$

where $i=0,1,2,\ldots,|n-m|-1$.

9-2-5 Intersect of the Asymptotes (Centroid)

The intersect of the $2|n-m|$ asymptotes of the RL lies on the real axis of the *s*-plane, at

$$\sigma_1 = \frac{\sum \text{finite poles of } G(s)H(s) - \sum \text{finite zeros of } G(s)H(s)}{n-m} \tag{9-35}$$

where *n* is the number of finite poles and *m* is the number of finite zeros of $G(s)H(s)$, respectively. The intersect of the asymptotes σ_1 represents the center of gravity of the root loci and is always a real number, or

$$\sigma_1 = \frac{\sum \text{real parts of poles of } G(s)H(s) - \sum \text{real parts of zeros of } G(s)H(s)}{n-m} \tag{9-36}$$

EXAMPLE 9-2-5 The root loci and their asymptotes for Eq. (9-26) for $-\infty \leq K \leq \infty$ are shown in Fig. 9-5.

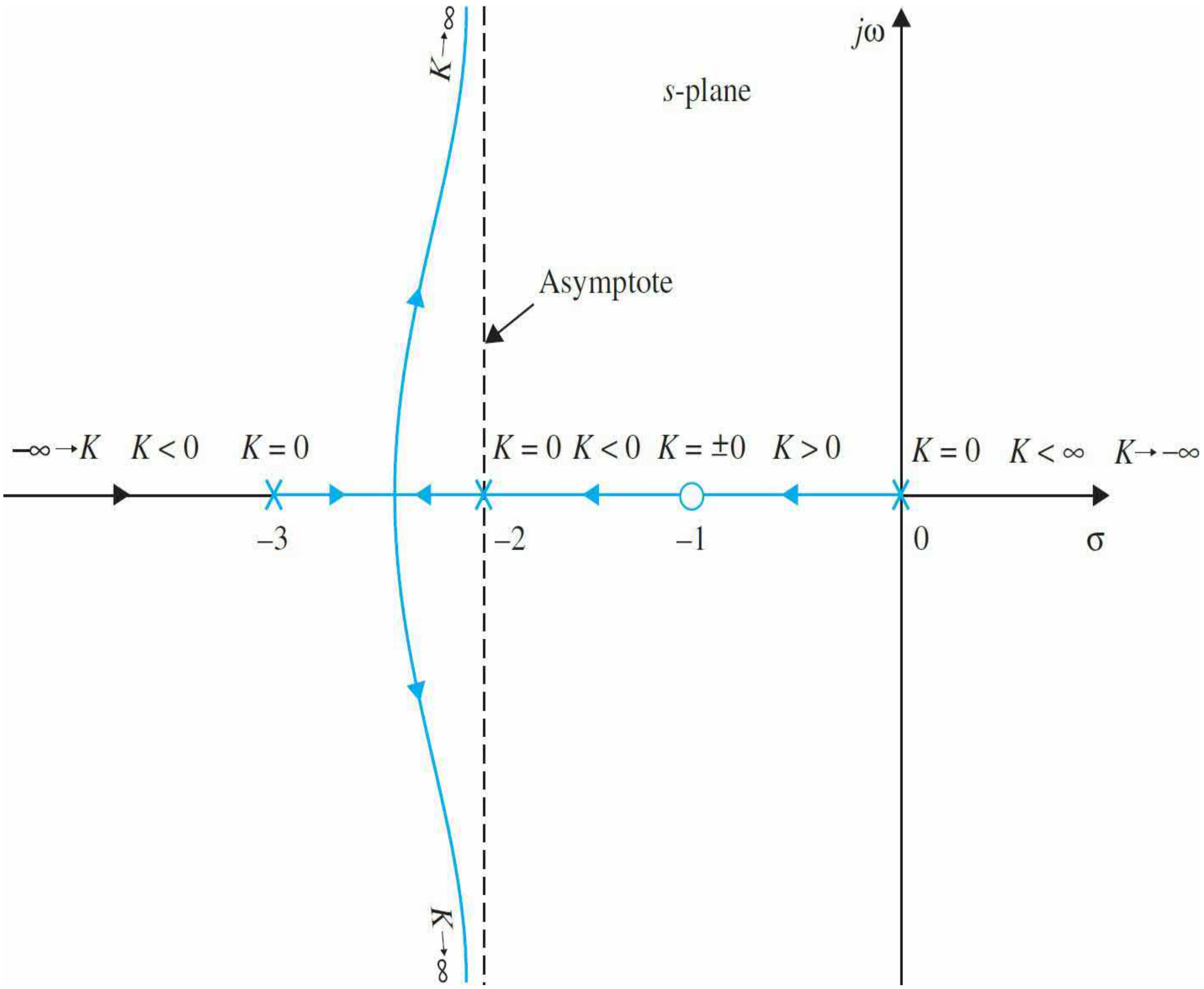
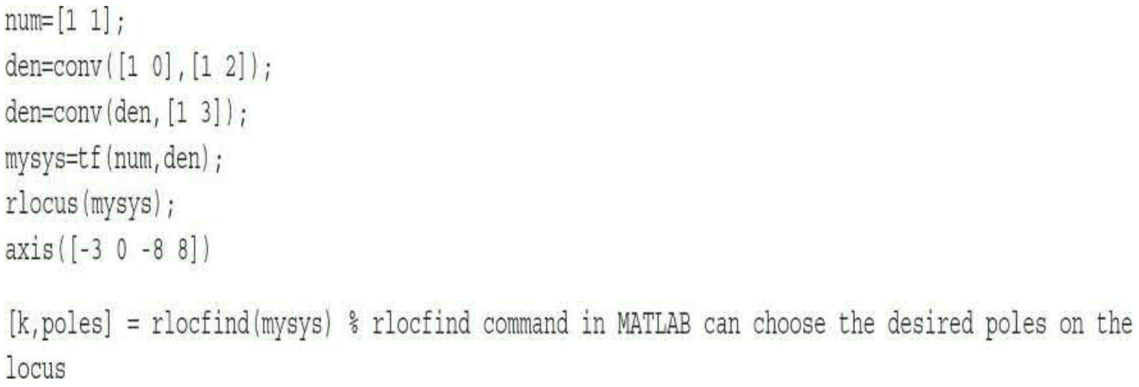


Figure 9-5 Root loci and asymptotes of $s(s + 2)(s + 3) + K(s + 1) = 0$ for $-\infty \le K \le \infty$.

Toolbox 9-2-1 MATLAB code for root loci in Fig. 9-5.



```
num=[1 1];
den=conv([1 0],[1 2]);
den=conv(den,[1 3]);
mysys=tf(num,den);
rlocus(mysys);
axis([-3 0 -8 8])

[k,poles] = rlocfind(mysys) % rlocfind command in MATLAB can choose the desired poles on the
locus
```

EXAMPLE 9-2-6 Consider the transfer function

$$G(s)H(s) = \frac{K(s+1)}{s(s+4)(s^2+2s+2)} \tag{9-37}$$

which corresponds to the characteristic equation

$$s(s+4)(s^2+2s+2)+K(s+1)=0 \tag{9-38}$$

The pole-zero configuration of $G(s)H(s)$ is shown in Fig. 9-6. From the six properties of the root loci discussed so far, the following information concerning the root loci of Eq. (9-38) when K varies from −∞ to ∞ is obtained:

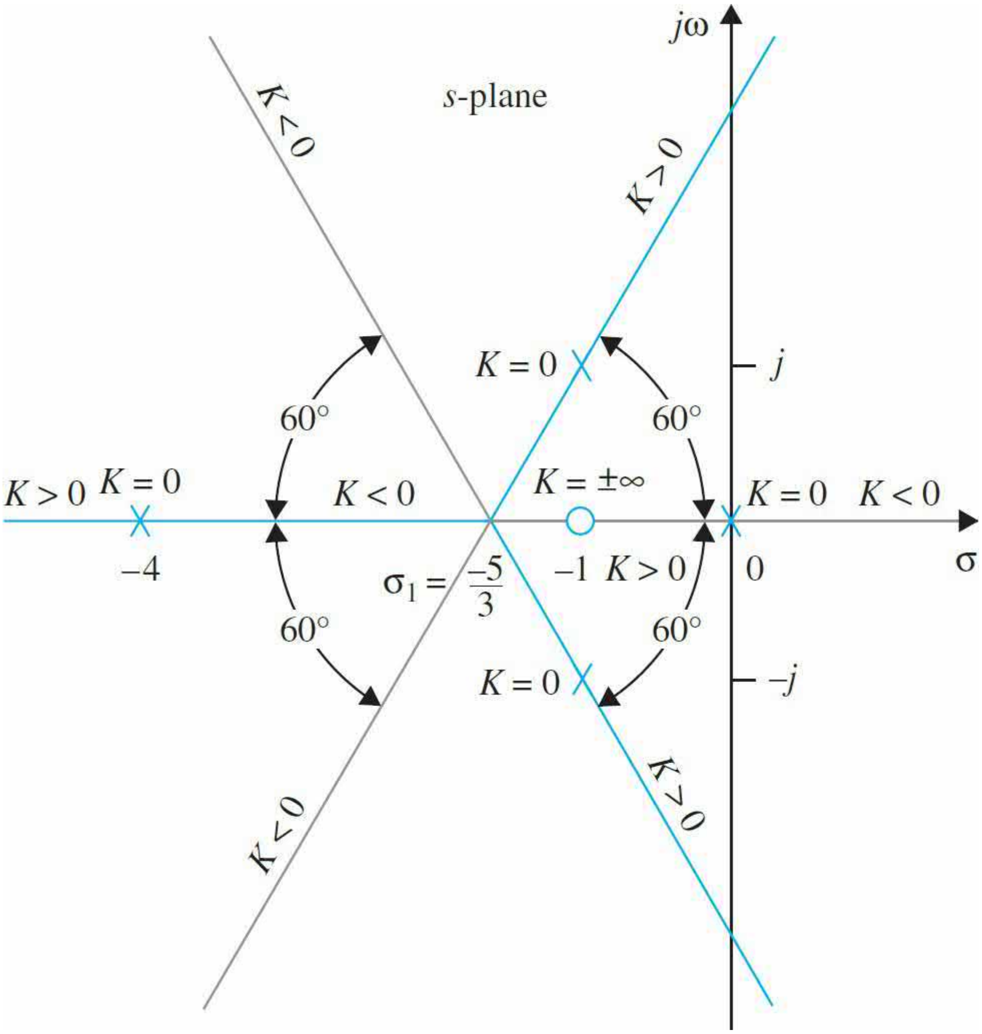


Figure 9-6 Asymptotes of the root loci of $s(s + 4)(s^2 + 2s + 2) + K(s + 1) = 0$.

1. $K = 0$: The points at which $K = 0$ on the root loci are at the poles of $G(s)H(s)$: $s = 0, -4, -1 + j$, and $-1 - j$.
2. $K = \pm\infty$: The points at which $K = \pm\infty$ on the root loci are at the zeros of $G(s)H(s$: $s = -1, \infty, \infty$, and ∞.
3. There are four root loci branches, since Eqs. (9-37) and (9-38) are of the fourth order.
4. The root loci are symmetrical to the real axis.
5. Since the number of finite poles of $G(s)H(s)$ exceeds the number of finite zeros of $G(s)H(s)$ by three ($n - m = 4 - 1 = 3$), when $K = \pm\infty$, three root loci approach $s = \infty$.

   The angles of the asymptotes of the RL ($K \geq 0$) are given by Eq. (9-33):

$$j=0: \quad \theta_0 = \frac{180^\circ}{3} = 60^\circ$$

$$j=1: \quad \theta_1 = \frac{540^\circ}{3} = 180^\circ$$

$$j=2: \quad \theta_2 = \frac{900^\circ}{3} = 300^\circ$$

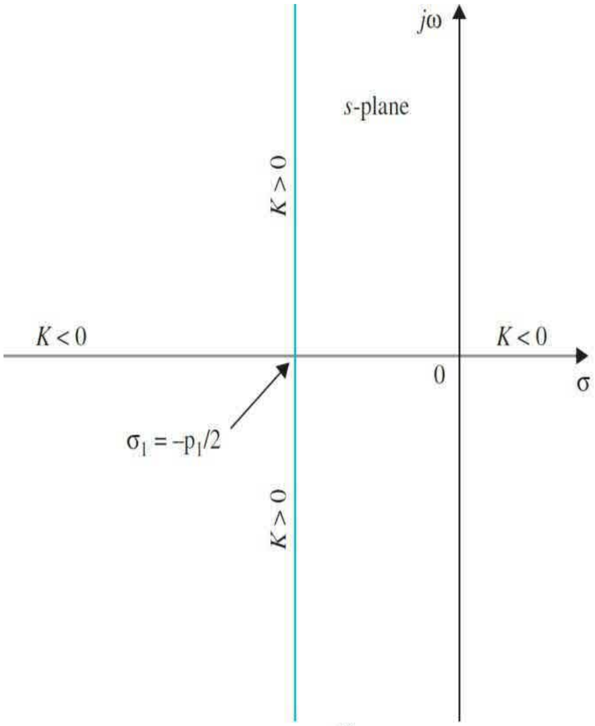
The angles of the asymptotes of the root loci for $K \leq 0$ are given by Eq. (9-34), and are calculated to be 0°, 120°, and 240°.

6. The intersection of the asymptotes is given by Eq. (9-36):

$$\sigma_1 = \frac{(-4-1-1)-(-1)}{4-1} = -\frac{5}{3} \qquad (9\text{-}39)$$

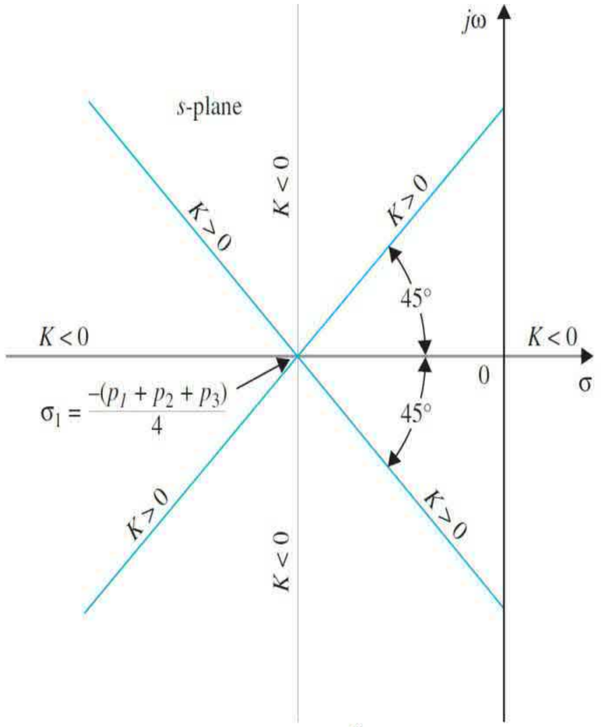
The asymptotes of the root loci are shown in Fig. 9-6. ▲

EXAMPLE 9-2-7 The asymptotes of the root loci of several equations are shown in Fig. 9-7. ▲



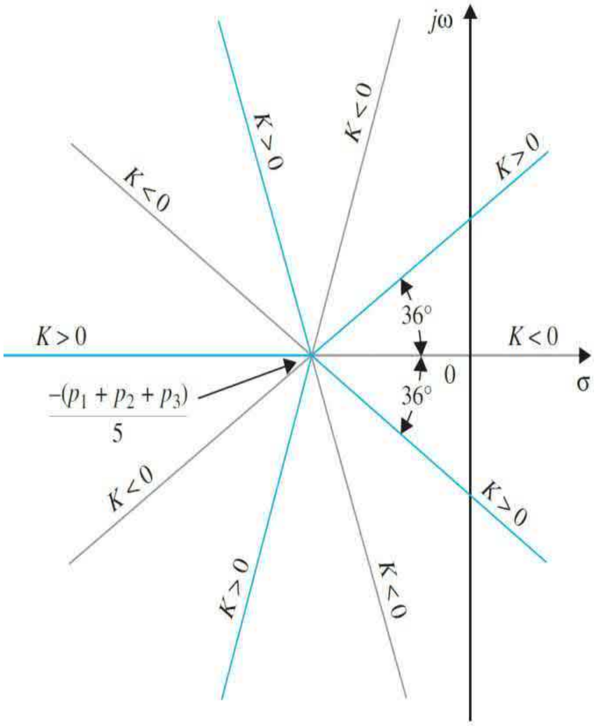
$$G(s)H(s) = \frac{K}{s(s+p_1)}$$

— $K > 0$ — $K < 0$



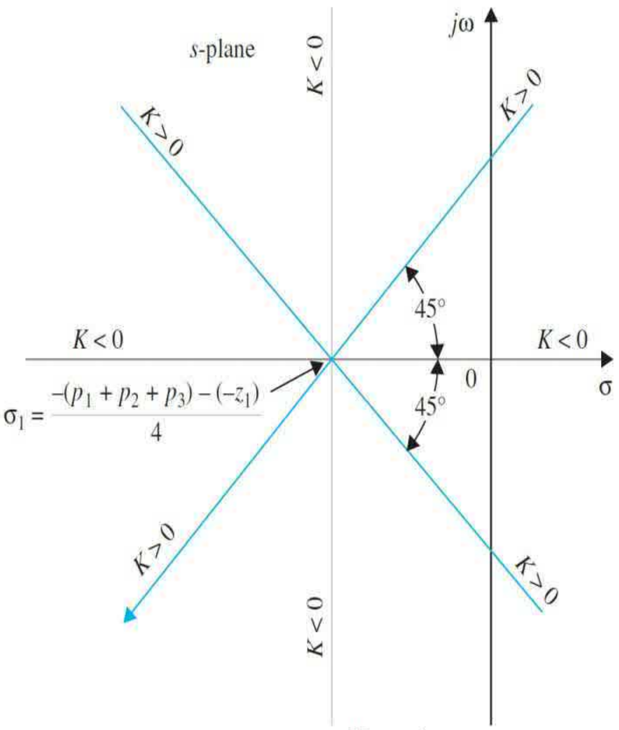
$$G(s)H(s) = \frac{K}{s(s+p_1)(s+p_2)(s+p_3)}$$

— $K > 0$ — $K < 0$



$$G(s)H(s) = \frac{K}{s^2(s+p_1)(s+p_2)(s+p_3)}$$

— $K > 0$ — $K < 0$



$$G(s)H(s) = \frac{K(s+z_1)}{s^2(s+p_1)(s+p_2)(s+p_3)}$$

— $K > 0$ — $K < 0$

Figure 9-7 Examples of the asymptotes of the root loci.

9-2-6 Root Loci on the Real Axis

The entire real axis of the *s*-plane is occupied by the RL for all values *K*. On a given section of the real axis, RL for $K \geq 0$ are found in the section only if the total number of poles and zeros of $G(s)H(s)$ to the right of the section is odd. Note that the remaining sections of the real axis are occupied by the RL for $K \leq 0$. Complex poles and zeros of $G(s)H(s)$ do not affect the type of RL found on the real axis.

The entire real axis of the *s*-plane is occupied by root loci.

These properties are arrived at based on the following observations:

1. At any point s_1 on the real axis, the angles of the vectors drawn from the complex-conjugate poles and zeros of $G(s)H(s)$ add up to zero. Thus, only the real zeros and poles of $G(s)H(s)$ contribute to the angular relations in Eqs. (9-18) and (9-19).
2. Only the real poles and zeros of $G(s)H(s)$ that lie to the right of the point s_1 contribute to Eqs. (9-18) and (9-19) because real poles and zeros that lie to the left of the point contribute nothing.
3. Each real pole of $G(s)H(s)$ to the right of s_1 contributes −180 degrees, and each real zero of $G(s)H(s)$ to the right of s_1 contributes +180 degrees to Eqs. (9-18) and (9-19).

The last observation shows that for s_1 to be a point on the root locus, there must be an **odd** number of poles and zeros of $G(s)H(s)$ to the right of the point. For s_1 to be a point on the branch of the root loci for $K \leq 0$, the total number of poles and zeros of $G(s)H(s)$ to the right of the point must be **even**. The following example illustrates the determination of the properties of the root loci on the real axis of the *s*-plane.

EXAMPLE 9-2-8 The root loci on the real axis for two pole-zero configurations of $G(s)H(s)$ are shown in Fig. 9-8.

Notice that the entire real axis is occupied by the root loci for all values of K. ▲

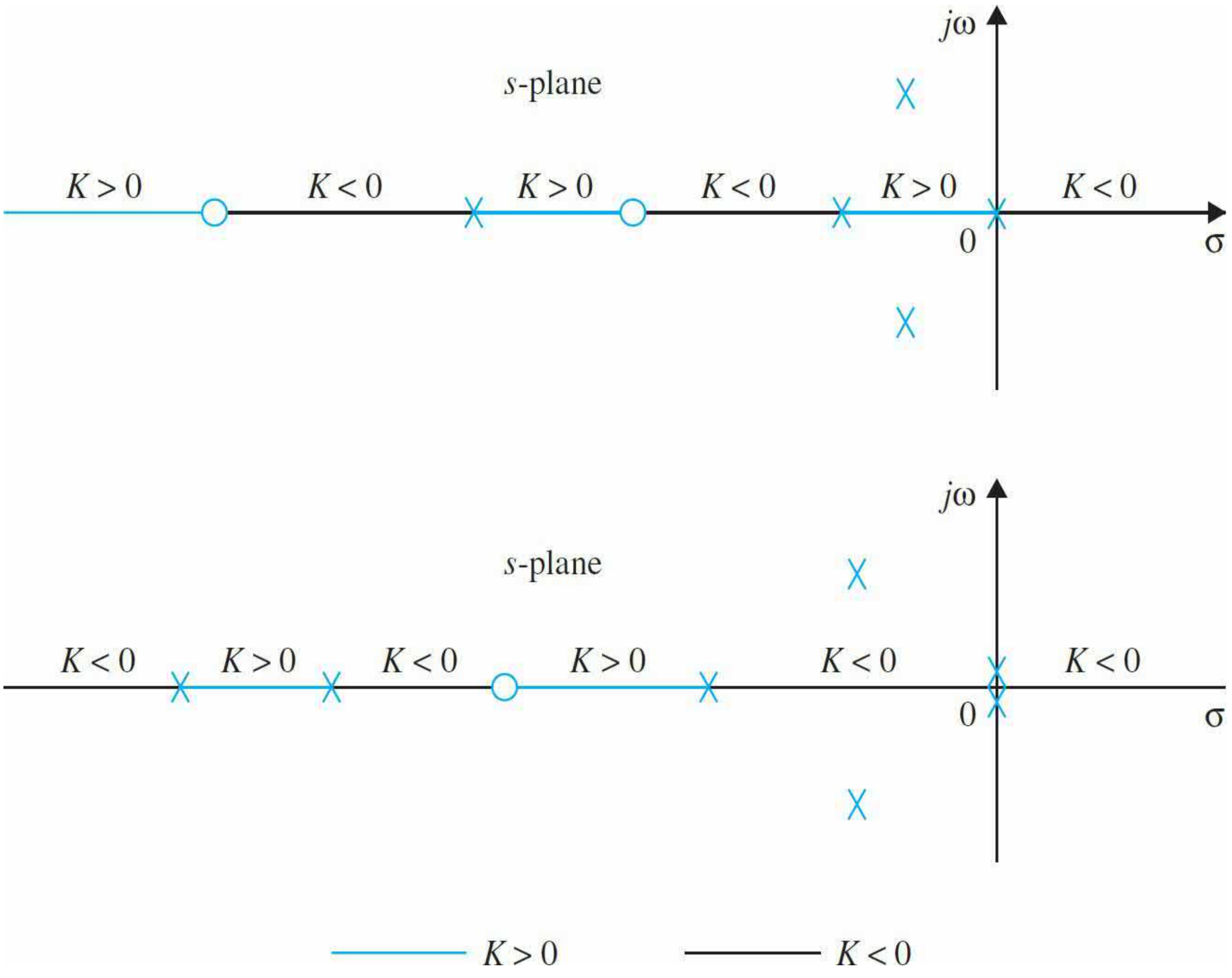


Figure 9-8 Properties of root loci on the real axis.

9-2-7 Angles of Departure and Angles of Arrival of the RL

The angle of departure or arrival of a root locus at a pole or zero, respectively, of $G(s)H(s)$ denotes the angle of the tangent to the locus near the point.

The angles of departure and arrival are determined using Eq. (9-18) for root loci for positive K and Eq. (9-19) for root loci for negative K. The details are illustrated by the following example.

EXAMPLE 9-2-9 For the root-locus diagram shown in Fig. 9-9, the root locus near the pole $s = -1 + j$ may be more accurately sketched by knowing the angle at which the root locus leaves the pole. As shown in Fig. 9-10, the angle of departure of the root locus at $s = -1 + j$ is represented by θ_2, measured with respect to the real axis. Let us assign s_1 to be a point on the RL leaving the pole at $-1 + j$ and is very close to the pole. Then, s_1 must satisfy Eq. (9-18). Thus,

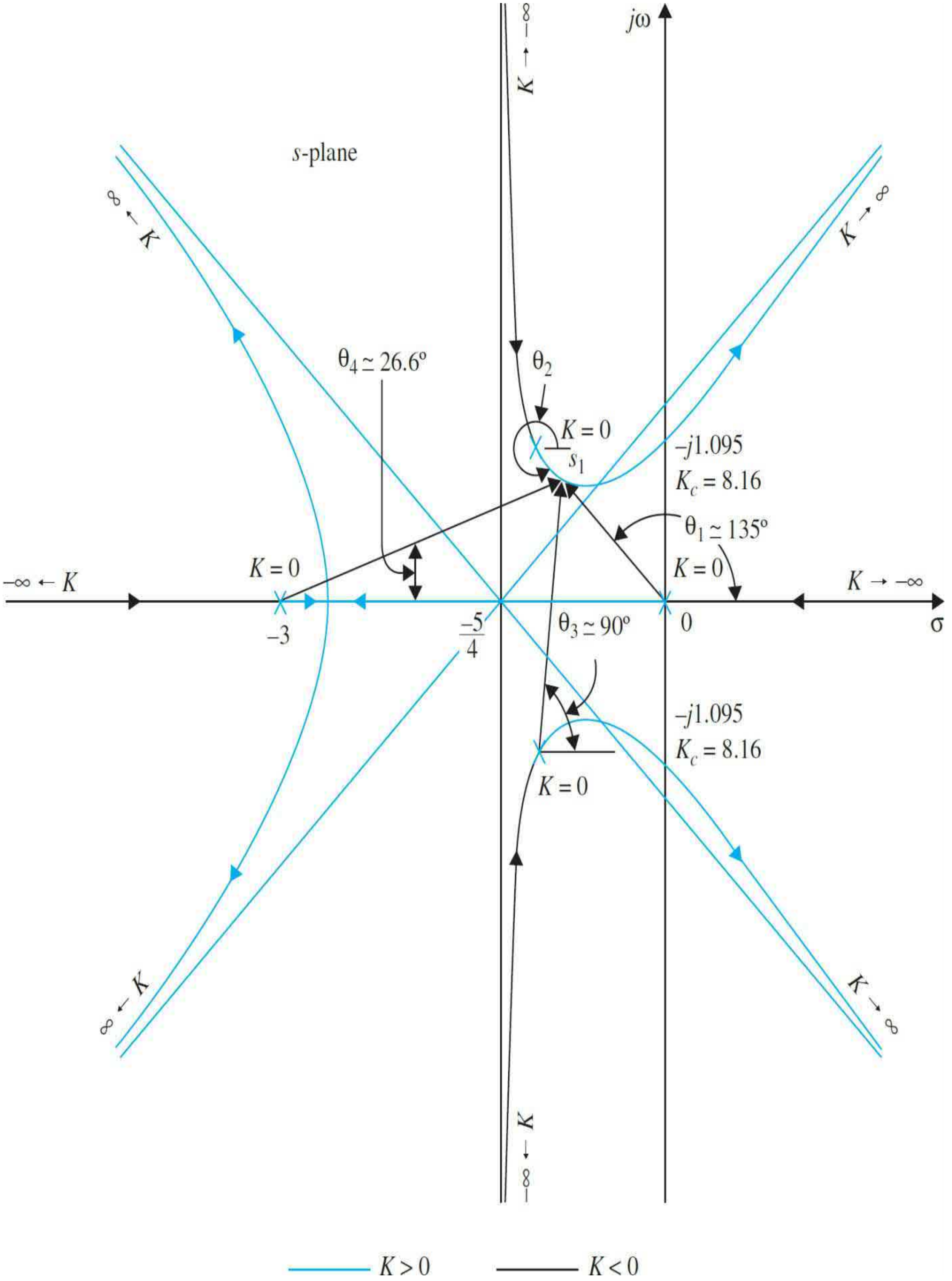

jω
s-plane
K → −∞
∞ ← K
K → ∞
θ₄ ≃ 26.6°
θ₂
K = 0
s₁
–j1.095
K_c = 8.16
θ₁ ≃ 135°
−∞ ← K
K = 0
K = 0
K → −∞
−3
−5/4
0
σ
θ₃ ≃ 90°
–j1.095
K_c = 8.16
K = 0
∞ ← K
K → ∞
−∞ ← K
K > 0
K < 0

Figure 9-9 Root loci of $s(s + 3)(s^2 + 2s + 2) + K = 0$ to illustrate the angles of departure or arrival.

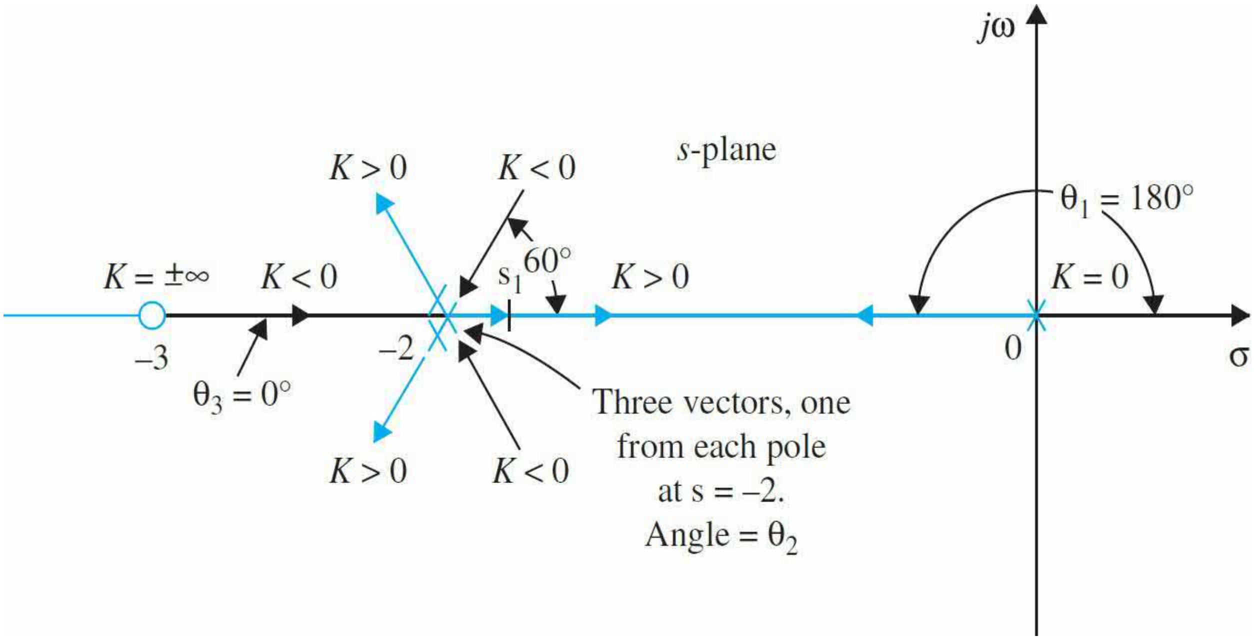


Figure 9-10 Angles of departure and arrival at a third-order pole.

$$\angle G(s_1)H(s_1) = -(\theta_1 + \theta_2 + \theta_3 + \theta_4) = (2i+1)180^\circ \tag{9-40}$$

where i is any integer. Since s_1 is assumed to be very close to the pole at −1 + j, the angles of the vectors drawn from the other three poles are approximated by considering that s_1 is at −1 + j. From Fig. F-7, Eq. (F-16) is written as

$$-(135^\circ + \theta_2 + 90^\circ + 26.6^\circ) = (2i+1)180^\circ \tag{9-41}$$

where θ_2 is the only unknown angle. In this case, we can set i to be −1, and the result for θ_2 is –71.6°.

When the angle of departure or arrival of a root locus for positive K at a simple pole or zero of $G(s)H(s)$ is determined, the angle of arrival or departure of the root locus for negative K at the same point differs from this angle by 180°, and Eq. (9-19) is now used. Figure 9-9 shows that the angle of arrival of the root locus for negative *K* at −1 + *j* is 108.4°, which is 180° − 71.6°. Similarly, for the root-locus diagram in Fig. 9-10, we can show that the root locus for negative *K* arrives at the pole *s* = −3 with an angle of 180°, and the root locus for positive *K* leaves the same pole at 0°. For the pole at *s* = 0, the angle of arrival of the negative-*K* root locus is 180°, whereas the angle of departure of the positive-*K* root locus is 180°. These angles are also determined from the knowledge of the type of root loci on sections of the real axis separated by the poles and zeros of *G*(*s*)*H*(*s*). Since the total angles of the vectors drawn from complex-conjugate poles and zeros to any point on the real axis add up to be zero, the angles of arrival and departure of root loci on the real axis are not affected by the complex poles and zeros of *G*(*s*)*H*(*s*). ▲

EXAMPLE 9-2-10 In this example, we examine the angles of departure and arrival of the root loci at a multiple-order pole or zero of *G*(*s*)*H*(*s*). Consider that a *G*(*s*)*H*(*s*) has a multiple-order (third-order) pole on the real axis, as shown in Fig. 9-10. Only the real poles and zeros of *G*(*s*)*H*(*s*) are shown, since the complex ones do not affect the type or the angles of arrival and departure of the root loci on the real axis. For the third-order pole at *s* = −2, there are three positive-*K* loci leaving and three negative-*K* loci arriving at the point. To find the angles of departure of the positive-*K* root loci, we assign a point s_1 on one of the loci near *s* = −2, and apply Eq. (9-18). The result is

$$-\theta_1 - 3\theta_2 + \theta_3 = (2i+1)180^\circ \tag{9-42}$$

where θ_1 and θ_3 denote the angles of the vectors drawn from the pole at 0 and the zero at −3, respectively, to s_1. The angle θ_2 is multiplied by 3, since there are three poles at *s* = −2, so that there are three vectors drawn from −2 to s_1. Setting *i* to zero in Eq. (9-42), and since θ_1 = 180°, θ_3=

0°, we have θ_2 = 0°, which is the angle of departure of the positive-*K* root loci that lies between *s* = 0 and *s* = −2. For the angles of departure of the other two positive-*K* loci, we set *i* = 1 and *i* = 2 successively in Eq. (F-18), and we have θ_2 = 120° and −120°. Similarly, for the three negative-*K* root loci that arrive at *s* = −2, Eq. (9-19) is used, and the angles of arrivals are found to be 60°, 180°, and −60°. ▲

9-2-8 Intersection of the RL with the Imaginary Axis

Routh-Hurwitz criterion may be used to find the intersection of the root loci on the imaginary axis.

The points where the RL intersect the imaginary axis of the *s*-plane, and the corresponding values of *K* may be determined by means of the Routh-Hurwitz criterion. For complex situations, when the RL have multiple number of intersections on the imaginary axis, the intersects and the critical values of *K* can be determined with the help of the root-locus computer program. The Bode diagram method in Chap. 10, associated with the frequency response, can also be used for this purpose.

EXAMPLE 9-2-11 The root loci shown in Fig. F-7 is for the equation

$$s(s+3)(s^2+2s+2)+K=0 \qquad (9\text{-}43)$$

Figure 9-9 shows that the root loci intersect the *jω* axis at two points. Applying the Routh-Hurwitz criterion to Eq. (9-43), and by solving the auxiliary equation, we have the critical value of *K* for stability at *K* = 8.16, and the corresponding crossover points on the *jω*-axis are at ±j1.095. ▲

9-2-9 Breakaway Points (Saddle Points) on the RL

Breakaway points on the RL of an equation correspond to multiple-order roots of the equation.

Figure 9-11*a* illustrates a case in which two branches of the root loci meet at the breakaway point on the real axis and then depart from the axis in opposite directions. In this case, the breakaway point represents a double root of the equation when the value of *K* is assigned the value corresponding to the point. Fig. 9-11*b* shows another common situation when two complex-conjugate root loci approach the real axis, meet at the breakaway point, and then depart in opposite directions along the real axis. In general, a breakaway point may involve more than two root loci. Figure 9-11*c* illustrates a situation when the breakaway point represents a fourth-order root.

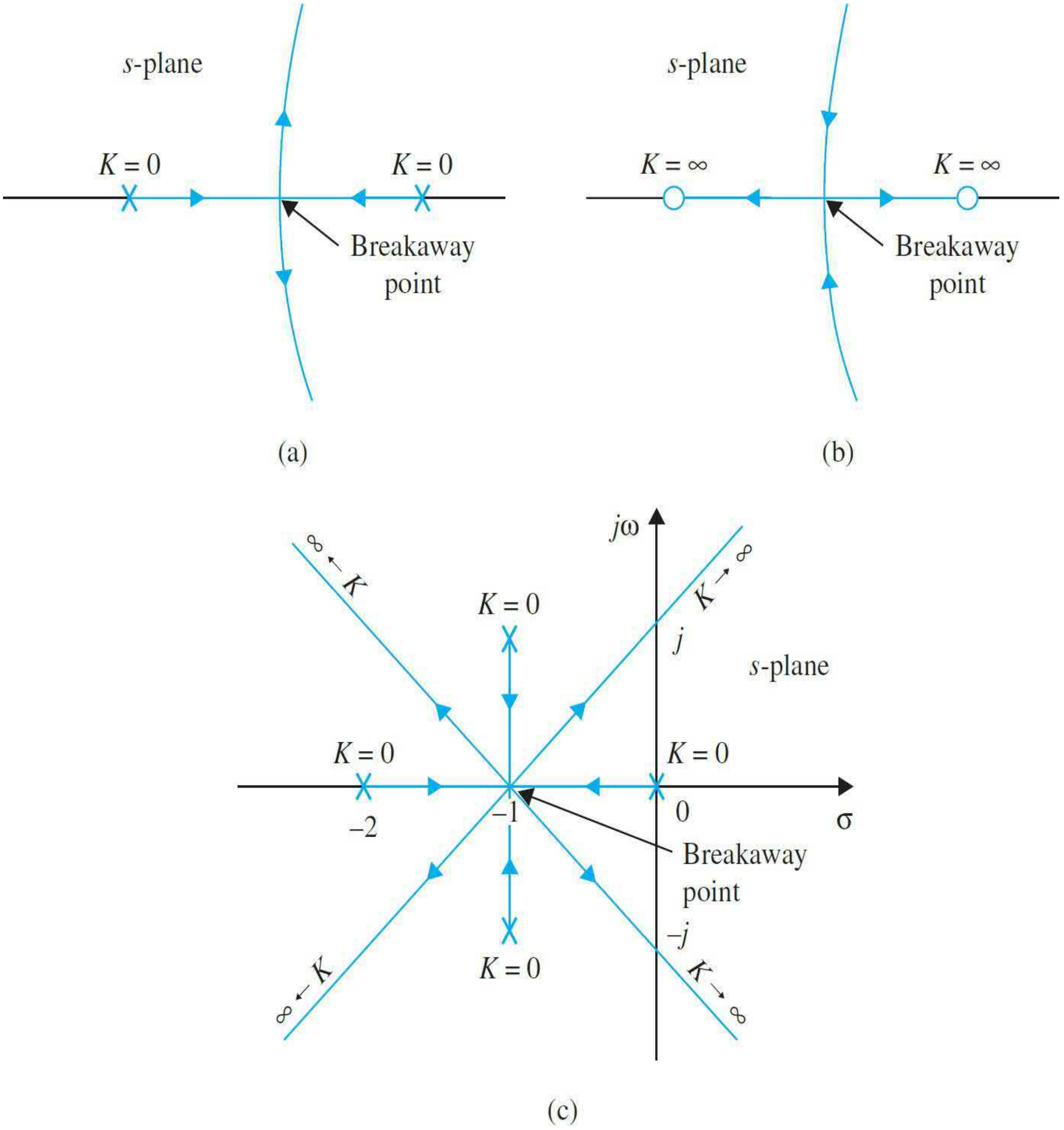


Figure 9-11 Examples of breakaway points on the real axis in the *s*-plane.

- A root-locus plot may have more than one breakaway points.
- Breakaway points may be complex conjugates in the *s*-plane.

A root-locus diagram can have, of course, more than one breakaway point.

Moreover, the breakaway points need not always be on the real axis. Because of the conjugate symmetry of the root loci, the breakaway points not on the real axis must be in complex-conjugate pairs. Refer to Fig. 9-14 for an example of root loci with complex breakaway points. The properties of the breakaway points of root loci are given as follows:

The breakaway points on the RL of $1+KG_1(s)H_1(s)=0$ must satisfy

$$\frac{dG_1(s)H_1(s)}{ds}=0 \tag{9-44}$$

It is important to point out that the condition for the breakaway point given in Eq. (9-44) is *necessary* but *not sufficient*. In other words, all breakaway points on the root loci must satisfy Eq. (9-44), but not all solutions of Eq. (9-44) are breakaway points. To be a breakaway point, the solution of Eq. (9-44) must also satisfy Eq. (9-5), that is, must also be a point on the root loci for some real *K*.

If we take the derivatives on both sides of Eq. (9-12) with respect to *s*, we get

$$\frac{dK}{ds}=\frac{dG_1(s)H_1(s)/ds}{[G_1(s)H_1(s)]^2} \tag{9-45}$$

Thus, the condition in Eq. (9-44) is equivalent to

$$\frac{dK}{ds}=0 \tag{9-46}$$

9-2-10 Angles of Arrival and Departure of Root Loci at the Breakaway Point

The angles at which the root loci arrive at or depart from a breakaway point depend on the number of loci that are involved at the point. For example, the root loci shown in Fig. 9-11*a* and *b* all arrive and break away at 180° apart, whereas in Fig. 9-11*c*, the four root loci arrive and depart with angles 90° apart. In general, ***n* root loci** ($-\infty \le K \le \infty$) **arrive at or depart from a breakaway point 180/*n* degrees apart**.

Many root-locus computer programs have features that will obtain the breakaway points, which is a rather tedious task to do manually.

EXAMPLE 9-2-12 Consider the second-order equation (similar to the PD control system in Example 7-7-1)

$$s(s+2)+K(s+4)=0 \tag{9-47}$$

Based on some of the properties of the root loci described thus far, the root loci of Eq. (9-47) are sketched as shown in Fig. 9-12 for −∞ < K < ∞. It can be proven that the complex portion of the root loci is a circle. The two breakaway points are on the real axis, one between 0 and −2 and the other between −4 and −∞. From Eq. (9-48), we have

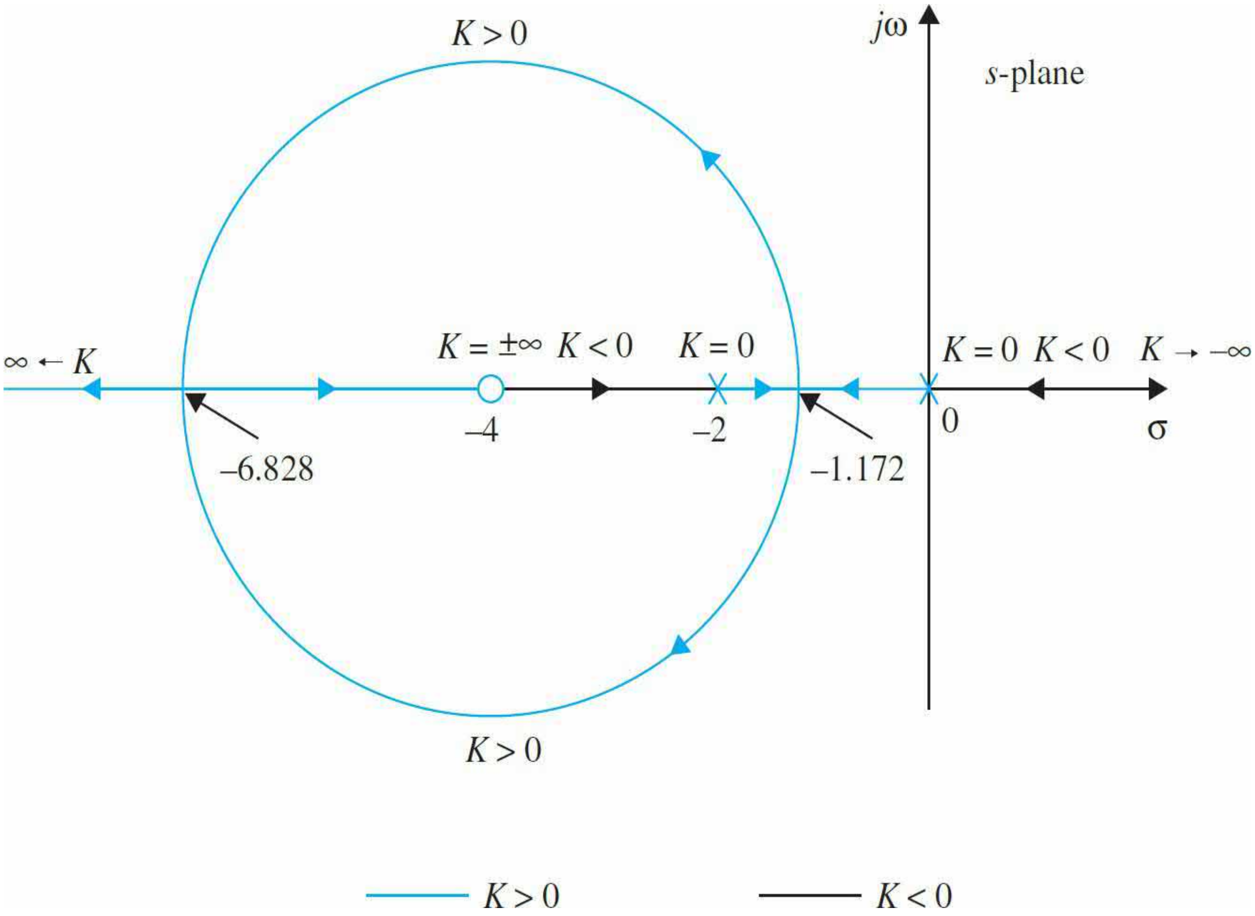


Figure 9-12 Root loci of $s(s+2)+K(s+4)=0$.

Applying Eq. (9-44), the breakaway points on the root loci must satisfy

$$\frac{dG_1(s)H_1(s)}{ds}=\frac{s(s+2)-2(s+1)(s+4)}{s^2(s+2)^2}=0 \tag{9-49}$$

or

$$s^2+8s+8=0 \tag{9-50}$$

Solving Eq. (9-50), we find the two breakaway points of the root loci at s = −1.172 and −6.828. Figure 9-12 shows that the two breakaway points are all on the root loci for positive K. ▲

EXAMPLE 9-2-13 Consider the equation (another example of a PD control system)

$$s^2+2s+2+K(s+2)=0 \tag{9-51}$$

The equivalent $G(s)H(s)$ is obtained by dividing both sides of Eq. (9-51) by the terms that do not contain K. We have

$$G(s)H(s)=\frac{K(s+2)}{s^2+2s+2} \tag{9-52}$$

Based on the poles and zeros of $G(s)H(s)$, the root loci of Eq. (9-52) are plotted as shown in Fig. 9-13. The plot shows that there are two breakaway points, one for $K > 0$ and one for $K < 0$. These breakaway points are determined from

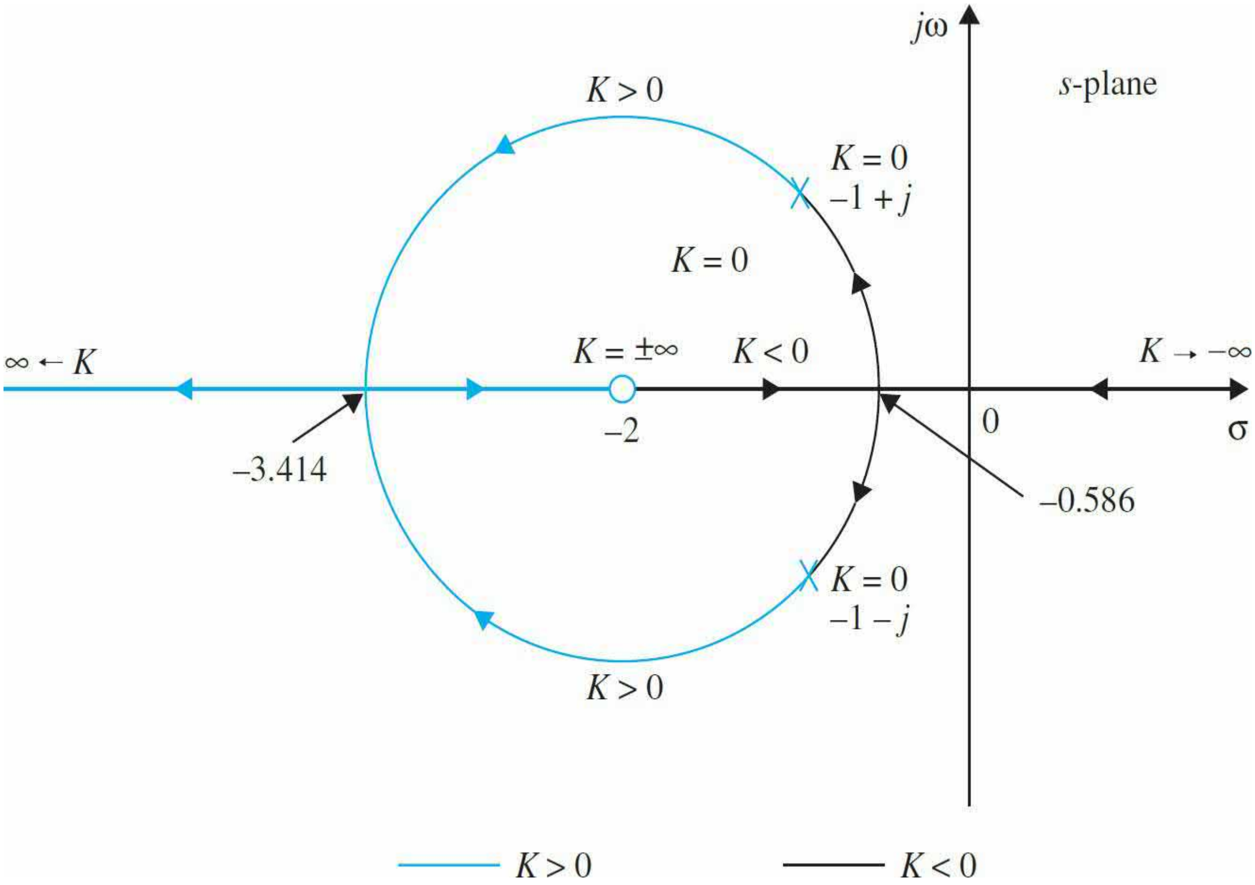


Figure 9-13 Root loci of $s^2 + 2s + 2 + K(s + 2) = 0$.

$$\frac{dG_1(s)H_1(s)}{ds} = \frac{d}{ds}\left(\frac{s+2}{s^2+2s+2}\right) = \frac{s^2+2s+2-2(s+1)(s+2)}{(s^2+2s+2)^2} = 0 \tag{9-53}$$

or

$$s^2+4s+2=0 \tag{9-54}$$

The solution of this equation gives the breakaway point as $s = -0.586$ and $s = -3.414$. ▲

EXAMPLE 9-2-14 Figure 9-14 shows the root loci of the equation

$$s(s+4)(s^2+4s+20)+K=0 \tag{9-55}$$

Dividing both sides of the last equation by the terms that do not contain K, we have

$$1+KG_1(s)H_1(s)=1+\frac{K}{s(s+4)(s^2+4s+20)}=0 \tag{9-56}$$

Since the poles of $G_1(s)H_1(s)$ are symmetrical about the axes s = −2 and $\omega = 0$ in the *s*-plane, the root loci of the equation are also symmetrical with respect to these two axes. Taking the derivative of $G_1(s)H_1(s)$ with respect to *s*, we get

$$\frac{dG_1(s)H_1(s)}{ds}=-\frac{4s^3+24s^2+72s+80}{[s(s+4)(s^2+4s+20)]^2}=0 \tag{9-57}$$

or

$$s^3+6s^2+18s+20=0 \tag{9-58}$$

The solutions of the last equation are $s = -2$, $-2 + j2.45$, and $-2 - j2.45$. In this case, Fig. 9-14 shows that all the solutions of Eq. (9-58) are breakaway points on the root loci, and two of these points are complex. ▲

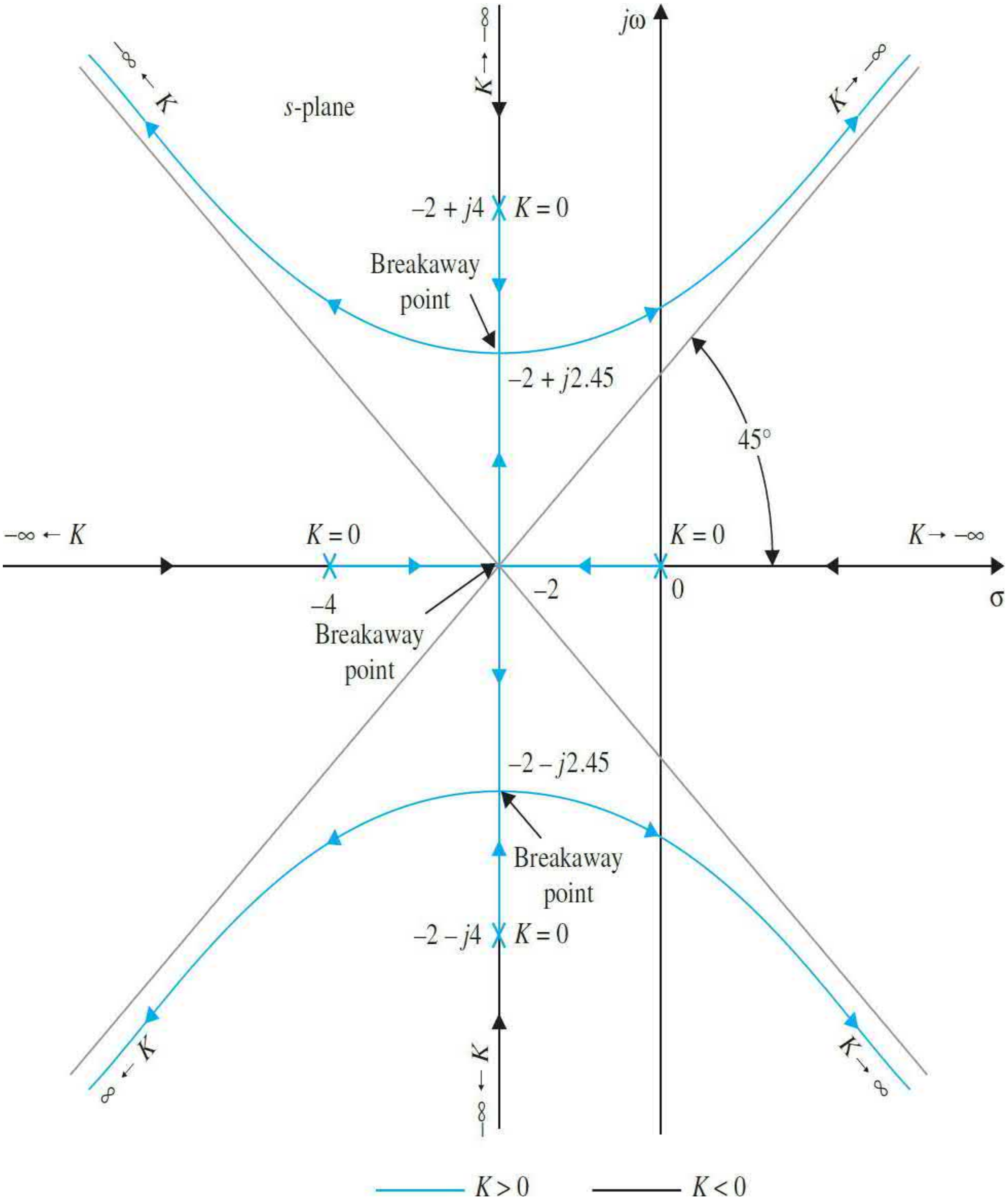


Figure 9-14 Root loci of $s(s + 4)(s^2 + 4s + 20) + K = 0$.

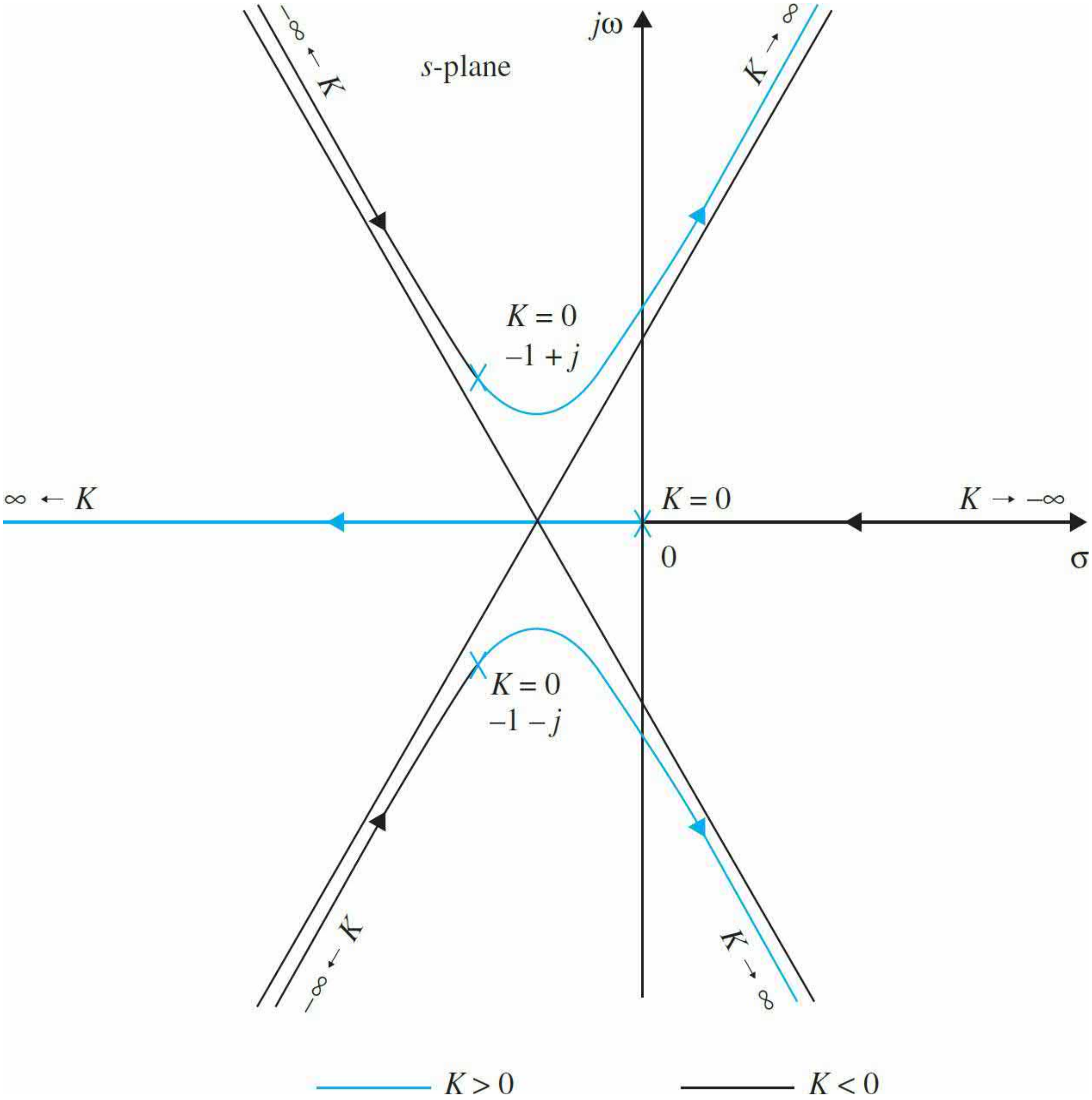
EXAMPLE 9-2-15 In this example, we demonstrate that not all the solutions of Eq. (9-44) are breakaway points on the root loci. The root loci of the equation

$$s(s^2+2s+2)+K=0 \qquad (9\text{-}59)$$

are shown in Fig. 9-15. The root loci show that neither the $K \geq 0$ loci nor the $K \leq 0$ loci has any breakaway point in this case. However, writing Eq. (F-36) as

$$1+KG_1(s)H_1(s)=1+\frac{K}{s(s^2+2s+2)}=0 \qquad (9\text{-}60)$$

Figure 9-15 Root loci of $s(s^2 + 2s + 2) + K = 0$.



and applying Eq. (9-44), we have the equation for the breakaway points:

$$3s^2 + 4s + 2 = 0 \qquad (9\text{-}61)$$

The roots of Eq. (9-61) are $s = -0.667 + j0.471$ and $-0.667 - j0.471$. These two roots are **not** breakaway points on the root loci, since they do not satisfy Eq. (9-59) for any real value of K. ▲

9-2-11 Calculation of *K* on the Root Loci

Once the root loci are constructed, the values of *K* at any point s_1 on the loci can be determined by use of the defining equation of Eq. (9-20). Graphically, the magnitude of *K* can be written as

$$|K| = \frac{\Pi \text{ lengths of vectors drawn from the poles of } G_1(s)H_1(s) \text{ to } s_1}{\Pi \text{ lengths of vectors drawn from the zeros of } G_1(s)H_1(s) \text{ to } s_1} \qquad (9\text{-}62)$$

EXAMPLE 9-2-16 As an illustration on the determination of the value of *K* on the root loci, the root loci of the equation

$$s^2 + 2s + 2 + K(s+2) = 0 \qquad (9\text{-}63)$$

are shown in Fig. 9-16. The value of *K* at the point s_1 is given by

$$K = \frac{A \times B}{C} \qquad (9\text{-}64)$$

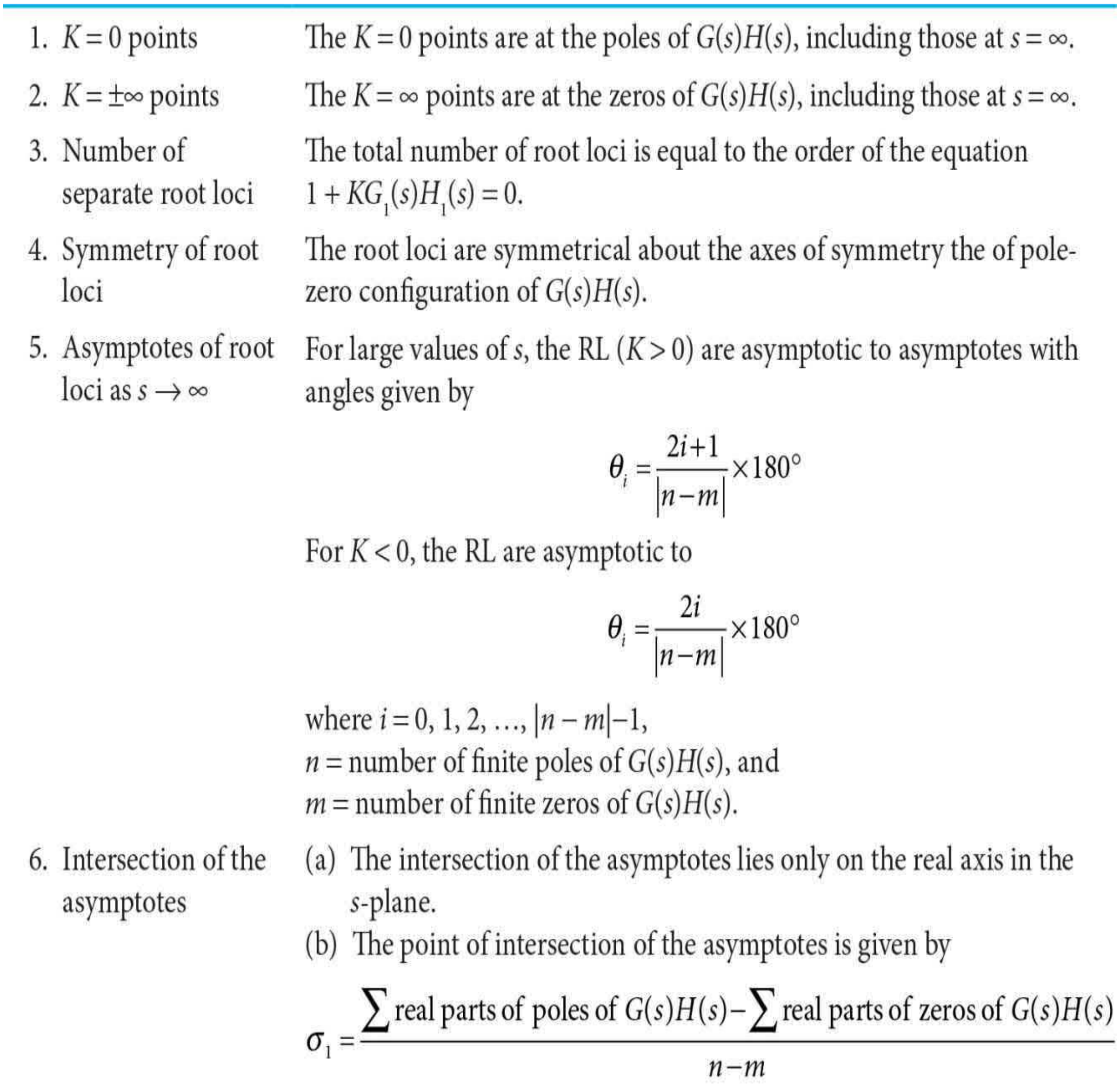
Figure 9-16 Graphical method of finding the values of *K* on the real axis.

where *A* and *B* are the lengths of the vectors drawn from the poles of $G(s)H(s) = K(s + 2)/(s^2 + 2s + 2)$ to the point s_1, and *C* is the length of the vector drawn from the zero of $G(s)H(s)$ to s_1. In this case, s_1 is on the locus where *K* is positive. In general, the value of *K* at the point where the root loci intersect the imaginary axis can also be found by the method just described. Figure 9-16 shows that the value of *K* at $s = 0$ is −1. The computer method and the Routh-Hurwitz criterion are other convenient alternatives of finding the critical value of *K* for stability. ▲

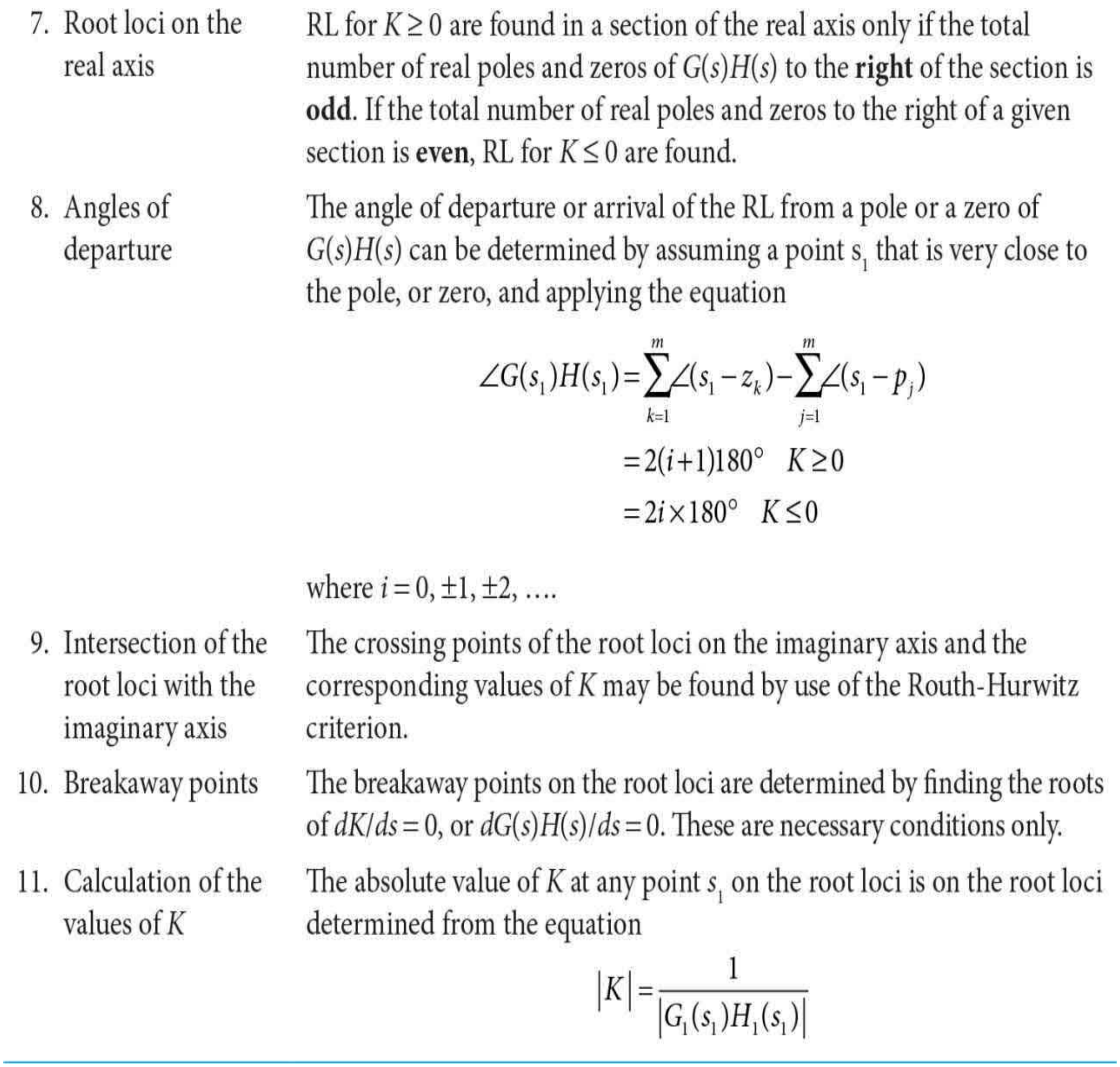
9-2-12 Summary: Properties of the Root Loci

In summary, except for extremely complex cases, the properties on the root loci just presented should be adequate for making a reasonably accurate sketch of the root-locus diagram short of plotting it point by point. The computer program can be used to solve for the exact root locations, the breakaway points, and some of the other specific details of the root loci, including the plotting of the final loci. However, one cannot rely on the computer solution completely, since the user still has to decide on the range and resolution of *K* so that the root-locus plot has a reasonable appearance. For quick reference, the important properties described are summarized in Table 9-1.

Table 9-1 Properties of the Root Loci of 1 + KG$_1$(s)H$_1$ = 0



1. $K = 0$ points	The $K = 0$ points are at the poles of $G(s)H(s)$, including those at $s = \infty$.
2. $K = \pm\infty$ points	The $K = \infty$ points are at the zeros of $G(s)H(s)$, including those at $s = \infty$.
3. Number of separate root loci	The total number of root loci is equal to the order of the equation $1 + KG_1(s)H_1(s) = 0$.
4. Symmetry of root loci	The root loci are symmetrical about the axes of symmetry the of pole-zero configuration of $G(s)H(s)$.
5. Asymptotes of root loci as $s \to \infty$	For large values of s, the RL ($K > 0$) are asymptotic to asymptotes with angles given by $\theta_i = \dfrac{2i+1}{\lvert n-m \rvert} \times 180°$ For $K < 0$, the RL are asymptotic to $\theta_i = \dfrac{2i}{\lvert n-m \rvert} \times 180°$ where $i = 0, 1, 2, \ldots, \lvert n - m \rvert - 1$, n = number of finite poles of $G(s)H(s)$, and m = number of finite zeros of $G(s)H(s)$.
6. Intersection of the asymptotes	(a) The intersection of the asymptotes lies only on the real axis in the s-plane. (b) The point of intersection of the asymptotes is given by $\sigma_1 = \dfrac{\sum \text{real parts of poles of } G(s)H(s) - \sum \text{real parts of zeros of } G(s)H(s)}{n-m}$



7. Root loci on the real axis	RL for $K \geq 0$ are found in a section of the real axis only if the total number of real poles and zeros of $G(s)H(s)$ to the **right** of the section is **odd**. If the total number of real poles and zeros to the right of a given section is **even**, RL for $K \leq 0$ are found.
8. Angles of departure	The angle of departure or arrival of the RL from a pole or a zero of $G(s)H(s)$ can be determined by assuming a point s_1 that is very close to the pole, or zero, and applying the equation $\angle G(s_1)H(s_1) = \sum_{k=1}^{m} \angle(s_1 - z_k) - \sum_{j=1}^{m} \angle(s_1 - p_j)$ $= 2(i+1)180^\circ \quad K \geq 0$ $= 2i \times 180^\circ \quad K \leq 0$ where $i = 0, \pm 1, \pm 2, \ldots$
9. Intersection of the root loci with the imaginary axis	The crossing points of the root loci on the imaginary axis and the corresponding values of K may be found by use of the Routh-Hurwitz criterion.
10. Breakaway points	The breakaway points on the root loci are determined by finding the roots of $dK/ds = 0$, or $dG(s)H(s)/ds = 0$. These are necessary conditions only.
11. Calculation of the values of K	The absolute value of K at any point s_1 on the root loci is on the root loci determined from the equation $\lvert K \rvert = \dfrac{1}{\lvert G_1(s_1)H_1(s_1) \rvert}$

EXAMPLE 9-2-17 Consider the equation

$$s(s+5)(s+6)(s^2+2s+2)+K(s+3)=0 \qquad \text{(9-65}$$

Dividing both sides of the last equation by the terms that do not contain K, we have

$$G(s)H(s) = \frac{K(s+3)}{s(s+5)(s+6)(s^2+2s+2)} \qquad \text{(9-66)}$$

The following properties of the root loci are determined:

1. The $K = 0$ points are at the poles of $G(s)H(s)$: $s = -5, -6, -1 + j$, and $-1 - j$.

2. The $K = \pm\infty$ points are at the zeros of $G(s)H(s)$: $s = -3, \infty\cdot, \infty, \infty$.

3. There are five separate branches on the root loci.

4. The root loci are symmetrical with respect to the real axis of the s-plane.

5. Since $G(s)H(s)$ has five poles and one finite zero, four RL and CRL should approach infinity along the asymptotes. The angles of the asymptotes of the RL are given by [Eq. (9-33)]

$$\theta_i = \frac{2i+1}{|n-m|}180^\circ = \frac{2i+1}{|5-1|}180^\circ \quad 0 \le K < \infty \tag{9-67}$$

for $i = 0, 1, 2, 3$. Thus, the four root loci that approach infinity as K approaches infinity should approach asymptotes with angles of 45°, −45°, 135°, and −135°, respectively. The angles of the asymptotes of the CRL at infinity are given by Eq. (9-34):

$$\theta_i = \frac{2i}{|n-m|}180^\circ = \frac{2i}{|5-1|}180^\circ \quad -\infty < K \le 0 \tag{9-68}$$

for $i = 0, 1, 2, 3$. Thus, as K approaches $-\infty$, four root loci for $K < 0$ should approach infinity along asymptotes with angles of 0°, 90°, 180°, and 270°.

6. The intersection of the asymptotes is given by [Eq. (9-36)]

$$\sigma_1 = \frac{\Sigma(-5-6-1-1)-(-3)}{4} = -2.5 \tag{9-69}$$

The results from these six steps are illustrated in Fig. 9-17. It should be pointed out that in general the properties of the asymptotes do not indicate on which side of the asymptotes the root loci lie. The asymptotes indicate nothing more than the behavior of the root loci as $|s| \to \infty$. In fact, the root locus can even cross an asymptote in the finite *s* domain. The segments of the root loci shown in Fig. 9-17 can be accurately plotted only if additional information is obtained.

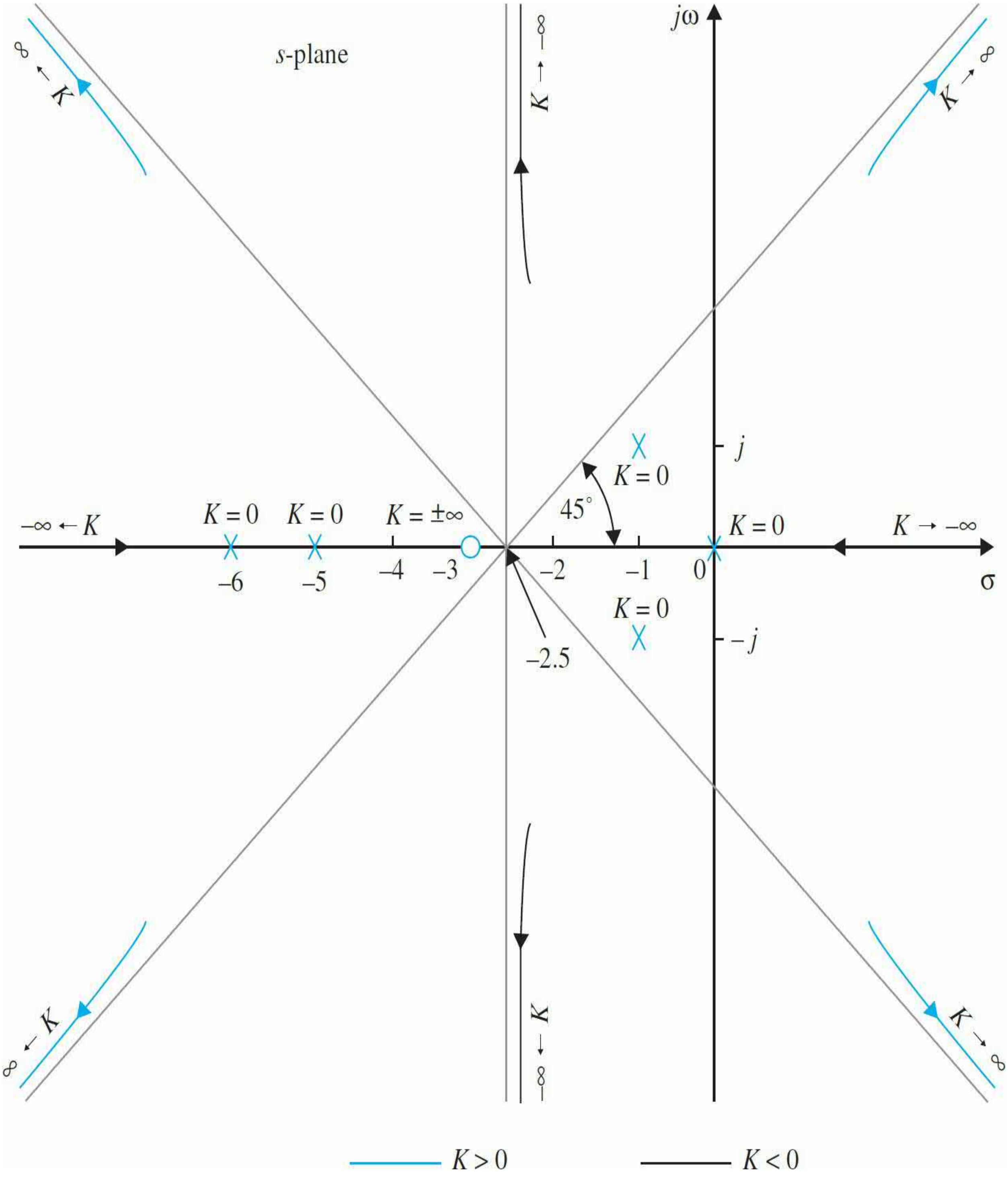


Figure 9-17 Preliminary calculation of the root loci of $s(s + 5)(s + 6)(s^2 + 2s + 2) + K(s + 3) = 0$.

7. Root loci on the real axis: There are $K \geq 0$ root loci on the real axis between $s = 0$ and -3, and $s = -5$ and -6. There are $K \leq 0$ root loci on the remaining portions of the real axis, that is, between $s = -3$ and -5, and $s = -6$ and $-\infty$, as shown in Fig. 9-18.

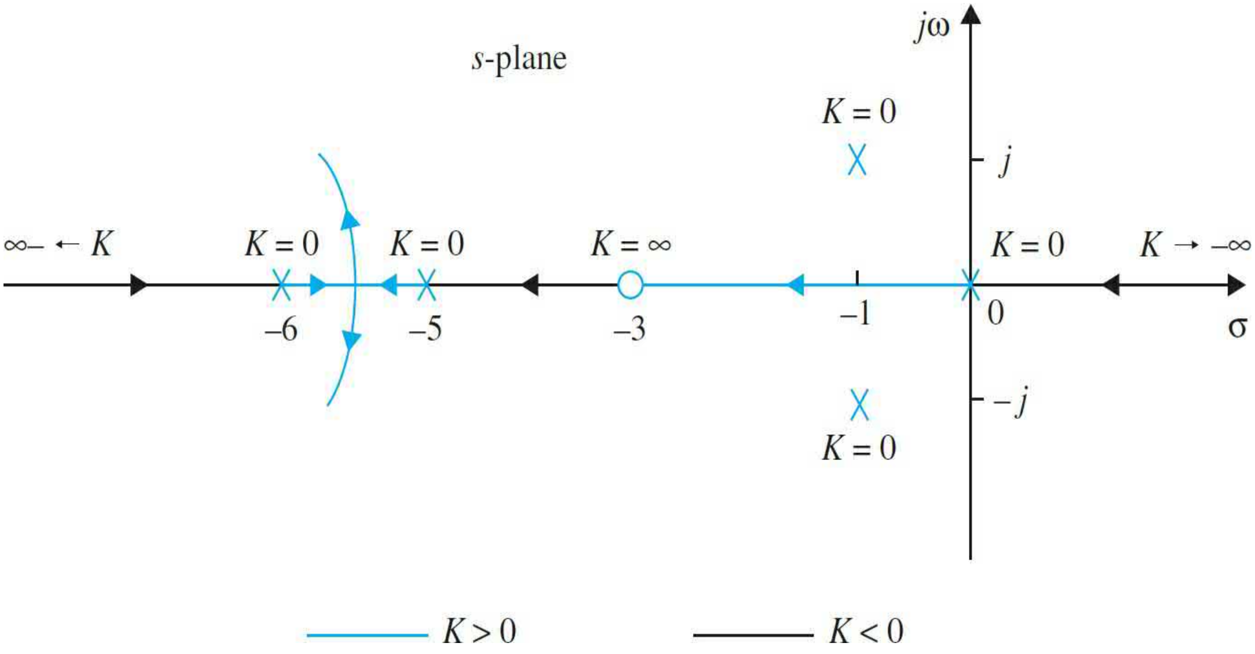


Figure 9-18 Root loci of $s(s + 5)(s + 6)(s^2 + 2s + 2) + K(s + 3) = 0$ on the real axis.

8. Angles of departure: The angle of departure θ of the root loci leaving the pole at $-1 + j$ is determined using Eq. (9-18). If s_1 is a point on the root loci leaving the pole at $-1 + j$, and s_1 is very close to the pole, as shown in Fig. 9-19, Eq. (9-18) gives

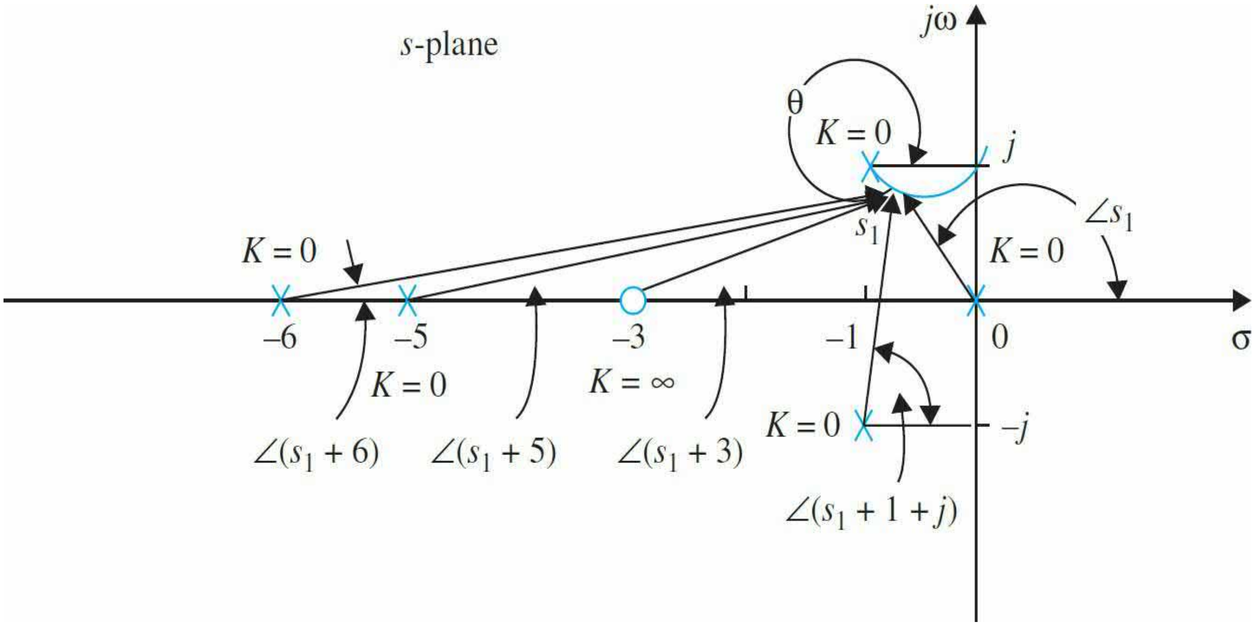


Figure 9-19 Computation of angle of departure of the root loci of $s(s + 5)(s + 6)(s^2 + 2s + 2) + K(s + 3) = 0$.

$$\angle(s_1+3)-\angle s_1-\angle(s_1+1+j)-\angle(s_1+5)-\angle(s_1+1-j)=(2i+1)180^\circ \tag{9-70}$$

or

$$26.6^\circ-135^\circ-90^\circ-14^\circ-11.4^\circ-\theta\cong(2i+1)180^\circ \tag{9-71}$$

for $i = 0, \pm1, \pm2, \ldots$ Therefore, selecting $i = 2$, $\theta \cong -43.8^\circ$

Similarly, Eq. (9-19) is used to determine the angle of arrival θ’ ′ of the $K \le 0$ root loci arriving at the pole $-1 + j$. It is easy to see that θ' differs from θ by 180°; thus,

$$\theta'=180^\circ-43.8^\circ=136.2^\circ \tag{9-72}$$

9. The intersection of the root loci on the imaginary axis is determined using Routh’s tabulation. Equation (F-42) is written as

$$s^5+13s^4+54s^3+82s^2+(60+K)s+3K=0 \tag{9-73}$$

Routh’s tabulation is

$$\begin{array}{llll}
s^5 & 1 & 54 & 60+K \\
s^4 & 13 & 82 & 3K \\
s^3 & 47.7 & 0.769K & 0 \\
s^2 & 65.6-0.212K & 3K & 0 \\
s^1 & \dfrac{3940-105K-0.163K^2}{65.6-0.212K} & 0 & 0 \\
s^0 & 3K & 0 & 0
\end{array}$$

For Eq. (9-73) to have no roots on the imaginary axis or in the right-half of the *s*-plane, the elements in the first column of Routh’s tabulation must all be of the same sign. Thus, the following inequalities must be satisfied:

$$65.6-0.212K>0 \quad \text{or} \quad K<309 \tag{9-74}$$

$$3940-105K-0.163K^2>0 \quad \text{or} \quad K<35 \tag{9-75}$$

$$K>0 \tag{9-76}$$

Thus, all the roots of Eq. (9-73) will stay in the left-half *s*-plane if *K* lies between 0 and 35, which means that the root loci of Eq. (9-73) cross the imaginary axis when *K* = 35 and *K* = 0.

The coordinates at the crossover points on the imaginary axis that correspond to *K* = 35 are determined from the auxiliary equation:

$$A(s)=(65.6-0.212K)s^2+3K=0 \tag{9-77}$$

which is obtained by using the coefficients from the row just above the row of zeros in the s^1 row that would have happened when *K* is set to 35. Substituting *K* = 35 in Eq. (9-77), we get

$$58.2s^2+105=03 \tag{9-78}$$

The roots of Eq. (9-78) are *s* = *j*1.34 and −*j*1.34, which are the points at which the root loci cross the *jω-axis*.

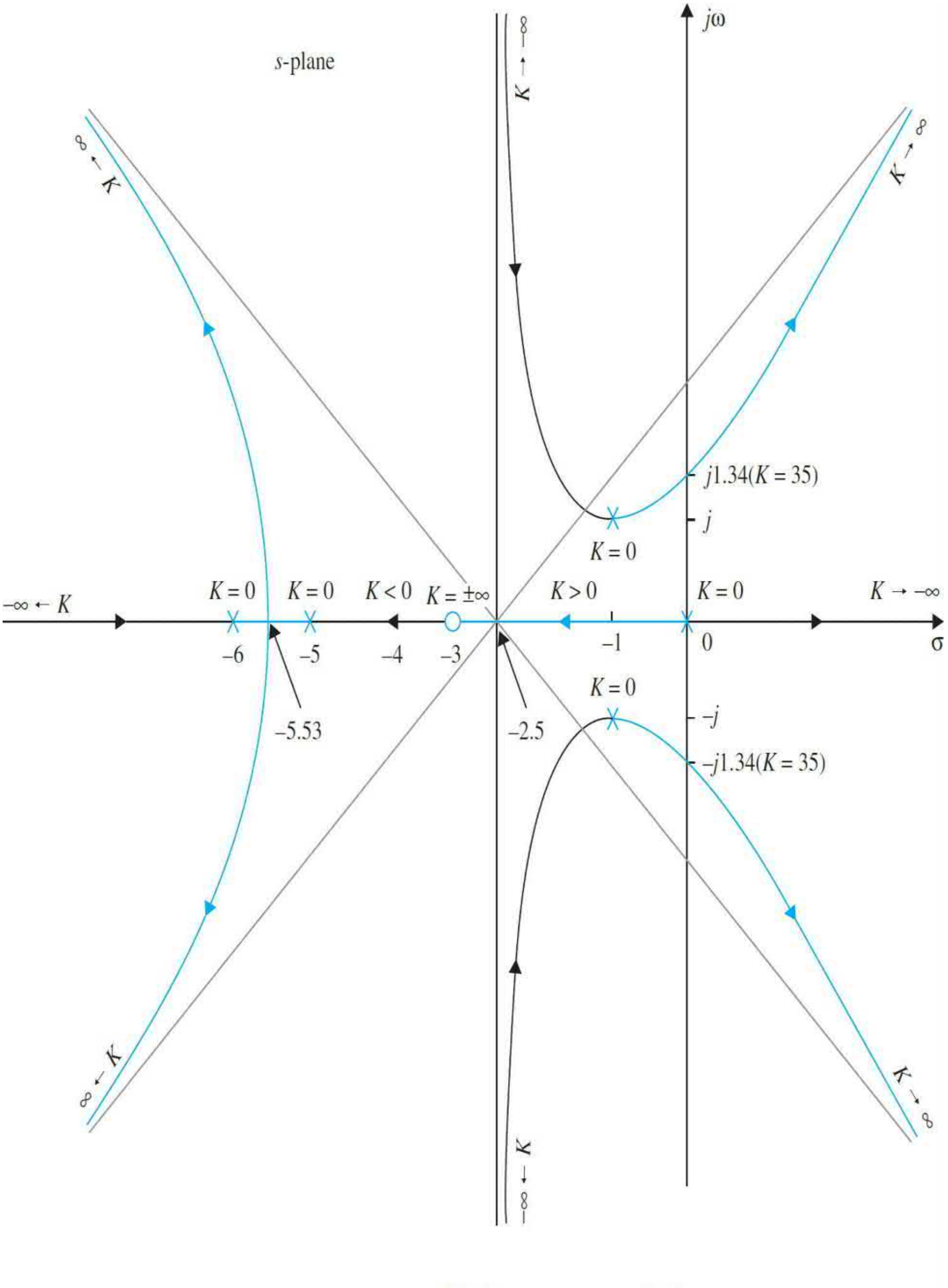
10. Breakaway points: Based on the information gathered from the preceding nine steps, a trial sketch of the root loci indicates that there can be only one breakaway point on the entire root loci, and the point should lie between the two poles of $G(s)H(s)$ at $s = -5$ and -6. To find the breakaway point, we take the derivative on both sides of Eq. (9-65) with respect to s and set it to zero; the resulting equation is

$$s^5 + 13.5s^4 + 66s^3 + 142s^2 + 123s + 45 = 03 \tag{9-79}$$

Since there is only one breakaway expected, only one root of the last equation is the correct solution of the breakaway point. The five roots of Eq. (9-79) are

$$s = 3.33 + j1.204 \qquad s = 3.33 - j1.204$$
$$s = -0.656 + j0.468 \qquad s = -0.656 - j0.468$$
$$s = -5.53$$

Clearly, the breakaway point is at −5.53. The other four solutions do not satisfy Eq. (9-73) and are not breakaway points. Based on the information obtained in the last 10 steps, the root loci of Eq. (9-73) are sketched as shown in Fig. 9-20.



s-plane
jω
σ
K → ∞
K → −∞
−∞ ← K
∞ ← K
j1.34(K = 35)
j
K = 0
K = 0
K = 0
K < 0
K = ±∞
K > 0
K = 0
K → −∞
−6
−5
−4
−3
−1
0
−5.53
−2.5
K = 0
−j
−j1.34(K = 35)
∞ ← K
K → ∞
−∞ ← K
K > 0
K < 0

Figure 9-20 Root loci of $s(s + 5)(s + 6)(s^2 + 2s + 2) + K(s + 3) = 0$.

9-3 THE ROOT SENSITIVITY

The condition on the breakaway points on the RL in Eq. (9-47) leads to the **root sensitivity**[17,19] of the characteristic equation. The sensitivity of the roots of the characteristic equation when K varies is defined as the **root sensitivity** and is given by

$$S_K = \frac{ds/s}{dK/K} = \frac{K}{s}\frac{ds}{dK} \tag{9-80}$$

Thus, Eq. (9-47) shows that *the root sensitivity at the breakaway points is infinite*. From the root-sensitivity standpoint, we should avoid selecting the value of K to operate at the breakaway points, which correspond to multiple-order roots of the characteristic equation. In the design of control systems, not only it is important to arrive at a system that has the desired characteristics but also, just as important, the system should be insensitive to parameter variations. For instance, a system may perform satisfactorily at a certain K, but if it is very sensitive to the variation of K, it may get into the undesirable performance region or become unstable if K varies by only a small amount. In formal control-system terminology, a system that is insensitive to parameter variations is called a **robust system**. Thus, the root-locus study of control systems must involve not only the shape of the root loci with respect to the variable parameter K but also how the roots along the loci vary with the variation of K.

EXAMPLE 9-3-1 Figure 9-21 shows the root-locus diagram of

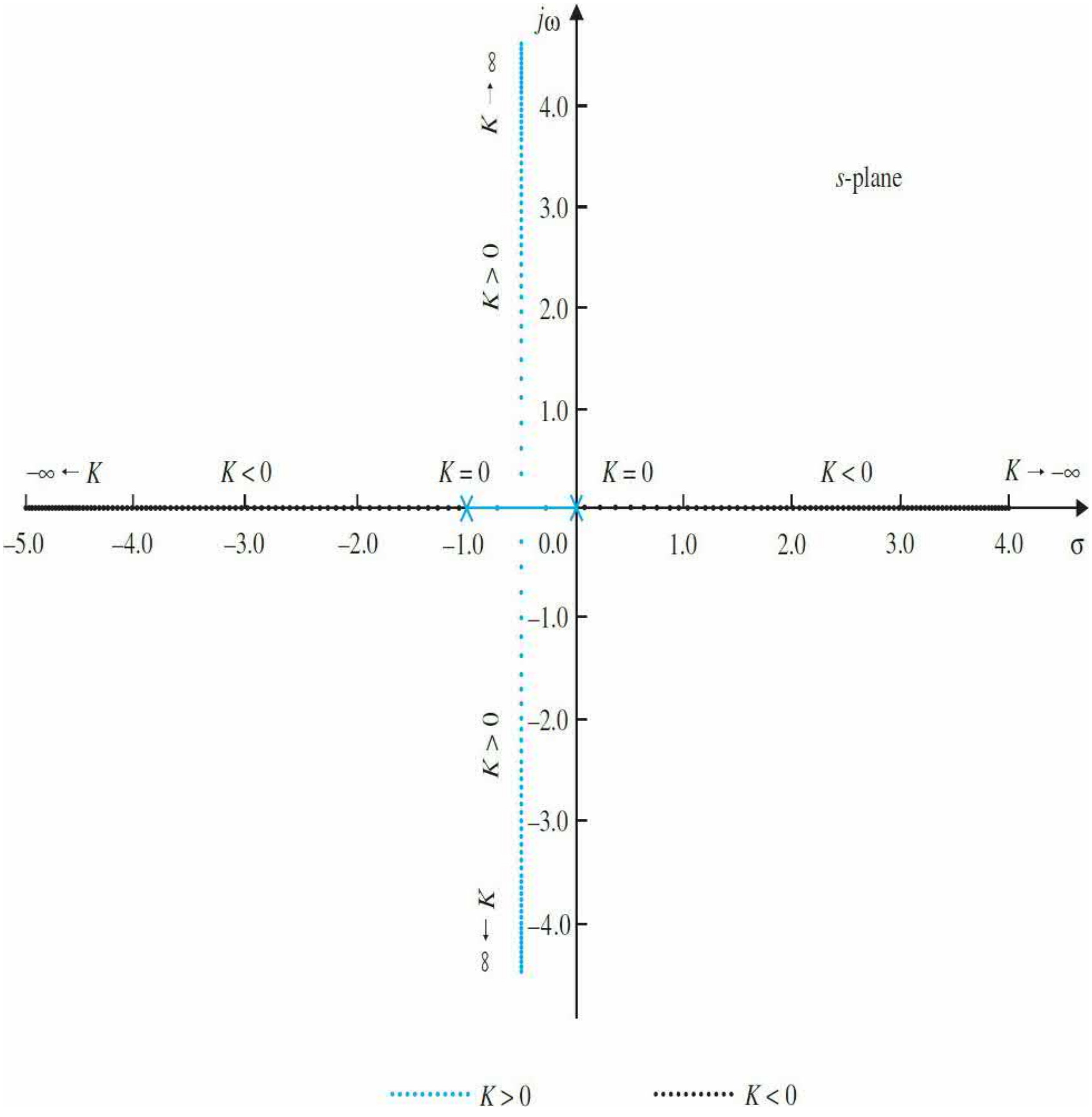


Figure 9-21 RL of *s*(*s* + 1) + *K* = 0 showing the root sensitivity with respect to *K*.

$$s(s+1)+K=0 \tag{9-81}$$

with *K* incremented uniformly over 100 values from −20 to 20. Each dot on the root-locus plot represents one root for a distinct value of *K*. Thus, we see that the root sensitivity is low when the magnitude of *K* is large. As the magnitude of *K* decreases, the movements of the roots become larger for the same incremental change in K. At the breakaway point, $s = -0.5$, the root sensitivity is infinite.

Figure 9-22 shows the RL of

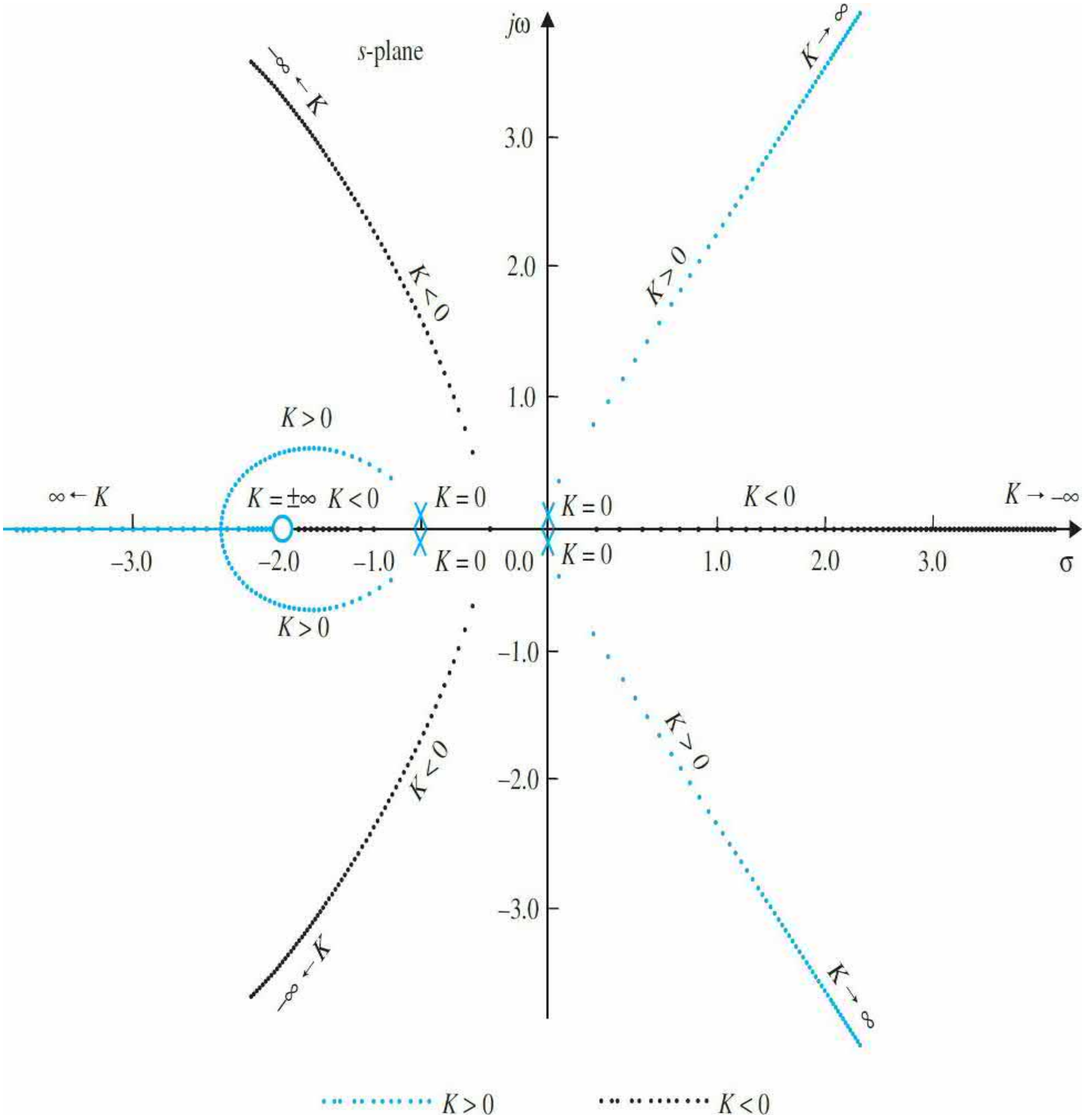


Figure 9-22 RL of $s^2(s+1)^2 + K(s+2) = 0$, showing the root sensitivity with respect to K.

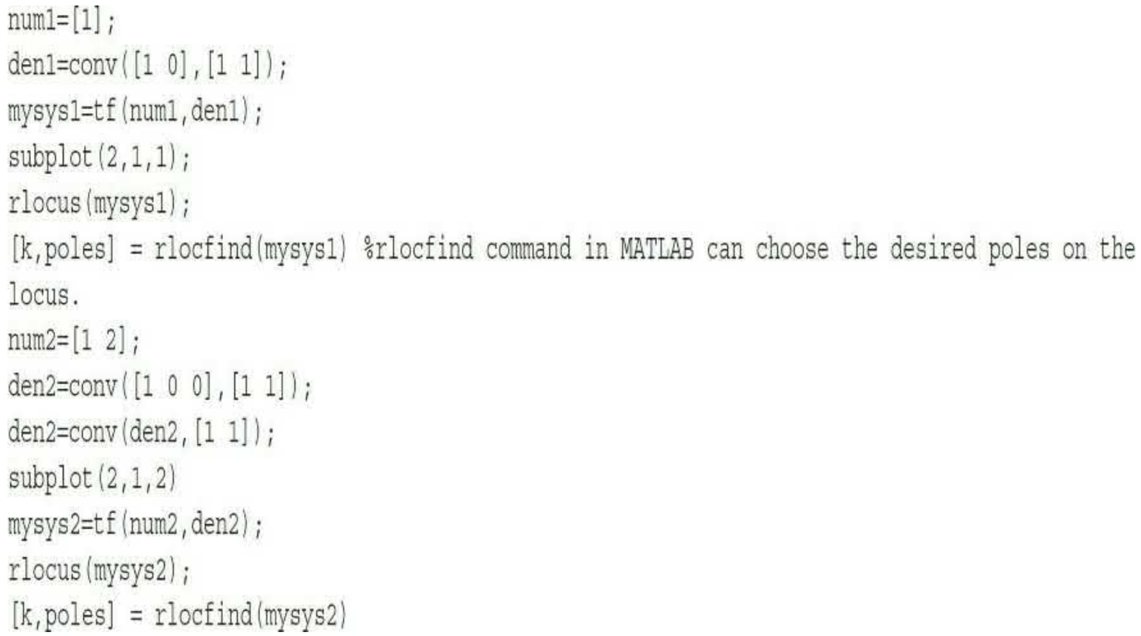
$$s(s+1) + K = 0 \tag{9-81}$$

with K incremented uniformly over 200 values from −40 to 50. Again, the loci show that the root sensitivity increases as the roots approach the breakaway points at $s = 0, -0.543, -1.0$, and -2.457. We can investigate the root sensitivity further by using the expression in Eq. (9-47). For the second-order equation in Eq. (9-81),

$$\frac{dK}{ds} = -2s - 1 \tag{9-83}$$

Toolbox 9-3-1

MATLAB statements for Eqs. (9-81) and (9-82).



```
num1=[1];
den1=conv([1 0],[1 1]);
mysys1=tf(num1,den1);
subplot(2,1,1);
rlocus(mysys1);
[k,poles] = rlocfind(mysys1) %rlocfind command in MATLAB can choose the desired poles on the
locus.
num2=[1 2];
den2=conv([1 0 0],[1 1]);
den2=conv(den2,[1 1]);
subplot(2,1,2)
mysys2=tf(num2,den2);
rlocus(mysys2);
[k,poles] = rlocfind(mysys2)
```

From Eq. (9-81), $K = -s(s + 1)$; the root sensitivity becomes

$$S_K = \frac{ds}{dK}\frac{K}{s} = \frac{s+1}{2s+1} \tag{9-84}$$

where $s = \sigma + j\omega$, and s must take on the values of the roots of Eq. (9-84). For the roots on the real axis, $\omega = 0$. Thus, Eq. (9-84) leads to

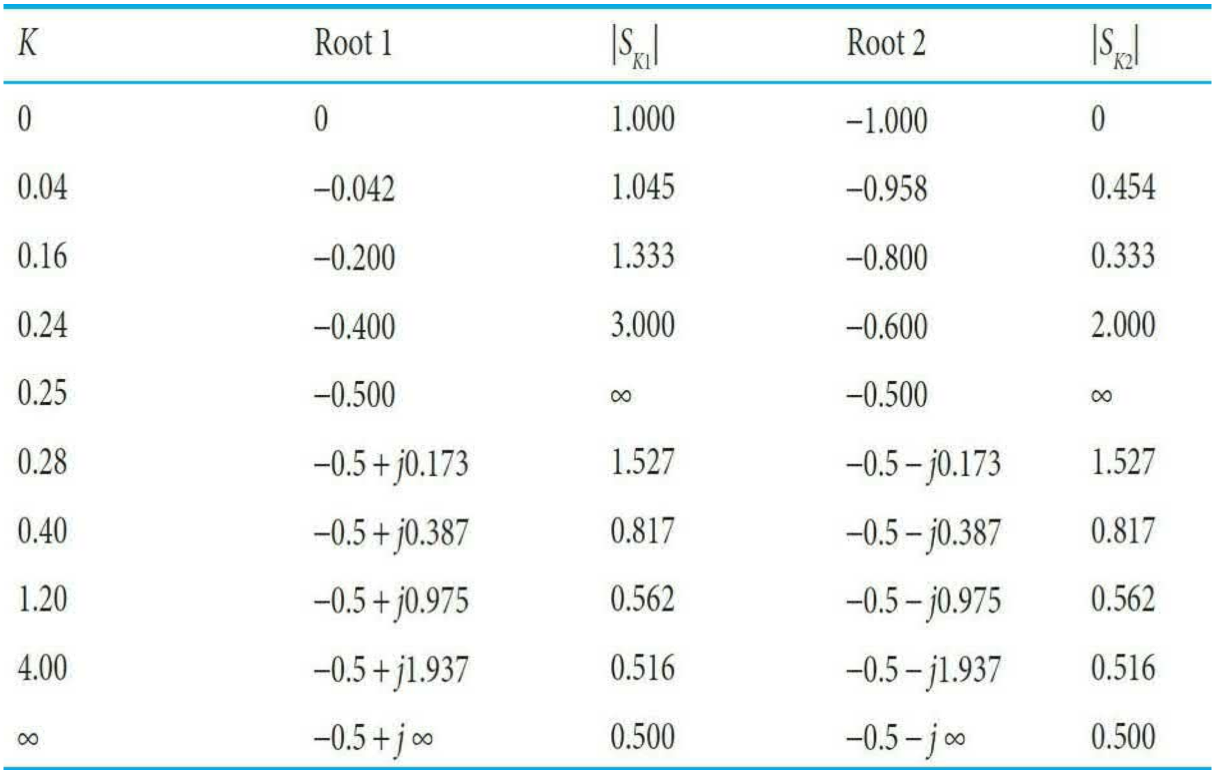
$$|S_K|_{\omega=0} = \left|\frac{\sigma+1}{2\sigma+1}\right| \tag{9-85}$$

When the two roots are complex, *s* = –0.5 for all values of *ω* ; Eq. (9-84) gives

$$|S_K|_{\sigma=-0.5} = \left(\frac{0.25+\omega^2}{4\omega^2}\right)^{1/2} \tag{9-86}$$

From Eq. (9-86), it is apparent that the sensitivities of the pair of complex-conjugate roots are the same, since *ω* appears only as ω^2 in the equation. Equation (9-85) indicates that the sensitivities of the two real roots are different for a given value of *K*. Table 9-2 gives the magnitudes of the sensitivities of the two roots of Eq. (9-81) for several values of *K*, where $|S_{K1}|$ denotes the root sensitivity of the first root, and $|S_{K2}|$ denotes that of the second root. These values indicate that although the two real roots reach *s* = –0.5 for the same value of *K* = 0.25, and each root travels the same distance from *ω* = 0 and *s* = –1, respectively, the sensitivities of the two real roots are not the same. ▲

Table 9-2 Root Sensitivity



K	Root 1	$\lvert S_{K1} \rvert$	Root 2	$\lvert S_{K2} \rvert$
0	0	1.000	−1.000	0
0.04	−0.042	1.045	−0.958	0.454
0.16	−0.200	1.333	−0.800	0.333
0.24	−0.400	3.000	−0.600	2.000
0.25	−0.500	∞	−0.500	∞
0.28	$-0.5+j0.173$	1.527	$-0.5-j0.173$	1.527
0.40	$-0.5+j0.387$	0.817	$-0.5-j0.387$	0.817
1.20	$-0.5+j0.975$	0.562	$-0.5-j0.975$	0.562
4.00	$-0.5+j1.937$	0.516	$-0.5-j1.937$	0.516
∞	$-0.5+j\infty$	0.500	$-0.5-j\infty$	0.500

9-4 DESIGN ASPECTS OF THE ROOT LOCI

One of the important aspects of the root-locus technique is that, for most control systems with moderate complexity, the analyst or designer can obtain vital information on the performance of the system by making a quick sketch of the RL using some or all of the properties of the root loci. It is of importance to understand all the properties of the RL even when the diagram is to be plotted with the help of a digital computer program. From the design standpoint, it is useful to learn the effects on the RL when poles and zeros of $G(s)H(s)$ are added or moved around in the s-plane. Some of these properties are helpful in the construction of the root-locus diagram. The design of the PI, PID, phase-lead, phase-lag, and lead-lag controllers discussed in Chap. 11 all have implications of adding poles and zeros to the loop transfer function in the s-plane.

9-4-1 Effects of Adding Poles and Zeros to $G(s)H(s)$

The general problem of controller design in control systems may be treated as an investigation of the effects to the root loci when poles and zeros are added to the loop transfer function *G*(*s*)*H*(*s*).

9-4-2 Addition of Poles to *G*(*s*)*H*(*s*)

Adding a pole to *G*(*s*)*H*(*s*) has the effect of pushing the root loci toward the right-half *s*-plane. The effect of adding a zero to *G*(*s*)*H*(*s*) can be illustrated with several examples.

EXAMPLE 9-4-1 Consider the function

$$G(s)H(s)=\frac{K}{s(s+a)}a>0 \tag{9-87}$$

The RL of 1 + *G*(*s*)*H*(*s*) = 0 are shown in Fig. 9-23*a*. These RL are constructed based on the poles of *G*(*s*)*H*(*s*), which are at *s* = 0 and –*a*. Now let us introduce a pole at *s* = –*b*, with *b* > *a*. The function *G*(*s*)*H*(*s*) now becomes

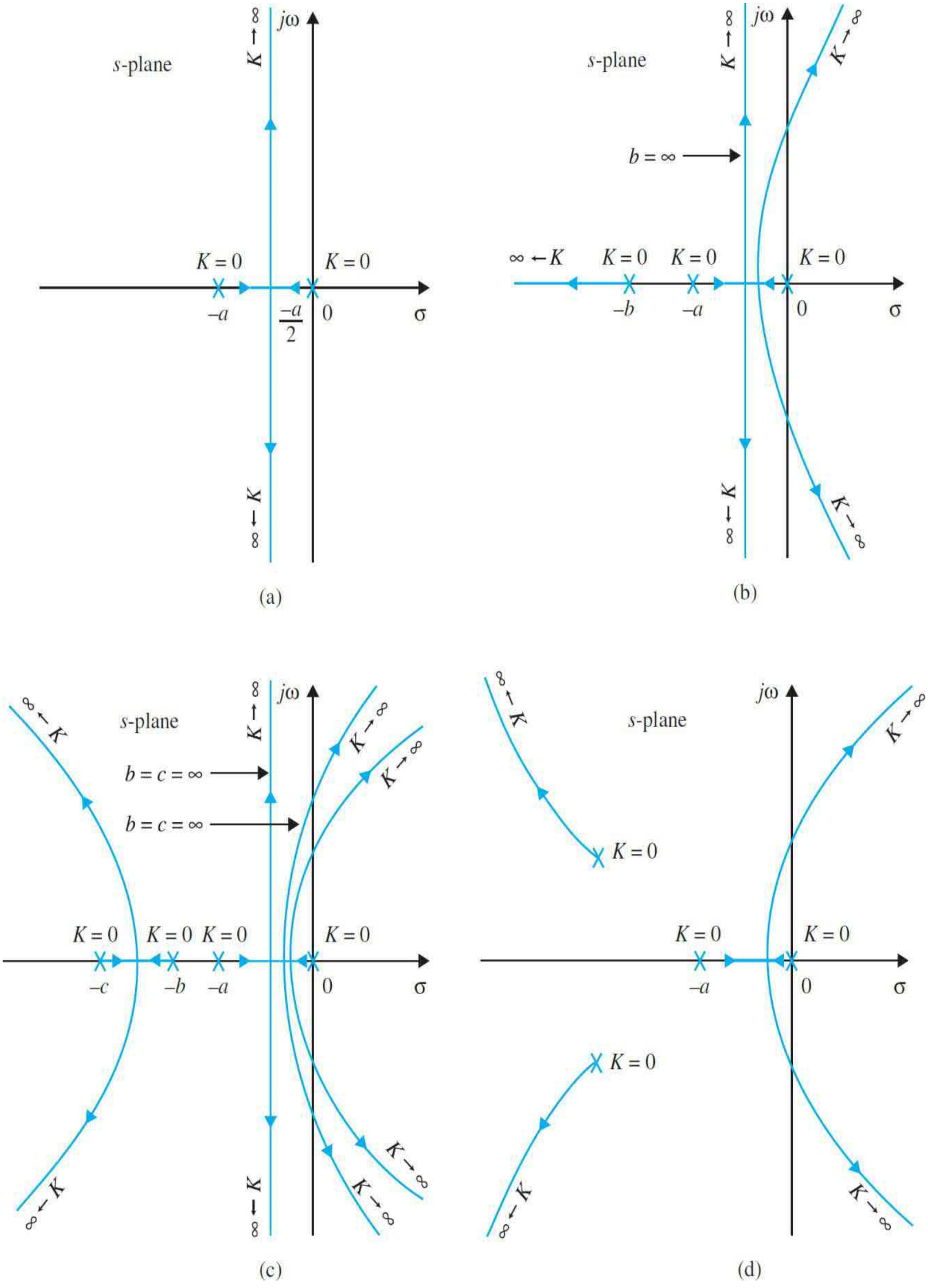

s-plane
jω
K → ∞
K = 0
K = 0
−a
−a/2
0
σ
K → ∞
(a)
s-plane
jω
K → ∞
K → ∞
b = ∞
∞ ← K
K = 0
K = 0
K = 0
−b
−a
0
σ
K → ∞
K → ∞
(b)
s-plane
jω
∞ ← K
K → ∞
K → ∞
K → ∞
b = c = ∞
b = c = ∞
K = 0
K = 0
K = 0
K = 0
−c
−b
−a
0
σ
∞ ← K
K → ∞
K → ∞
K → ∞
(c)
s-plane
jω
∞ ← K
K → ∞
K = 0
K = 0
K = 0
K = 0
−a
0
σ
∞ ← K
K → ∞
(d)

Figure 9-23 Root-locus diagrams that show the effects of adding poles to *G*(*s*)*H*(*s*).

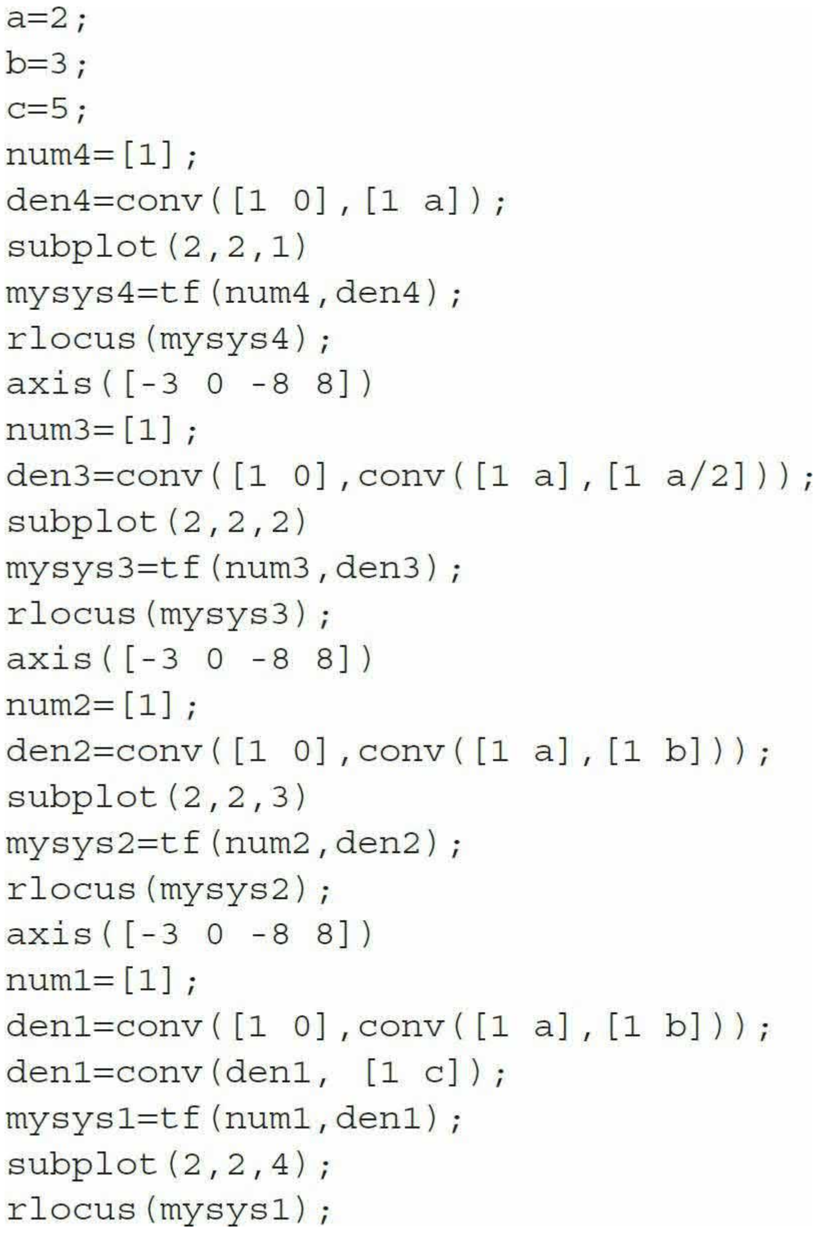
$$G(s)H(s)=\frac{K}{s(s+a)(s+b)} \tag{9-88}$$

Figure 9-23*b* shows that the pole at *s* = –*b* causes the complex part of the root loci to bend toward the right-half *s*-plane. The angles of the asymptotes for the complex roots are changed from ±90° to ±60°. The intersect of the asymptotes is also moved from –*a*/2 to –(*a* + *b*)/2 on the real axis.

If *G*(*s*)*H*(*s*) represents the loop transfer function of a control system, the system with the root loci in Fig. 9-23*b* may become unstable if the value of *K* exceeds the critical value for stability, whereas the system represented by the root loci in Fig. 9-23*a* is always stable for *K* > 0. Figure 9-23*c* shows the root loci when another pole is added to *G*(*s*)*H*(*s*) at *s* = –*c*, *c* > *b*. The system is now of the fourth order, and the two complex root loci are bent farther to the right. The angles of asymptotes of these two complex loci are now ±45°. The stability condition of the fourth-order system is even more acute than that of the third-order system. Figure 9-23*d* illustrates that the addition of a pair of complex-conjugate poles to the transfer function of Eq. (9-87) will result in a similar effect. Therefore, we may draw a general conclusion that the addition of poles to *G*(*s*)*H*(*s*) has the effect of moving the dominant portion of the root loci toward the right-half *s*-plane.

Toolbox 9-4-1

The results for Fig. 9-23 can be obtained by the following MATLAB code:



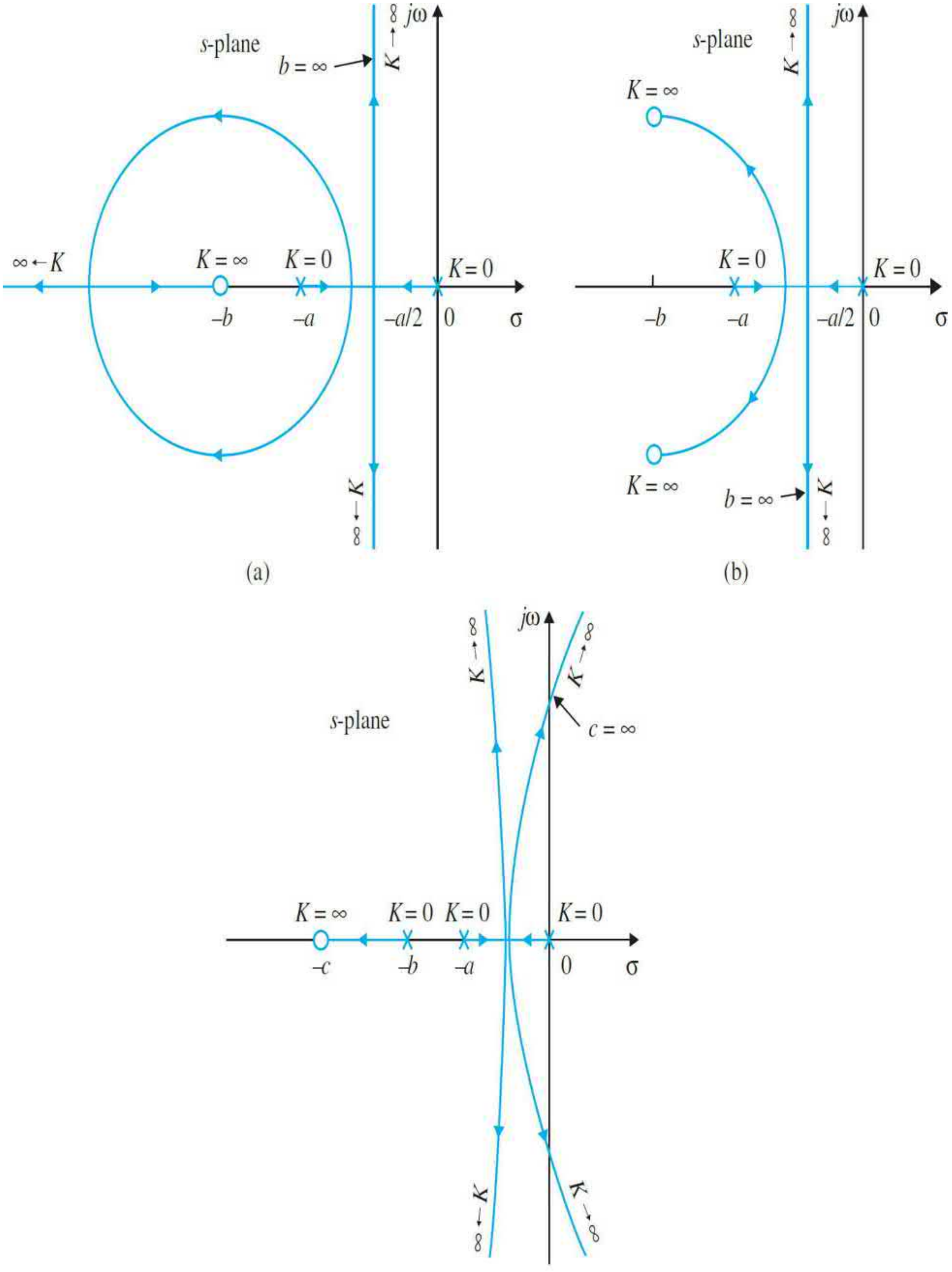
```
a=2;
b=3;
c=5;
num4=[1];
den4=conv([1 0],[1 a]);
subplot(2,2,1)
mysys4=tf(num4,den4);
rlocus(mysys4);
axis([-3 0 -8 8])
num3=[1];
den3=conv([1 0],conv([1 a],[1 a/2]));
subplot(2,2,2)
mysys3=tf(num3,den3);
rlocus(mysys3);
axis([-3 0 -8 8])
num2=[1];
den2=conv([1 0],conv([1 a],[1 b]));
subplot(2,2,3)
mysys2=tf(num2,den2);
rlocus(mysys2);
axis([-3 0 -8 8])
num1=[1];
den1=conv([1 0],conv([1 a],[1 b]));
den1=conv(den1, [1 c]);
mysys1=tf(num1,den1);
subplot(2,2,4);
rlocus(mysys1);
```

9-4-3 Addition of Zeros to $G(s)H(s)$

Adding left-half plane zeros to the function $G(s)H(s)$ generally has the effect of moving and bending the root loci toward the left-half s-plane.

The following example illustrates the effect of adding a zero and zeros to $G(s)H(s)$ on the RL.

EXAMPLE 9-4-2 Figure 9-24*a* shows the RL of the $G(s)H(s)$ in Eq. (9-87) with a zero added at $s = -b(b > a)$. The complex-conjugate part of the RL of the original system is bent toward the left and forms a circle. Thus, if $G(s)H(s)$ is the loop transfer function of a control system, the relative stability of the system is improved by the addition of the zero. Figure 9-24*b* shows that a similar effect will result if a pair of complex-conjugate zeros is added to the function of Eq. (9-87). Figure 9-24*c* shows the RL when a zero at $s = -c$ is added to the transfer function of Eq. (9-88).



(a)

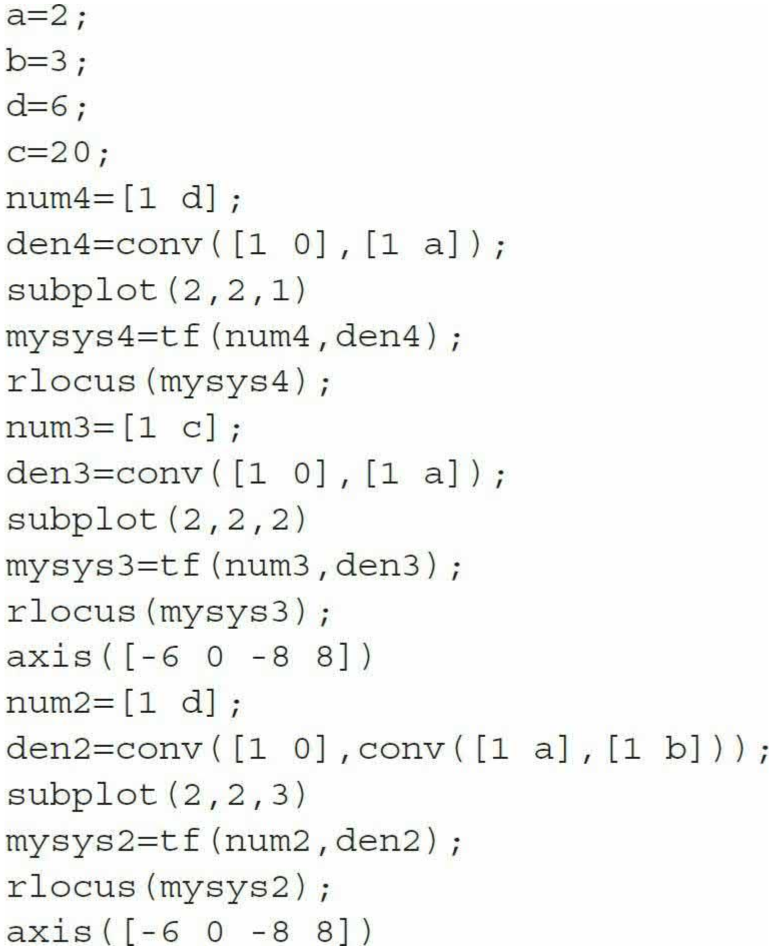
(b)

(c)

Figure 9-24 Root-locus diagrams that show the effects of adding zeros to $G(s)H(s)$.

Toolbox 9-4-2

MATLAB code for Fig. 9-24.



```
a=2;
b=3;
d=6;
c=20;
num4=[1 d];
den4=conv([1 0],[1 a]);
subplot(2,2,1)
mysys4=tf(num4,den4);
rlocus(mysys4);
num3=[1 c];
den3=conv([1 0],[1 a]);
subplot(2,2,2)
mysys3=tf(num3,den3);
rlocus(mysys3);
axis([-6 0 -8 8])
num2=[1 d];
den2=conv([1 0],conv([1 a],[1 b]));
subplot(2,2,3)
mysys2=tf(num2,den2);
rlocus(mysys2);
axis([-6 0 -8 8])
```

EXAMPLE 9-4-3 Consider the equation

$$s^2(s+a)+K(s+b)=0 \qquad (9\text{-}89)$$

Dividing both sides of Eq. (9-89) by the terms that do not contain K, we have the loop transfer function

$$G(s)H(s)=\frac{K(s+b)}{s^2(s+a)} \tag{9-90}$$

It can be shown that the nonzero breakaway points depend on the value of *a* and are

$$s=-\frac{a+3}{4}\pm\frac{1}{4}\sqrt{a^2-10a+9} \tag{9-91}$$

Figure 9-25 shows the RL of Eq. (9-44) with *b* = 1 and several values of *a*. The results are summarized as follows:

Figure 9-25*a*: *a* = 10. Breakaway points: *s* = –2.5 and –40.

Figure 9-25*b*: *a* = 9. The two breakaway points given by Eq. (9-91) converge to one point at *s* = –3. Note the change in the RL when the pole at –*a* is moved from –10 to –9.

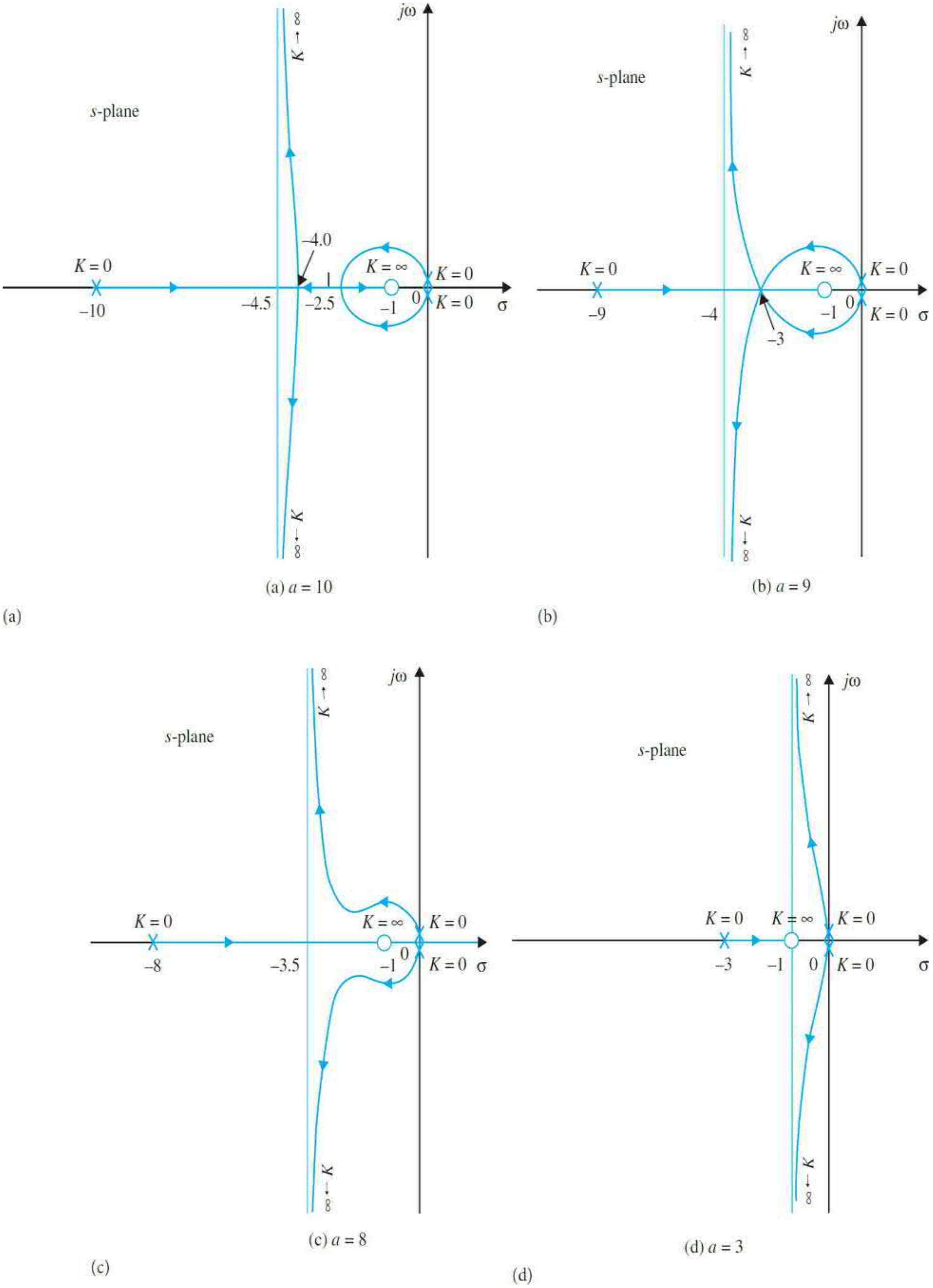
For values of *a* less than 9, the values of *s* as given by Eq. (9-91) no longer satisfy Eq. (9-89), which means that there are no finite, nonzero, breakaway points.

Figure 9-25*c*: *a* = 8. No breakaway point on RL.

As the pole at *s* = –*a* is moved farther to the right along the real axis, the complex portion of the RL is pushed farther toward the right-half plane.

Figure 9-25*d*: *a* = 3.

Figure 9-25*e*: *a* = *b* = 1. The pole at *s* = –*a* and the zero at –*b* cancel each other out, and the RL degenerate into a second-order case and lie entirely on the *jω* -axis.



(a) $a = 10$

(a)

(b) $a = 9$

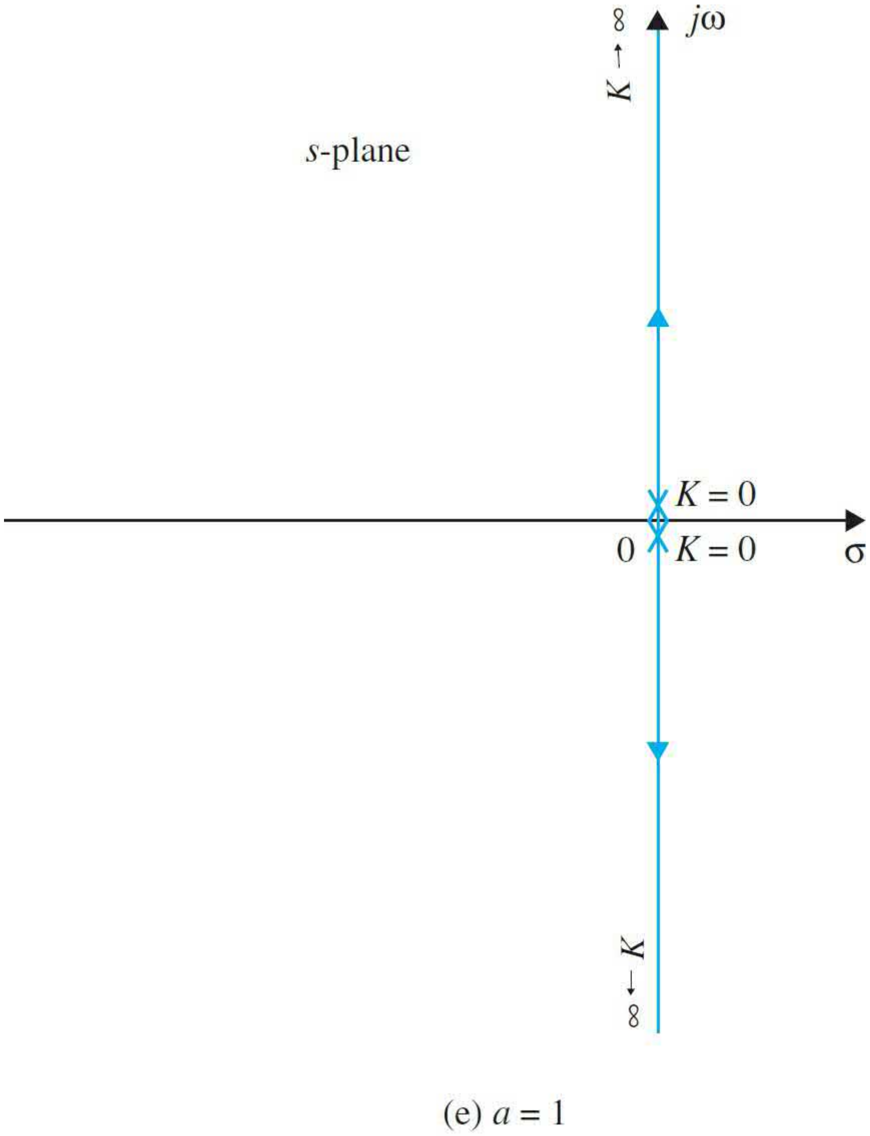
(b)

(c) $a = 8$

(c)

(d) $a = 3$

(d)

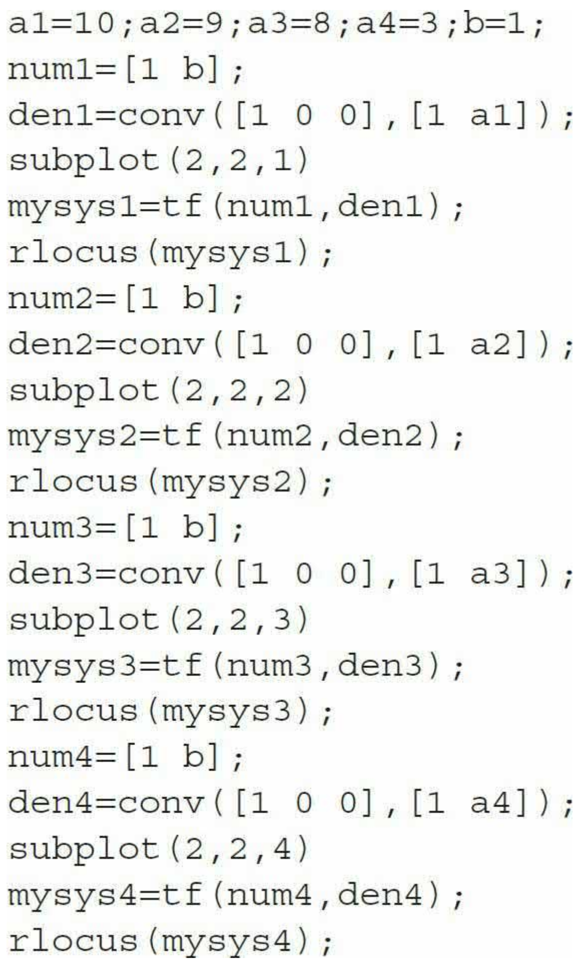


(e)

Figure 9-25 Root-locus diagrams that show the effects of moving a pole of $G(s)H(s)$ · $G(s)H(s) = K(s + 1)/[s^2(s + a)]$.

Toolbox 9-4-3

MATLAB code for Fig. 9-25.



```
a1=10;a2=9;a3=8;a4=3;b=1;
num1=[1 b];
den1=conv([1 0 0],[1 a1]);
subplot(2,2,1)
mysys1=tf(num1,den1);
rlocus(mysys1);
num2=[1 b];
den2=conv([1 0 0],[1 a2]);
subplot(2,2,2)
mysys2=tf(num2,den2);
rlocus(mysys2);
num3=[1 b];
den3=conv([1 0 0],[1 a3]);
subplot(2,2,3)
mysys3=tf(num3,den3);
rlocus(mysys3);
num4=[1 b];
den4=conv([1 0 0],[1 a4]);
subplot(2,2,4)
mysys4=tf(num4,den4);
rlocus(mysys4);
```

EXAMPLE 9-4-4 Consider the equation

$$s(s^2+2s+a)+K(s+2)=0 \tag{9-92}$$

which leads to the equivalent $G(s)H(s)$ as

$$G(s)H(s)=\frac{K(s+2)}{s(s^2+2s+a)} \tag{9-93}$$

The objective is to study the RL for various values of $a(>0)$. The breakaway point equation of the RL is determined as

$$s^3+4s^2+4s+a=0 \tag{9-94}$$

Figure 9-26 shows the RL of Eq. (9-92) under the following conditions.

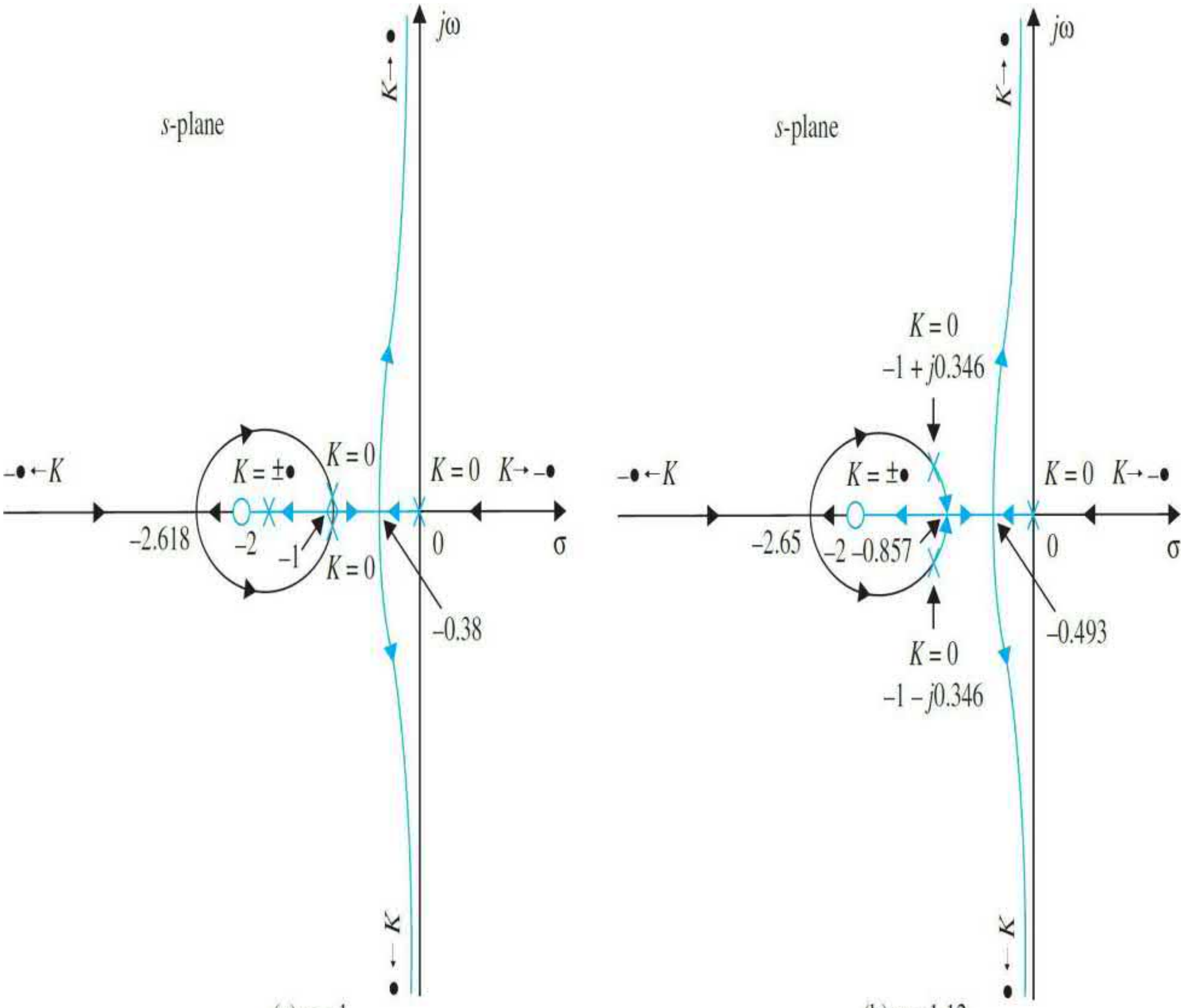
Figure 9-26*a*: $a = 1$. Breakaway points: $s = -0.38, -1.0$, and -2.618, with the last point being on the RL for $K \geq 0$. As the value of a is increased from unity, the two double poles of $G(s)H(s)$ at $s = -1$ will move vertically up and down with the real parts equal to -1. The breakaway points at $s = -0.38$ and $s = -2.618$ will move to the left, whereas the breakaway point at $s = -1$ will move to the right.

Figure 9-26*b*: $a = 1.12$. Breakaway points: $s = -0.493, -0.857$, and -2.65. Because the real parts of the poles and zeros of $G(s)H(s)$ are not affected by the value of a, the intersect of the asymptotes is always at $s = 0$.

Figure 9-26*c*: $a = 1.185$. Breakaway points: $s = -0.667, -0.667$, and -2.667. The two breakaway points of the RL that lie between $s = 0$ and -1 converge to a point.

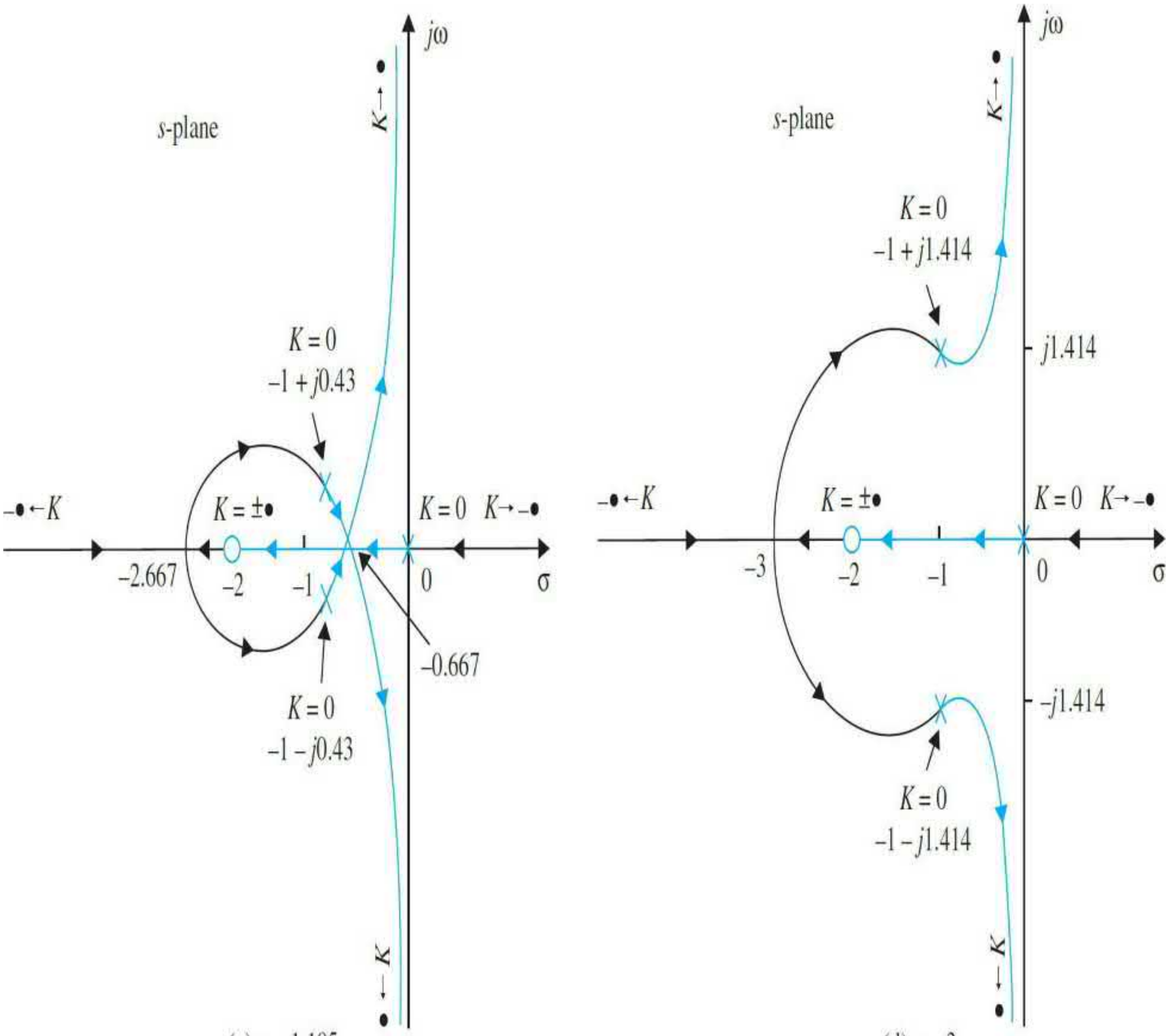
Figure 9-26*d*: $a = 3$. Breakaway point: $s = -3$. When a is greater than 1.185, Eq. (9-94) yields only one solution for the breakaway point.

The reader may investigate the difference between the RL in Fig. 9-26*c* and *d* and fill in the evolution of the loci when the value of a is gradually changed from 1.185 to 3 and beyond. ▲


jω
s-plane
K→
K = 0
K = ±•
−•←K
K = 0
K→ −•
−2.618
−2
−1
0
σ
K = 0
−0.38
K →
jω
s-plane
K→
K = 0
−1 + j0.346
−•←K
K = ±•
K = 0
K→ −•
−2.65
−2
−0.857
0
σ
−0.493
K = 0
−1 − j0.346
K →

(a) $a = 1$

(b) $a = 1.12$



(c) $a = 1.185$

(d) $a = 3$

Figure 9-26 Root-locus diagrams that show the effects of moving a pole of $G(s)H(s) = K(s + 2)/[s(s^2 + 2s + a)]$.

9-5 ROOT CONTOURS: MULTIPLE-PARAMETER VARIATION

The root-locus technique discussed thus far is limited to only one variable parameter in K. In many control-systems problems, the effects of varying several parameters should be investigated. For example, when designing a controller that is represented by a transfer function with poles and zeros, it would be useful to investigate the effects on the characteristic equation roots when these poles and zeros take on various values. In Sec. 9-4, the root loci of equations with two variable parameters are studied by fixing one parameter and assigning different values to the other. In this section, the multiparameter problem is investigated through a more systematic method of embedding. When more than one parameter varies continuously from –∞ to ∞, the root loci are referred to as the ***root contours (RC)***. It will be shown that the root contours still possess the same properties as the single-parameter root loci, so that the methods of construction discussed thus far are all applicable.

The principle of root contour can be described by considering the equation

$$P(s)+K_1Q_1(s)+K_2Q_2(s)=0 \tag{9-95}$$

where K_1 and K_2 are the variable parameters, and P(s), Q_1(s), and Q_2(s) are polynomials of s. The first step involves setting the value of one of the parameters to zero. Let us set K_2 to zero. Then, Eq. (9-95) becomes

$$P(s)+K_1Q_1(s)=0 \tag{9-96}$$

which now has only one variable parameter in K_1. The root loci of Eq. (9-96) may be determined by dividing both sides of the equation by $P(s)$. Thus,

$$1+\frac{K_1Q_1(s)}{P(s)}=0 \tag{9-97}$$

Equation (9-97) is of the form of $1 + K_1G_1(s)H_1(s) = 0$, so we can construct the RL of the equation based on the pole-zero configuration of $G_1(s)H_1(s)$. Next, we restore the value of K_2, while considering the value of K_1 fixed, and divide both sides of Eq. (9-95) by the terms that do not contain K_2. We have

$$1+\frac{K_2Q_2(s)}{P(s)+K_1Q_1(s)}=0 \tag{9-98}$$

which is of the form of $1 + K_2G_2(s)H_2(s) = 0$. The root contours of Eq. (9-95) when K_2 varies (while K_1 is fixed) are constructed based on the pole-zero configuration of

$$G_2(s)H_2(s) = \frac{Q_2(s)}{P(s)+K_1Q_1(s)} \tag{9-99}$$

It is important to note that the poles of $G_2(s)H_2(s)$ are identical to the roots of Eq. (9-96). Thus, the root contours of Eq. (9-95) when K_2 varies must all start ($K_2 = 0$) at the points that lie on the root loci of Eq. (9-96). This is the reason why one root-contour problem is considered to be embedded in another. The same procedure may be extended to more than two variable parameters. The following examples illustrate the construction of RCs when multiparameter-variation situations exist.

EXAMPLE 9-5-1 Consider the equation

$$s^3 + K_2s^2 + K_1s + K_1 = 0 \tag{9-100}$$

where K_1 and K_2 are the variable parameters, which vary from 0 to ∞.

As the first step, we let $K_2 = 0$, and Eq. (9-100) becomes

$$s^3 + K_1s + K_1 = 0 \tag{9-101}$$

Dividing both sides of the last equation by s^3, which is the term that does not contain K_1, we have

$$1 + \frac{K_1(s+1)}{s^3} = 0 \tag{9-102}$$

The root contours of Eq. (9-101) are drawn based on the pole-zero configuration of

$$G_1(s)H_1(s) = \frac{s+1}{s^3} \tag{9-103}$$

as shown in Fig. 9-27*a*. Next, we let K_2 vary between 0 and ∞ while holding K_1 at a constant nonzero value. Dividing both sides of Eq. (9-100) by the terms that do not contain K_2, we have

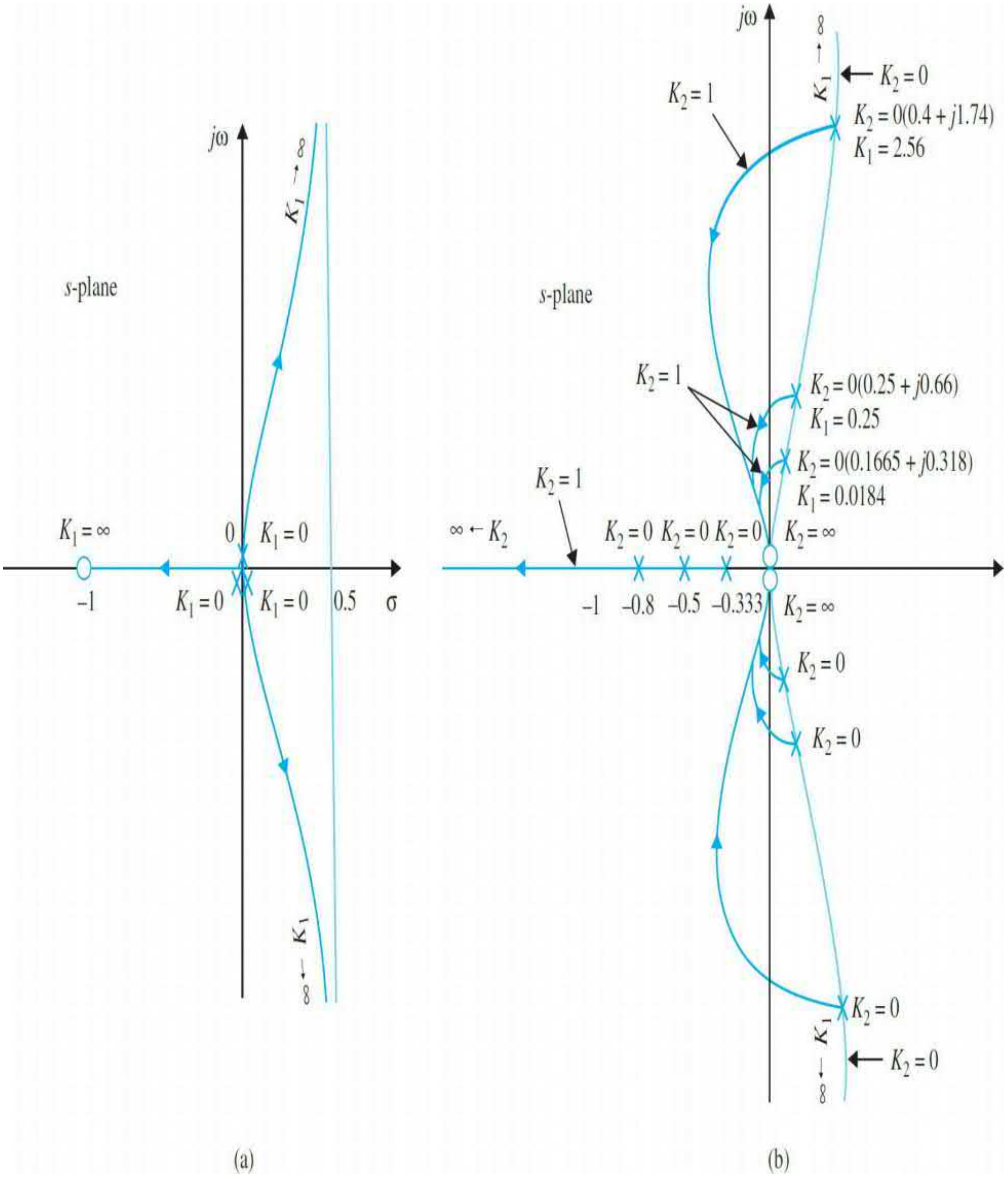


Figure 9-27 Root contours of $s^2 + K_2 s^2 + K_1 s + K_1 = 0$. (*a*) $K_2 = 0$. (*b*) K_2 varies and K_1 is a constant.

$$1+\frac{K_2 s^2}{s^3+K_1 s+K_1}=0 \tag{9-104}$$

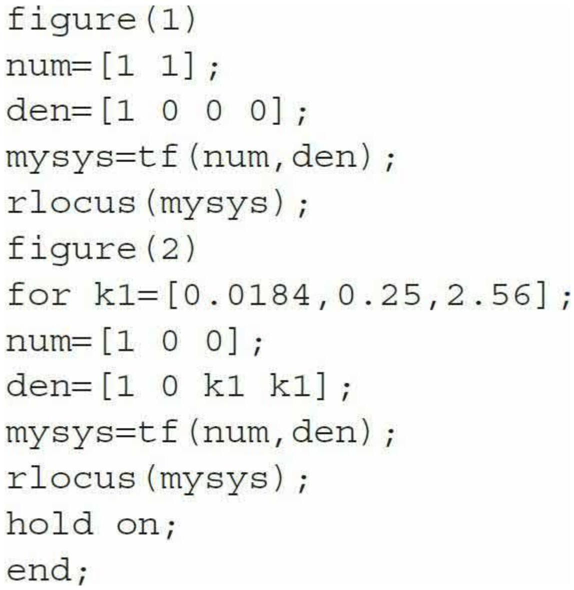
Thus, the root contours of Eq. (9-100) when K_2 varies may be drawn from the pole-zero configuration of

$$G_2(s)H_2(s)=\frac{s^2}{s^3+K_1 s+K_1} \tag{9-105}$$

The zeros of $G_2(s)H_2(s)$ are at $s = 0, 0$; but the poles are at the zeros of $1 + K_1G_1(s)H_1(s)$, which are found on the RL of Fig. 9-27*a*. Thus, for fixed K_1, the RC when K_2 varies must all emanate from the root contours of Eq. 9-27*a*. Figure 9-27*b* shows the root contours of Eq. (9-100) when K_2 varies from 0 to ∞, for $K_1 = 0.0184, 0.25$, and 2.56.

Toolbox 9-5-1

MATLAB code for Fig. 9-27.



```
figure(1)
num=[1 1];
den=[1 0 0 0];
mysys=tf(num,den);
rlocus(mysys);
figure(2)
for k1=[0.0184,0.25,2.56];
num=[1 0 0];
den=[1 0 k1 k1];
mysys=tf(num,den);
rlocus(mysys);
hold on;
end;
```

EXAMPLE 9-5-2 Consider the loop transfer function

$$G(s)H(s)=\frac{K}{s(1+Ts)(s^2+2s+2)} \tag{9-106}$$

of a closed-loop control system. It is desired to construct the root contours of the characteristic equation with *K* and *T* as variable parameters. The characteristic equation of the system is

$$s(1+Ts)(s^2+2s+2)+K=0 \tag{9-107}$$

First, we set the value of *T* to zero. The characteristic equation becomes

$$s(s^2+2s+2)+K=0 \tag{9-108}$$

The root contours of this equation when *K* varies are drawn based on the pole-zero configuration of

$$G_1(s)H_1(s)=\frac{1}{s(s^2+2s+2)} \tag{9-109}$$

as shown in Fig. 9-28*a*. Next, we let *K* be fixed, and consider that *T* is the variable parameter.

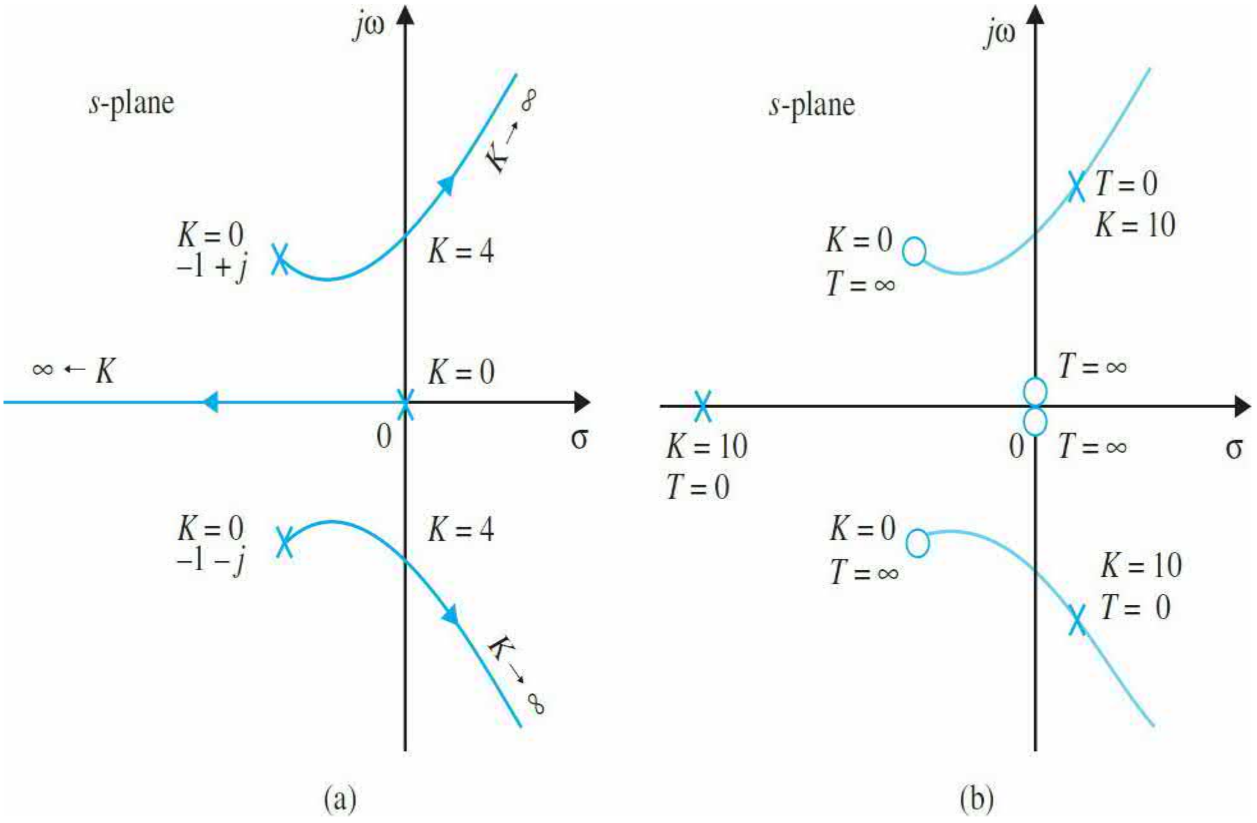


Figure 9-28 (*a*) RL for $s(s^2+2s+2)+K=0$. (*b*) Pole-zero configuration of $G_2(s)H_2(s) = Ts^2(s^2 + 2s + 2)/[s(s^2 + 2s +2) + K]$.

Dividing both sides of Eq. (9-62) by the terms that do not contain *T*, we get

$$1+TG_2(s)H_2(s)=1+\frac{Ts^2(s^2+2s+2)}{s(s^2+2s+2)+K}=0 \tag{9-110}$$

The root contours when *T* varies are constructed based on the pole-zero configuration of $G_2(s)H_2(s)$. When $T = 0$, the points on the root contours are at the poles of $G_2(s)H_2(s)$, which are on the root contours of Eq. (9-108). When $T = \infty$, the roots of Eq. (9-107) are at the zeros of $G_2(s)H_2(s)$, which are at $s = 0$, 0, −1 +*j*, and −1 +*j*. Figure 9-28*b* shows the pole-zero configuration of $G_2(s)H_2(s)$ for $K = 10$. Notice that $G_2(s)H_2(s)$ has three finite poles and four finite zeros. The root contours for Eq. (9-107) when *T* varies are shown in Figs. 9-29, 9-30, and 9-31 for three different values of *K*.

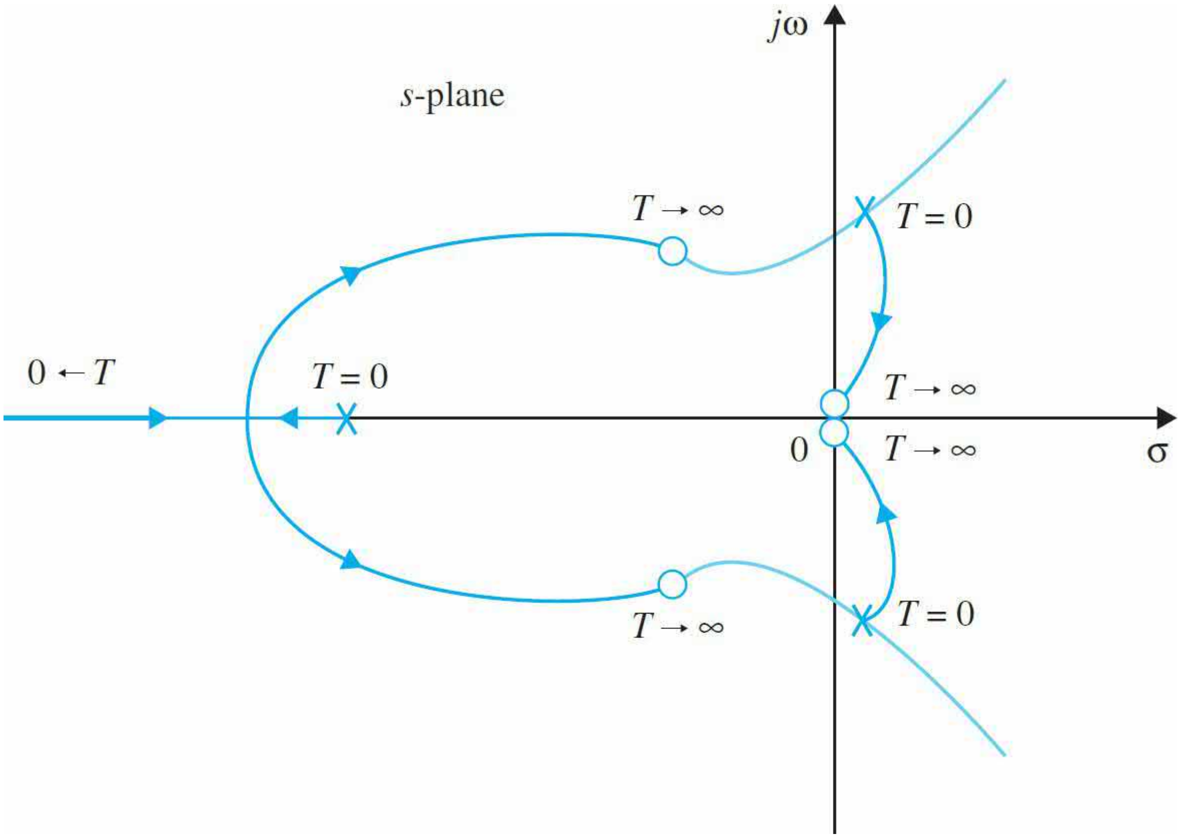


Figure 9-29 Root contours for $s(1+\mathrm{T}s)(s^2+2s+2)+K=0$. $K > 4$.

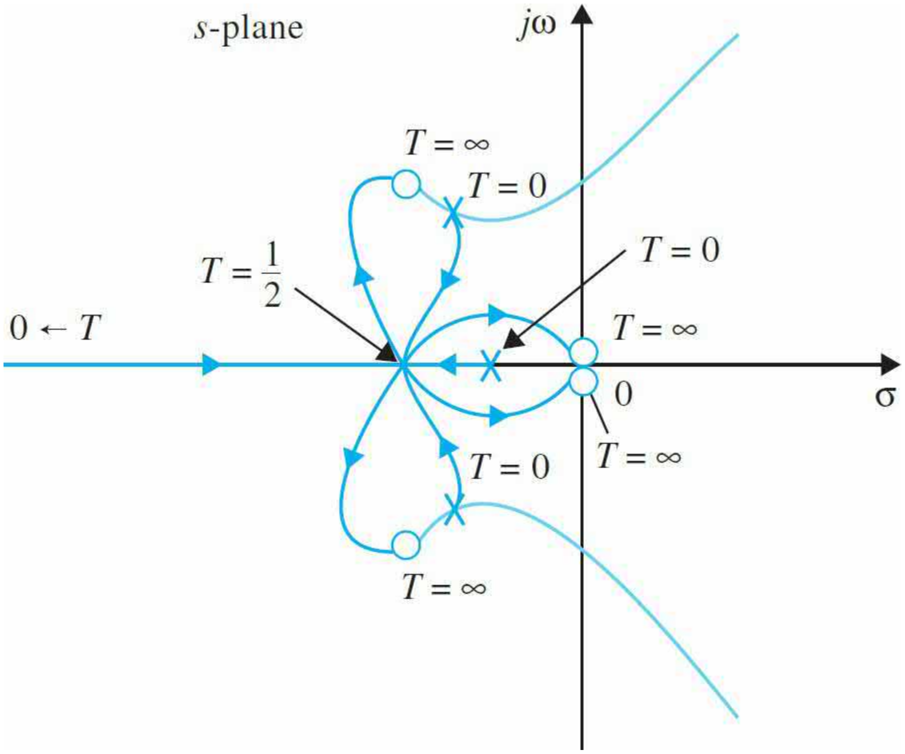


Figure 9-30 Root contours for $s(1+Ts)(s^2+2s+2)+K=0$. $K = 0.5$.

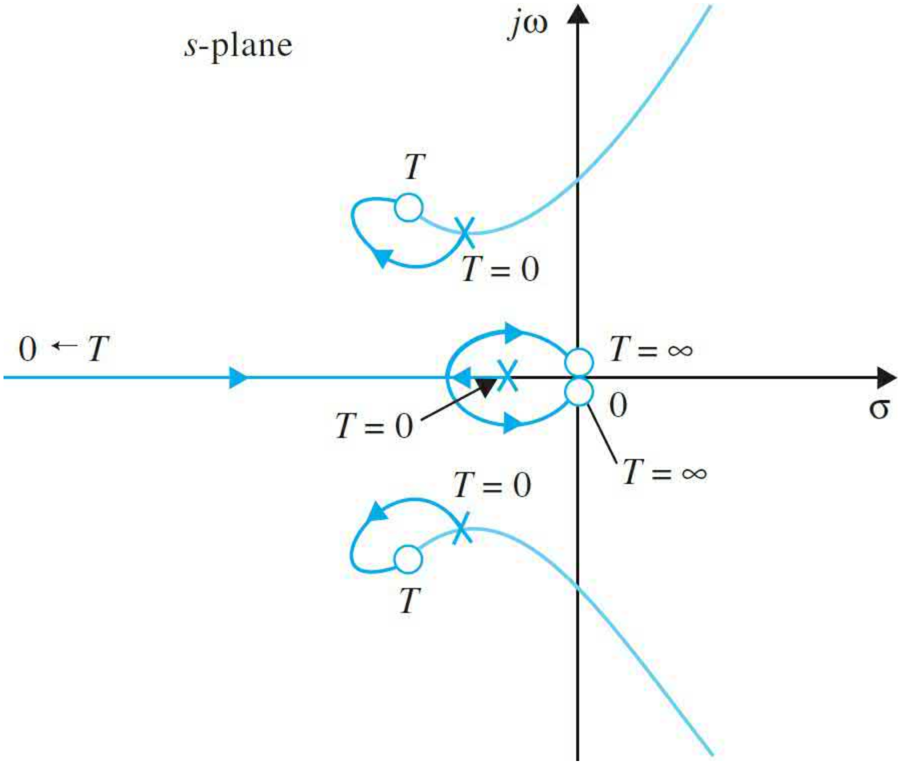
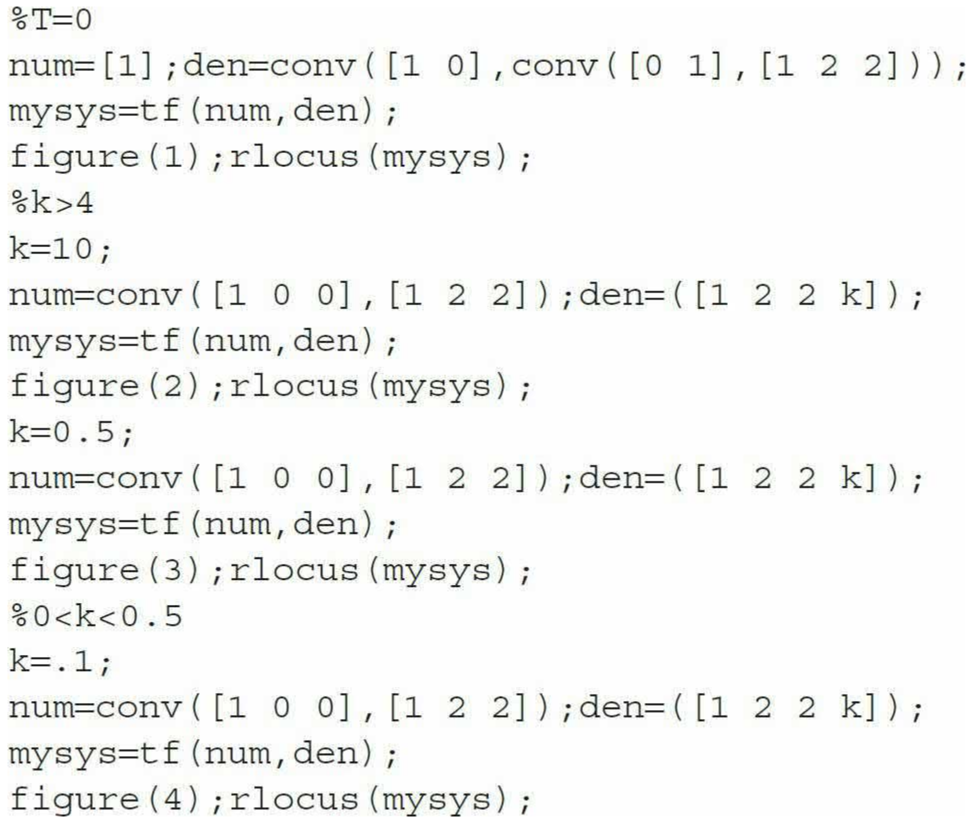


Figure 9-31 Root contours for $s(1+Ts)(s^2+2s+2)+K=0$. $0 < K < 0.5$.

The root contours in Fig. 9-30 show that when $K = 0.5$ and $T = 0.5$, the characteristic equation in Eq. (9-107) has a quadruple root at $s = -1$.

Toolbox 9-5-2

MATLAB code for Example 9-5-2.



```
%T=0
num=[1];den=conv([1 0],conv([0 1],[1 2 2]));
mysys=tf(num,den);
figure(1);rlocus(mysys);
%k>4
k=10;
num=conv([1 0 0],[1 2 2]);den=([1 2 2 k]);
mysys=tf(num,den);
figure(2);rlocus(mysys);
k=0.5;
num=conv([1 0 0],[1 2 2]);den=([1 2 2 k]);
mysys=tf(num,den);
figure(3);rlocus(mysys);
%0<k<0.5
k=.1;
num=conv([1 0 0],[1 2 2]);den=([1 2 2 k]);
mysys=tf(num,den);
figure(4);rlocus(mysys);
```

EXAMPLE 9-5-3 As an example to illustrate the effect of the variation of a zero of $G(s)H(s)$, consider the function

$$G(s)H(s)=\frac{K(1+Ts)}{s(s+1)(s+2)} \tag{9-111}$$

The characteristic equation is

$$s(s+1)(s+2)+K(1+Ts)=0 \tag{9-112}$$

Let us first set T to zero and consider the effect of varying K. Equation (9-112) becomes

$$s(s+1)(s+2)+K=0 \tag{9-113}$$

This leads to

$$G_1(s)H_1(s) = \frac{1}{s(s+1)(s+2)} \qquad (9\text{-}114)$$

The root contours of Eq. (9-113) are drawn based on the pole-zero configuration of Eq. (9-114), and are shown in Fig. 9-32.

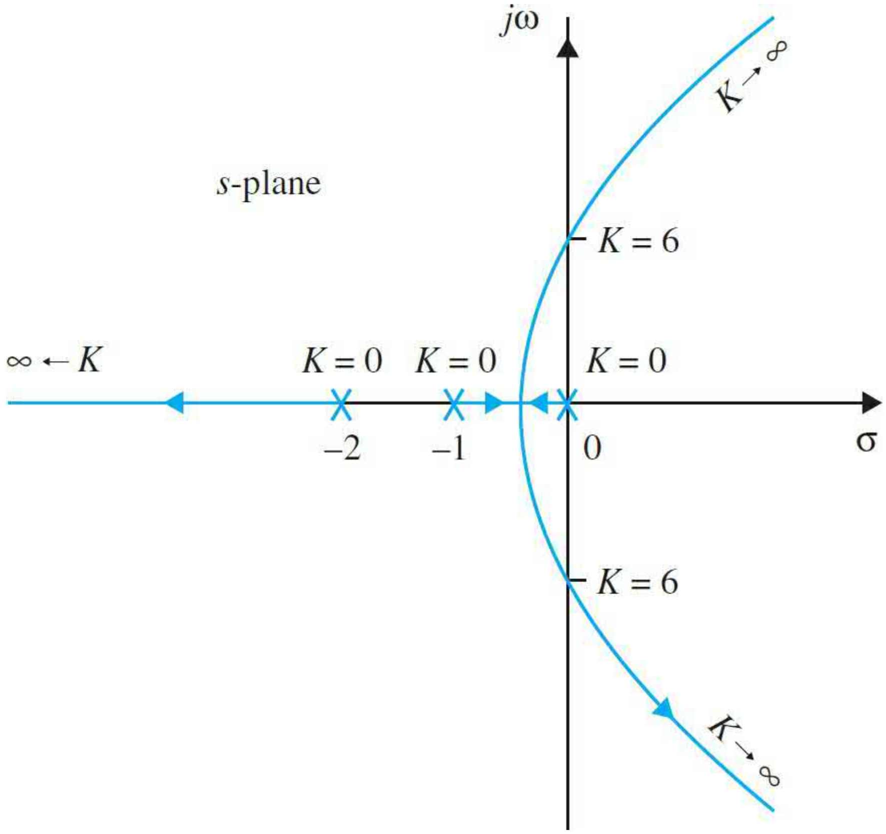


Figure 9-32 Root loci for *s*(*s* + 1)(*s* + 2) + *K* = 0.

When the *K* is fixed and nonzero, we divide both sides of Eq. (9-115) by the terms that do not contain *T*, and we get

$$1 + TG_2(s)H_2(s) = 1 + \frac{TKs}{s(s+1)(s+2)+K} = 0 \qquad (9\text{-}115)$$

The points that correspond to $T = 0$ on the root contours are at the poles of $G_2(s)H_2(s)$ or the zeros of $s(s + 1)(s + 2) + K$, whose root contours are sketched as shown in Fig. 9-32, where *K* varies. If we choose $K = 20$ just as an illustration, the pole-zero configuration of $G_2(s)H_2(s)$ is shown in Fig. 9-33. The root contours of Eq. (9-112) for $0 \le T < \infty$ are shown in Fig. 9-34 for three different values of *K*.

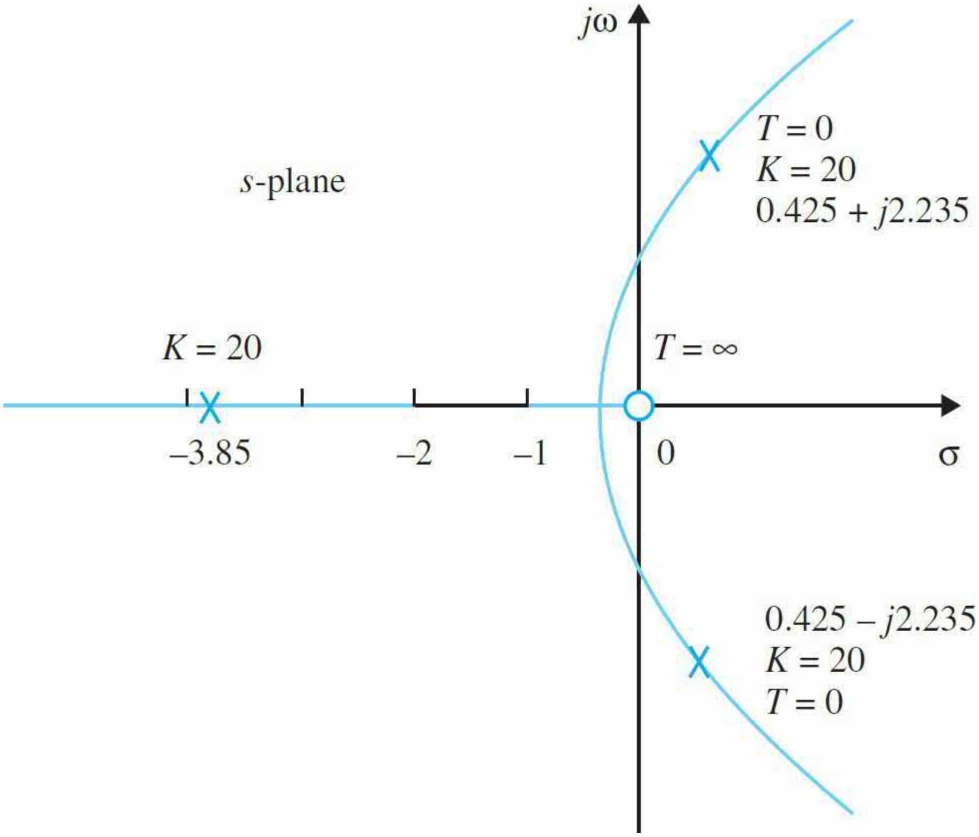


Figure 9-33 Pole-zero configuration of $G_2(s)H_2(s) = KS\ [s(s+1)(s+2)+K\ ]$. $K = 20$.

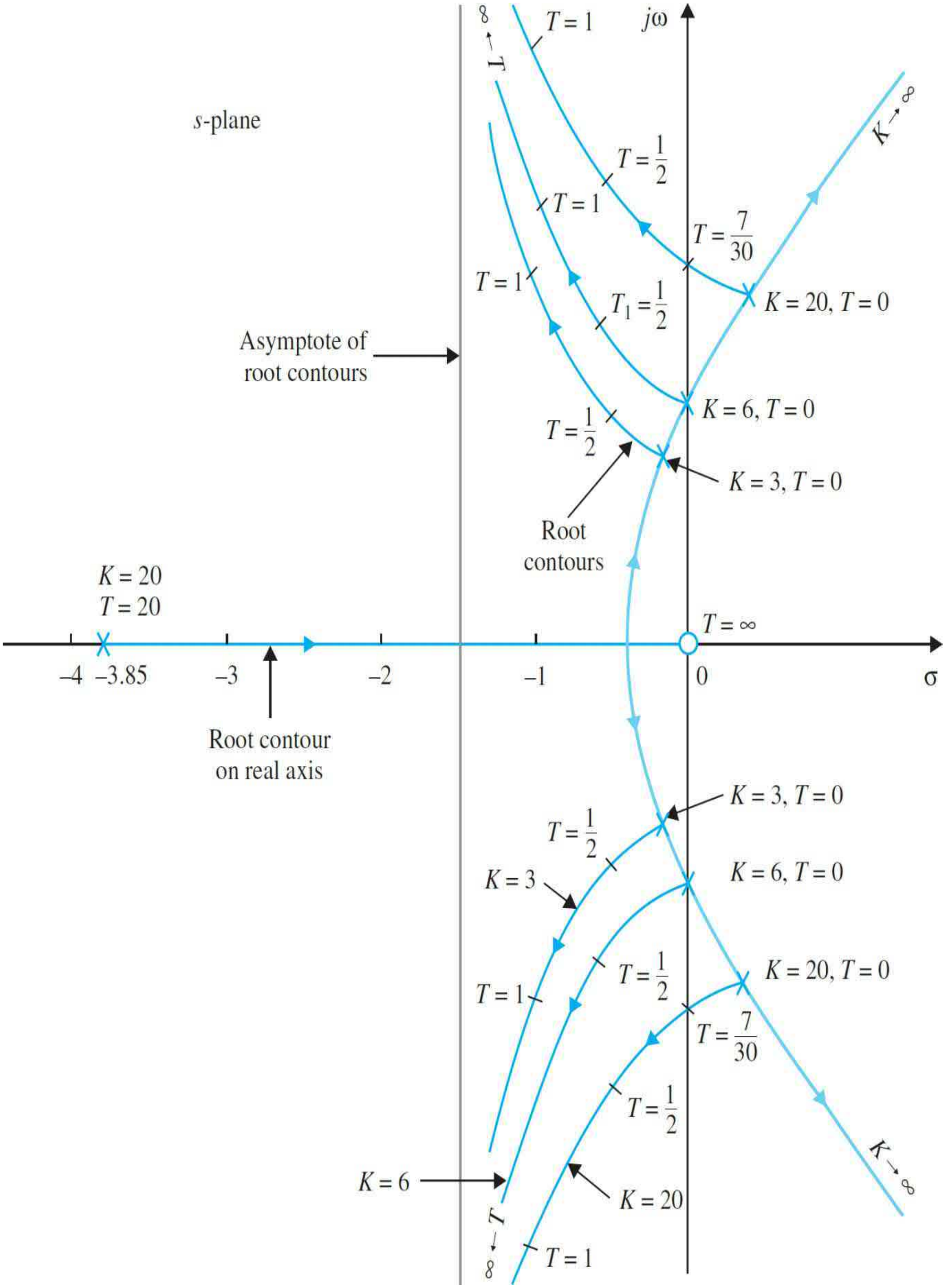
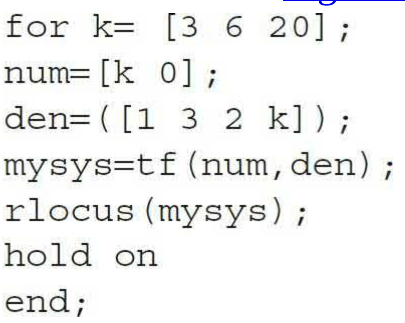

s-plane
jω
σ
T → ∞
T = 1
T = $\frac{1}{2}$
T = 1
T = $\frac{7}{30}$
T = 1
T_1 = $\frac{1}{2}$
K = 20, T = 0
K → ∞
Asymptote of root contours
K = 6, T = 0
T = $\frac{1}{2}$
K = 3, T = 0
Root contours
K = 20
T = 20
T = ∞
–4
–3.85
–3
–2
–1
0
Root contour on real axis
K = 3, T = 0
T = $\frac{1}{2}$
K = 3
K = 6, T = 0
T = $\frac{1}{2}$
K = 20, T = 0
T = 1
T = $\frac{7}{30}$
T = $\frac{1}{2}$
K = 6
K = 20
K → ∞
T → ∞
T = 1

Figure 9-34 Root contours of $s(s + 1)(s + 2) + K + KTs = 0$.

Because $G_2(s)H_2(s)$ has three poles and one zero, the angles of the asymptotes of the root contours when T varies are at 90° and –90°. We can show that the intersection of the asymptotes is always at $s = 1.5$. This is because the sum of the poles of $G_2(s)H_2(s)$, which is given by the negative of the coefficient of the s^2 term in the denominator polynomial of Eq. (9-70), is 3; the sum of the zeros of $G_2(s)H_2(s)$ is 0; and $n – m$ in Eq. (9-30) is 2.

Toolbox 9-5-3

MATLAB code for Fig. 9-25.



```
for k= [3 6 20];
num=[k 0];
den=([1 3 2 k]);
mysys=tf(num,den);
rlocus(mysys);
hold on
end;
```

The root contours in Fig. 9-17 show that adding a zero to the loop transfer function generally improves the relative stability of the closed-loop system by moving the characteristic equation roots toward the left in the s-plane. As shown in Fig. 9-17, for $K = 20$, the system is stabilized for all values of T greater than 0.2333. However, the largest relative damping ratio that the system can have by increasing T is only approximately 30 percent.

9-6 MATLAB TOOLS

Apart from the MATLAB toolboxes appearing in this chapter, this chapter does not contain any software because of its focus on theoretical development. In Chap. 11, when we address control system design, we will introduce the MATLAB SISO design tool that can allow you solve root-locus problems with a great deal of ease.

9-7 SUMMARY

In this chapter, we introduced the root-locus technique for linear continuous data control systems. The technique represents a graphical method of investigating the roots of the characteristic equation of a linear time-invariant system when one or more parameters vary. In Chap. 11 the root-locus method will be used heavily for the design of control systems. However, keep in mind that, although the characteristic equation roots give exact indications on the absolute stability of linear SISO systems, they give only qualitative information on the relative stability, since the zeros of the closed-loop transfer function, if any, play an important role on the dynamic performance of the system.

The root-locus technique can also be applied to discrete-data systems with the characteristic equation expressed in the z-transform. As shown in App. H, the properties and construction of the root loci in the z-plane are essentially the same as those of the continuous-data systems in the s-plane, except that the interpretation of the root location to system performance must be made with respect to the unit circle $|z| = 1$ and the significance of the regions in the z-plane.

The majority of the material in this chapter is designed to provide the basics of constructing the root loci. Computer programs, such as the MATLAB toolboxes used throughout this chapter, can be used to plot the root loci and provide details of the plot. The final section of Chap. 11 deals with the root-locus tools of MATLAB. However, the authors believe that a computer program can be used only as a tool, and the intelligent investigator should have a thorough understanding of the fundamentals of the subject.

The root-locus technique can also be applied to linear systems with pure time delay in the system loop. The subject is not treated here, since systems with pure time delays are more easily treated with the frequency-domain methods discussed in Chap. 10.

REFERENCES

General Subjects

1. W. R. Evans, “Graphical Analysis of Control Systems,” *Trans. AIEE,*

Vol. 67, pp. 548–551, 1948.

2. W. R. Evans, “Control System Synthesis by Root Locus Method,” *Trans. AIEE*, Vol. 69, pp. 66–69, 1950.

3. W. R. Evans, *Control System Dynamics*, McGraw-Hill Book Company, New York, 1954.

Construction and Properties of Root Loci

4. C. C. MacDuff, *Theory of Equations*, John Wiley & Sons, New York, pp. 29–104, 1954.

5. C. S. Lorens and R. C. Titsworth, “Properties of Root Locus Asymptotes,” *IRE Trans. Automatic Control*, AC-5, pp. 71–72, Jan. 1960.

6. C. A. Stapleton, “On Root Locus Breakaway Points,” *IRE Trans. Automatic Control*, Vol. AC-7, pp. 88–89, April 1962.

7. M. J. Remec, “Saddle-Points of a Complete Root Locus and an Algorithm for Their Easy Location in the Complex Frequency Plane,” *Proc. Natl. Electronics Conf.*, Vol. 21, pp. 605–608, 1965.

8. C. F. Chen, “A New Rule for Finding Breaking Points of Root Loci Involving Complex Roots,” *IEEE Trans. Automatic Control*, AC-10, pp. 373–374, July 1965.

9. V. Krishran, “Semi-Analytic Approach to Root Locus,” *IEEE Trans. Automatic Control*, Vol. AC-11, pp. 102–108, Jan. 1966.

10. R. H. Labounty and C. H. Houpis, “Root Locus Analysis of a High-Grain Linear System with Variable Coefficients; Application of Horowitz’s Method,” *IEEE Trans. Automatic Control*, Vol. AC-11, pp. 255–263, April 1966.

11. A. Fregosi and J. Feinstein, “Some Exclusive Properties of the Negative Root Locus,” *IEEE Trans. Automatic Control*, Vol. AC-14, pp. 304–305, June 1969.

Analytical Representation of Root Loci

12. G. A. Bendrikov and K. F. Teodorchik, “The Analytic Theory of Constructing Root Loci,” *Automation and Remote Control*, pp. 340–344, March 1959.

13. K. Steiglitz, “Analytical Approach to Root Loci,” *IRE Trans. Automatic Control*, Vol. AC-6, pp. 326–332, Sept. 1961.

14. C. Wojcik, “Analytical Representation of Root Locus,” *Trans. ASME*, J. Basic Engineering, Ser. D. Vol. 86, March 1964.

15. C. S. Chang, “An Analytical Method for Obtaining the Root Locus with Positive and Negative Gain,” *IEEE Trans. Automatic Control*, Vol. AC-10, pp. 92–94, January 1965.

16. B. P. Bhattacharyya, “Root Locus Equations of the Fourth Degree,” *Interna. J. Control*, Vol. 1, No. 6, pp. 533–556, 1965.

Root Sensitivity

17. J. G. Truxal and M. Horowitz, “Sensitivity Consideration in Active Network Synthesis,” *Proc. Second Midwest Symposium on Circuit Theory*, East Lansing, MI, 1956.

18. R. Y. Huang, “The Sensitivity of the Poles of Linear Closed-Loop Systems,” *IEEE Trans. Appl. Ind.*, Vol. 77, Part 2, pp. 182–187, September 1958.

19. H. Ur, “Root Locus Properties and Sensitivity Relations in Control Systems,” *IRE Trans. Automatic Control*, Vol. AC-5, pp. 58–65, January 1960.

PROBLEMS

9-1. Find the angles of the asymptotes and the intersect of the asymptotes of the root loci of the following equations when K varies from $-\infty$ to ∞.

(a) $s^4+4s^3+4s^2+(K+8)s+K=0$

(b) $s^3+5s^2+(K+1)s+K=0$

(c) $s^2+K(s^3+3s^2+2s+8)=0$

(d) $s^3+2s^2+3s+K(s^2-1)(s+3)=0$

(e) $s^5+2s^4+3s^3+K(s^2+3s+5)=0$

(f) $s^4+2s^2+10+K(s+5)=0$

9-2. Use MATLAB to solve Prob. 9-1.

9-3. Show that the asymptotes angles are

$$\begin{cases} \theta_i = \dfrac{(2i+1)}{|n-m|}\times 180^\circ \ K>0 \\ \theta_i = \dfrac{(2i)}{|n-m|}\times 180^\circ \ K<0 \end{cases}$$

9-4. Prove that the asymptotes center is

$$\sigma_1 = \frac{\sum \text{finite poles of } G(s)H(s) - \sum \text{finite zeros of } G(s)H(s)}{n-m}$$

9-5. Plot the asymptotes for $K > 0$ and $K < 0$ for

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

9-6. For the loop transfer functions that follow, find the angle of departure or arrival of the root loci at the designated pole or zero.

(a) $$G(s)H(s) = \frac{Ks}{(s+1)(s^2+1)}$$

Angle of arrival ($K < 0$) and angle of departure ($K > 0$) at $s = j$.

(a) $G(s)H(s)=\dfrac{Ks}{(s-1)(s^2+1)}$

Angle of arrival ($K < 0$) and angle of departure ($K > 0$) at $s = j$.

(a) $G(s)H(s)=\dfrac{K}{s(s+2)(s^2+2s+2)}$

Angle of departure ($K > 0$) at $s = -1 + j$.

(a) $G(s)H(s)=\dfrac{K}{s^2(s^2+2s+2)}$

Angle of departure ($K > 0$) at $s = -1 + j$.

(a) $G(s)H(s)=\dfrac{K(s^2+2s+2)}{s^2(s+2)(s+3)}$

Angle of arrival ($K > 0$) at $s = -1 + j$.

9-7. Prove that

(a) The departure angle of the root locus from a complex pole is ω_D = 180° – argGH', where argGH' is the phase angle of GH at the complex pole, ignoring the effect of that pole.

(b) The arrival angle of the root locus at the complex zero is ω_D = 180° – argGH'', where argGH'' is the phase angle of GH at the complex zero, ignoring the contribution of that particular zero.

9-8. Find the angles of departure and arrival for all complex poles and zeros of the open-loop transfer function of

$$G(s)H(s)=\frac{K(s^2+2s+2)}{s(s^2+4)}K>0$$

9-9. Mark the $K = 0$ and $K = \pm\infty$ points and the RL and complementary root locus (CRL) on the real axis for the pole-zero configurations shown in Fig. 9P-9. Add arrows on the root loci on the real axis in the direction of increasing *K*.

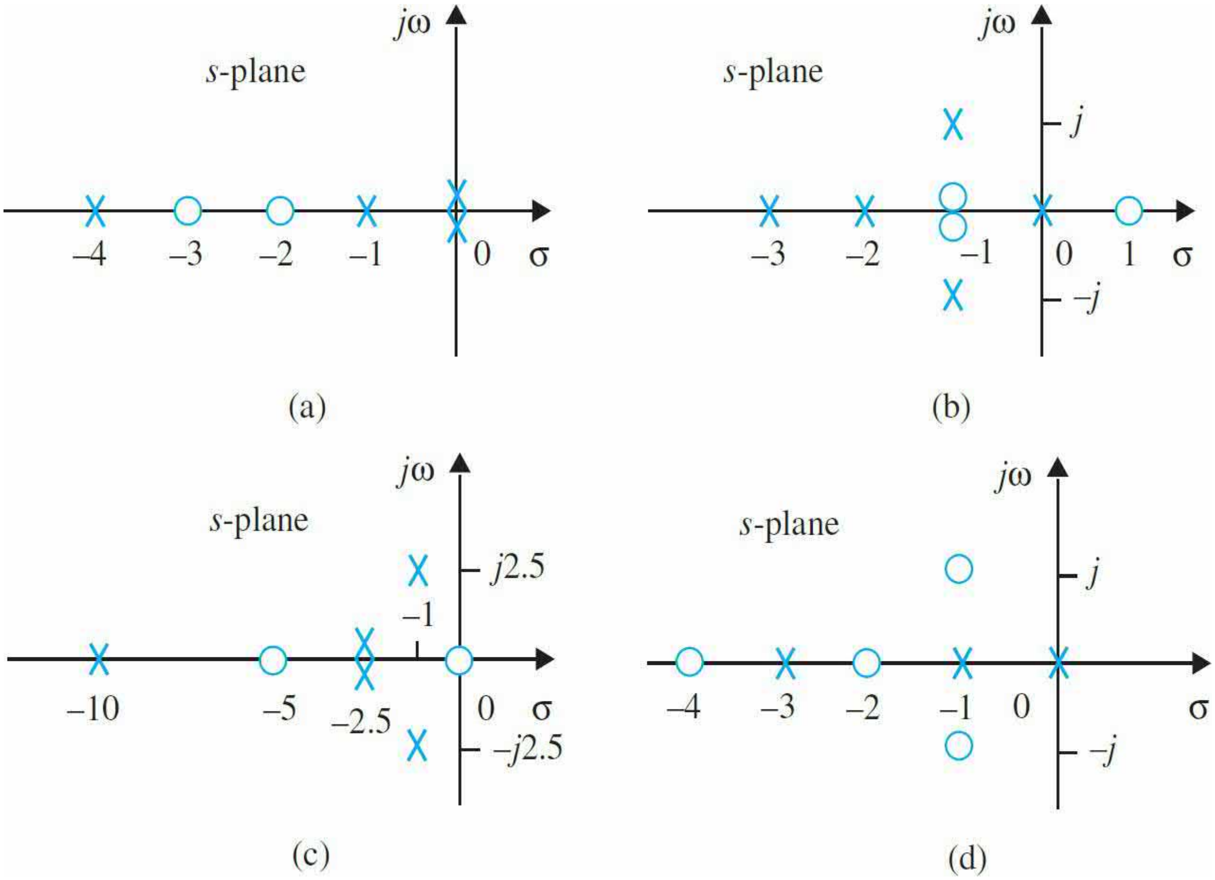


Figure 9P-9

9-10. Prove that a breakaway ω satisfies the following:

$$\sum_{i=1}^{n} \frac{1}{\alpha + P_i} = \sum_{i=1}^{m} \frac{1}{\alpha + Z_i}$$

9-11. Find all the breakaway points of the root loci of the systems described by the pole–zero configurations shown in Fig. 9P-9.

9-12. Construct the root-locus diagram for each of the following control systems for which the poles and zeros of $G(s)H(s)$ are given. The characteristic equation is obtained by equating the numerator of $1 + G(s)H(s)$ to zero.

(a) Poles at 0, –5, –6; zero at –8

(b) Poles at 0, –1, –3, –4; no finite zeros

(c) Poles at 0, 0, –2, –2; zero at –4

(d) Poles at 0, –1 + *j*, –1 –*j*; zero at –2

(e) Poles at 0, –1 + *j*, –1 –*j*; zero at –5

(f) Poles at 0, –1 + *j*, –1 –*j*; no finite zeros

(g) Poles at 0, 0, –8, –8; zeros at –4, –4

(h) Poles at 0, 0, –8, –8; no finite zeros

(i) Poles at 0, 0, –8, –8; zeros at –4 + *j*2, –4 –*j*2

(j) Poles at –2, 2; zeros at 0, 0

(k) Poles at *j*, –*j*, *j*2, –*j*2; zeros at –2, 2

(l) Poles at *j*, –*j*, *j*2, –*j*2; zeros at –1, 1

(m) Poles at 0, 0, 0, 1; zeros at –1, –2, –3

(n) Poles at 0, 0, 0, –100, –200; zeros at –5, –40

(o) Poles at 0, –1, –2; zero at 1

9-13. Use MATLAB to solve Prob. 9-12.

9-14. The characteristic equations of linear control systems are given as follows. Construct the root loci for $K \geq 0$.

(a) $s^3+3s^2+(K+2)s+5K=0$

(b) $s^3+s^2+(K+2)s+3K=0$

(c) $s^3+5Ks^2+10=0$

(d) $s^4+(K+3)s^3+(K+1)s^2+(2K+5)+10=0$

(e) $s^3+2s^2+2s+K(s^2-1)(s+2)=0$

(f) $s^3-2s+K(s+4)(s+1)=0$

(g) $s^4+6s^3+9s^2+K(s^2+4s+5)=0$

(h) $s^3+2s^2+2s+K(s^2-2)(s+4)=0$

(i) $s(s^2-1)+K(s+2)(s+0.5)=0$

(j) $s^4+2s^3+2s^2+2Ks+5K=0$

(k) $s^5+2s^4+3s^3+2s^2+s+K=0$

9-15. Use MATLAB to solve Prob. 9-14.

9-16. The forward-path transfer functions of a unity-feedback control system are given in the following.

(a) $$G(s)=\frac{K(s+3)}{s(s^2+4s+4)(s+5)(s+6)}$$

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

(c) $$G(s)=\frac{K(s^2+2s+8)}{s(s+5)(s+10)}$$

(d) $$G(s)=\frac{K(s^2+4)}{(s+2)^2(s+5)(s+6)}$$

(e) $$G(s)=\frac{K(s+10)}{s^2(s+2.5)(s^2+2s+2)}$$

(f) $$G(s)=\frac{K}{(s+1)(s^2+4s+5)}$$

(g) $$G(s)=\frac{K(s+2)}{(s+1)(s^2+6s+10)}$$

(h) $$G(s)=\frac{K(s+2)(s+3)}{s(s+1)}$$

(i) $$G(s)=\frac{K}{s(s^2+4s+5)}$$

Construct the root loci for $K \geq 0$. Find the value of K that makes the relative damping ratio of the closed-loop system (measured by the dominant complex characteristic equation roots) equal to 0.707, if such a solution exists.

9-17. Use MATLAB to verify your answer to Prob. 9-16.

9-18. A unity-feedback control system has the forward-path transfer functions given in the following. Construct the root-locus diagram for $K \geq 0$. Find the values of K at all the breakaway points.

(a) $$G(s)=\frac{K(s+3)}{s(s^2+4s+4)(s+5)(s+6)}$$

(b) $$G(s)=\frac{K(s+3)}{s(s^2+4s+4)(s+5)(s+6)}$$

(c) $$G(s)=\frac{K(s+3)}{s(s^2+4s+4)(s+5)(s+6)}$$

(e) $$G(s)=\frac{K(s+3)}{s(s^2+4s+4)(s+5)(s+6)}$$

(d) $$G(s)=\frac{K(s+3)}{s(s^2+4s+4)(s+5)(s+6)}$$

(f) $$G(s)=\frac{K(s+3)}{s(s^2+4s+4)(s+5)(s+6)}$$

9-19. Use MATLAB to verify your answer to Prob. 9-18.

9-20. The forward-path transfer function of a unity-feedback control system is

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

Construct the root loci of the characteristic equation of the closed-loop system for $K \geq \infty$, with

(a) n = 1, **(b)** n = 2, **(c)** n = 3, **(d)** n = 4, and **(e)** n = 5.

9-21. Use MATLAB to solve Prob. 9-20.

9-22. The characteristic equation of the control system shown in Fig. 7P-16 when K = 100 is

$$s^3+25s^2+(100K_t+2)s+100=0$$

Construct the root loci of the equation for $K_t \geq 0$.

9-23. Use MATLAB to verify your answer to Prob. 9-22.

9-24. The block diagram of a control system with tachometer feedback is shown in Fig. 9P-24.

(a) Construct the root loci of the characteristic equation for $K \geq 0$ when $K_t \geq 0$.

(b) Set $K = 10$. Construct the root loci of the characteristic equation for $K_t \geq 0$.

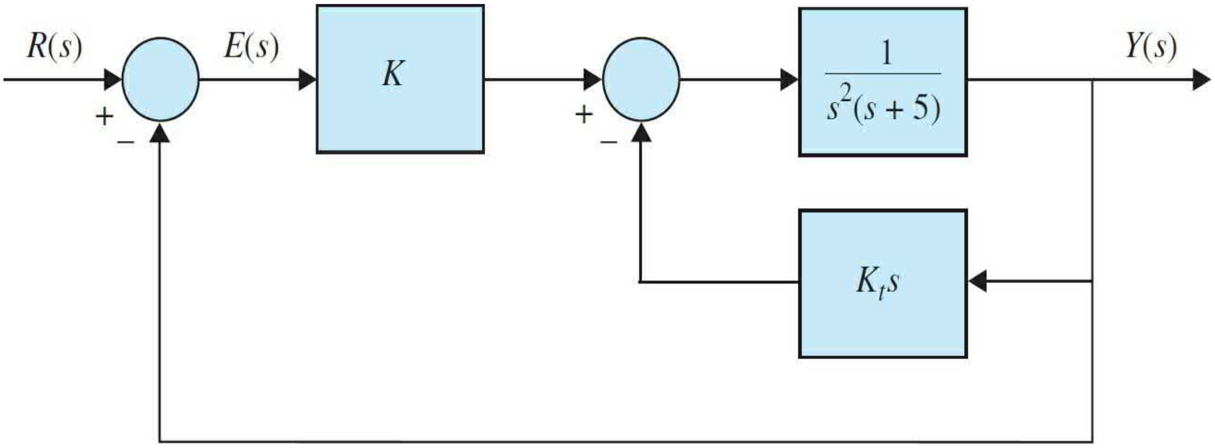


Figure 9P-24

9-25. Use MATLAB to solve Prob. 9-24.

9-26. The characteristic equation of the dc-motor control system described in Probs. 4-49 and 5-40 can be approximated as

$$2.05J_L s^3 + (1 + 10.25J_L)s^2 + 116.84s + 1843 = 0$$

when $K_L = \infty$ and the load inertia J_L is considered as a variable parameter. Construct the root loci of the characteristic equation for $K_L \geq 0$.

9-27. Use MATLAB to verify your answer to Prob. 9-26.

9-28. The forward-path transfer function of the control system shown in Fig. 9P-24 is

$$G(s) = \frac{K(s+\alpha)(s+3)}{s(s^2-1)}$$

(a) Construct the root loci for $K \geq 0$ with $\omega = 5$.

(b) Construct the root loci for $\omega \geq 0$ with $K = 10$.

9-29. Use MATLAB to solve Prob. 9-28.

9-30. The forward-path transfer function of a control system is

$$G(s)=\frac{K(s+0.4)}{s^2(s+3.6)}$$

(a) Construct the root loci for $K \geq 0$.

(b) Use MATLAB to verify your answer to part (a).

9-31. The characteristic equation of the liquid-level control system described in Prob. 5-42 is written

$$0.06s(s+12.5)(As+K_o)+250N=0$$

(a) For $A = K_o = 50$, construct the root loci of the characteristic equation as N varies from 0 to ∞

(b) For $N = 10$ and $K_o = 50$, construct the root loci of the characteristic equation for $A \geq 0$.

(c) For $A = 50$ and $N = 20$, construct the root loci for $K_o \geq 0$.

9-32. Use MATLAB to solve Prob. 9-31.

9-33. Repeat Prob. 9-31 for the following cases.

(a) $A = K_o = 100$ **(b)** $N = 20$ and $K_o = 50$ **(c)** $A = 100$ and $N = 20$.

9-34. Use MATLAB to verify your answer to Prob. 9-33.

9-35. The feedforward transfer function of a unity-feedback system is

$$G(s)=\frac{K(s+2)^2}{(s^2+4)(s+5)^2}$$

(a) Construct the root loci for $K = 25$.

(b) Find the range of K value for which the system is stable.

(c) Use MATLAB to verify your answer to part (a).

9-36. The transfer functions of a single-feedback-loop control system are

$$G(s)=\frac{K}{s^2(s+1)(s+5)}\quad H(s)=1$$

(a) Construct the loci of the zeros of 1 + $G(s)$ for $K \geq 0$.

(b) Repeat part (a) when $H(s) = 1 + 5s$.

9-37. Use MATLAB to solve Prob. 9-36.

9-38. The forward-path transfer function of a unity-feedback system is

$$G(s)=\frac{Ke^{-Ts}}{s+1}$$

(a) Construct the root loci for $T = 1$ sec and $K > 1$.

(b) Find the values of K where the system is stable.

(c) Use MATLAB to verify your answer to part (a).

9-39. The transfer functions of a single-feedback-loop control system are

$$G(s)=\frac{10}{s^2(s+1)(s+5)}\quad H(s)=1+T_d s$$

(a) Construct the root loci of the characteristic equation for $T_d \geq 0$.

(b) Use MATLAB to verify your answer to part (a).

9-40. For the dc-motor control system described in Probs. 4-49 and 5-40, it is of interest to study the effects of the motor-shaft compliance K_L on the system performance.

(a) Let $K = 1$, with the other system parameters as given in Probs. 4-49 and 5-40. Find an equivalent $G(s)H(s)$ with K_L as the gain factor. Construct the root loci of the characteristic equation for $K_L \geq 0$. The system can be approximated as a fourth-order system by canceling the large negative pole and zero of $G(s)H(s)$ that are very close to each other.

(b) Repeat part (a) with $K = 1000$.

9-41. Use MATLAB to verify your answer to Prob. 9-40.

9-42. The characteristic equation of the dc-motor control system described in Probs. 4-49 and 5-40 is given in the following when the motor shaft is considered to be rigid ($K_L = \infty$). Let $K=1, J_m=0.001, L_a=0.001, n=0.1, R_a=5, K_i=9, K_b=0.0636, B_m=0,$ and K_s = 1.

$$L_a(J_m+n^2J_L)s^3+(R_aJ_m+n^2R_aJ_L+B_mL_a)s^2+(R_aB_m+K_iK_b)s+nK_sK_iK=0$$

(a) Construct the root loci for $J_L \geq 0$ to show the effects of variation of the load inertia on system performance.

(b) Use MATLAB to verify your answer to part (a).

9-43. Given the equation $s^3+\alpha s^2+Ks+K=0$, it is desired to investigate the root loci of this equation for $-\infty < K < \infty$ and for various values of ω .

(a) Construct the root loci for $-\infty < K < \infty$ when $\omega = 12$.

(b) Repeat part (a) when $\omega = 4$.

(c) Determine the value of ω so that there is only one nonzero breakaway point on the entire root loci for $-\infty < K < \infty$. Construct the root loci.

9-44. Use MATLAB to solve Prob. 9-43.

9-45. The forward-path transfer function of a unity-feedback control system is

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

Determine the values of ω so that the root loci ($-\infty < K < \infty$) will have zero, one, and two breakaway points, respectively, not including the one at $s = 0$. Construct the root loci for $-\infty < K < \infty$ for all three cases.

9-46. Figure 9P-46 shows the block diagram of a unity-feedback control system.

Design a proper controller $H(s)$ for the system.

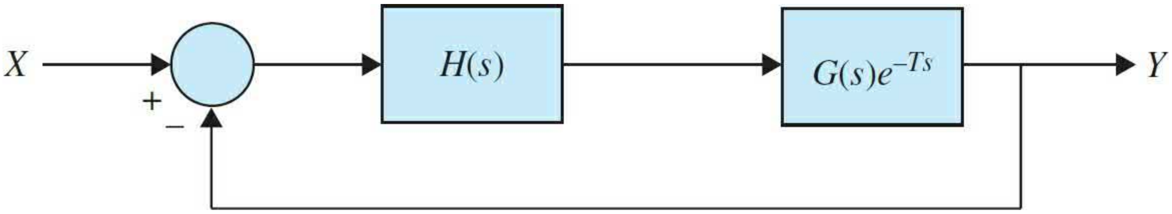


Figure 9P-46

9-47. The pole-zero configuration of $G(s)H(s)$ of a single-feedback-loop control system is shown in Fig. 9P-47*a*. Without actually plotting, apply the angle-of-departure (and-arrival) property of the root loci to determine which root-locus diagram shown is the correct one.

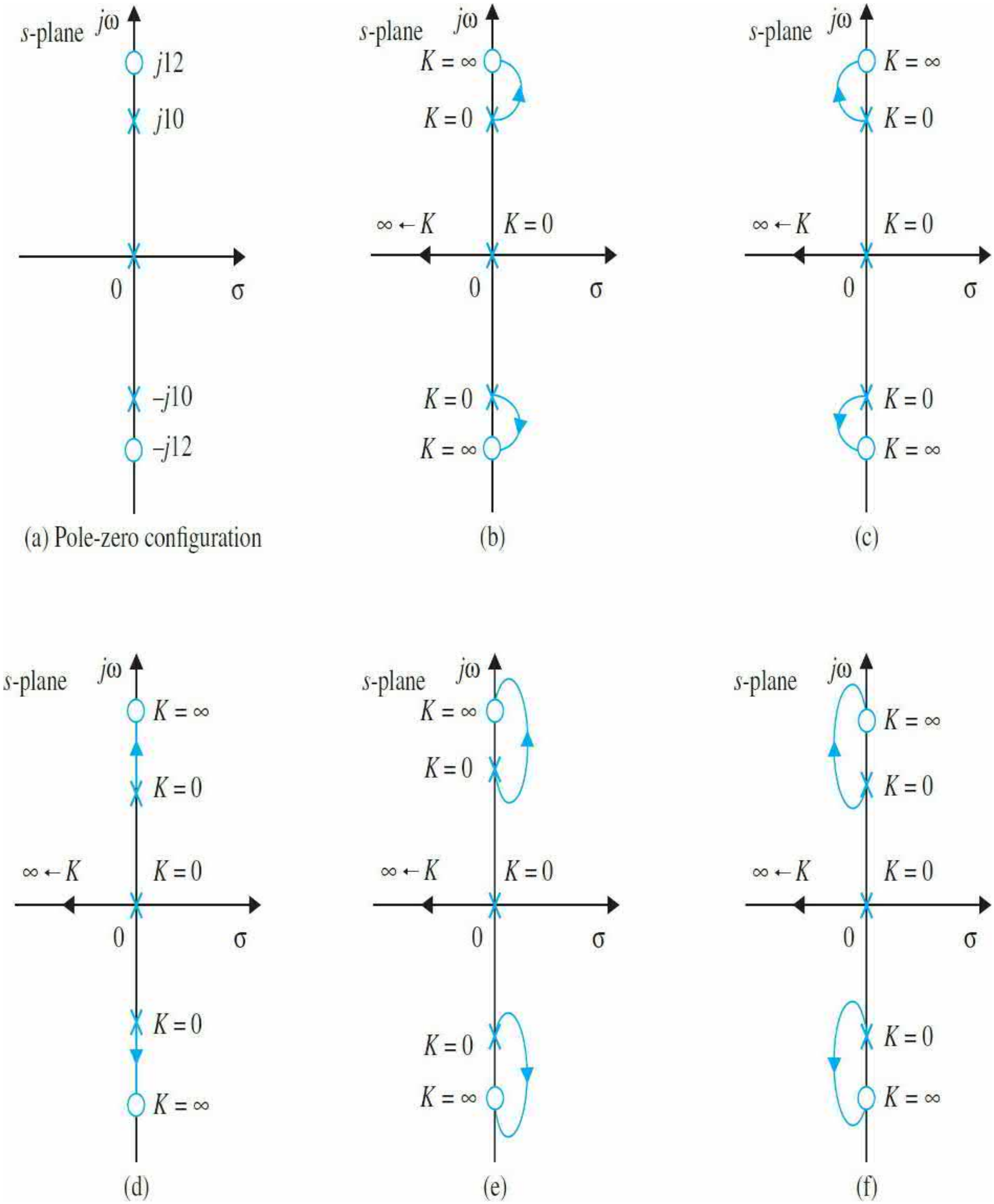

s-plane
jω
j12
j10
0
σ
−j10
−j12
(a) Pole-zero configuration
K = ∞
K = 0
∞ ← K
(b)
(c)
(d)
(e)
(f)

Figure 9P-47

CHAPTER 10

Frequency-Domain Analysis

Before starting this chapter, the reader is encouraged to refer to App. B to review the theoretical background related to complex variables and frequency-domain analysis and frequency plots (including asymptotic approximations—Bode plots).

In practice, the performance of a control system is more realistically measured by its time-domain characteristics. The reason is that the performance of most control systems is judged based on the time responses due to certain test signals. This is in contrast to the analysis and design of communication systems for which the frequency response is of more importance, since most of the signals to be processed are either sinusoidal or composed of sinusoidal components. We learned in Chap. 7 that the time response of a control system is usually more difficult to determine analytically, especially for high-order systems. In design problems, there are no unified methods of arriving at a designed system that meets the time-domain performance specifications, such as maximum overshoot, rise time, delay time, settling time, and so on. On the other hand, in the frequency domain, there is a wealth of graphical methods available that are not limited to low-order systems. It is important to realize that there are correlating relations between the frequency-domain and the time-domain performances in a linear system, so the time-domain properties of the system can be predicted based on the frequency-domain characteristics. The frequency domain is also more convenient for measurements of system sensitivity to noise and parameter variations. With these concepts in mind, we consider the primary motivation for conducting control systems analysis and design in the frequency domain to be convenience and the availability of the existing analytical tools. Another reason is that it presents an alternative point of view to control-system problems, which often provides valuable or crucial information in the complex analysis and design of control systems. Therefore, to conduct a frequency-domain analysis of a linear control system does not imply that the system will only be subject to a sinusoidal input. It may never be. Rather, from the frequency-response studies, we will be able to project the time-domain performance of the system.

Learning Outcomes

After successful completion of this chapter, you will be able to

1. Conduct a frequency response analysis of a system.
2. Plot the frequency response of a system using polar, magnitude, and phase diagrams.
3. Identify the resonant peak frequency and magnitude, bandwidth in a frequency response.
4. For a prototype second-order system, relate the frequency-domain specifications to the time-response specifications.
5. Plot the system Nyquist diagram.
6. Perform system stability analysis using the Nyquist stability criterion.
7. Use MATLAB to construct plot frequency response, Nyquist and Nichols plots.
8. Use frequency-response techniques to design control systems.

10-1 INTRODUCTION TO FREQUENCY RESPONSE

The starting point for frequency-domain analysis of a linear system is its transfer function. Let us consider the following practical example. It is well known from linear system theory that when the input to a stable linear time-invariant system is sinusoidal with amplitude *R* and frequency *ω*,

$$r(t) = R\sin\omega t \tag{10-1}$$

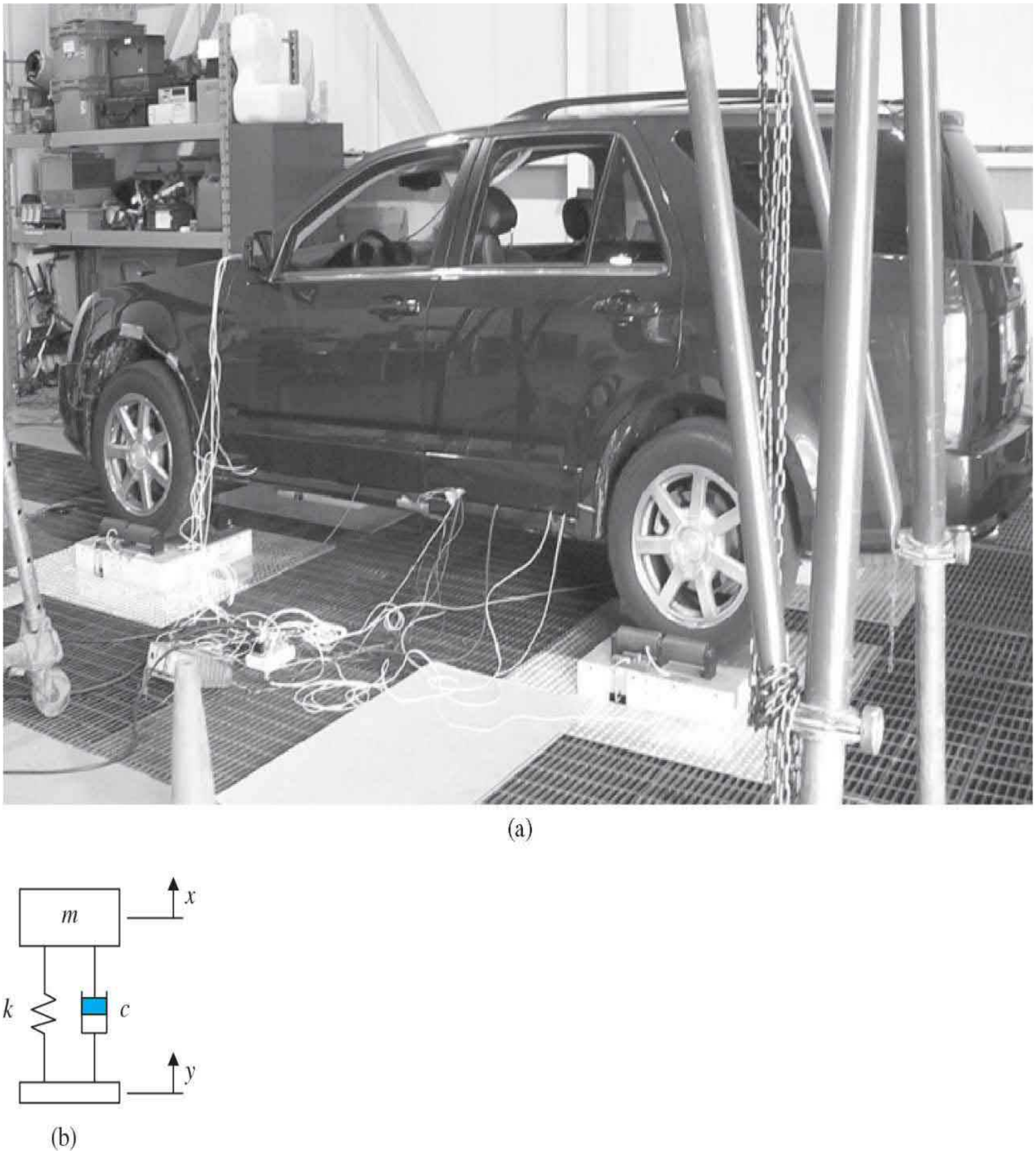
the steady-state output of the system, *y*(*t*), will be a sinusoid with the same frequency *w* but possibly with different amplitude and phase; that is,

$$y(t) = Y\sin(\omega t + \phi) \tag{10-2}$$

where *Y* is the amplitude of the output sine wave and *ϕ* is the phase shift in degrees or radians.

Let us examine this concept through a practical example.

EXAMPLE 10-1-1 For the test vehicle suspension system, modeled in Example 6-5-2, as shown in Fig. 10-1*a*, a four post shaker is used to test the performance of the vehicle suspension system by applying various excitations to the vehicle. Using the 1-DOF quarter-car model, shown in Fig. 10-1*b*, we can study the performance of each wheel suspension and its response to various road inputs. In this case we apply a sinusoidal input to each wheel, and study the behavior of the system as the excitation frequency varies from zero to very high (depending on shaker frequency limitations).


(a)

(b)

Figure 10-1 (*a*) A Cadillac SRX 2005 model on a four-post shaker test facility (from author’s research on active suspension system). (*b*) A 1° of freedom quarter-car model of one wheel suspension.

The equation of motion of the quarter-car system, categorized as a base excitation system (see Chap. 2), is defined as follows:

$$\ddot{z}(t)+2\zeta\omega_n\dot{z}(t)+\omega_n^2 z(t)=-\ddot{y}(t) \tag{10-3}$$

where the system parameters are scaled for simpler treatment of the example.

m	Effective ¼ car mass	10 kg
k	Effective stiffness	160 N/m
c	Effective damping	100 N·m/s^{-1}
$x(t)$	Absolute displacement of the mass m	m
$y(t)$	Absolute displacement of the base	m
$z(t)$	Relative displacement $z(t)=x(t)-y(t)$	M

Equation (10-3) reflects the bounce of the vehicle subject to road excitations. The transfer function of the relative motion is shown as

$$G(s)=\frac{1}{(s+2)(s+8)}=\frac{1}{s^2+10s+16}=\frac{1}{s^2+2\zeta\omega_n s+\omega_n{}^2} \tag{10-4}$$

where the system is overdamped ζ = 1.25. and ω_n = 4 red/s. The time response of the system for $-\ddot{y}(t)=A\sin(\omega t)$ can be obtained using the inverse Laplace transforms, see Chap. 3. From Eq. (10-3) and the Laplace transformation table in App. C, $\mathcal{L}[\sin\omega t]=\dfrac{\omega}{s^2+\omega^2}$,

$$Z(s)=\frac{A\omega}{s^2+\omega^2}\frac{1}{(s+2)(s+8)} \tag{10-5}$$

Hence,

$$z(t)=\{\text{Transients }\rightarrow 0\}+A\frac{1}{\sqrt{(\omega^2+4)(\omega^2+64)}}\sin\left(\omega t+\tan^{-1}\frac{-10\omega}{(16-\omega^2)}\right) \tag{10-6}$$

where, because the system is inherently stable, the transient response vanishes, and the steady-state response is *z*(*t*) = *Z* sin(*ωt* + ϕ), and can be represented in polar form as

$$\begin{aligned} z(t) &= Z\sin(\omega t+\phi) \\ &= A\left|G(j\omega)\right|\angle G(j\omega) \\ &= AM\angle\phi \\ &= Z\angle\phi \end{aligned} \tag{10-7}$$

where *G*(*jω*) = |*G*(*jω*) is the frequency response function. We define

$$M = \left|G(j\omega)\right| = \frac{1}{\sqrt{(\omega^2+4)(\omega^2+64)}} \tag{10-8a}$$

$$\phi = \angle G(j\omega) = \tan^{-1}\frac{-10\omega}{(16-\omega^2)} \tag{10-8b}$$

as the frequency response magnitude and phase, respectively, see App. B for review of complex variables. Note that care must be taken in calculating the phase because of the sign change in the denominator of Eq. (10-8a), as *w* varies. The polar representation of the frequency response function $G(j\omega)=\frac{1}{(\omega j+2)(\omega j+8)}$ for *ω* = 1 is shown in Fig. 10-2. As shown the frequency response is defined by the vector *G*(*jω*) = |*G*(*jω*) <*G*(*jω*) = *M*<*ϕ* of magnitude *M* and phase *ϕ*. Considering the input and output time responses, shown in Fig. 10-3, the magnitude and phase values are identified in this case.

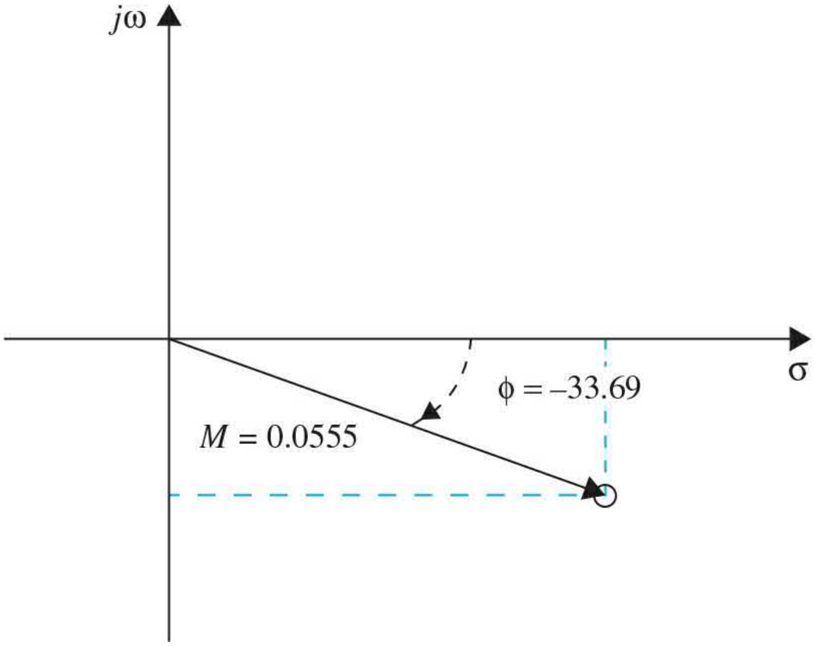


Figure 10-2 Polar representation of the frequency response function $G(j\omega)=\dfrac{1}{(\omega j+2)(\omega j+8)}$ for *ω* = 1 rad/s.

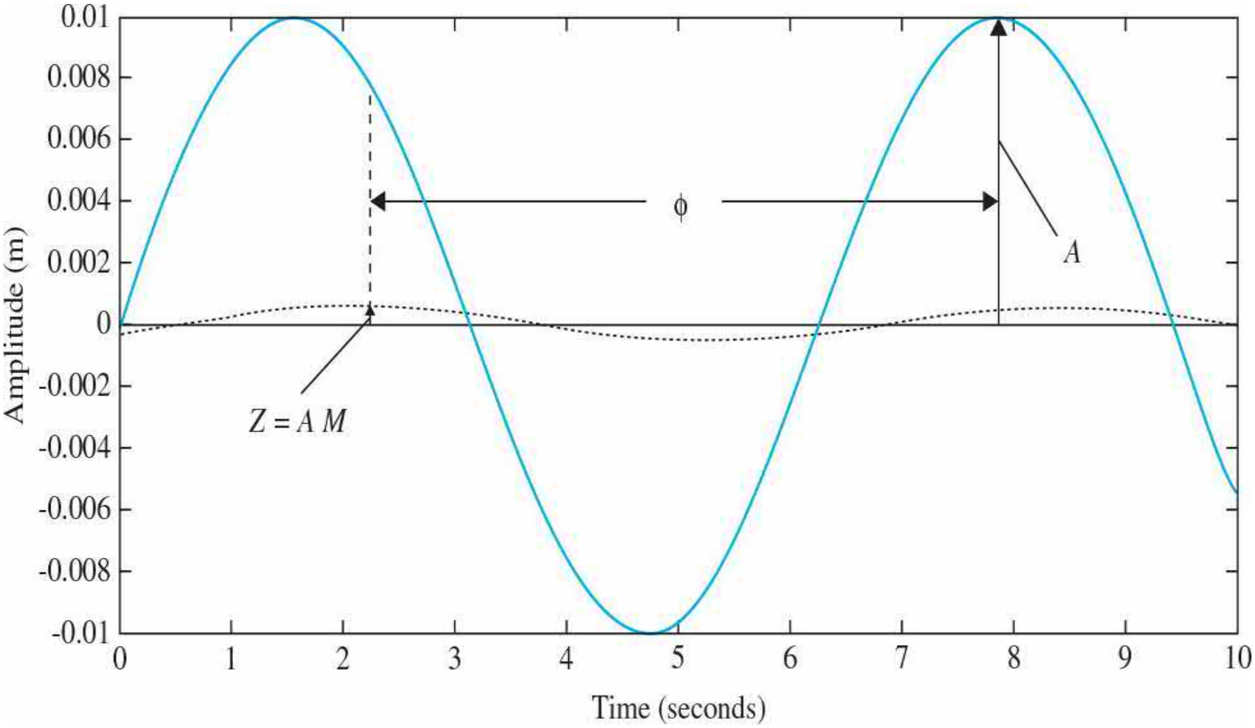
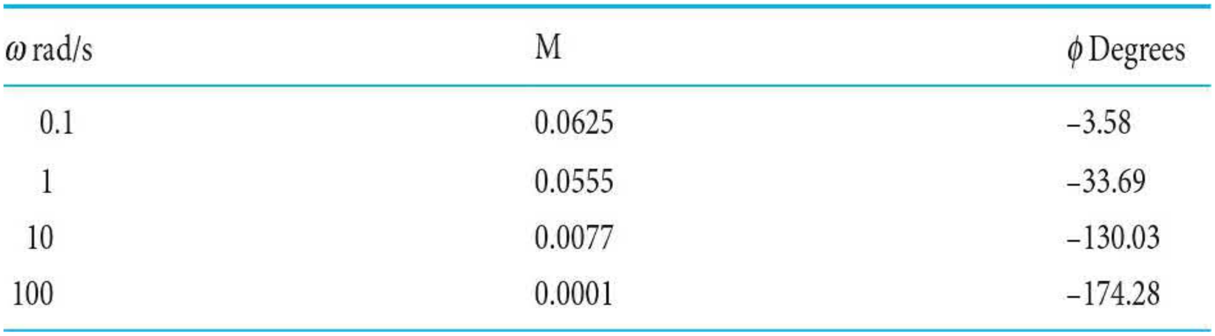


Figure 10-3 Graphical representation of input *A sin(ωt)* and output *z(t)* = *Z* sin(*ωt* + *ϕ*) for *A* = 0.01 and *ω* = 1 rad/s.

Table 10-1 describes different *M* and ϕ values as *w* increases. As shown, the magnitude *M* decreases as the frequency increases, and the phase *f* changes from 0° to –180°. The polar representation of the frequency response function $G(j\omega)=\dfrac{1}{(\omega j+2)(\omega j+8)}$ for *w* varying from 0.1 to 100 rad/s is shown in Fig. 10-4, where the “x” values correspond to the excitation frequencies shown in Table 10-1. Generally speaking, from Fig. 10-4, we can think of the polar frequency response plot as how a vector of magnitude *M* and phase *f* changes its magnitude and direction as *w* varies from zero to infinity.

TABLE 10-1 Numerical Values of Sample Magnitude and Phase of the System in Example 10-1-1



ω rad/s	M	ϕ Degrees
0.1	0.0625	−3.58
1	0.0555	−33.69
10	0.0077	−130.03
100	0.0001	−174.28

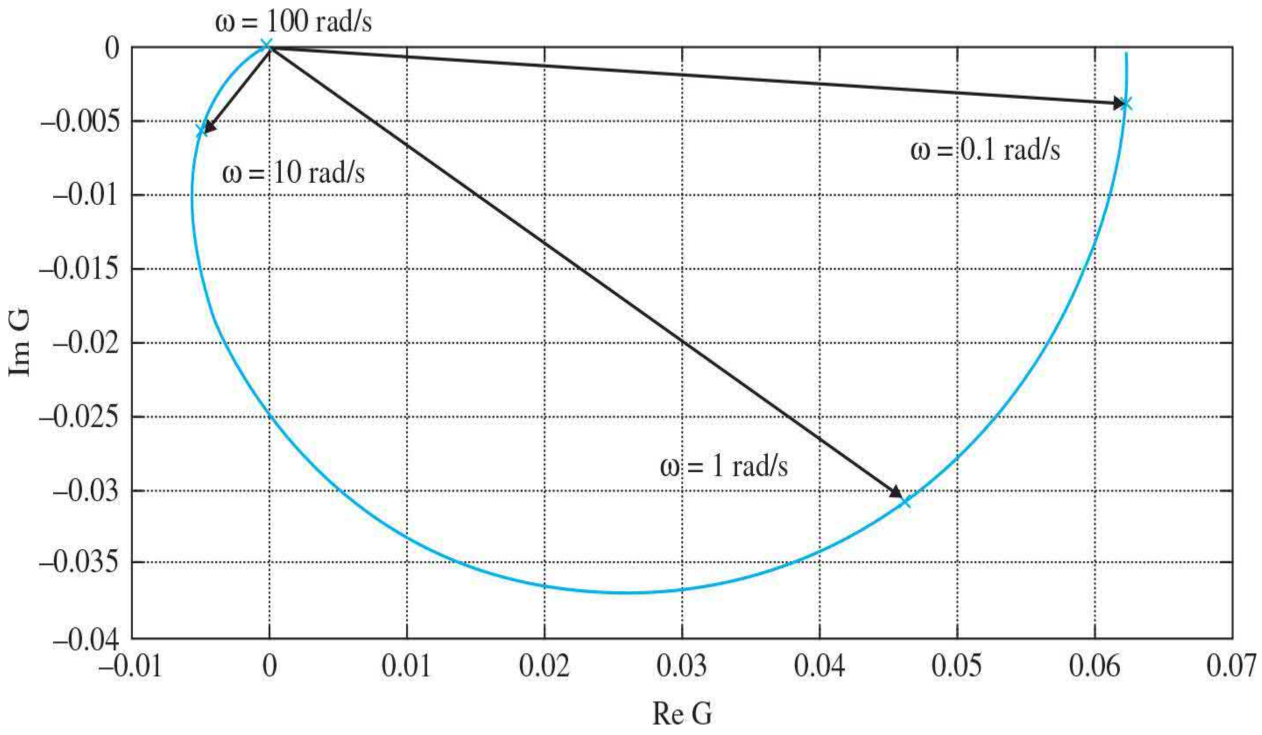


Figure 10-4 Polar representation of the frequency response function $G(j\omega)=\dfrac{1}{(\omega j+2)(\omega j+8)}$ for *w* varying from 0.1 to 100 rad/s.

From App. B, we know that the polar plot in Fig. 10-4 can be represented as separate magnitude and phase frequency response plots—also known as the Bode plot of the system. Figure 10-5 shows the frequency response plots. Note that, as discussed in App. B, in the Bode plot, it is customary to use log scale for the abscissa and dB = (20 log *M*) scale to represent the magnitude and degrees to represent the phase.

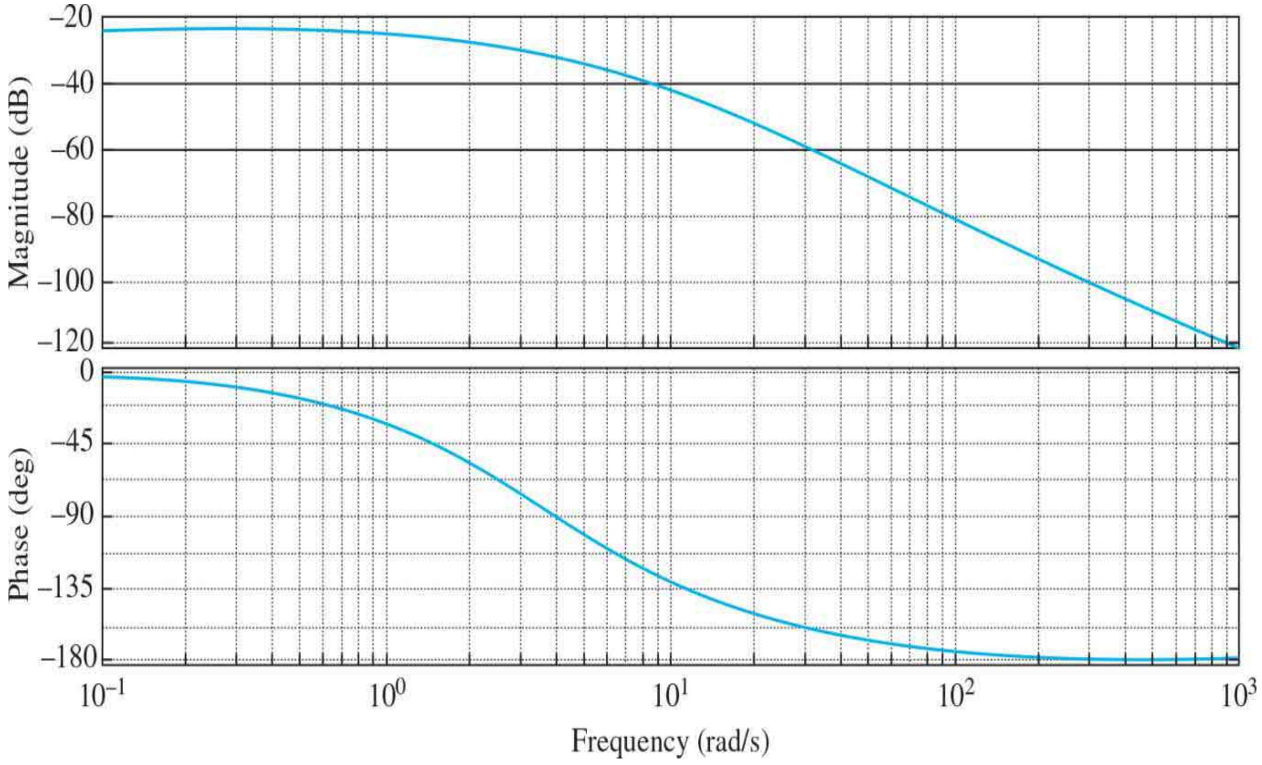


Figure 10-5 Magnitude and phase frequency response plots for $G(j\omega)=\dfrac{1}{(\omega j+2)(\omega j+8)}$ for *ω* varying from 0.1 to 100 rad/s.

MATLAB Toolbox 10-1-1 was used to arrive at Figs. 10-3 through 10-5.

In general, for the transfer function of a linear SISO system *M*(*s*); the Laplace transforms of the input and the output, see Eqs. (10-1) and (10-2), are related through

$$Y(s)=M(s)R(s) \tag{10-9}$$

For sinusoidal steady-state analysis, we replace *s* by *jω*, and the last equation becomes

$$Y(j\omega)=M(j\omega)R(j\omega) \tag{10-10}$$

By writing the function *Y*(*jω*) as

$$Y(j\omega)=\left|Y(j\omega)\right|\angle Y(j\omega) \tag{10-11}$$

with similar definitions for *M*(*jω*) and *R*(*jω*), Eq. (10-10) leads to the magnitude relation between the input and the output:

$$|Y(j\omega)| = |M(j\omega)||R(j\omega)| \tag{10-12}$$

and the phase relation:

$$\angle Y(j\omega) = \angle M(j\omega) + \angle R(j\omega) \tag{10-13}$$

Thus, for the input and output signals described by Eqs. (10-1) and (10-2), respectively, the amplitude of the output sinusoid is

$$Y = R|M(j\omega_0)| \tag{10-14}$$

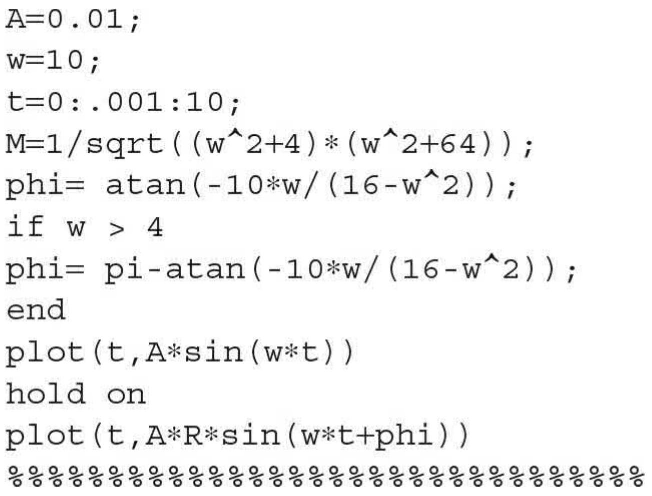
and the phase of the output is

$$\phi = \angle M(j\omega_0) \tag{10-15}$$

Thus, by knowing the transfer function *M*(*s*) of a linear system, the magnitude characteristic, |*M*(*jω*)|, and the phase characteristic, <*M*(*jω*), completely describe the steady-state performance when the input is a sinusoid. The crux of frequency-domain analysis is that the amplitude and phase characteristics of a closed-loop system can be used to predict both time-domain transient and steady-state system performances.

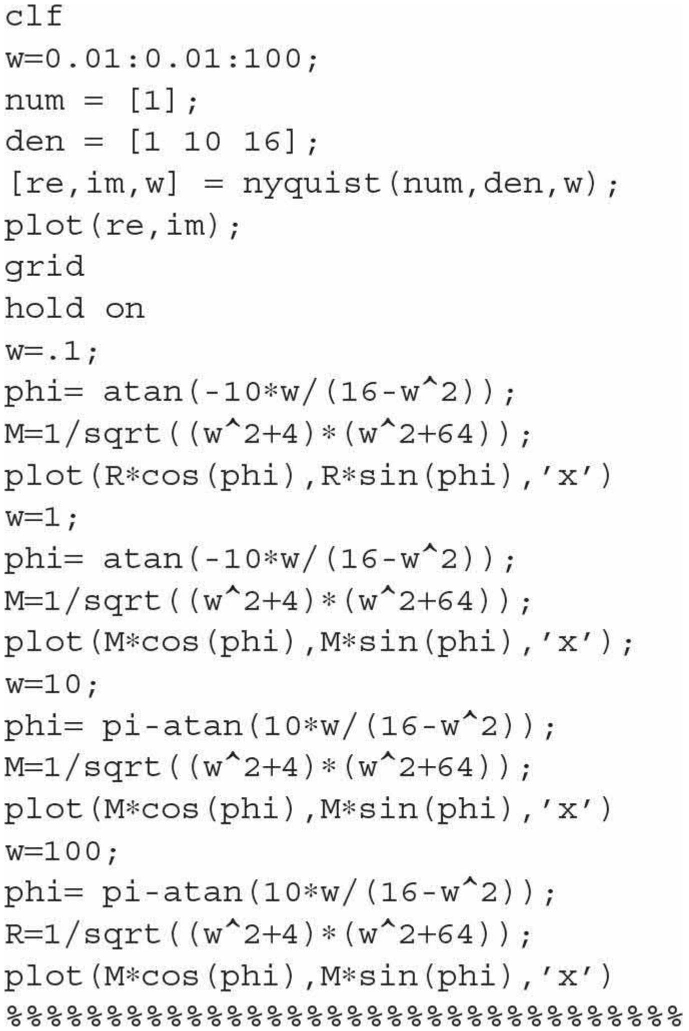
Toolbox 10-1-1

MATLAB code for Fig. 10-3 for the input and output time response plots.



```
A=0.01;
w=10;
t=0:.001:10;
M=1/sqrt((w^2+4)*(w^2+64));
phi= atan(-10*w/(16-w^2));
if w > 4
phi= pi-atan(-10*w/(16-w^2));
end
plot(t,A*sin(w*t))
hold on
plot(t,A*R*sin(w*t+phi))
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

MATLAB code for Fig. 10-4—polar plot.



```
clf
w=0.01:0.01:100;
num = [1];
den = [1 10 16];
[re,im,w] = nyquist(num,den,w);
plot(re,im);
grid
hold on
w=.1;
phi= atan(-10*w/(16-w^2));
M=1/sqrt((w^2+4)*(w^2+64));
plot(R*cos(phi),R*sin(phi),'x')
w=1;
phi= atan(-10*w/(16-w^2));
M=1/sqrt((w^2+4)*(w^2+64));
plot(M*cos(phi),M*sin(phi),'x');
w=10;
phi= pi-atan(10*w/(16-w^2));
M=1/sqrt((w^2+4)*(w^2+64));
plot(M*cos(phi),M*sin(phi),'x')
w=100;
phi= pi-atan(10*w/(16-w^2));
R=1/sqrt((w^2+4)*(w^2+64));
plot(M*cos(phi),M*sin(phi),'x')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

MATLAB code for Fig. 10-5—Bode plots.

```
clf
num = [1];
den = [1 10 16];
G = tf(num,den);
bode(G);
grid
```

In the end, to put these results into perspective, by looking at Figs. 10-3 through 10-5, you can see that the suspension system works as a filter to significantly reduce the effect of road excitations. The performance of the suspension system, however, is frequency dependent where the magnitude of the suspension transfer function reduces with higher frequencies while its phase varies from 0° to –180°.

EXAMPLE 10-1-2 In frequency-domain analyses of control systems, often we have to determine the basic properties of a polar plot. Consider the following transfer function (see also Example B-2-4):

$$G(s)=\frac{10}{s(s+1)(s+2)} \tag{10-16}$$

By substituting $s = j\omega$ in Eq. (10-16), the magnitude and phase of $G(j\omega)$ at $\omega = 0$ and $\omega = \infty$ are computed as follows:

$$\lim_{\omega\to 0}\left|G(j\omega)\right|=\lim_{\omega\to 0}\frac{5}{\omega}=\infty \tag{10-17}$$

$$\lim_{\omega\to 0}\angle G(j\omega)=\lim_{\omega\to 0}\angle 5/j\omega=-90^\circ \tag{10-18}$$

$$\lim_{\omega\to\infty}\left|G(j\omega)\right|=\lim_{\omega\to\infty}\frac{10}{\omega^3}=0 \tag{10-19}$$

Thus, the properties of the polar plot of $G(j\omega)$ at $\omega = 0$ and $\omega = \infty$ are ascertained.

To express $G(j\omega)$ as the sum of its real and imaginary parts, we must rationalize $G(j\omega)$ by multiplying its numerator and denominator by the complex conjugate of its denominator. Therefore, *G*(*jω*) is written to find the intersections of the *G*(*jω*) plot on the real and imaginary axes of the *G*(*jω*)-plane, we rationalize *G*(*jω*) to give

$$G(j\omega)=\frac{10(-j\omega)(-j\omega+1)(-j\omega+2)}{j\omega(j\omega+1)(j\omega+2)(-j\omega)(-j\omega+1)(-j\omega+2)} \tag{10-20}$$

After simplification, the last equation is written

$$G(j\omega)=\mathrm{Re}[G(j\omega)]+j\,\mathrm{Im}[G(j\omega)]=\frac{-30}{9\omega^2+(2-\omega^2)^2}-\frac{j10(2-\omega^2)}{9\omega^3+\omega(2-\omega^2)^2} \tag{10-21}$$

Next, we determine the intersections, if any, of the polar plot with the two axes of the *G*(*jω*)-plane. If the polar plot of *G*(*jω*) intersects the real axis, at the point of intersection, the imaginary part of *G*(*jω*) is zero, that is,

$$\mathrm{Im}[G(j\omega)]=0 \tag{10-22}$$

Similarly, the intersection of *G*(*jω*) with the imaginary axis is found by setting Re[*G*(*jω*)] of Eq. (10-21) to zero, that is,

$$\mathrm{Re}[G(j\omega)]=0 \tag{10-23}$$

Setting Re[*G*(*jω*)] to zero, we have *ω* = ∞, and *G*(*j*∞) = 0, which means that the *G*(*jω*) plot intersects the imaginary axis only at the origin. Setting [*G*(*jω*)] to zero, we have $\omega=\pm\sqrt{2}$ rad/s. This gives the point of intersection on the real axis at

$$G(\pm j\sqrt{2})=-5/3 \tag{10-24}$$

The result, $\omega=\pm\sqrt{2}$ rad/s, has no physical meaning because the frequency is negative; it simply represents a mapping point on the negative *jω*-axis of the *s*-plane. In general, if *G*(*s*) is a rational function of *s* (a quotient of two polynomials of *s*), the polar plot of *G*(*jω*) for negative values of ω is the mirror image of that for positive ω, with the mirror placed on the real axis of the $G(j\omega)$-plane. From Eq. (B-58), we also see that $\text{Re}[G(j0)] = \infty$ and $\text{Im}[G(j0)] = \infty$. With this information, it is now possible to make a sketch of the polar plot for the transfer function in Eq. (10-16), as shown in Fig. 10-6.

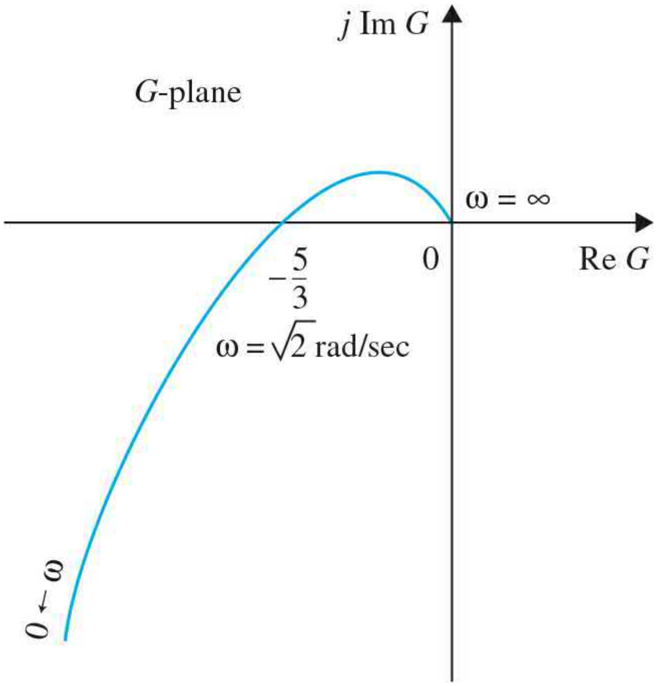


Figure 10-6 Polar plot of $G(s) = \dfrac{10}{s(s+1)(s+2)}$.

You should also be able to sketch the asymptotic magnitude or phase plots of the system. Please refer to App. B for further discussions on the subject.

MATLAB Toolbox 10-1-2 can be used to arrive at Fig. 10-6 and the Bode plots of the system.

Toolbox 10-1-2

MATLAB code for Fig. 10-6—polar plot.

```
clf
w=0.01:0.01:100;
num = [10];
den = [1 3 2 0];
[re,im,w] = nyquist(num,den,w);
plot(re,im);
grid
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

MATLAB code for Fig. 10-7—Bode plots.

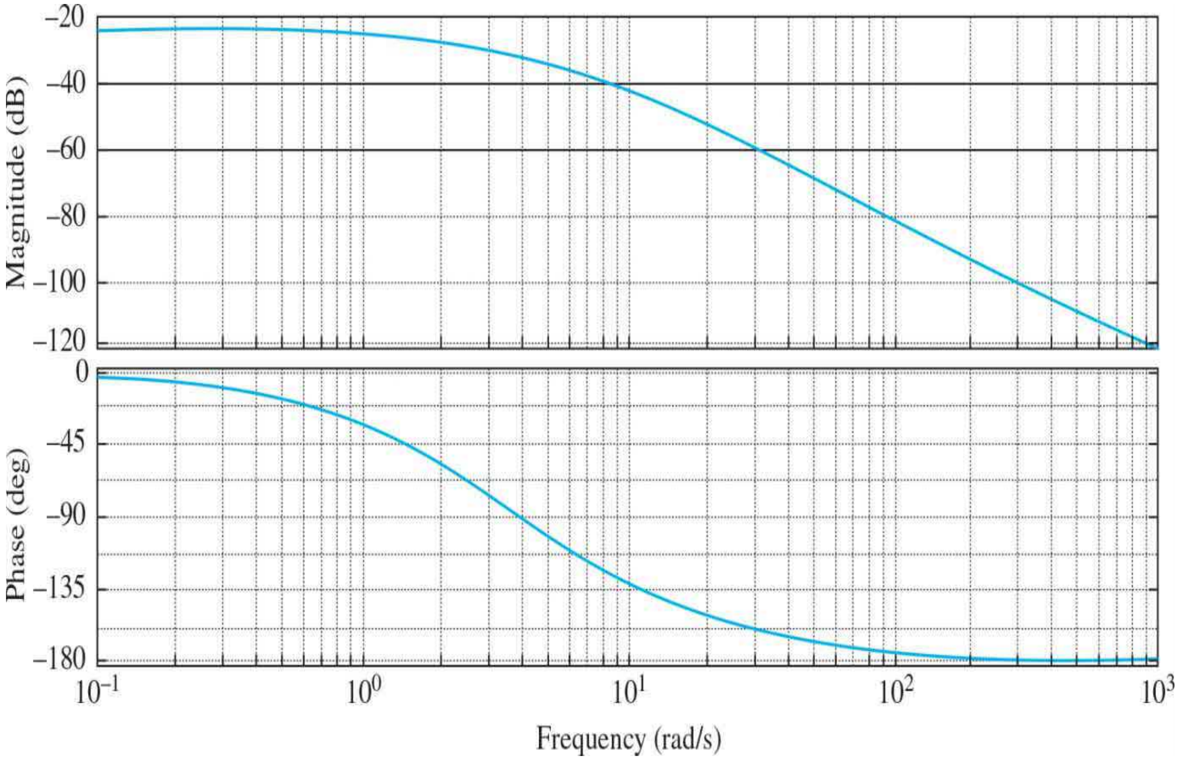


Figure 10-7 Bode diagrams for $G(s)=\dfrac{10}{s(s+1)(s+2)}$.

```
clf
num = [10];
den = [1 3 2 0];
G = tf(num,den);
bode(G);
grid
```

10-1-1 Frequency Response of Closed-Loop Systems

For the single-loop control-system configuration studied in the preceding chapters, the closed-loop transfer function is

$$M(s)=\frac{Y(s)}{R(s)}=\frac{G(s)}{1+G(s)H(s)} \tag{10-25}$$

Under the sinusoidal steady state, $s = j\omega$, Eq. (10-25) becomes

$$M(j\omega)=\frac{Y(j\omega)}{R(j\omega)}=\frac{G(j\omega)}{1+G(j\omega)H(j\omega)} \tag{10-26}$$

The sinusoidal steady-state transfer function $M(j\omega)$ may be expressed in terms of its magnitude and phase, that is,

$$M(j\omega)=|M(j\omega)|\angle M(j\omega) \tag{10-27}$$

Or $M(j\omega)$ can be expressed in terms of its real and imaginary parts:

$$M(j\omega)=\mathrm{Re}[M(j\omega)]+j\mathrm{Im}[M(j\omega)] \tag{10-28}$$

The magnitude of $M(j\omega)$ is

$$|M(j\omega)|=\left|\frac{G(j\omega)}{1+G(j\omega)H(j\omega)}\right|=\frac{|G(j\omega)|}{|1+G(j\omega)H(j\omega)|} \tag{10-29}$$

and the phase of $M(j\omega)$ is

$$\angle M(j\omega)=\phi_M(j\omega)=\angle G(j\omega)-\angle[1+G(j\omega)H(j\omega)] \tag{10-30}$$

If $M(s)$ represents the input-output transfer function of an electric filter, then the magnitude and phase of $M(j\omega)$ indicate the filtering characteristics on the input signal. Figure 10-8 shows the gain and phase characteristics of an ideal low-pass filter that has a sharp cutoff frequency at ωc. It is well known that an ideal filter characteristic is physically unrealizable. In many ways, the design of control systems is quite similar to filter design, and the control system is regarded as a signal processor. In fact, if the ideal low-pass-filter characteristics shown in Fig. 10-8 were physically realizable, they would be highly desirable for a control system, since all signals would be passed without distortion below the frequency *ωc*, and completely eliminated at frequencies above *ωc* where noise may lie.

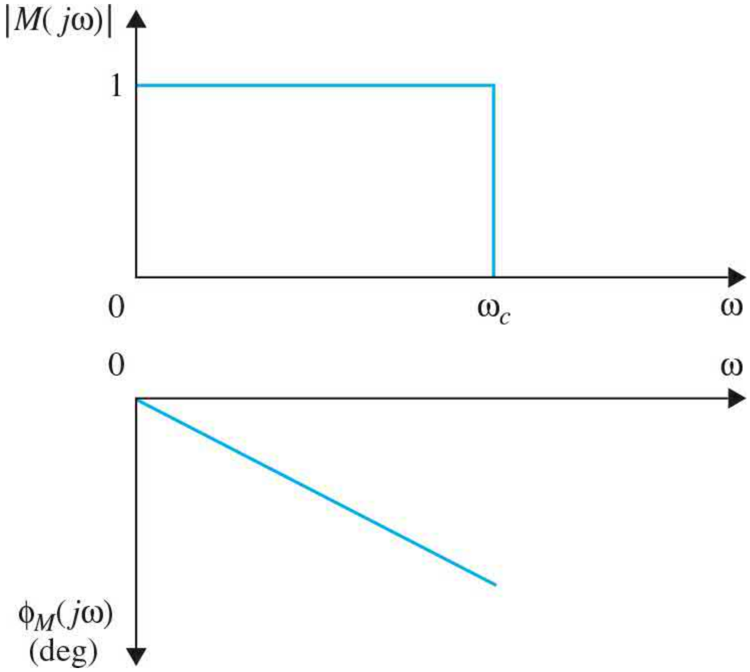


Figure 10-8 Gain-phase characteristics of an ideal low-pass filter.

If *ωc* is increased indefinitely, the output *Y*(*jω*) would be identical to the input *R*(*jω*) for all frequencies. Such a system would follow a step-function input in the time domain exactly. From Eq. (10-29), we see that, for |*M*(*jω*)| to be unity at all frequencies, the magnitude of *G*(*jω*) must be infinite. An infinite magnitude of *G*(*jω*) is, of course, impossible to achieve in practice, nor would it be desirable, since most control systems may become unstable when their loop gains become very high. Furthermore, all control systems are subject to noise during operation. Thus, in addition to responding to the input signal, the system should be able to reject and suppress noise and unwanted signals. For control systems with high-frequency noise, such as air-frame vibration of an aircraft, the frequency response should have a finite cutoff frequency *ωc*.

M_r indicates the relative stability of a stable closed-loop system.

The phase characteristics of the frequency response of a control system are also of importance, as we shall see that they affect the stability of the system.

Figure 10-9 illustrates typical gain and phase characteristics of a control system. As shown by Eqs. (10-29) and (10-30), the gain and phase of a closed-loop system can be determined from the forward-path and loop transfer functions. In practice, the frequency responses of $G(s)$ and $H(s)$ can often be determined by applying sine-wave inputs to the system and sweeping the frequency from 0 to a value beyond the frequency range of the system.

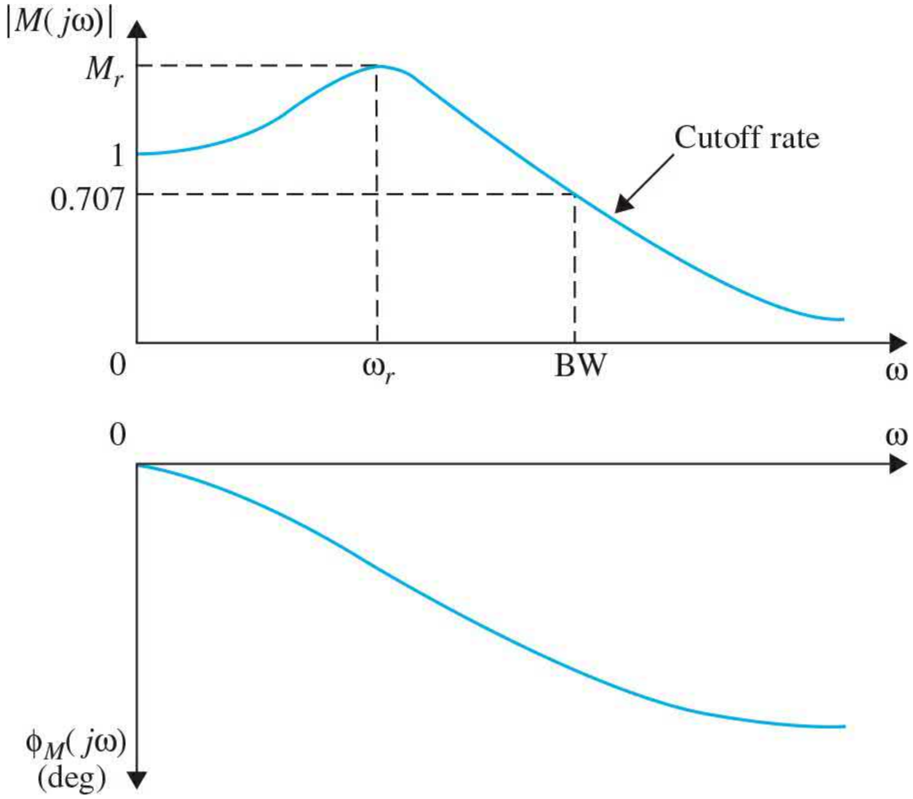


Figure 10-9 Typical gain-phase characteristics of a feedback control system.

10-1-2 Frequency-Domain Specifications

In the design of linear control systems using the frequency-domain methods, it is necessary to define a set of specifications so that the performance of the system can be identified. Specifications such as the maximum overshoot, damping ratio, and the like used in the time domain can no longer be used directly in the frequency domain. The following frequency-domain specifications are often used in practice.

Resonant Peak M_r

The resonant peak *Mr* is the maximum value of $|M(j\omega)|$.

In general, the magnitude of *Mr* gives indication on the relative stability of a stable closed-loop system. Normally, a large *Mr* corresponds to a large maximum overshoot of the step response. For most control systems, it is generally accepted in practice that the desirable value of *Mr* should be between 1.1 and 1.5.

Resonant Frequency ω

The resonant frequency ω_r is the frequency at which the peak resonance M_r occurs.

Bandwidth BW

The bandwidth BW is the frequency at which $|M(jw)|$ drops to 70.7% of, or 3 dB down from, its zero-frequency value.

In general, the bandwidth of a control system gives indication on the transient-response properties in the time domain. A large bandwidth corresponds to a faster rise time, since higher-frequency signals are more easily passed through the system. Conversely, if the bandwidth is small, only signals of relatively low frequencies are passed, and the time response will be slow and sluggish. Bandwidth also indicates the noise-filtering characteristics and the robustness of the system. The robustness represents a measure of the sensitivity of a system to parameter variations. A robust system is one that is insensitive to parameter variations.

- BW gives an indication of the transient response properties of a control system.
- BW gives an indication of the noise-filtering characteristics and robustness of the system.

Cutoff Rate

Often, bandwidth alone is inadequate to indicate the ability of a system in distinguishing signals from noise. Sometimes it may be necessary to look at the slope of $|M(j\omega)|$, which is called the cutoff rate of the frequency response, at high frequencies. Apparently, two systems can have the same bandwidth, but the cutoff rates may be different.

The performance criteria for the frequency-domain defined above are illustrated in Fig. 10-9. Other important criteria for the frequency domain will be defined in later sections of this chapter.

10-2 M_R, ω_R , AND BANDWIDTH OF THE PROTOTYPE SECOND-ORDER SYSTEM

10-2-1 Resonant Peak and Resonant Frequency

For the prototype second-order system, the resonant peak *Mr*, the resonant frequency *ωr*, and the bandwidth BW are all uniquely related to the damping ratio ζ and the natural undamped frequency *ωn* of the system.

Consider the closed-loop transfer function of the prototype second-order system

$$M(s)=\frac{Y(s)}{R(s)}=\frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2} \tag{10-31}$$

At sinusoidal steady state, $s = j\omega$, Eq. (10-31) becomes

$$\begin{aligned} M(j\omega)&=\frac{Y(j\omega)}{R(j\omega)}=\frac{\omega_n^2}{(j\omega)^2+2\zeta\omega_n(j\omega)+\omega_n^2} \\ &=\frac{1}{1+j2(\omega/\omega_n)\zeta-(\omega/\omega_n)^2} \end{aligned} \tag{10-32}$$

We can simplify Eq. (10-32) by letting $u = \omega/\omega_n$. Then, Eq. (10-32) becomes

$$M(ju)=\frac{1}{1+j2u\zeta-u^2} \tag{10-33}$$

The magnitude and phase of $M(ju)$ are

$$|M(ju)| = \frac{1}{[(1-u^2)^2 + (2\zeta u)^2]^{1/2}} \tag{10-34}$$

and

$$\angle M(ju) = \phi_M(ju) = -\tan^{-1}\frac{2\zeta u}{1-u^2} \tag{10-35}$$

respectively. The resonant frequency is determined by setting the derivative of $|M(ju)|$ with respect to u to zero. Thus,

$$\frac{d|M(ju)|}{du} = -\frac{1}{2}[(1-u^2)^2 + (2\zeta u)^2]^{-3/2}(4u^3 - 4u + 8u\zeta^2) = 0 \tag{10-36}$$

from which we get

$$4u^3 - 4u + 8u\zeta^2 = 4u(u^2 - 1 + 2\zeta^2) = 0 \tag{10-37}$$

In normalized frequency, the roots of Eq. (10-37) are u_r 0 and

$$u_r = \sqrt{1-2\zeta^2} \tag{10-38}$$

- For the prototype second-order system, M_r is a function of ζ only.
- For the prototype second-order system, $M_r = 1$ and $\omega_r = 0$ when $\zeta \geq 0.707$.

The solution of $u_r = 0$ merely indicates that the slope of the $|M(ju)|$-versus-ω curve is zero at $\omega = 0$; it is not a true maximum if ζ is less than 0.707. Equation (10-38) gives the resonant frequency

$$\omega_r = \omega_n\sqrt{1-2\zeta^2} \tag{10-39}$$

Because frequency is a real quantity, Eq. (10-39) is meaningful only for $2\zeta^2 \leq 1$, or $\zeta \leq 0.707$. This means simply that, for all values of ζ greater than 0.707, the resonant frequency is $\Omega_r = 0$ and $M_r = 1$.

Substituting Eq. (10-38) into Eq. (10-35) for *u* and simplifying, we get

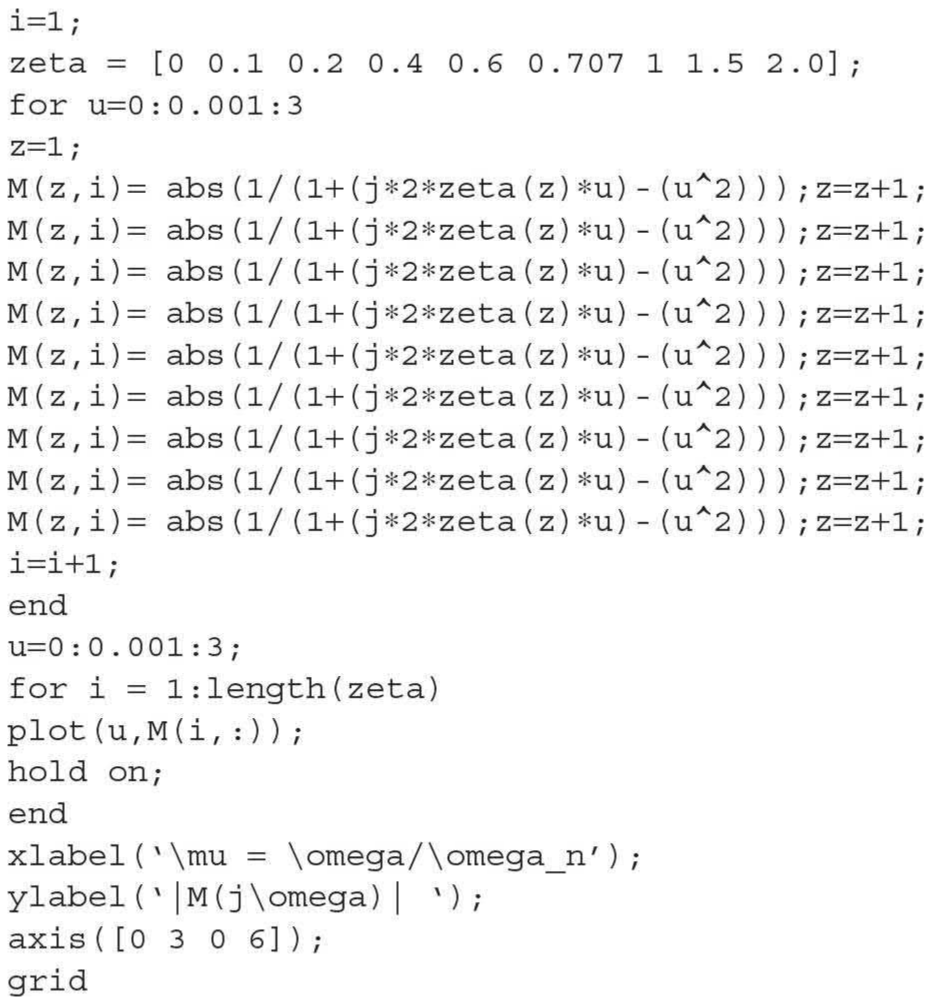
$$M_r = \frac{1}{2\zeta\sqrt{1-\zeta^2}} \qquad \zeta \leq 0.707 \tag{10-40}$$

It is important to note that, for the prototype second-order system, *Mr* is a function of the damping ratio ζ only, and *ωr* is a function of both ζ and *ωn*. Furthermore, although taking the derivative of |*M*(*ju*) | with respect to *u* is a valid method of determining *Mr* and *ωr*, for higher-order systems, this analytical method is quite tedious and is not recommended. Graphical methods to be discussed and computer methods are much more efficient for high-order systems.

BW/ω_n decreases monotonically as the damping ratio ζ decreases.

Toolbox 10-2-1

MATLAB code for Fig. 10-10.



```
i=1;
zeta = [0 0.1 0.2 0.4 0.6 0.707 1 1.5 2.0];
for u=0:0.001:3
z=1;
M(z,i)= abs(1/(1+(j*2*zeta(z)*u)-(u^2)));z=z+1;
M(z,i)= abs(1/(1+(j*2*zeta(z)*u)-(u^2)));z=z+1;
M(z,i)= abs(1/(1+(j*2*zeta(z)*u)-(u^2)));z=z+1;
M(z,i)= abs(1/(1+(j*2*zeta(z)*u)-(u^2)));z=z+1;
M(z,i)= abs(1/(1+(j*2*zeta(z)*u)-(u^2)));z=z+1;
M(z,i)= abs(1/(1+(j*2*zeta(z)*u)-(u^2)));z=z+1;
M(z,i)= abs(1/(1+(j*2*zeta(z)*u)-(u^2)));z=z+1;
M(z,i)= abs(1/(1+(j*2*zeta(z)*u)-(u^2)));z=z+1;
M(z,i)= abs(1/(1+(j*2*zeta(z)*u)-(u^2)));z=z+1;
i=i+1;
end
u=0:0.001:3;
for i = 1:length(zeta)
plot(u,M(i,:));
hold on;
end
xlabel('\mu = \omega/\omega_n');
ylabel('|M(j\omega)|  ');
axis([0 3 0 6]);
grid
```

Figure 10-10 illustrates the plots of $|M(ju)|$ of Eq. (10-34) versus u for various values of ζ. Notice that, if the frequency scale were unnormalized, the value of $\omega_r = u_r\omega$ would increase when ζ decreases, as indicated by Eq. (10-39). When $\zeta = 0$, ω_r ω_n. Figures 10-11 and 10-12 illustrate the relationship between Mr and ζ, and $u_r(=\omega_r/\omega_n)$ and ζ, respectively.

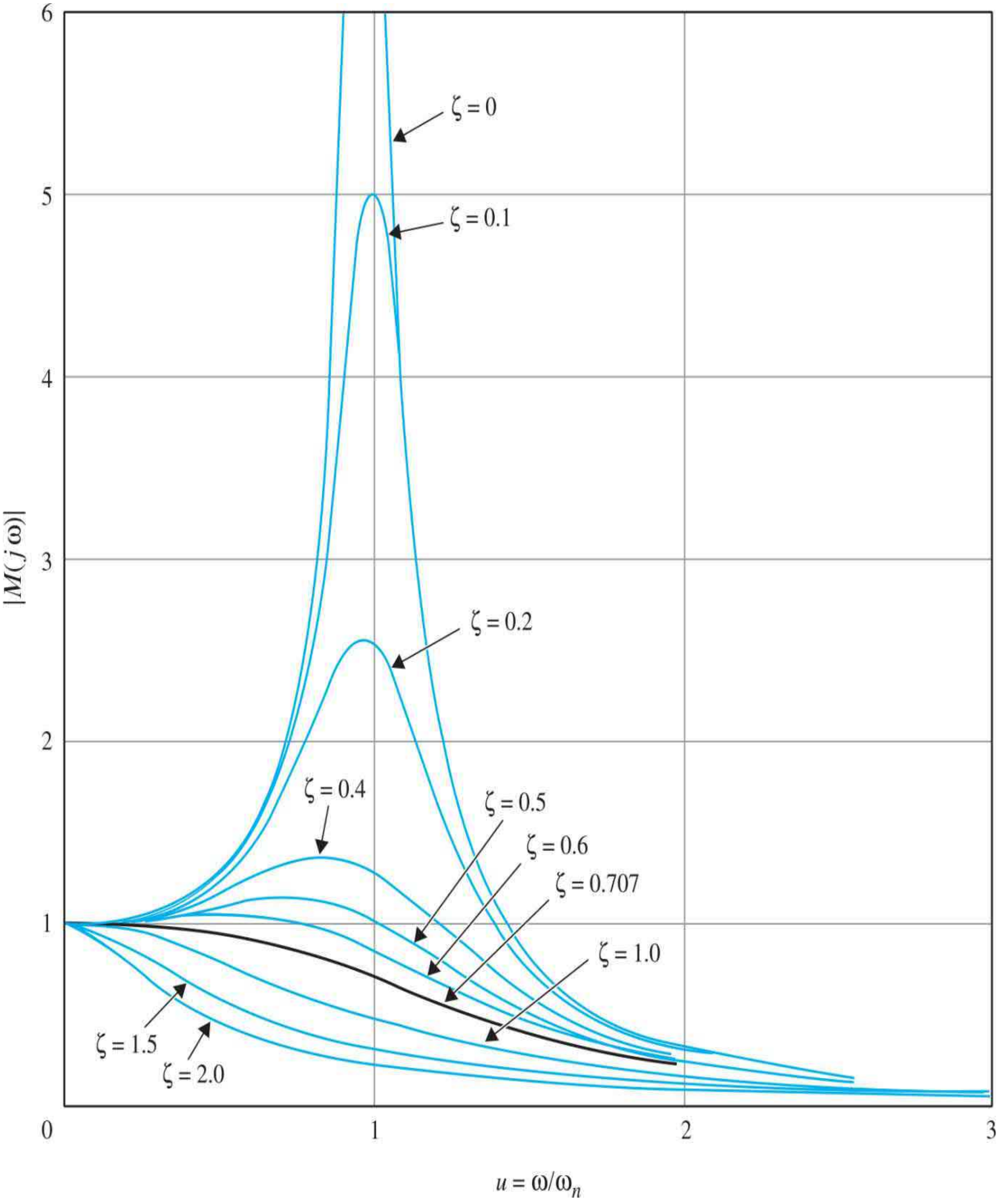


Figure 10-10 Magnification versus normalized frequency of the prototype second-order control system.

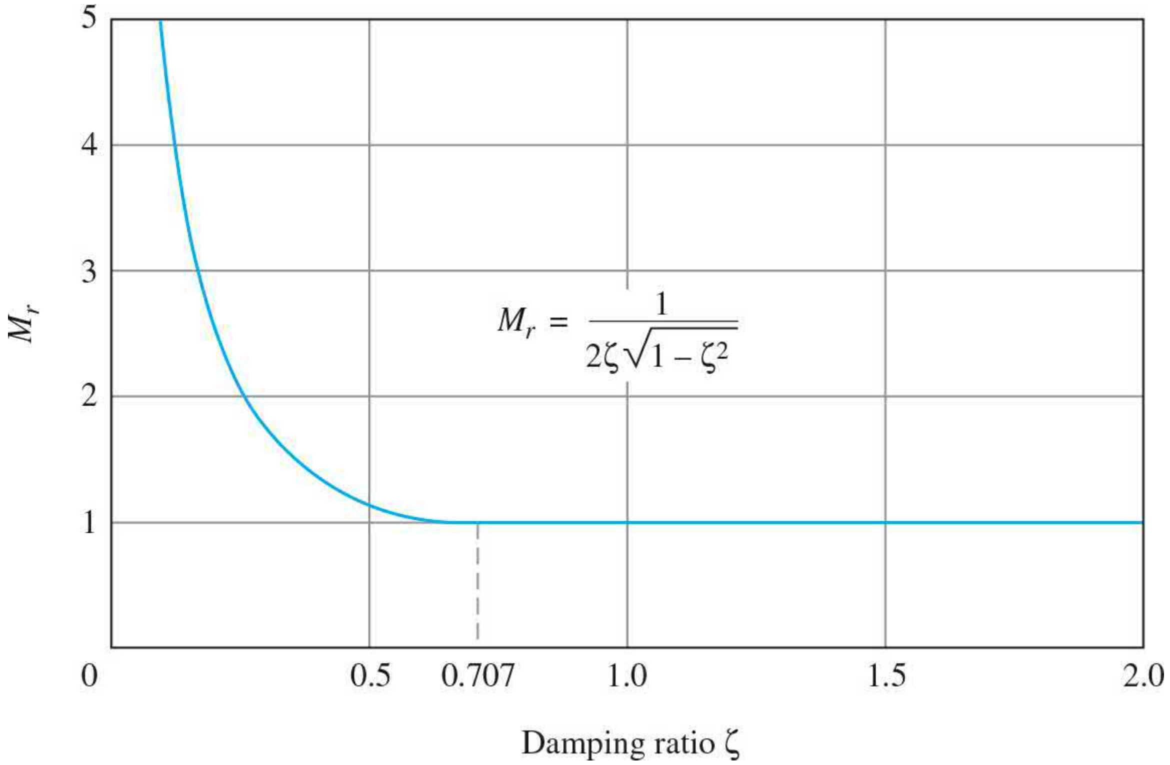


Figure 10-11 M_r versus damping ratio for the prototype second-order system.

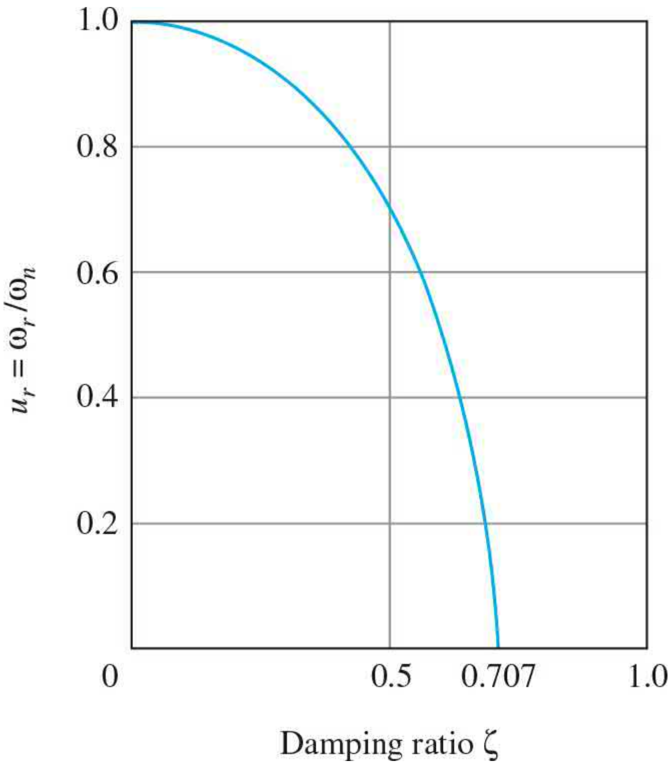


Figure 10-12 Normalized resonant frequency versus damping ratio for the prototype second-order system. $u_r = \sqrt{1-2\zeta^2}$.

10-2-2 Bandwidth

In accordance with the definition of bandwidth, we set the value of $|M(ju)|$ to $1/\sqrt{2} \cong 0.707$..

$$|M(ju)| = \frac{1}{[(1-u^2)^2 + (2\zeta u)^2]^{1/2}} = \frac{1}{\sqrt{2}} \tag{10-41}$$

Thus,

$$[(1-u^2)^2 + (2\zeta u)^2]^{1/2} = \sqrt{2} \tag{10-42}$$

- BW is directly proportional to ω_n.
- When a system is unstable, M_r no longer has any meaning.
- Bandwidth and rise time are inversely proportional to each other.

which leads to

$$u^2 = (1-2\zeta^2) \pm \sqrt{4\zeta^4 - 4\zeta^2 + 2} \tag{10-43}$$

The plus sign should be chosen in the last equation, since *u* must be a positive real quantity for any ζ. Therefore, the bandwidth of the prototype second-order system is determined from Eq. (10-43) as

$$\text{BW} = \omega_n \left[(1-2\zeta^2) + \sqrt{4\zeta^4 - 4\zeta^2 + 2} \right]^{1/2} \tag{10-44}$$

Figure 10-13 shows a plot of BW/*ωn* as a function of ζ. Notice that, as ζ increases, BW/*ωn* decreases monotonically. Even more important, Eq. (10-44) shows that BW is directly proportional to *ωn*.

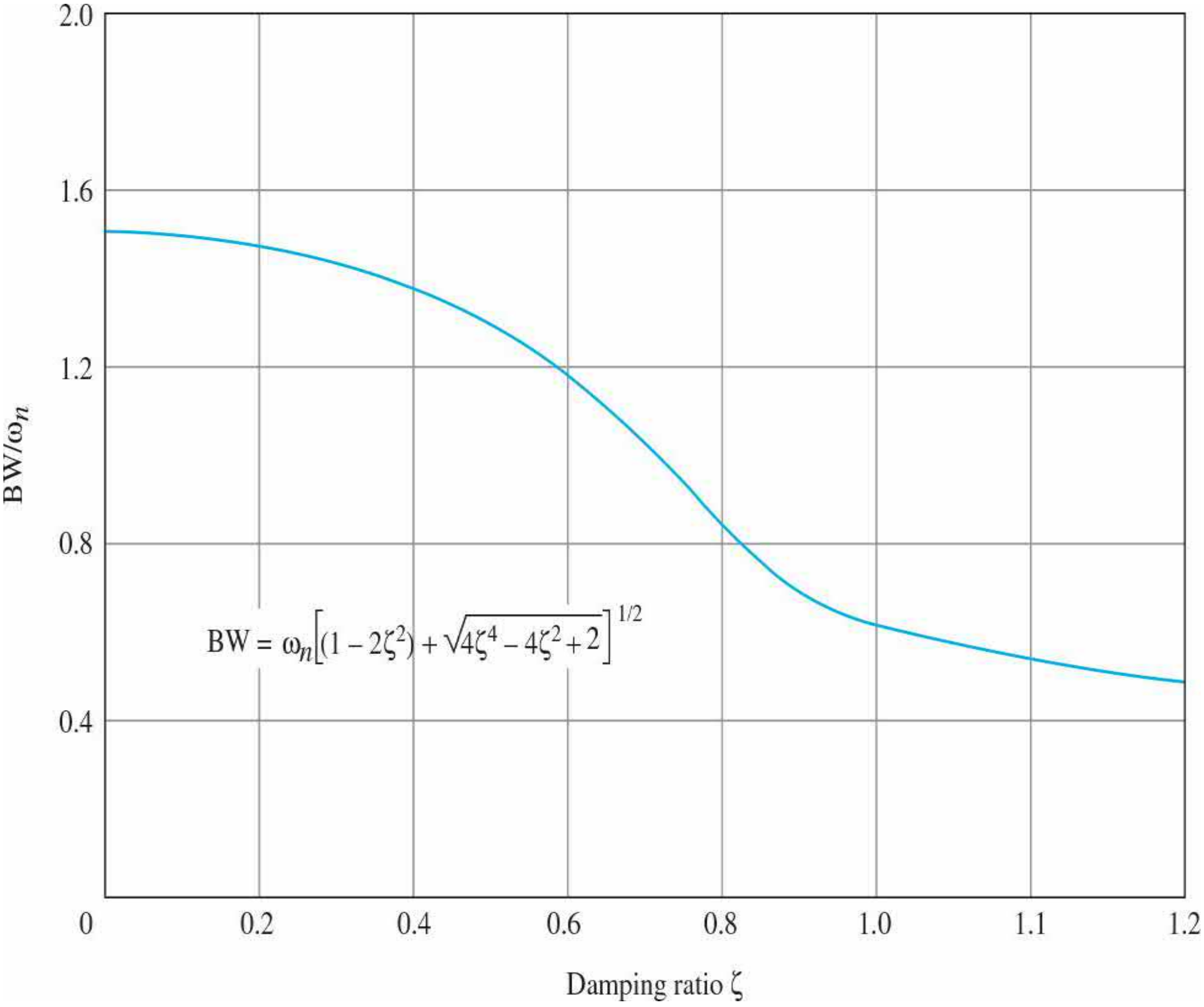


Figure 10-13 Bandwidth/ω_n versus damping ratio for the prototype second-order system.

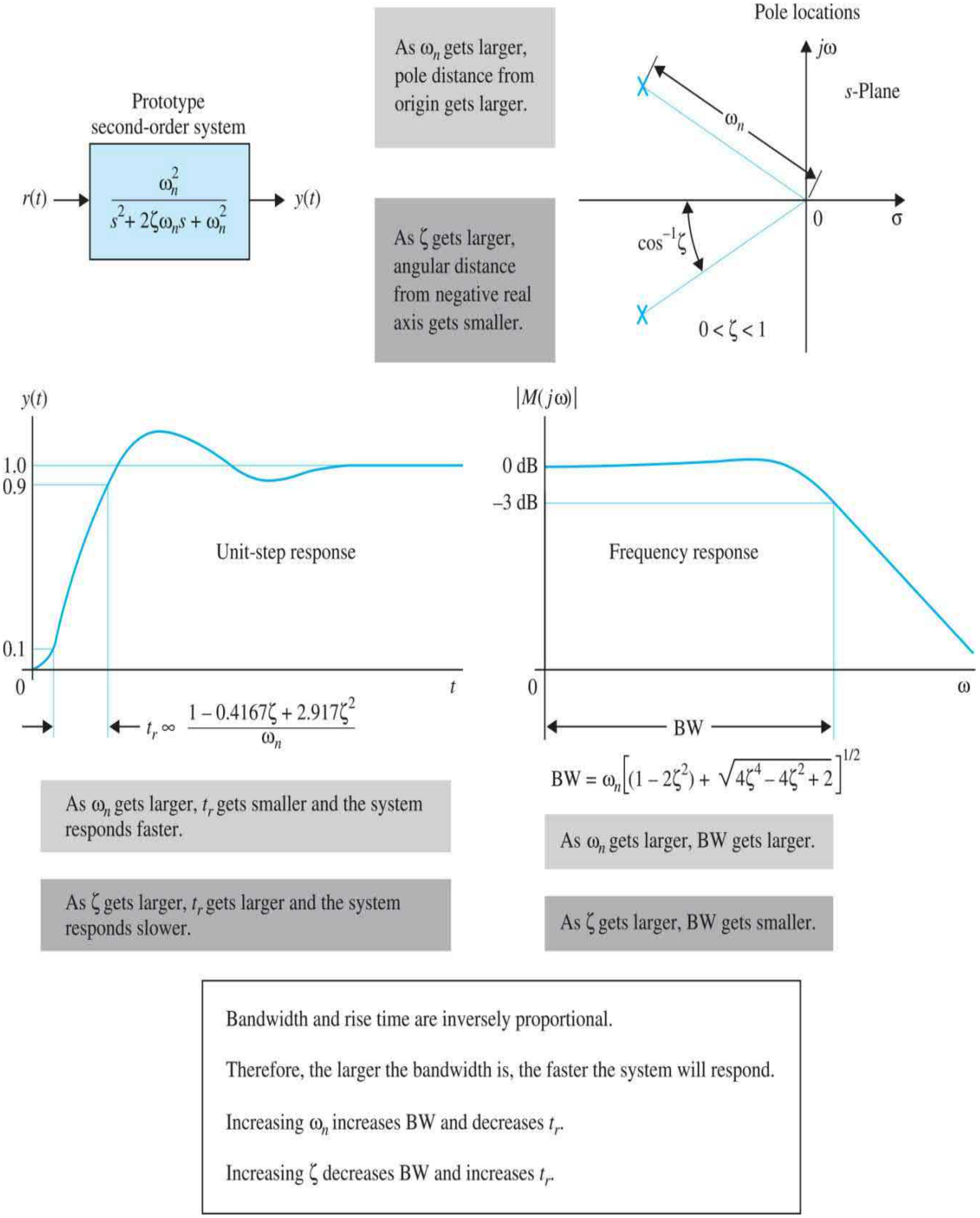
We have established some simple relationships between the time-domain response and the frequency-domain characteristics of the prototype second-order system. The summary of these relationships is as follows.

1. The resonant peak *Mr* of the closed-loop frequency response depends on ζ only [Eq. (10-40)]. When ζ is zero, *Mr* is infinite. When ζ is negative, the system is unstable, and the value of *Mr* ceases to have any meaning. As ζ increases, *Mr* decreases.
2. For ζ≥0.707, $M_r = 1$ (see Fig. 10-11), and $\omega_r = 0$ (see Fig. 10-12). In comparison with the unit-step time response, the maximum overshoot in Eq. (7-40) also depends only on ζ. However, the maximum overshoot is

zero when $\zeta \geq 1$.

3. Bandwidth is directly proportional to ωn [Eq. (10-44)]; that is, BW increases and decreases linearly with ωn. BW also decreases with an increase in ζ for a fixed ωn (see Fig. 10-13). For the unit-step response, rise time increases as ωn decreases, as demonstrated in Eq. (7-46) and Fig. 7-15. Therefore, BW and rise time are inversely proportional to each other.
4. Bandwidth and Mr are proportional to each other for $0 \zeta \leq 0.707$.

The correlations among pole locations, unit-step response, and the magnitude of the frequency response for the prototype second-order system are summarized in Fig. 10-14.



As ω_n gets larger, pole distance from origin gets larger.

As ζ gets larger, angular distance from negative real axis gets smaller.

As ω_n gets larger, t_r gets smaller and the system responds faster.

As ζ gets larger, t_r gets larger and the system responds slower.

As ω_n gets larger, BW gets larger.

As ζ gets larger, BW gets smaller.

Bandwidth and rise time are inversely proportional.

Therefore, the larger the bandwidth is, the faster the system will respond.

Increasing ω_n increases BW and decreases t_r.

Increasing ζ decreases BW and increases t_r.

Figure 10-14 Correlation among pole locations, unit-step response, and the magnitude of frequency response of the prototype second-order system.

10-3 EFFECTS OF ADDING POLES AND ZEROS TO THE FORWARD-PATH TRANSFER FUNCTION

The relationships between the time-domain and the frequency-domain responses arrived at in the preceding section apply only to the prototype second-order system described by Eq. (10-31). When other second-order or higher-order systems are involved, the relationships are different and may be more complex. It is of interest to consider the effects on the frequency-domain response when poles and zeros are added to the prototype second-order transfer function. It is simpler to study the effects of adding poles and zeros to the closed-loop transfer function; however, it is more realistic from a design standpoint to modify the forward-path transfer function.

The objective of the next two sections is to demonstrate the simple relationships between BW, *Mr*, and the time-domain response of the prototype second-order forward-path transfer function. Typical effects on BW of adding a pole and a zero to the forward-path transfer function are investigated.

10-3-1 Effects of Adding a Zero to the Forward-Path Transfer Function

The closed-loop transfer function of Eq. (10-31) may be considered as that of a unity-feedback control system with the prototype second-order forward-path transfer function

$$G(s)=\frac{\omega_n^2}{s(s+2\zeta\omega_n)} \tag{10-45}$$

Let us add a zero at s = –1/T to the transfer function so that Eq. (10-45) becomes

The general effect of adding a zero to the forward-path transfer function is to increase the BW of the closed-loop system.

$$G(s) = \frac{\omega_n^2(1+Ts)}{s(s+2\zeta\omega_n)} \tag{10-46}$$

The closed-loop transfer function is

$$M(s) = \frac{\omega_n^2(1+Ts)}{s^2+(2\zeta\omega_n+T\omega_n^2)s+\omega_n^2} \tag{10-47}$$

In principle, *Mr*, *ωr*, and BW of the system can all be derived using the same steps used in the previous section. However, because there are now three parameters in *ζ*, *ωn*, and *T*, the exact expression for *Mr*, *ωr*, and BW are difficult to obtain analytically even though the system is still second order. After a length derivation, the bandwidth of the system is found to be

$$\text{BW} = \left(-b+\frac{1}{2}\sqrt{b^2+4\omega_n^4}\right)^{1/2} \tag{10-48}$$

where

$$b = 4\zeta^2\omega_n^2+4\zeta\omega_n^3T-2\omega_n^2-\omega_n^4T^2 \tag{10-49}$$

While it is difficult to see how each of the parameters in Eq. (10-48) affects the bandwidth, Fig. 10-15 shows the relationship between BW and *T* for ζ = 0.707 and ω_n = 1.

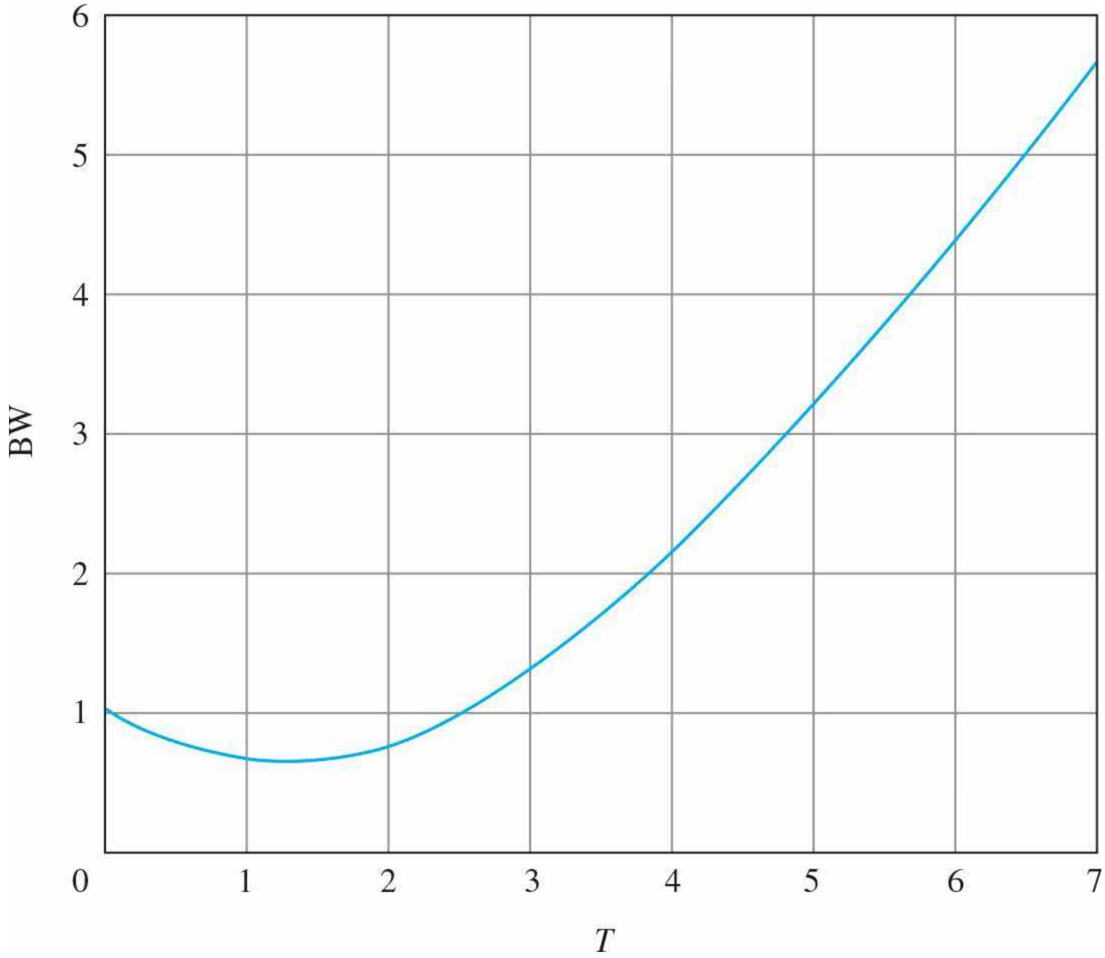
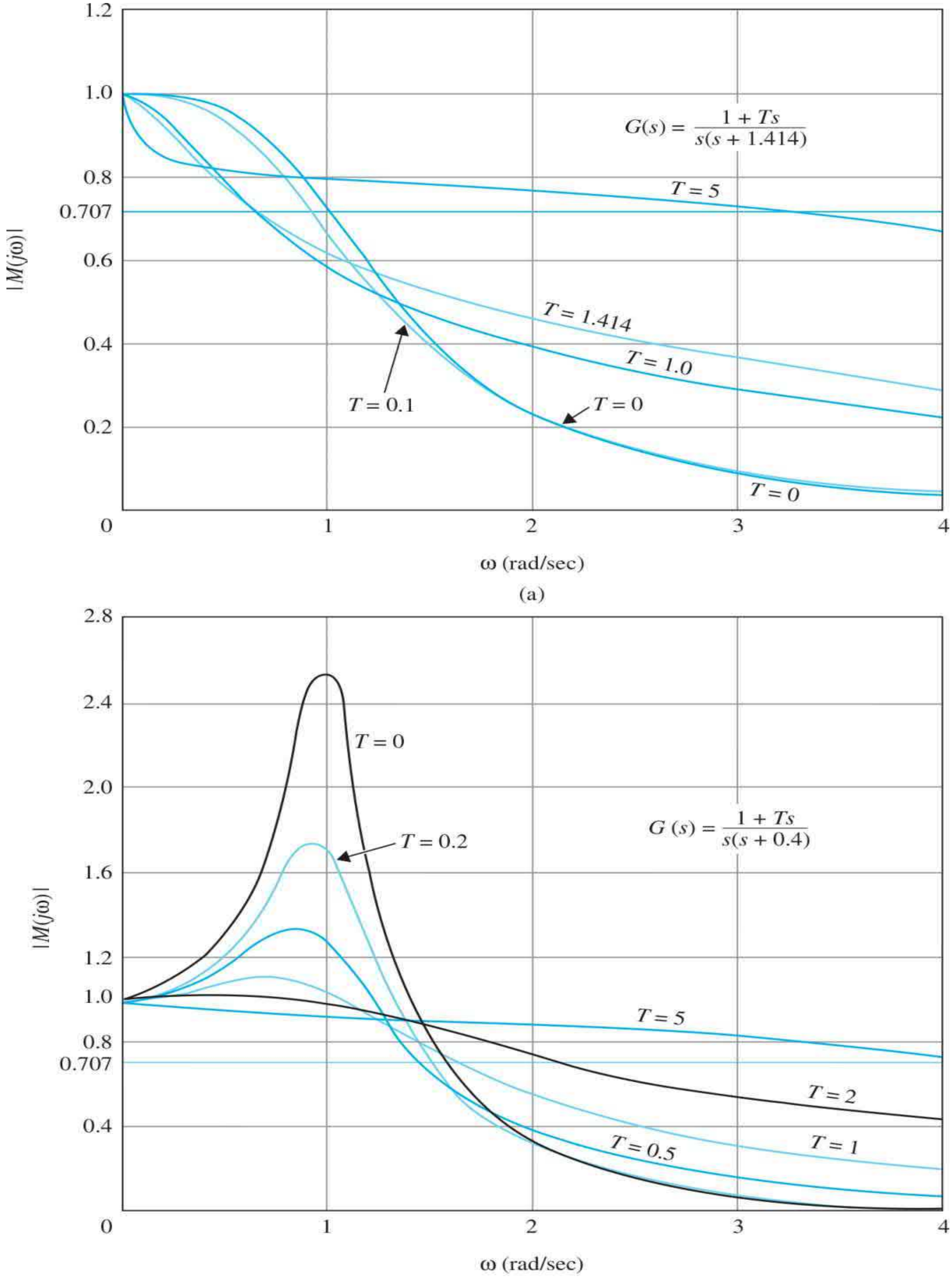


Figure 10-15 Bandwidth of a second-order system with open-loop transfer function $G(s) = (1 + Ts)/[s(s + 1.414)]$.

Notice that **the general effect of adding a zero to the forward-path transfer function is to increase the bandwidth of the closed-loop system.**

However, as shown in Fig. 10-15, over a range of small values of T, the bandwidth is actually decreased. Figure 10-16*a* and *b* gives the plots of $|M(j\omega)|$ of the closed-loop system that has the $G(s)$ of Eq. (10-46) as its forward-path transfer function: $\omega_n = 1$; $\zeta = 0.707$ and 0.2, respectively; and T takes on various values. These curves verify that the bandwidth generally increases with the increase of T by the addition of a zero to $G(s)$, except for a range of small values of T, for which BW is actually decreased.


1.2
1.0
0.8
0.707
0.6
0.4
0.2
0
1
2
3
4
|M(jω)|
ω (rad/sec)
$G(s) = \frac{1 + Ts}{s(s + 1.414)}$
T = 5
T = 1.414
T = 1.0
T = 0.1
T = 0
T = 0

(a)

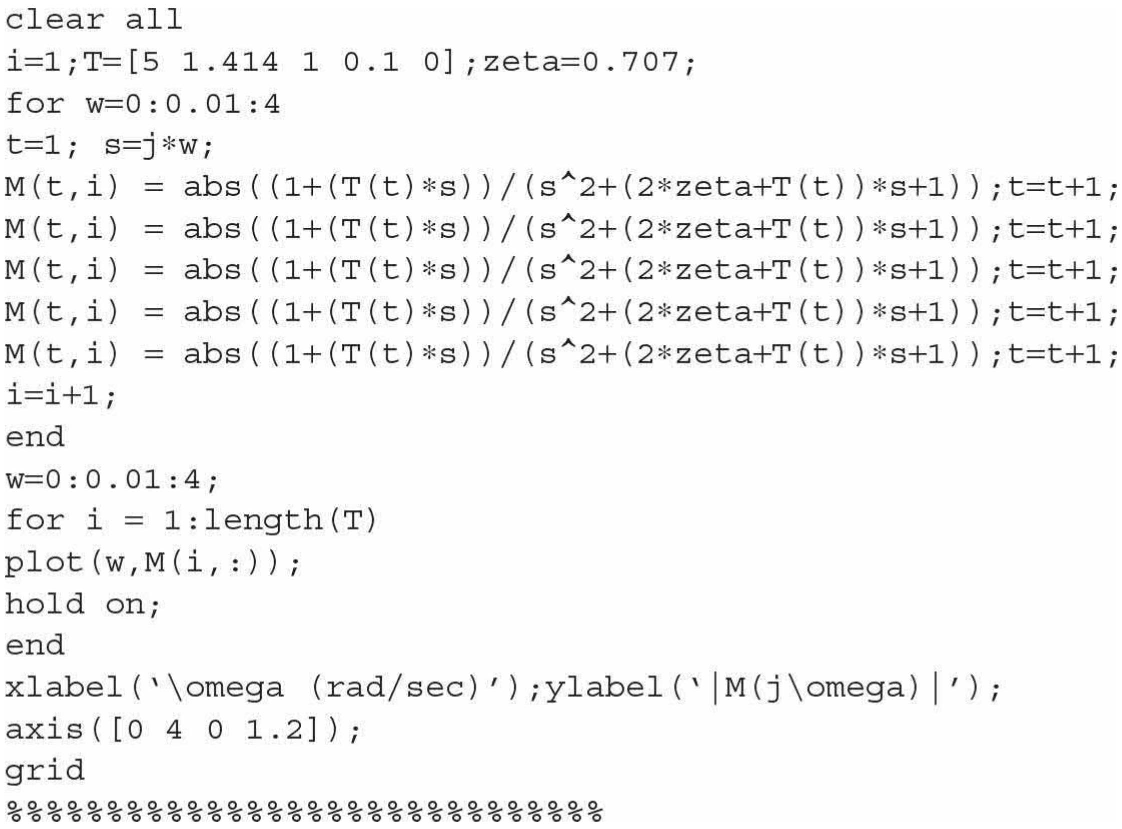
2.8
2.4
2.0
1.6
1.2
1.0
0.8
0.707
0.4
0
1
2
3
4
|M(jω)|
ω (rad/sec)
$G\ (s) = \frac{1 + Ts}{s(s + 0.4)}$
T = 0
T = 0.2
T = 5
T = 2
T = 1
T = 0.5

(b)

Figure 10-16 Magnification curves for the second-order system with the forward-path transfer function *G*(*s*) in Eq. (10-46). (*a*) ω*n* = 1, ζ = 0.707 (*b*) ω*n* = 1, ζ = 0.2.

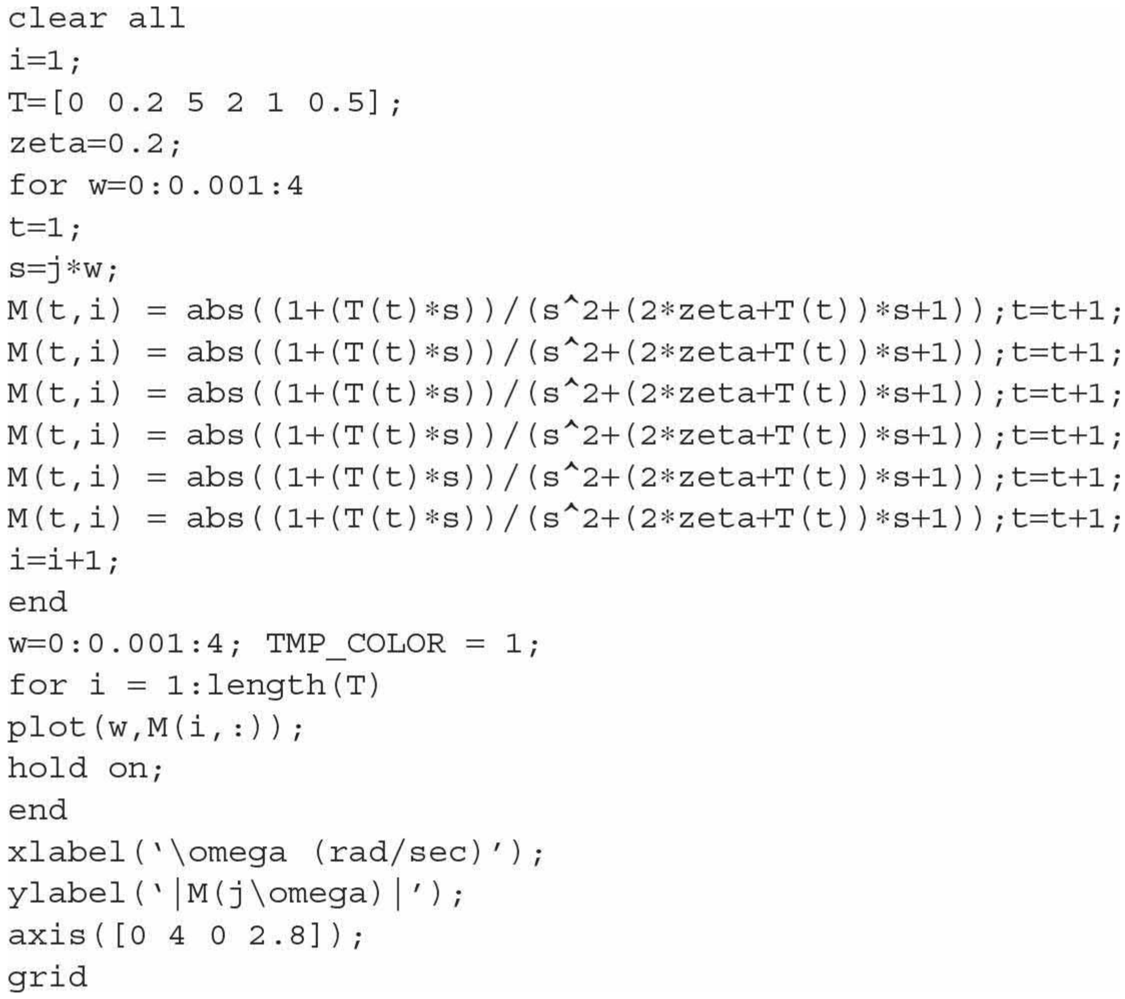
Toolbox 10-3-1

MATLAB code for Fig. 10-16*a*.



```
clear all
i=1;T=[5 1.414 1 0.1 0];zeta=0.707;
for w=0:0.01:4
t=1; s=j*w;
M(t,i) = abs((1+(T(t)*s))/(s^2+(2*zeta+T(t))*s+1));t=t+1;
M(t,i) = abs((1+(T(t)*s))/(s^2+(2*zeta+T(t))*s+1));t=t+1;
M(t,i) = abs((1+(T(t)*s))/(s^2+(2*zeta+T(t))*s+1));t=t+1;
M(t,i) = abs((1+(T(t)*s))/(s^2+(2*zeta+T(t))*s+1));t=t+1;
M(t,i) = abs((1+(T(t)*s))/(s^2+(2*zeta+T(t))*s+1));t=t+1;
i=i+1;
end
w=0:0.01:4;
for i = 1:length(T)
plot(w,M(i,:));
hold on;
end
xlabel(‘\omega (rad/sec)’);ylabel(‘|M(j\omega)|’);
axis([0 4 0 1.2]);
grid
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

MATLAB code for Fig. 10-16*b*.



```
clear all
i=1;
T=[0 0.2 5 2 1 0.5];
zeta=0.2;
for w=0:0.001:4
t=1;
s=j*w;
M(t,i) = abs((1+(T(t)*s))/(s^2+(2*zeta+T(t))*s+1));t=t+1;
M(t,i) = abs((1+(T(t)*s))/(s^2+(2*zeta+T(t))*s+1));t=t+1;
M(t,i) = abs((1+(T(t)*s))/(s^2+(2*zeta+T(t))*s+1));t=t+1;
M(t,i) = abs((1+(T(t)*s))/(s^2+(2*zeta+T(t))*s+1));t=t+1;
M(t,i) = abs((1+(T(t)*s))/(s^2+(2*zeta+T(t))*s+1));t=t+1;
M(t,i) = abs((1+(T(t)*s))/(s^2+(2*zeta+T(t))*s+1));t=t+1;
i=i+1;
end
w=0:0.001:4; TMP_COLOR = 1;
for i = 1:length(T)
plot(w,M(i,:));
hold on;
end
xlabel(‘\omega (rad/sec)’);
ylabel(‘|M(j\omega)|’);
axis([0 4 0 2.8]);
grid
```

Figures 10-17 and 10-18 show the corresponding unit-step responses of the closed-loop system. These curves show that a high bandwidth corresponds to a faster rise time. However, as *T* become very large, the zero of the closed-loop transfer function, which is at $s = -1/T$, moves very close to the origin, causing the system to have a large time constant. Thus, Fig. 10-17 illustrates the situation that the rise time is fast, but the large time constant of the zero near the origin of the *s*-plane causes the time response to drag out in reaching the final steady state (i.e., the settling time will be longer).

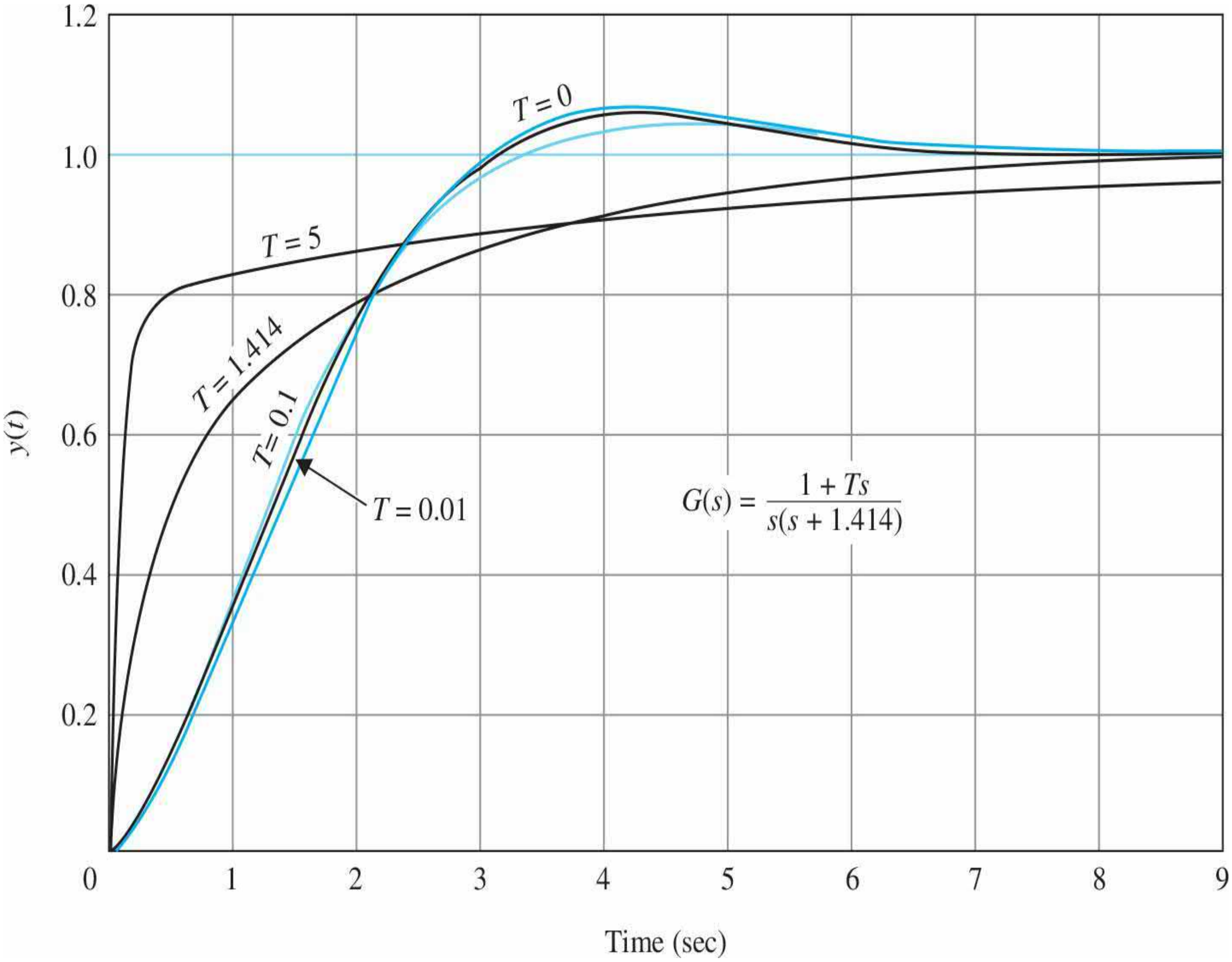


Figure 10-17 Unit-step responses of a second-order system with a forward-path transfer function $G(s)$.

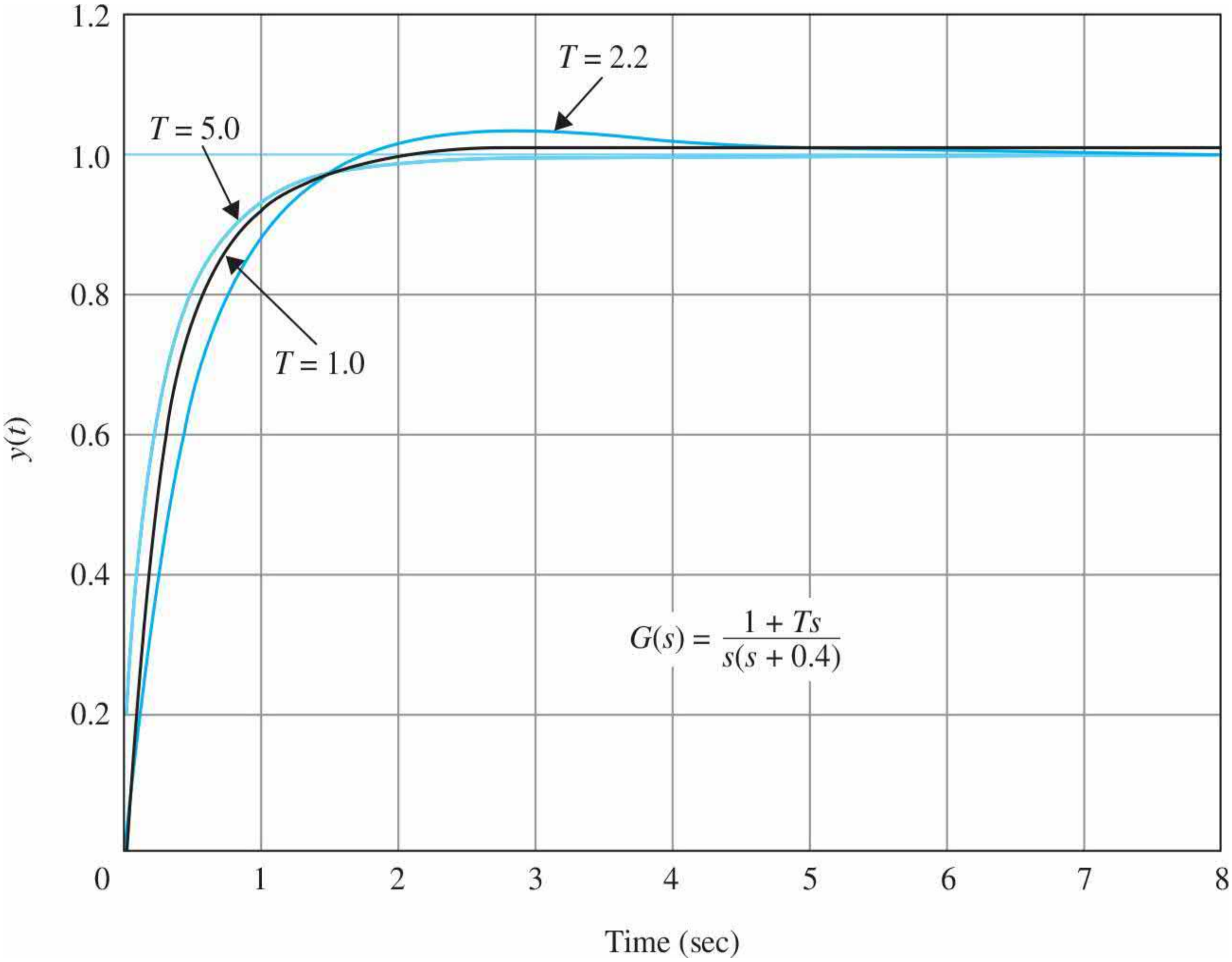
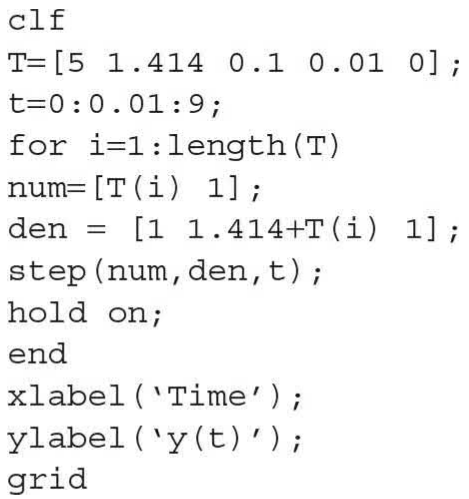


Figure 10-18 Unit-step responses of a second-order system with a forward-path transfer function $G(s)$.

Toolbox 10-3-2

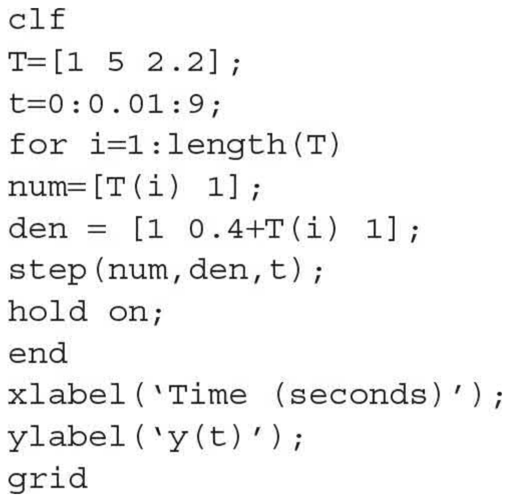
MATLAB code for Fig. 10-17—use clear all, close all, and clc if necessary.



```
clf
T=[5 1.414 0.1 0.01 0];
t=0:0.01:9;
for i=1:length(T)
num=[T(i) 1];
den = [1 1.414+T(i) 1];
step(num,den,t);
hold on;
end
xlabel(‘Time’);
ylabel(‘y(t)’);
grid
```

Toolbox 10-3-3

MATLAB code for Fig. 10-18—use clear all, close all, and clc if necessary.



```
clf
T=[1 5 2.2];
t=0:0.01:9;
for i=1:length(T)
num=[T(i) 1];
den = [1 0.4+T(i) 1];
step(num,den,t);
hold on;
end
xlabel(‘Time (seconds)’);
ylabel(‘y(t)’);
grid
```

10-3-2 Effects of Adding a Pole to the Forward-Path Transfer Function

Adding a pole at $s = -1/T$ to the forward-path transfer function of Eq. (10-45) leads to

$$G(s) = \frac{\omega_n^2}{s(s+2\zeta\omega_n)(1+Ts)} \tag{10-50}$$

> Adding a pole to the forward-path transfer function makes the closed-loop system less stable, and decreases the bandwidth.

The derivation of the bandwidth of the closed-loop system with $G(s)$ given in Eq. (10-50) is quite tedious. We can obtain a qualitative indication on the bandwidth properties by referring to Fig. 10-19, which shows the plots of $|M(j\omega)|$ ω for $\omega_n = 1, \zeta = 0.707$, and versus values of T. Because the system is now of the third order, it can be unstable for a certain set of system parameters. It can be shown that, for $\omega_n = 1$ and ζ, the system is stable for all positive values of T. The $|M(j\omega)|$-versus-ω curves of Fig. 10-19 show that, for small values of T, the bandwidth of the system is slightly increased by the addition of the pole, but Mr is also increased. When T becomes large, the pole added to $G(s)$ has the effect of decreasing the bandwidth but increasing Mr. Thus, we can conclude that, in general, **the effect of adding a pole to the forward-path transfer function is to make the closed-loop system less stable while decreasing the bandwidth.**

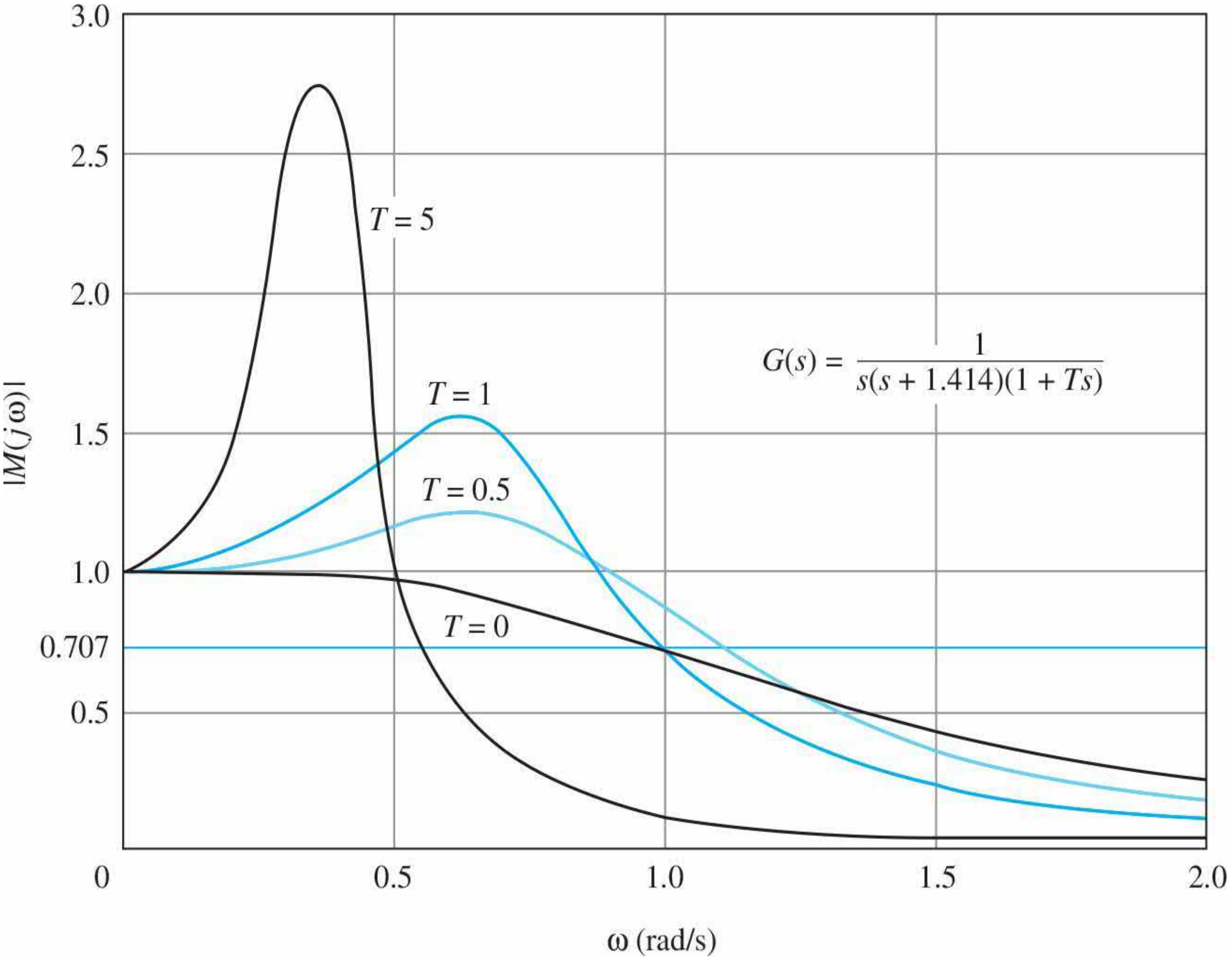


Figure 10-19 Magnification curves for a third-order system with a forward-path transfer function $G(s)$.

The unit-step responses of Fig. 10-20 show that, for larger values of T, T = 1 and T = 5, the following relations are observed:

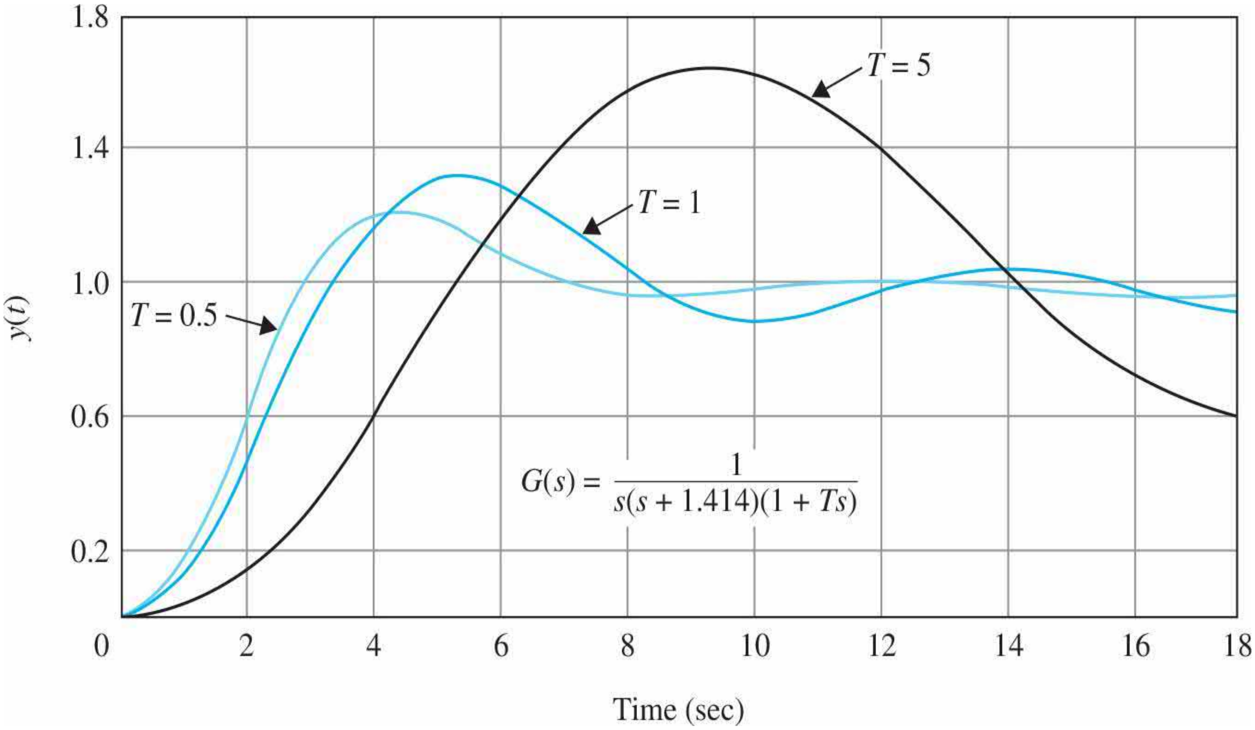


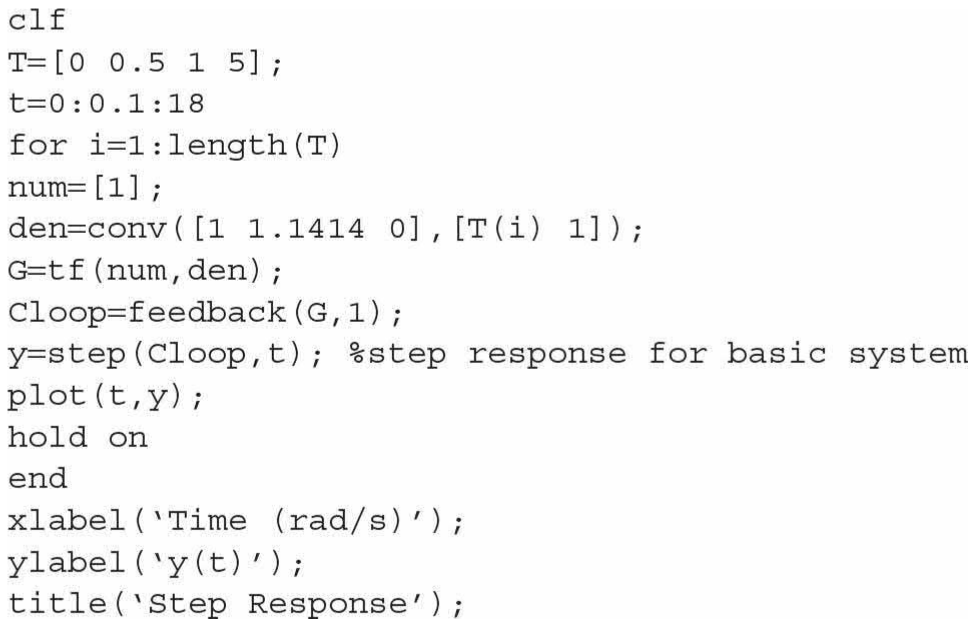
Figure 10-20 Unit-step responses of a third-order system with a forward-path transfer function $G(s)$.

1. The rise time increases with the decrease of the bandwidth.
2. The larger values of *Mr* also correspond to a larger maximum overshoot in the unit-step responses.

The correlation between *Mr* and the maximum overshoot of the step response is meaningful only when the system is stable. When $G(j\omega) = -1|M(j\omega)|$ is infinite, and the closed-loop system is marginally stable. On the other hand, when the system is unstable, the value of $|M(j\omega)|$ is analytically finite, but it no longer has any significance.

Toolbox 10-4-1

MATLAB code for Fig. 10-12—use clear all, close all, and clc if necessary.



```
clf
T=[0 0.5 1 5];
t=0:0.1:18
for i=1:length(T)
num=[1];
den=conv([1 1.1414 0],[T(i) 1]);
G=tf(num,den);
Cloop=feedback(G,1);
y=step(Cloop,t); %step response for basic system
plot(t,y);
hold on
end
xlabel(‘Time (rad/s)’);
ylabel(‘y(t)’);
title(‘Step Response’);
```

When $M_r = \infty$, the closed-loop system is marginally stable. When the system is unstable, M_r no longer has any meaning.

10-4 NYQUIST STABILITY CRITERION: FUNDAMENTALS

- The Nyquist plot of $L(j\omega)$ is done in polar coordinates as ω varies from 0 to ∞.
- The Nyquist criterion also gives indication on relative stability

Thus far we have presented two methods of determining the stability of linear SISO systems: the Routh-Hurwitz criterion and the root-locus method of determining stability by locating the roots of the characteristic equation in the s-plane. Of course, if the coefficients of the characteristic equation are all known, we can solve for the roots of the equation by use of MATLAB.

The Nyquist criterion is a semigraphical method that determines the stability of a closed-loop system by investigating the properties of the frequency-domain plot, the **Nyquist plot,** of the loop transfer function $G(s)H(s)$, or $L(s)$. Specifically, the Nyquist plot of $L(s)$ is a plot of $L(j\omega)$ in the polar coordinates of Im[$L(j\omega)$] versus Re[$L(j\omega)$] as ω varies from 0 to ∞. This is another example of using the properties of the loop transfer function to find the performance of the closed-loop system. The Nyquist criterion has the following features that make it an alternative method that is attractive for the analysis and design of control systems.

1. In addition to providing the absolute stability, like the Routh-Hurwitz criterion, the Nyquist criterion also gives information on the relative stability of a stable system and the degree of instability of an unstable system. It also gives an indication of how the system stability may be improved, if needed.
2. The Nyquist plot of $G(s)H(s)$ or of $L(s)$ is very easy to obtain, especially with the aid of a computer.
3. The Nyquist plot of $G(s)H(s)$ gives information on the frequency-domain characteristics such as Mr, ωr, BW, and others with ease.
4. The Nyquist plot is useful for systems with pure time delay that cannot be treated with the Routh-Hurwitz criterion and are difficult to analyze with the root-locus method.

This subject is also treated in App. G for the general case where the loop transfer function is of nonminimum-phase type.

10-4-1 Stability Problem

The Nyquist criterion represents a method of determining the location of the characteristic equation roots with respect to the left half and the right half of the s-plane. Unlike the root-locus method, the Nyquist criterion does not give the exact location of the characteristic equation roots.

Let us consider that the closed-loop transfer function of a SISO system is

$$M(s)=\frac{G(s)}{1+G(s)H(s)} \tag{10-51}$$

where $G(s)H(s)$ can assume the following form

$$G(s)H(s) = \frac{K(1+T_1 s)(1+T_2 s)\cdots(1+T_m s)}{s^p(1+T_a s)(1+T_b s)\cdots(1+T_n s)} e^{-T_d s} \tag{10-52}$$

where the *T*'s are real or complex-conjugate coefficients, and *Td* is a real time delay.[1]

$$G(s)H(s) = \frac{K(1+T_1 s)(1+T_2 s)\cdots(1+T_m s)}{s^p(1+T_a s)(1+T_b s)\cdots(1+T_n s)}$$

Because the characteristic equation is obtained by setting the denominator polynomial of *M*(*s*) to zero, the roots of the characteristic equation are also the zeros of 1 + *G*($_s$)*H*(*s*). Or, the characteristic equation roots must satisfy

$$\Delta(s) = 1 + G(s)H(s) = 0 \tag{10-53}$$

In general, for a system with multiple number of loops, the denominator of *M*(*s*) can be written as

$$\Delta(s) = 1 + L(s) = 0 \tag{10-54}$$

where *L*(*s*) is the loop transfer function and is of the form of Eq. (10-52).

Before embarking on the details of the Nyquist criterion, it is useful to summarize the pole-zero relationships of the various system transfer functions.

Identification of Poles and Zeros

- Loop transfer function zeros: zeros of *L*(*s*).
- Loop transfer function poles: poles of *L*(*s*).
- Closed-loop transfer function poles: zeros of 1 + *L*(*s*) = roots of the characteristic equation poles of 1 + *L*(*s*) = poles of *L*(*s*).

Stability Conditions

We define two types of stability with respect to the system configuration.

- ***Open-loop stability.*** A system is said to be **open-loop stable** if the poles of the loop transfer function *L*(*s*) are all in the left-half *s*-plane. For a single-loop system, this is equivalent to the system being stable when the loop is opened at any point.

- ***Closed-loop stability.*** A system is said to be **closed-loop stable**, or simply stable, if the poles of the closed-loop transfer function or the zeros of $1 + L(s)$ are all in the left-half s-plane. Exceptions to the above definitions are systems with poles or zeros intentionally placed at $s = 0$.

10-4-2 Definitions of Encircled and Enclosed

Because the Nyquist criterion is a graphical method, we need to establish the concepts of encircled and enclosed, which are used for the interpretation of the Nyquist plots for stability.

Encircled

A point or region in a complex function plane is said to be encircled by a closed path if it is found inside the path.

For example, point A in Fig. 10-21 is encircled by the closed path Γ because A is *inside* the closed path. Point B is not encircled by the closed path Γ because it is *outside* the path. Furthermore, when the closed path Γ has a direction assigned to it, the encirclement, if made, can be in the clockwise (CW) or the counterclockwise (CCW) direction. As shown in Fig. 10-21, point A is encircled by Γ in the CCW direction. We can say that the region *inside* the path is encircled in the prescribed direction, and the region *outside* the path is not encircled.

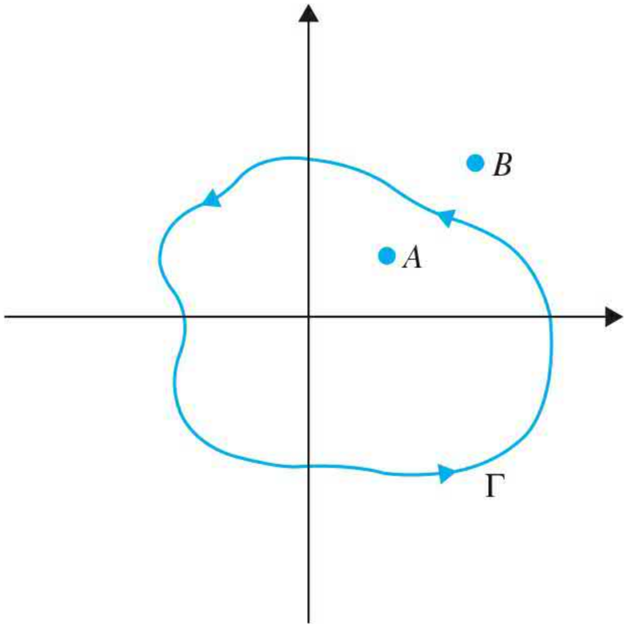


Figure 10-21 Definition of encirclement.

Enclosed

A point or region is said to be enclosed by a closed path if it is encircled in the CCW direction or the point or region lies to the left of the path when the path is traversed in the prescribed direction.

The concept of enclosure is particularly useful if only a portion of the closed path is shown. For example, the shaded regions in Fig. 10-22*a* and *b* are considered to be *enclosed* by the closed path Γ. In other words, point *A* in Fig. 10-22*a* is *enclosed* by Γ, but point *A* in Fig. 10-22*b* is not. However, point *B* and all the points in the shaded region outside Γ in Fig. 10-22*b* are *enclosed*.

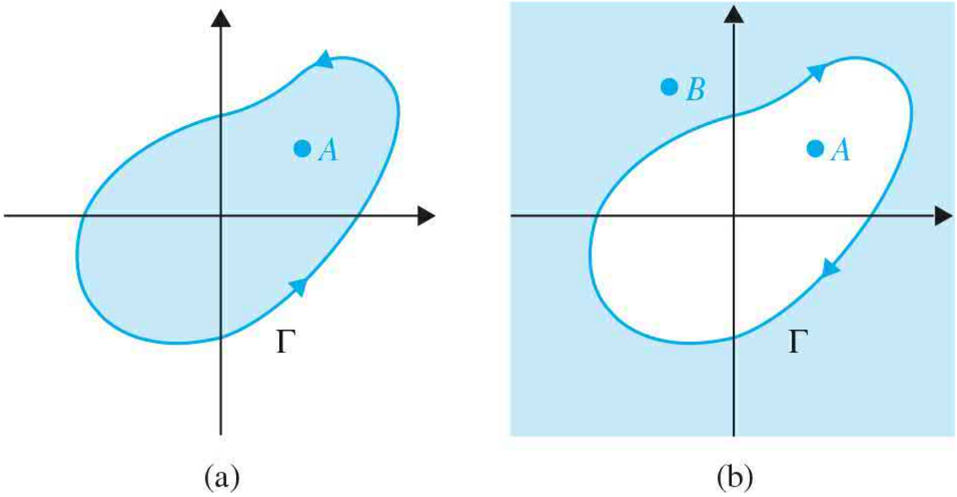


Figure 10-22 Definition of enclosed points and regions. (*a*) Point *A* is enclosed by *G*. (*b*) Point *A* is not enclosed, but *B* is enclosed by the locus *G*.

10-4-3 Number of Encirclements and Enclosures

When a point is encircled by a closed path Γ, a number *N* can be assigned to the number of times it is encircled. The magnitude of *N* can be determined by drawing an arrow from the point to any arbitrary point s_1 on the closed path Γ and then letting s_1 follow the path in the prescribed direction until it returns to the starting point. The total *net* number of revolutions traversed by this arrow is *N*, or the net angle is $2\pi N$ radians. For example, point *A* in Fig. 10-23*a* is *encircled once* or 2π radians by Γ, and point *B* is *encircled twice* or 4π radians, all in the CW direction. In Fig. 10-23*b*, point *A* is *enclosed once*, and point *B* is *enclosed twice* by Γ. By definition, *N* is positive for CCW encirclement and negative for CW encirclement.

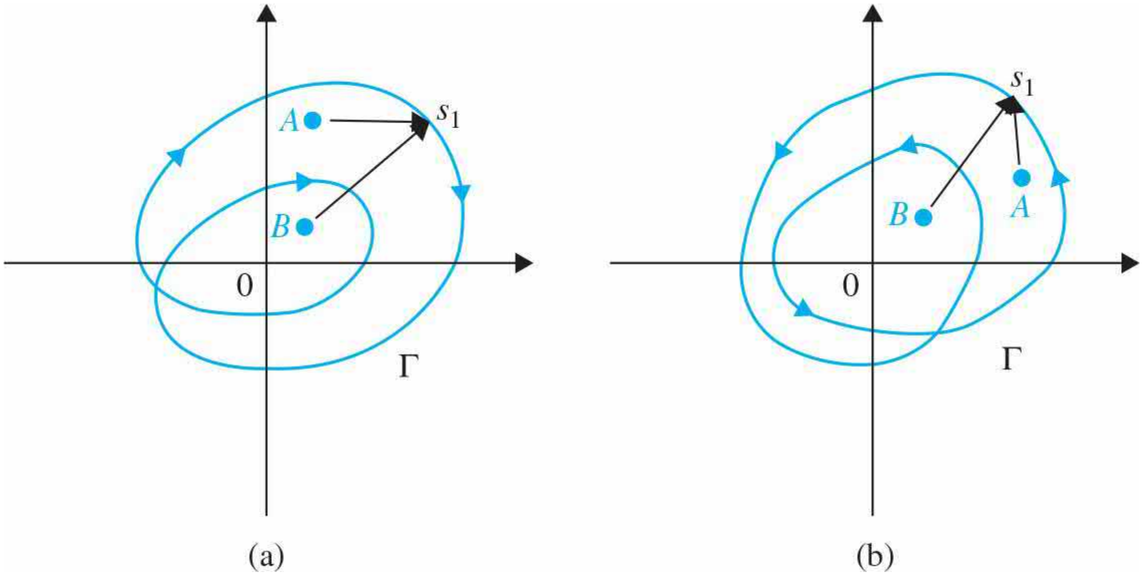


Figure 10-23 Definition of the number of encirclements and enclosures.

10-4-4 Principles of the Argument

The Nyquist criterion was originated as an engineering application of the well-known "principle of the argument" concept in complex-variable theory. The principle is stated in the following in a heuristic manner.

Let Δ(*s*) be a single-valued function of the form of the right-hand side of Eq. (10-52), which has a finite number of poles in the *s*-plane. Single valued means that, for each point in the *s*-plane, there is one and only one corresponding point, including infinity, in the complex Δ(*s*)-plane. As defined in Chap. 9, infinity in the complex plane is interpreted as a point.

Do not attempt to relate Δ(*s*) with *L*(*s*). They are not the same.

Suppose that a continuous closed path Γ_s is arbitrarily chosen in the *s*-plane, as shown in Fig. 10-24*a*. If Γ_s does not go through any poles of Δ(*s*), then the trajectory ΓΔ mapped by Δ(*s*) into the Δ(*s*)-plane is also a closed one, as shown in Fig. 10-24*b*. Starting from a point s_1, the Γ_s locus is traversed in the arbitrarily chosen direction (CW in the illustrated case), through the points s_2 and s_3, and then returning to s_1 after going through all the points on the Γ_s locus, as shown in Fig. 10-24*a*, the corresponding Γ_Δ locus will start from the point $\Delta(s_1)$ and go through points $\Delta(s_2)$ and $\Delta(s_3)$, corresponding to s_1, s_2, and s_3, respectively, and finally return to the starting point, $\Delta(s_1)$. The direction of traverse of Γ_Δ can be either CW or CCW, that is, in the same direction or the opposite direction as that of Γ_s, depending on the function $\Delta(s)$. In Fig. 10-24*b*, the direction of Γ_Δ is arbitrarily assigned, for illustration purposes, to be CCW.

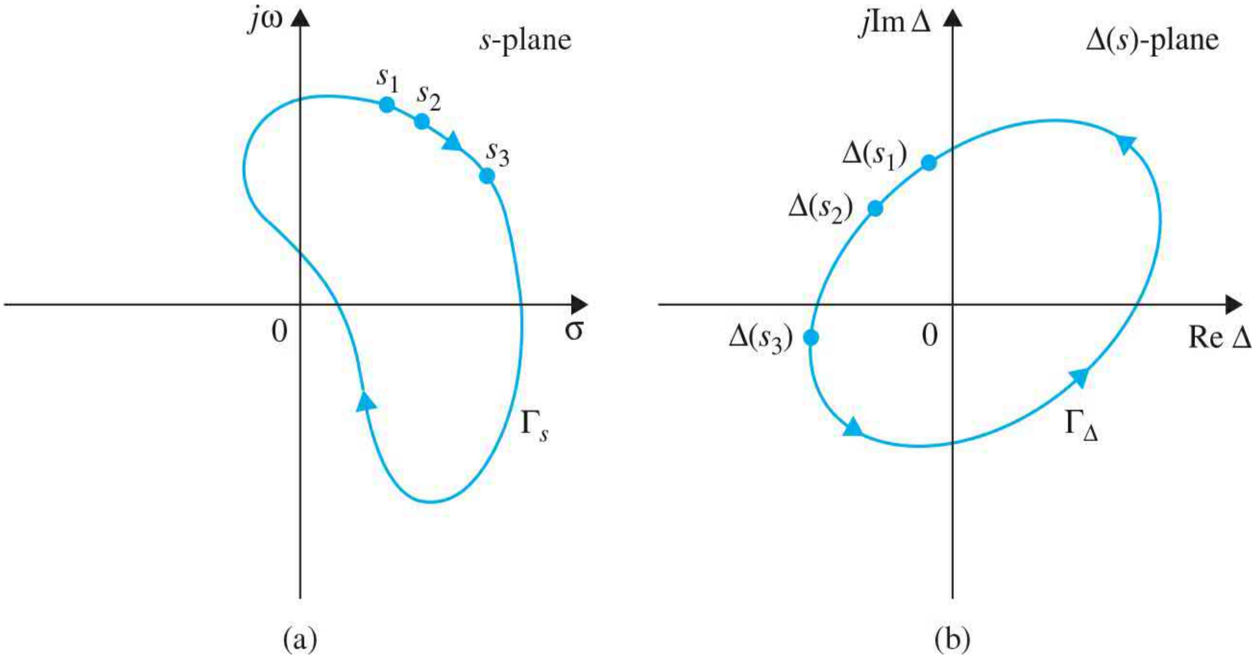


Figure 10-24 (*a*) Arbitrarily chosen closed path in the *s*-plane. (*b*) Corresponding locus Γ_s in the $\Delta(s)$-plane.

Although the mapping from the *s*-plane to the $\Delta(s)$-plane is single-valued, the reverse process is not a single-valued mapping. For example, consider the function

$$\Delta(s) = \frac{K}{s(s+1)(s+2)} \qquad (10\text{-}55)$$

which has poles $s = 0, -1$, and -2 in the *s*-plane. For each point in the *s*-plane, there is only one corresponding point in the $\Delta(s)$-plane. However, for each point in the $\Delta(s)$-plane, the function maps into three corresponding points in the *s*-plane. The simplest way to illustrate this is to write Eq. (10-55) as

$$s(s+1)(s+2)-\frac{K}{\Delta(s)}=0 \tag{10-56}$$

If Δ(*s*) is a real constant, which represents a point on the real axis in the Δ(*s*)-plane, the third-order equation in Eq. (10-56) gives three roots in the *s*-plane. The reader should recognize the parallel of this situation to the root-locus diagram that essentially represents the mapping of Δ(*s*) = −1 + *j*0 onto the loci of roots of the characteristic equation in the *s*-plane, for a given value of *K*. Thus, the root loci of Eq. (10-55) have three individual branches in the *s*-plane.

The principle of the argument can be stated:

Let Δ(*s*) **be a single-valued function that has a finite number of poles in the *s*-plane. Suppose that an arbitrary closed path** Γ_s **is chosen in the *s*-plane so that the path does not go through any one of the poles or zeros of** Γ(*s*)**; the corresponding Γ*ρ* locus mapped in the** Δ(*s*)**-plane will encircle the origin as many times as the difference between the number of zeros and poles of** Δ(*s*) **that are encircled by the *s*-plane locus** Γ_s**.**

In equation form, the principle of the argument is stated as

$$N=Z-P \tag{10-57}$$

where N = number of encirclements of the origin made by the Δ(*s*)-plane locus Γ_Δ
Z = number of zeros of Δ(*s*) encircled by the *s*-plane locus Γ_s in the *s*-plane
P = number of poles of Δ(*s*) encircled by the *s*-plane locus Γ_s in the *s*-plane

In general, *N* can be positive ($Z > P$), zero ($Z = P$), or negative ($Z < P$). These three situations are described in more detail as follows:

1. $N > 0$ ($Z > P$). If the *s*-plane locus encircles more zeros than poles of Δ(*s*) in a certain prescribed direction (CW or CCW), *N* is a positive integer. In this case, the Δ(*s*)-plane locus Γ_Δ will encircle the origin of the Δ(*s*)-plane *N* times in the same direction as that of Γ_s.
2. $N = 0$($Z = P$). If the *s*-plane locus encircles as many poles as zeros, or no poles and zeros, of Δ(*s*), the Δ(*s*)-plane locus Γ_Δ will not encircle the origin of the Δ(*s*)-plane.

3. $N<(Z<P)$. If the *s*-plane locus encircles more poles than zeros of $\Delta(s)$ in a certain direction, *N* is a negative integer. In this case, the $\Delta(s)$-plane locus Γ_Δ will encircle the origin *N* times in the *opposite* direction as that of Γ_s.

A convenient way of determining *N* with respect to the origin (or any point) of the $\Delta(s)$-plane is to draw a line from the point in any direction to a point as far as necessary; the number of *net* intersections of this line with the $\Delta(s)$ locus gives the magnitude of *N*. Figure 10-25 gives several examples of this method of determining *N*. In these illustrated cases, it is assumed that the Γ_s locus has a CCW sense.

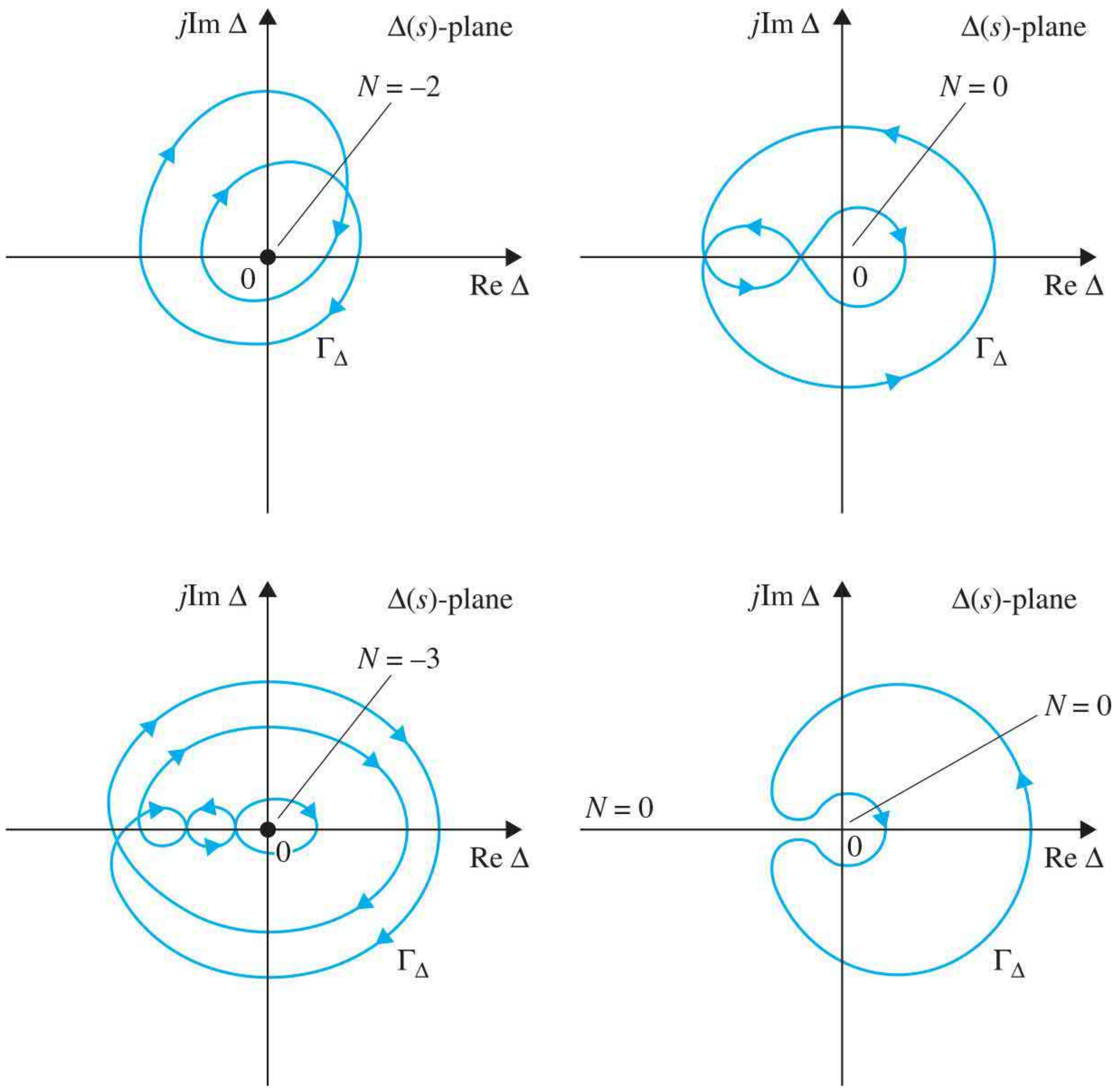


Figure 10-25 Examples of the determination of N in the Δ(s)-plane.

Critical Point

For convenience, we shall designate the origin of the Δ(s)-plane as the **critical point** from which the value of N is determined. Later, we shall designate other points in the complex-function plane as critical points, dependent on the way the Nyquist criterion is applied.

A rigorous proof of the principle of the argument is not given here. The following illustrative example may be considered a heuristic explanation of the principle.

Let us consider the function Δ(*s*) is of the form

$$\Delta(s) = \frac{K(s+z_1)}{(s+p_1)(s+p_2)} \tag{10-58}$$

where *K* is a positive real number. The poles and zeros of Δ(*s*) are assumed to be as shown in Fig. 10-19*a*. The function Δ(*s*) can be written as

$$\begin{aligned}\Delta(s) &= |\Delta(s)|\angle\Delta(s)\\ &= \frac{K|s+z_1|}{|s+p_1||s+p_2|}[\angle(s+z_1)-\angle(s+p_1)-\angle(s+p_2)]\end{aligned} \tag{10-59}$$

Figure 10-26 *a* shows an arbitrarily chosen trajectory Γ_s in the *s*-plane, with the arbitrary point s_1 on the path, and Γ_s does not pass through any of the poles and the zeros of Δ(*s*). The function Δ(*s*) evaluated at $s = s_1$ is

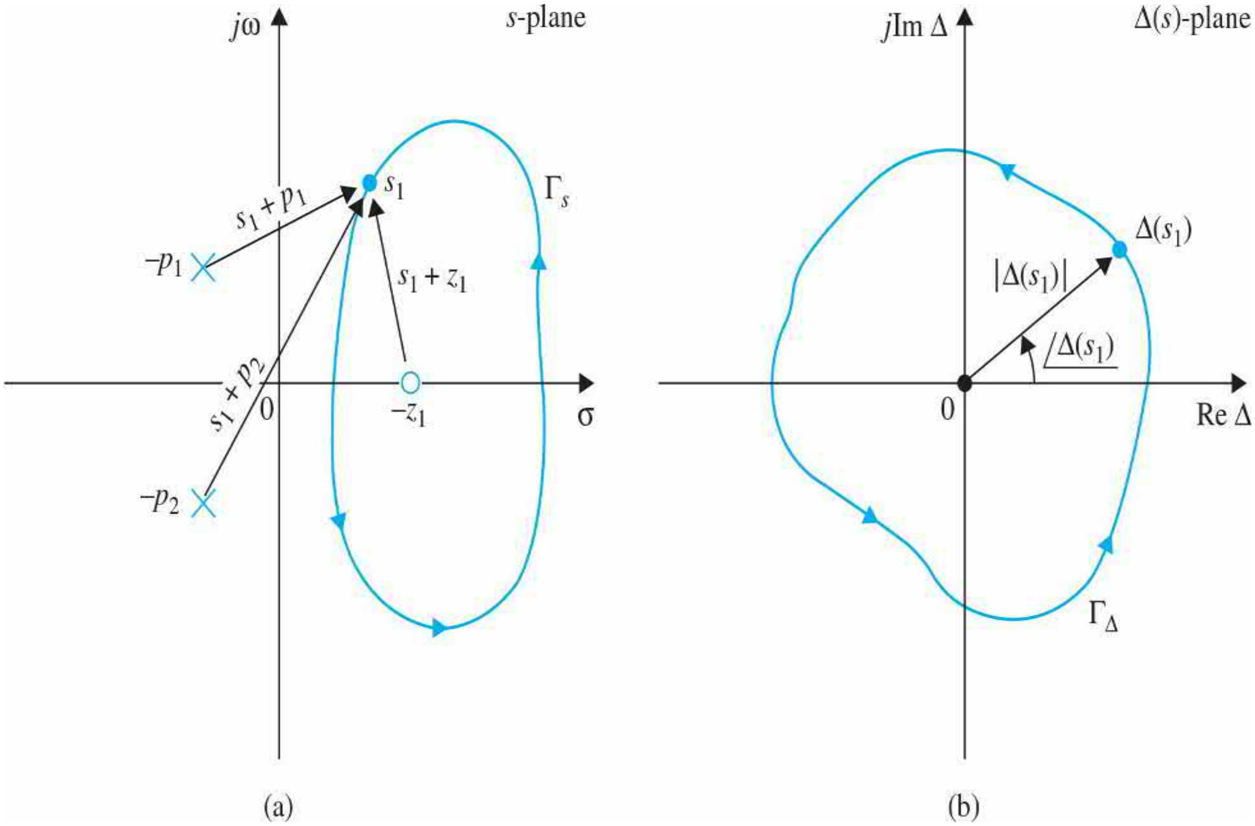


Figure 10-26 (*a*) Pole-zero configuration of Δ(s) in Eq. (10-59) and the s-plane trajectory Γ_s. (*b*) Δ(s)-plane locus Γ_Δ, which corresponds to the Γ_s locus

of (a) through the mapping of Eq. (10-59).

> Z and P refer to only the zeros and poles, respectively, of $\Delta(s)$ that are encircled by Γ_s.

The term $(s_1 + z_1)$ can be represented graphically by the vector drawn from $-z_1$ to s_1. Similar vectors can be drawn for $(s_1 + P_1)$ and $(s + P_2)$. Thus, $\Delta(s_1)$ is represented by the vectors drawn from the finite poles and zeros of $\Delta(s)$ to the point s_1, as shown in Fig. 10-26*a*. Now, if the point s_1 is moved along the locus Γ_s in the prescribed CCW direction until it returns to the starting point, the angles generated by the vectors drawn from the two poles that are not encircled by Γ_s when s_1 completes one roundtrip are zero, whereas the vector $(s_1 + z_1)$ drawn from the zero at $-z_1$, which is encircled by Γ_s, generates a positive angle (CCW) of 2π radians, which means that the corresponding $\Delta(s)$ plot must go around the origin 2π radians, or one revolution, in the CCW direction, as shown in Fig. 10-26*b*. This is why only the poles and zeros of $\Delta(s)$ that are inside the Γ_s trajectory in the s-plane will contribute to the value of N of Eq. (10-57). Because the poles of $\Delta(s)$ contribute to a negative phase, and zeros contribute to a positive phase, the value of N depends only on the difference between Z and P. For the case illustrated in Fig. 10-26*a*, $Z = 1$ and $P = 0$.

Thus,

$$N = Z - P = 1 \tag{10-61}$$

which means that the $\Delta(s)$-plane locus Γ_Δ should encircle the origin once in the same direction as that of the s-plane locus Γ_s . It should be kept in mind that Z and P refer only to the zeros and poles, respectively, of $\Delta(s)$ that are encircled by Γ_s and not the total number of zeros and poles of $\Delta(s)$.

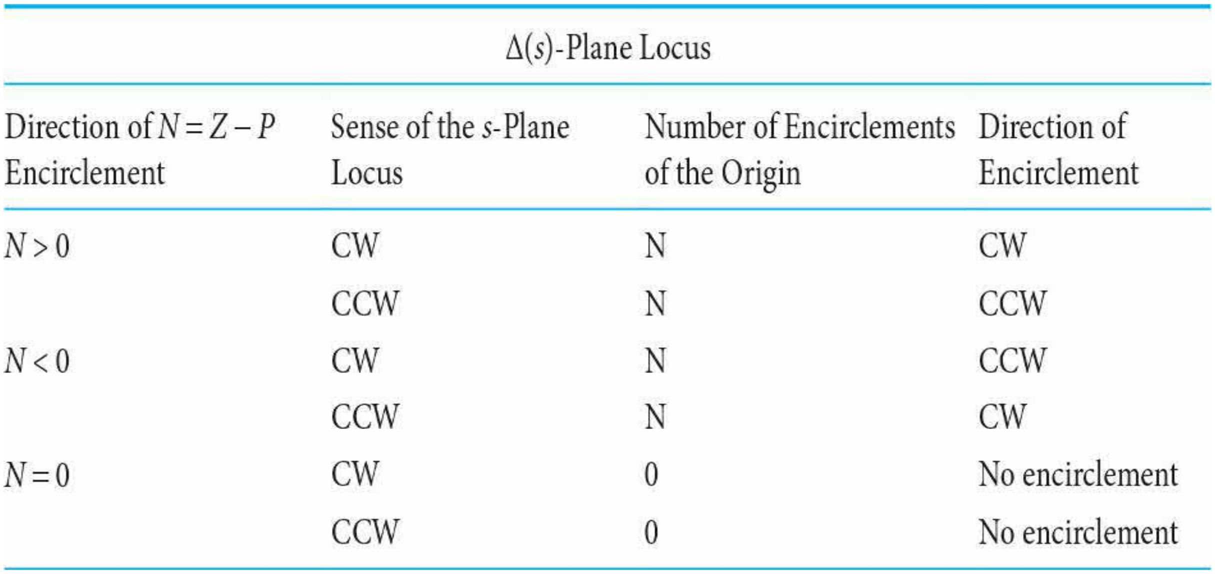
In general, the net angle traversed by the $\Delta(s)$-plane locus, as the s-plane locus is traversed once in any direction, is equal to

$$2\pi(Z - P) = 2\pi N \text{ radians} \tag{10-62}$$

This equation implies that if there are N more zeros than poles of $\Delta(s)$, which are encircled by the s-plane locus Γ_s, in a prescribed direction, the $\Delta(s)$-plane locus will encircle the origin N times in the *same direction* as that of Γ_s. Conversely, if N more poles than zeros are encircled by Γ_s in a given direction, N in Eq. (10-62) will be negative, and the $\Delta(s)$-plane locus must encircle the origin N times in the *opposite direction* to that of Γ_s.

A summary of all the possible outcomes of the principle of the argument is given in Table 10-2.

TABLE 10-2 Summary of All Possible Outcomes of the Principle of the Argument



		$\Delta(s)$-Plane Locus	
Direction of $N = Z - P$ Encirclement	Sense of the s-Plane Locus	Number of Encirclements of the Origin	Direction of Encirclement
$N > 0$	CW	N	CW
	CCW	N	CCW
$N < 0$	CW	N	CCW
	CCW	N	CW
$N = 0$	CW	0	No encirclement
	CCW	0	No encirclement

10-4-5 Nyquist Path

Years ago when Nyquist was faced with solving the stability problem, which involves determining if the function $\Delta(s) = 1 + (s)$ has zeros in the right-half s-plane, he apparently discovered that the principle of the argument could be applied to solve the stability problem if the s-plane locus Γ_s is taken to be one that encircles the entire right half of the s-plane. Of course, as an alternative, G_s can be chosen to encircle the entire left-half s-plane, as the solution is a relative one. Figure 10-27 illustrates a G_s locus with a CCW sense that encircles the entire right half of the s-plane. This path is chosen to be the s-plane trajectory G_s for the Nyquist criterion, since in mathematics, CCW is traditionally defined to be the positive sense. The path G_s shown in Fig. 10-27 is defined to be the **Nyquist path**. Because the Nyquist path must not pass through any poles and zeros of Δ(*s*), the small semicircles shown along the *jω*-axis in Fig. 10-27 are used to indicate that the path should go around these poles and zeros if they fall on the *jω*-axis. It is apparent that, if any pole or zero of Δ(*s*) lies inside the right-half *s*-plane, it will be encircled by the Nyquist path G_s.

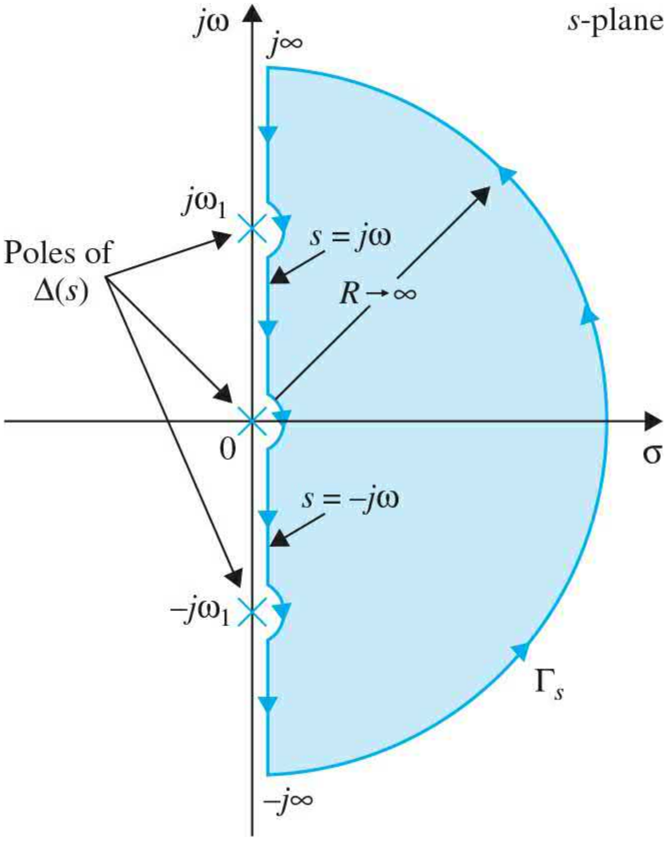


Figure 10-27 Nyquist path.

10-4-6 Nyquist Criterion and the *L*(*s*) or the *G*(*s*)*H*(*s*) Plot

The Nyquist path is defined to encircle the entire right-half *s*-plane.

The Nyquist criterion is a direct application of the principle of the argument when the *s*-plane locus is the Nyquist path of Fig. 10-27. In principle, once the Nyquist path is specified, the stability of a closed-loop system can be determined by plotting the Δ(*s*) = 1 + *L*(*s*) locus when *s* takes on values along the Nyquist path and investigating the behavior of the Δ(*s*) plot with respect to the **critical point,** which in this case is the origin of the $\Delta(s)$-plane.

Because the function $L(s)$ is generally known, **it would be simpler to construct the $L(s)$ plot that corresponds to the Nyquist path, and the same conclusion on the stability of the closed-loop system can be obtained by observing the behavior of the $L(s)$ plot with respect to the $(-1, j0)$ point in the $L(s)$-plane.**

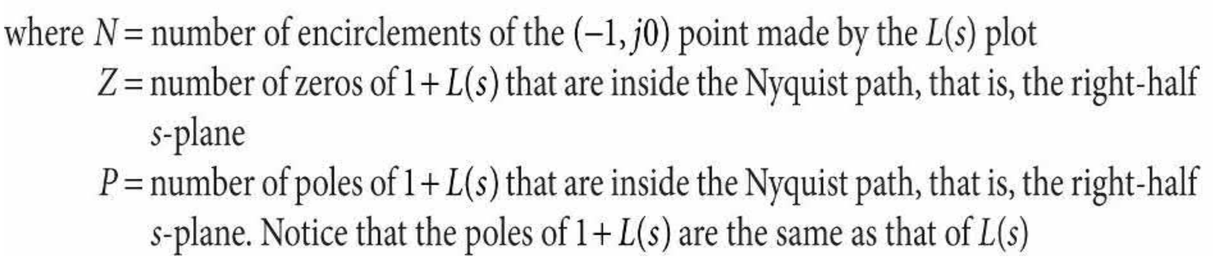
This is because the origin of the $\Delta(s) = 1 + L(s)$ plane corresponds to the $(-1,j0)$ point in the $L(s)$-plane. Thus the $(-1, j0)$ point in the $L(s)$-plane becomes the critical point for the determination of closed-loop stability.

For single-loop systems, $L(s) = G(s)H(s)$, the previous development leads to the determination of the closed-loop stability by investigating the behavior of the $G(s)H(s)$ plot with respect to the $(-1, j0)$ point of the $G(s)H(s)$-plane. Thus, the Nyquist stability criterion is another example of using the loop transfer function properties to find the behavior of closed-loop systems.

Thus, given a control system that has the characteristic equation given by equating the numerator polynomial of $1 + L(s)$ to zero, where $L(s)$ is the loop transfer function, the application of the Nyquist criterion to the stability problem involves the following steps.

1. The Nyquist path G_s is defined in the s-plane, as shown in Fig. 10-27.
2. The $L(s)$ plot corresponding to the Nyquist path is constructed in the $L(s)$-plane.
3. The value of N, the number of encirclement of the $(-1, j0)$ point made by the $L(s)$ plot, is observed.
4. The Nyquist criterion follows from Eq. (10-57),

$$N = Z - P \tag{10-63}$$



where N = number of encirclements of the $(-1, j0)$ point made by the $L(s)$ plot

Z = number of zeros of $1 + L(s)$ that are inside the Nyquist path, that is, the right-half s-plane

P = number of poles of $1 + L(s)$ that are inside the Nyquist path, that is, the right-half s-plane. Notice that the poles of $1 + L(s)$ are the same as that of $L(s)$

The stability requirements for the two types of stability defined earlier are interpreted in terms of *Z* and *P*.

For closed-loop stability, *Z* must equal zero. For open-loop stability, *P* must equal zero.

Thus, the condition of stability according to the Nyquist criterion is stated as

$$N = -P \tag{10-64}$$

That is, **for a closed-loop system to be stable, the *L*(*s*) plot must encircle the (−1, *j* 0) point as many times as the number of poles of *L* (*s*) that are in the right-half *s*-plane, and the encirclement, if any, must be made in the clockwise direction** (**if** Γ_s is defined in the CCW sense).

10-5 NYQUIST CRITERION FOR SYSTEMS WITH MINIMUM-PHASE TRANSFER FUNCTIONS

We shall first apply the Nyquist criterion to systems with *L*(*s*) that are **minimum-phase transfer functions.** The properties of the minimum-phase transfer functions are described in App. G and are summarized as follows:

1. A minimum-phase transfer function does not have poles or zeros in the right-half *s*-plane or on the *jω*-axis, excluding the origin.
2. For a minimum-phase transfer function *L*(*s*) with *m* zeros and *n* poles, excluding the poles at *s* = 0, when *s* = *j* ω and as ω varies from ∞ to 0, the total phase variation of *L*(*j*ω) is (*n* − *m*)π/2 rad.
3. The value of a minimum-phase transfer function cannot become zero or infinity at any finite nonzero frequency.
4. A nonminimum-phase transfer function will always have a more positive phase shift as *ω* varies from ∞ to 0. Or, equally true, it will always have a more negative phase shift as *ω* varies from 0 to ∞.

- A minimum-phase transfer function does not have poles or zeros in the right-half *s*-plane or on the *jω*-axis, except at *s* = 0.

- For $L(s)$ that is minimum-phase type, Nyquist criterion can be checked by plotting the segment of $L(j\omega)$ from $\omega = \infty$ to 0.

Because a majority of the loop transfer functions encountered in the real world satisfy condition 1 and are of the minimum-phase type, it would be prudent to investigate the application of the Nyquist criterion to this class of systems. As if turns out, this is quite simple.

Because a minimum-phase $L(s)$ does not have any poles or zeros in the right-half s-plane or on the $j\omega$-axis (except at $s = 0$) $P = 0$, and the poles of $\Delta(s) = 1+ L(s)$ also have the same properties. Thus, the Nyquist criterion for a system with $L(s)$ being a minimum-phase transfer function is simplified to

$$N = 0 \tag{10-65}$$

Thus, the Nyquist criterion can be stated: **For a closed-loop system with loop transfer function $L(s)$ that is of minimum-phase type, the system is closed-loop stable if the plot of $L(s)$ that corresponds to the Nyquist path does not encircle the critical point $(-1, j0)$ in the $L(s)$-plane.**

Furthermore, if the system is unstable, $Z \neq 0$; N in Eq. (10-65) would be a positive integer, which means that the critical point $(- j0)$ is **enclosed** N times (corresponding to the direction of the Nyquist path defined here). Thus, the Nyquist criterion of stability for systems with minimum-phase loop transfer functions can be further simplified:**For a closed-loop system with loop transfer function $L(s)$that is of minimum-phase type, the system is closed-loop stable if the $L(s)$plot that corresponds to the Nyquist path does not enclose the $(-1, j0)$ point. If the $(-1, j0)$ point is enclosed by the Nyquist plot, the system is unstable.**

Because the region that is enclosed by a trajectory is defined as the region that lies to the left when the trajectory is traversed in the prescribed direction, the *Nyquist criterion can be checked simply by plotting the segment of* $L(j\omega)$ *from $\omega = \infty$ to 0, or, points on the positive $j\omega$-axis*. This simplifies the procedure considerably, since the plot can be made easily on a computer. The only drawback to this method is that the Nyquist plot that corresponds to the $j\omega$-axis tells only whether the critical point is enclosed or not and, if it is, not how many times. Thus, if the system is found to be unstable, the enclosure property does not give information on how many roots of the characteristic equation are in the right-half *s*-plane. However, in practice, this information is not vital. From this point on, we shall define the *L*(*jω*) plot that corresponds to the positive *jω*-axis of the *s*-plane as the Nyquist plot of *L*(*s*).

10-5-1 Application of the Nyquist Criterion to Minimum-Phase Transfer Functions That Are Not Strictly Proper

Just as in the case of the root locus, it is often necessary in design to create an equivalent loop transfer function *Leq*(*s*) so that a variable parameter *K* will appear as a multiplying factor in *Leq*(*s*), that is, $L(s) = KL_{eq}(s)$. Because the equivalent loop transfer function does not correspond to any physical entity, it may not have more poles than zeros, and the transfer function is not strictly proper, as defined in App. G. In principle, there is no difficulty in constructing the Nyquist plot of a transfer function that is not strictly proper, and the Nyquist criterion can be applied for stability studies without any complications. However, some computer programs may not be prepared for handling improper transfer functions, and it may be necessary to reformulate the equation for compatibility with the computer program. To examine this case, consider that the characteristic equation of a system with a variable parameter *K* is conditioned to

$$1+KL_{\mathrm{eq}}(s)=0 \tag{10-66}$$

If *Leq*(*s*) does not have more poles than zeros, we can rewrite Eq. (10-51) as

$$1+\frac{1}{KL_{eq}(s)}=0 \tag{10-67}$$

by dividing both sides of the equation by *KLeq*(*s*). Now we can plot the Nyquist plot of 1/*Leq*(*s*), and the critical point is still (−1, *j*0) for $K > 0$. The variable parameter on the Nyquist plot is now 1/*K*. Thus, with this minor adjustment, the Nyquist criterion can still be applied.

The Nyquist criterion presented here is cumbersome when the loop transfer function is of the nonminimum-phase type, for example, when *L*(*s*) has poles or/and zeros in the right-half *s*-plane. A generalized Nyquist criterion that will take care of transfer functions of all types is presented in App. G.

10-6 RELATION BETWEEN THE ROOT LOCI AND THE NYQUIST PLOT

Because both the root locus analysis and the Nyquist criterion deal with the location of the roots of the characteristic equation of a linear SISO system, the two analyses are closely related. Exploring the relationship between the two methods will enhance the understanding of both methods. Given the characteristic equation

$$1+L(s)=1+KG_1(s)H_1(s)=0 \tag{10-68}$$

the Nyquist plot of $L(s)$ in the $L(s)$-plane is the mapping of the Nyquist path in the s-plane. Because the root loci of Eq. (10-53) must satisfy the conditions

$$\angle KG_1(s)H_1(s)=(2j+1)\pi \quad K\geq 0 \tag{10-69}$$

$$\angle KG_1(s)H_1(s)=2j\pi \quad K\leq 0 \tag{10-70}$$

for $j = 0, \pm1, \pm2, \ldots$, the root loci simply represent a mapping of the real axis of the $L(s)$-plane or the $G(s)H(s)$-plane onto the s-plane. In fact, for the RL $K \geq 0$, the mapping points are on the negative real axis of the $L(s)$-plane, and, for the RL $K \leq 0$, the mapping points are on the positive real axis of the $L(s)$-plane. It was pointed out earlier that the mapping from the s-plane to the function plane for a rational function is single valued, but the reverse process is multivalued. As a simple illustration, the Nyquist plot of a type-1 third-order transfer function $G(s)H(s)$ that corresponds to points on the $j\omega$-axis of the s-plane is shown in Fig. 10-28. The root loci for the same system are shown in Fig. 10-29 as a mapping of the real axis of the $G(s)H(s)$-plane onto the s-plane. Note that, in this case, each point of the $G(s)H(s)$-plane corresponds to three points in the s-plane. The $(-1,j0)$ point of the $G(s)H(s)$-plane corresponds to the two points where the root loci intersect the $j\omega$-axis and a point on the real axis.

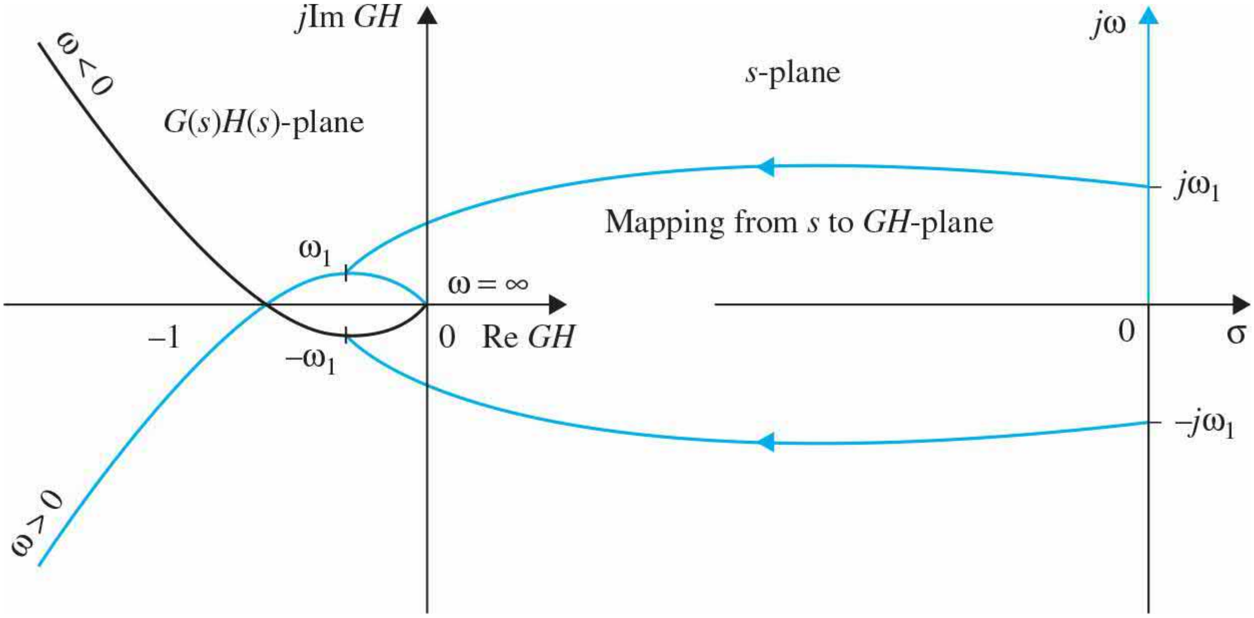


Figure 10-28 Polar plot of $G(s)H(s) = K/[s(s + a)(s + b)]$ interpreted as a mapping of the $j\omega$-axis of the s-plane onto the $G(s)H(s)$-plane.

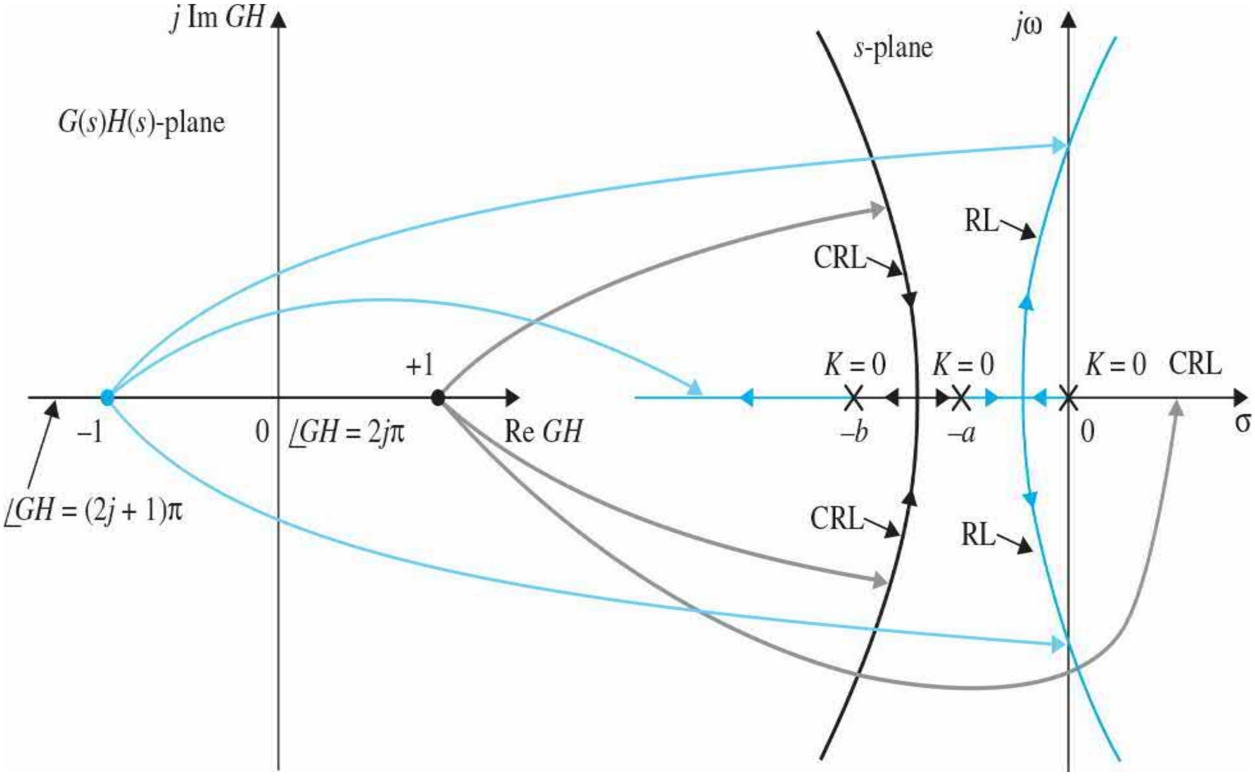


Figure 10-29 Root-locus diagram of $G(s)H(s) = K/[(s)(s+ a)(s + b)]$ interpreted as a mapping of the real axis of the $G(s)H(s)$-plane onto the s-plane.

The Nyquist plot and the root loci each represents the mapping of only a very limited portion of one domain to the other. In general, it would be useful to consider the mapping of points other than those on the $j\omega$-axis of the s-plane and on the real axis of the $G(s)H(s)$-plane. For instance, we may use the mapping of the constant-damping-ratio lines in the s-plane onto the $G(s)H(s)$-plane for the purpose of determining relative stability of the closed-loop system. Figure 10-30 illustrates the $G(s)H(s)$ plots that correspond to different constant-damping-ratio lines in the s-plane. As shown by curve (3) in Fig. 10-30, when the $G(s)H(s)$ curve passes through the $(-1, j0)$ point, it means that Eq. (10-67) is satisfied, and the corresponding trajectory in the s-plane passes through the root of the characteristic equation. Similarly, we can construct the root loci that correspond to the straight lines rotated at various angles from the real axis in the $G(s)H(s)$-plane, as shown in Fig. 10-31. Notice that these root loci now satisfy the condition of

$$\angle KG_1(s)H_1(s) = (2j+1)\pi - \theta \quad K \geq 0 \qquad (10\text{-}71)$$

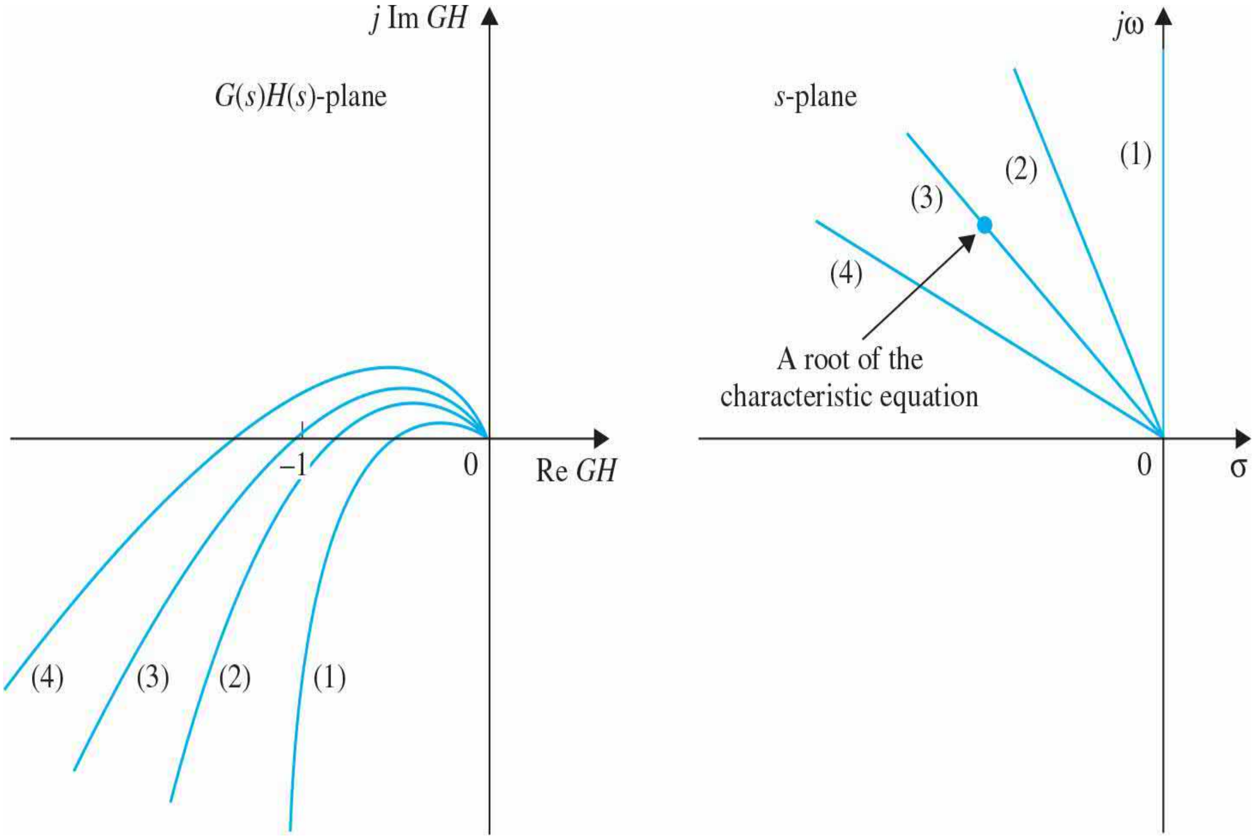


Figure 10-30 $G(s)H(s)$ plots that correspond to constant-damping-ratio

lines in the *s*-plane.

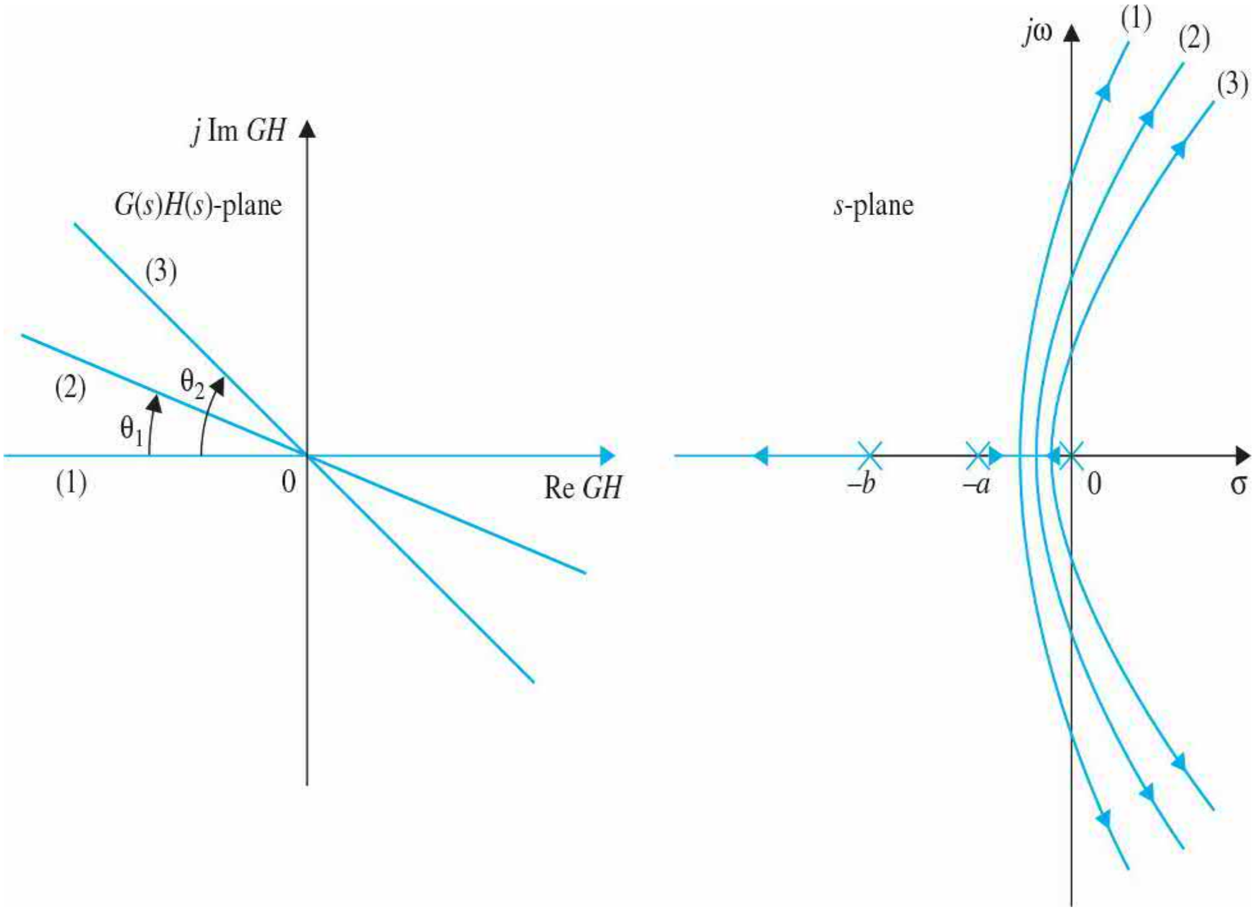


Figure 10-31 Root loci that correspond to different phase-angle loci in the *G*(*s*)*H*(*s*)-plane.

Or the root loci of Fig. 10-24 must satisfy the equation

$$1+G(s)H(s)e^{j\theta}=0 \tag{10-72}$$

for the various values of θ indicated.

10-7 ILLUSTRATIVE EXAMPLES: NYQUIST CRITERION FOR MINIMUM-PHASE TRANSFER FUNCTIONS

The following examples serve to illustrate the application of the Nyquist criterion to systems with minimum-phase loop transfer functions.

EXAMPLE 10-7-1 Consider that a single-loop feedback control system has the loop transfer function

$$L(s) = G(s)H(s) = \frac{K}{s(s+2)(s+10)} \tag{10-73}$$

which is of minimum-phase type. The stability of the closed-loop system can be conducted by investigating whether the Nyquist plot of $L(j\omega)/K$ for $\omega = \infty$ to 0 encloses the $(-1, j0)$ point. The Nyquist plot of $L(j\omega)/K$ may be plotted using freqtool. Figure 10-32 shows the Nyquist plot of $L(j\omega)/K$ for $\omega = \infty$ to 0. However, because we are interested only in whether the critical point is enclosed, in general, it is not necessary to produce an accurate Nyquist plot. Because the area that is enclosed by the Nyquist plot is to the left of the curve, traversed in the direction that corresponds to $\omega = \infty$ to 0 on the Nyquist path, all that is necessary to determine stability is to find the point or points at which the Nyquist plot crosses the real axis in the $L(j\omega)/K$-plane. In many cases, information on the intersection on the real axis and the properties of $L(j\omega)/K$ at $\omega = \infty$ and $\omega = 0$ would allow the sketching of the Nyquist plot without actual plotting. We can use the following steps to obtain a sketch of the Nyquist plot of $L(j\omega)/K$.

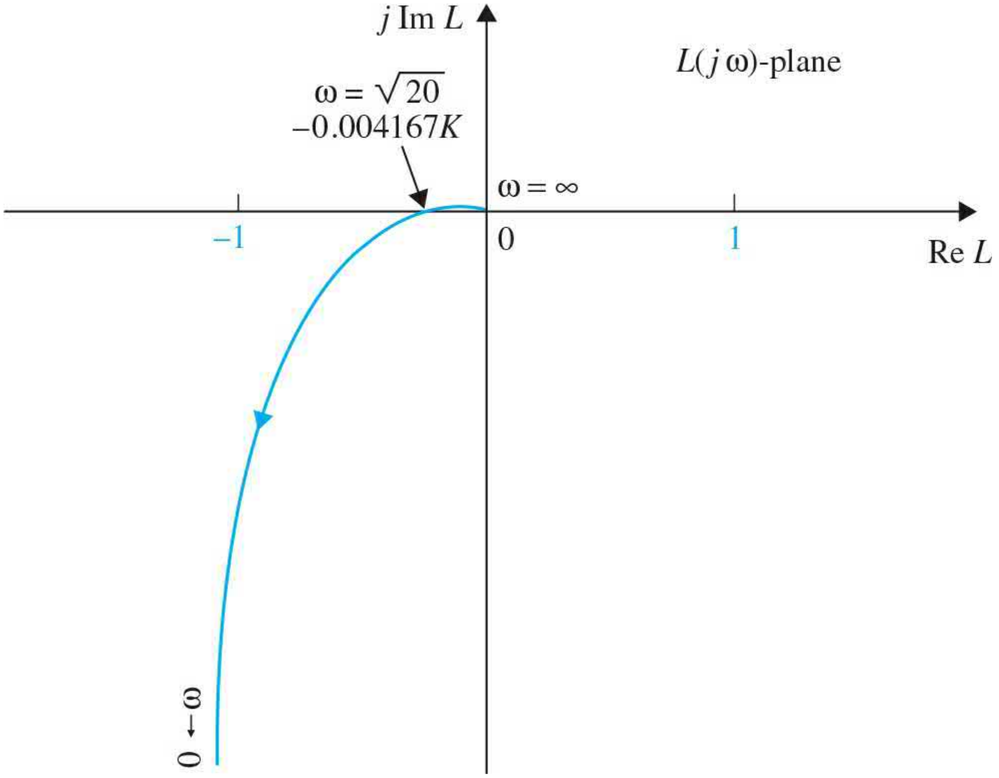


Figure 10-32 Nyquist plot of $L(s)/K = \dfrac{1}{s(s+2)(s+10)}$ for $\omega = \infty$ to $\omega = 0$.

1. Substitute $s = j\omega$ in $L(s)$. Setting $s = j\omega$ in Eq. (10-73), we get

$$L(j\omega)/K = \frac{1}{j\omega(j\omega+2)(j\omega+10)} \tag{10-74}$$

2. Substituting $\omega = 0$ in the last equation, we get the zero-frequency property of $L(j\omega)$,

$$L(j0)/K = \infty\angle -90 \tag{10-75}$$

3. Substituting $\omega = \infty$ in Eq. (10-74), the property of the Nyquist plot at infinite frequency is established.

$$L(j\infty)/K = 0\angle -270^\circ \tag{10-76}$$

These results are verified by the plot shown in Fig. 10-32—also check App. G for more information on polar plots.

4. To find the intersect(s) of the Nyquist plot with the real axis, if any,

we rationalize *L*(*j*ω)/*K* by multiplying the numerator and the denominator of the equation by the complex conjugate of the denominator. Thus, Eq. (10-74) becomes

$$L(j\omega)/K = \frac{[-12\omega^2 - j\omega(20-\omega^2)]}{[-12\omega^2 + j\omega(20-\omega^2)][-12\omega^2 - j\omega(20-\omega^2)]} \\ = \frac{[-12\omega - j(20-\omega^2)]}{\omega[144\omega^2 + (20-\omega^2)]} \qquad (10\text{-}77)$$

5. To find the possible intersects on the real axis, we set the imaginary part of *L*(*j*ω)/*K* to zero. The result is

$$\text{Im}[L(j\omega)/K] = \frac{-(20-\omega^2)}{\omega[144\omega^2 + (20-\omega^2)]} = 0 \qquad (10\text{-}78)$$

Toolbox 10-7-1

MATLAB code for Fig. 10-32.

```
w=0.1:0.1:1000;
num = [1];
den = conv(conv([1 10],[1,2]),[1 0]);
[re,im,w] = nyquist(num,den,w);
plot(re,im);
grid
```

The solutions of the last equation are ω = ∞, which is known to be a solution at *L*(*j*ω)/*K* = 0, and

$$\omega = \pm\sqrt{20}\ \text{rad/s} \qquad (10\text{-}79)$$

Because ω is positive, the correct answer is $\omega = \sqrt{20}$ rad/s. Substituting this frequency into Eq. (10-77), we have the intersect on the real axis of the *L*(*j*ω)-plane at

$$L(j\sqrt{20})/K = -\frac{12}{2880} = -0.004167 \qquad (10\text{-}80)$$

The last five steps should lead to an adequate sketch of the Nyquist plot of $L(j\omega)/K$ short of plotting it. Thus, we see that, if K is less than 240, the intersect of the $L(j\omega)$ locus on the real axis would be to the right of the critical point $(-1, j0)$; the latter is not enclosed, and the system is stable. If $K = 240$, the Nyquist plot of $L(j\omega)$ would intersect the real axis at the −1 point, and the system would be marginally stable. In this case, the characteristic equation would have two roots on the $j\omega$-axis in the s-plane at $s = \pm j\sqrt{20}$. If the gain is increased to a value beyond 240, the intersect would be to the left of the −1 point on the real axis, and the system would be unstable. When K is negative, we can use the $(+1, j0)$ point in the $L(jw)$-plane as the critical point. Figure 10-32 shows that, under this condition, the +1 point on the real axis would be enclosed for all negative values of K, and the system would always be unstable. Thus, the Nyquist criterion leads to the conclusion that the system is stable in the range of $0 < K < 2400$. Note that application of the Routh-Hurwitz stability criterion leads to this same result.

Figure 10-33 shows the root loci of the characteristic equation of the system described by the loop transfer function in Eq. (10-73). The correlation between the Nyquist criterion and the root loci is easily observed.

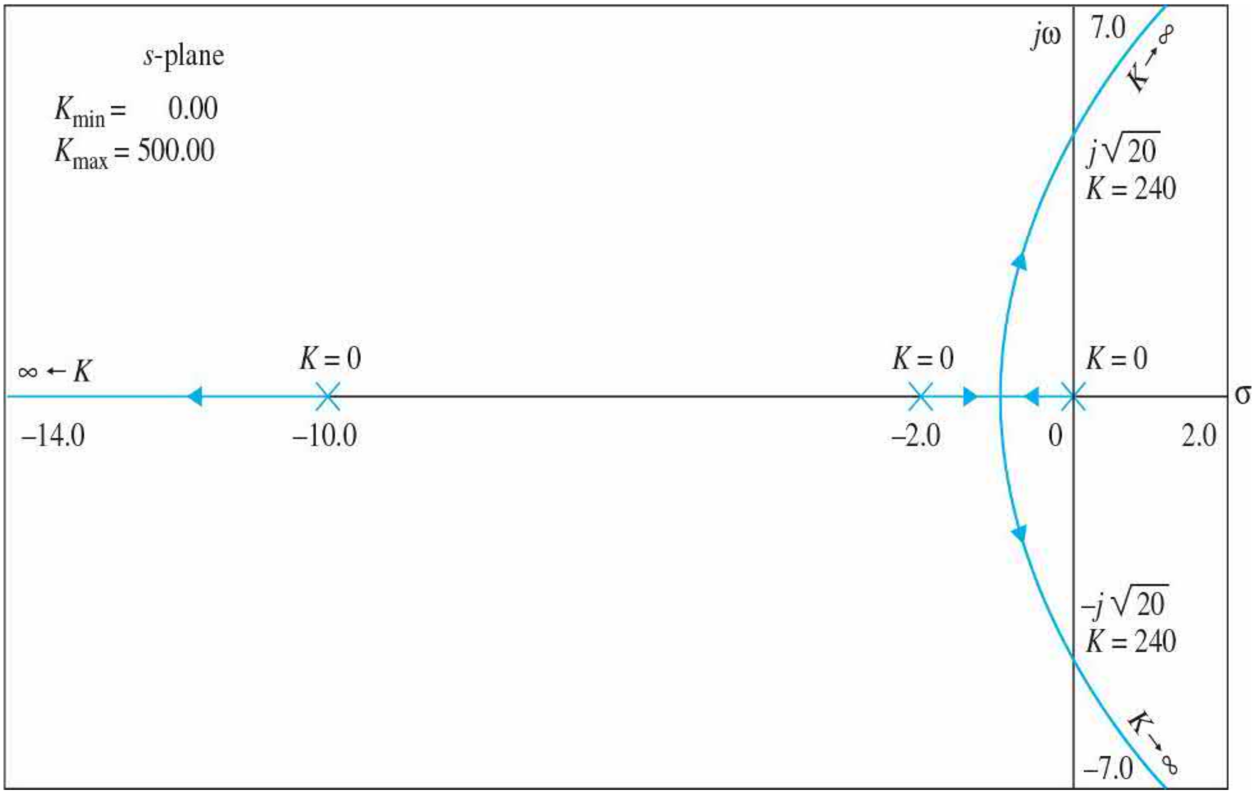


Figure 10-33 RL of $L(s)=\dfrac{K}{s(s+2)(s+10)}$.

Toolbox 10-7-2

MATLAB code for Fig. 10-33.

```
den=conv([1 2 0],[1 10]);
mysys=tf(.0001,den);
rlocus(mysys);
```

EXAMPLE 10-7-2 Consider the characteristic equation

$$Ks^3+(2K+1)s^2+(2K+5)s+1=0 \tag{10-81}$$

Dividing both sides of the last equation by the terms that do not contain *K*, we have

$$1+KL_{\text{eq}}(s)=1+\frac{Ks(s^2+2s+2)}{s^2+5s+1}=0 \tag{10-82}$$

Thus,

$$L_{\text{eq}}(s)=\frac{s(s^2+2s+2)}{s^2+5s+1} \tag{10-83}$$

which is an improper function. We can obtain the information to manually sketch the Nyquist plot of $L_{eq}(s)$ to determine the stability of the system. Setting $s = j\omega$ in Eq. (10-83), we get

$$\frac{L_{\text{eq}}(j\omega)}{K}=\frac{\omega[-2\omega+j(2-\omega^2)]}{(1-\omega^2)+5j\omega} \tag{10-84}$$

From the last equation, we obtain the two end points of the Nyquist plot:

$$L_{\text{eq}}(j0)=0\angle 90 \quad \text{and} \quad L_{\text{eq}}(j\infty)=\infty\angle 90^\circ \tag{10-85}$$

Rationalizing Eq. (10-84) by multiplying its numerator and denominator by the complex conjugate of the denominator, we get

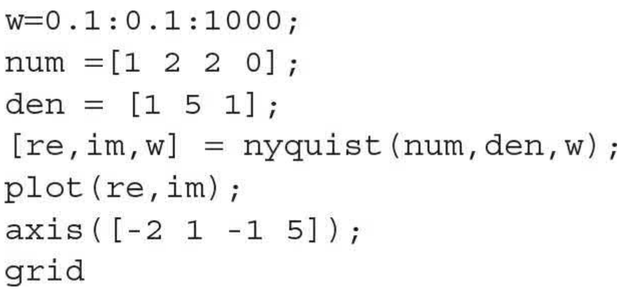
$$\frac{L_{\mathrm{eq}}(j\omega)}{K}=\frac{\omega^2[5(2-\omega^2)-2(1-\omega^2)]+j\omega[10\omega^2+(2-\omega^2)(1-\omega^2)]}{(1-\omega^2)^2+25\omega^2} \tag{10-86}$$

To find the possible intersects of the $L_{eq}(j\omega)$ plot on the real axis, we set the imaginary part of Eq. (10-86) to zero. We get $\omega = 0$ and

$$\omega^4+7\omega^2+2=0 \tag{10-87}$$

Toolbox 10-7-3

MATLAB code for Fig. 10-34.



```
w=0.1:0.1:1000;
num =[1 2 2 0];
den = [1 5 1];
[re,im,w] = nyquist(num,den,w);
plot(re,im);
axis([-2 1 -1 5]);
grid
```

Using the MATLAB command “roots([1 0 7 0 2])” we can show that all four roots of Eq. (10-87) are imaginary ((±j2.589, ±j0.546), which indicates that the $L_{eq}(j\omega)/K$ locus intersects the real axis only at $\omega = 0$. Using the information given by Eq. (10-70) and the fact that there are no other intersections on the real axis than at $\omega = 0$, the Nyquist plot of $L_{eq}(j\omega)/K$ is sketched as shown in Fig. 10-34. Notice that this plot is sketched without any detailed data computed on $L_{eq}(j\omega)/K$ and, in fact, could be grossly inaccurate. However, the sketch is adequate to determine the stability of the system. Because the Nyquist plot in Fig. 10-34 does not enclose the (−1, j0) point as ω varies from ∞ to 0, the system is stable for all finite positive values of K.

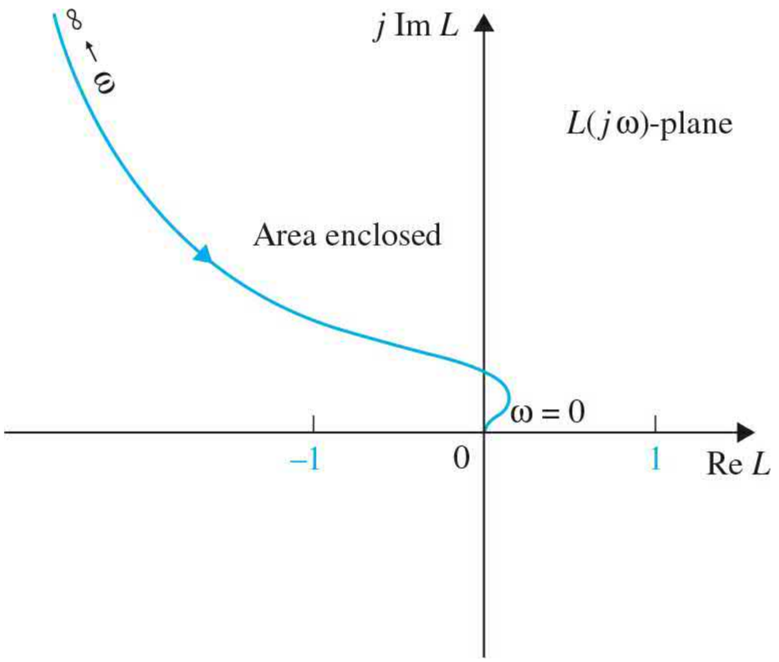


Figure 10-34 Nyquist plot of $\frac{L_{eq}(s)}{K} = \frac{s(s^2+2s+2)}{s^2+5s+1}$ for $\omega = \infty$ to $\omega = 0$.

Figure 10-35 shows the Nyquist plot of Eq. (10-81), based on the poles and zeros of *Leq*(*s*)/*K* in Eq. (10-83). Notice that the RL stays in the left-half *s*-plane for all positive values of *K*, and the results confirm the Nyquist criterion results on system stability.

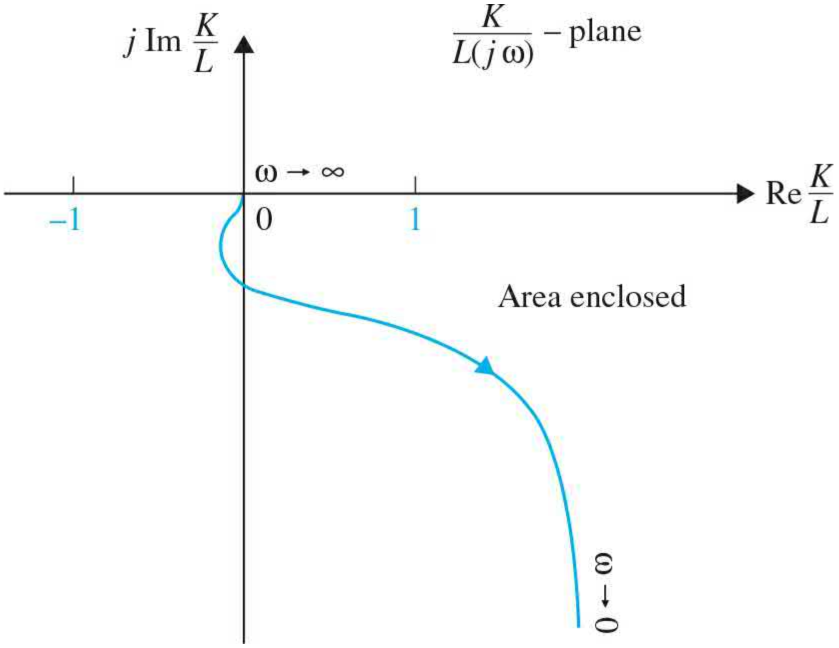


Figure 10-35 Nyquist plot of K/L_{eq} for $\frac{L_{\text{eq}}(s)}{K} = \frac{s(s^2+2s+2)}{s^2+5s+1}$ for $\omega = \infty$ to $\omega = 0$.

$$\frac{K}{L_{\text{eq}}(j\omega)} = \frac{(1-\omega^2)+5j\omega}{[-2\omega^2 + j\omega(2-\omega^2)]} \tag{10-88}$$

for $\omega = \infty$ to 0. The plot again does not enclose the $(-1, j0)$ point, and the system is again stable for all positive values of K by interpreting the Nyquist plot of $K/L_{eq}(j\omega)$.

Figure 10-36 shows the RL of Eq. (10-82) for $K > 0$, using the pole-zero configuration of $L_{eq}(s)$ of Eq. (10-83). Because the RL stays in the left-half s-plane for all positive values of K, the system is stable for $0 < K < \infty$, which agrees with the conclusion obtained with the Nyquist criterion.

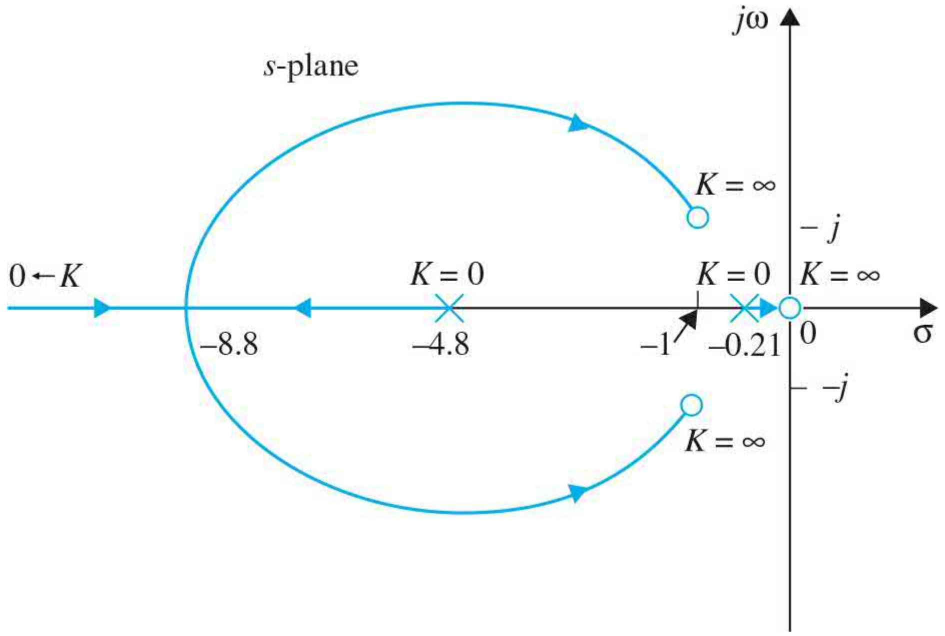


Figure 10-36 RL of $L(s) = \frac{Ks(s^2+2s+2)}{s^2+5s+1}$.

10-8 EFFECTS OF ADDING POLES AND ZEROS TO *L*(*S*) ON THE SHAPE OF THE

NYQUIST PLOT

Because the performance of a control system is often affected by adding and moving poles and zeros of the loop transfer function, it is important to investigate how the Nyquist plot is affected when poles and zeros are added to *L*(*s*).

Let us begin with a first-order transfer function

$$L(s)=\frac{K}{1+T_1 s} \tag{10-89}$$

where T_1 is a positive real constant. The Nyquist plot of *L*(*j*ω) for 0 ≤ ω ≤ ∞ is a semicircle, as shown in Fig. 10-37. The figure also shows the interpretation of the closed-loop stability with respect to the critical point for all values of *K* between -∞ and ∞.

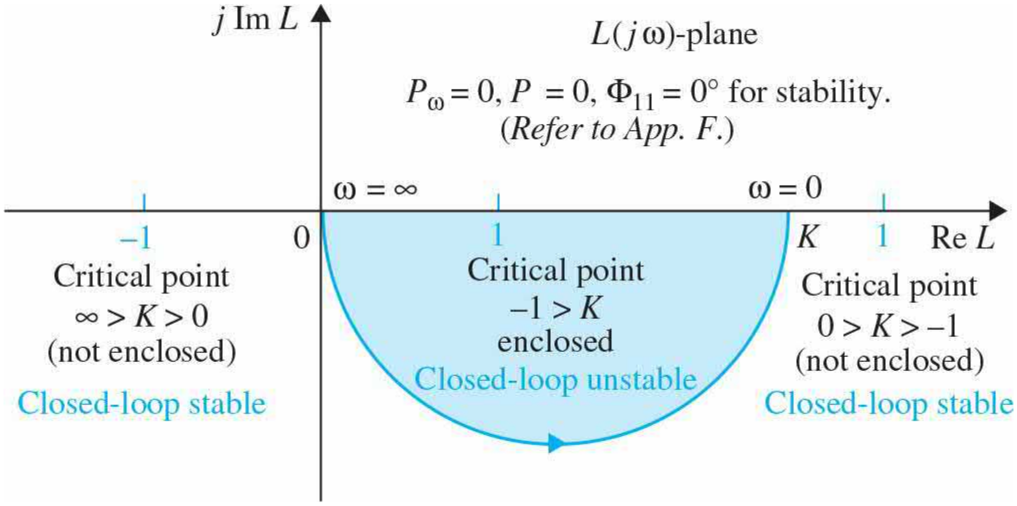


Figure 10-37 Nyquist plot of $L(s)=\dfrac{K}{s(1+T_1 s)}$.

10-8-1 ADDITION OF POLES AT *s* = 0

Consider that a pole at *s* = 0 is added to the transfer function of Eq. (10-74), then

$$L(s)=\frac{K}{s(1+T_1 s)} \tag{10-90}$$

Because adding a pole at $s = 0$ is equivalent to dividing $L(s)$ by $j\omega$, the phase of $L(j\omega)$ is reduced by 90° at both zero and infinite frequencies. In addition, the magnitude of $L(j\omega)$ at $\omega = 0$ becomes infinite. Figure 10-38 illustrates the Nyquist plot of $L(j\omega)$ in Eq. (10-90) and the closed-loop stability interpretations with respect to the critical points for $-\infty < K \infty$. In general, adding a pole of multiplicity p at $s = 0$ to the transfer function of Eq. (10-89) will give the following properties to the Nyquist plot of $L(j\omega)$:

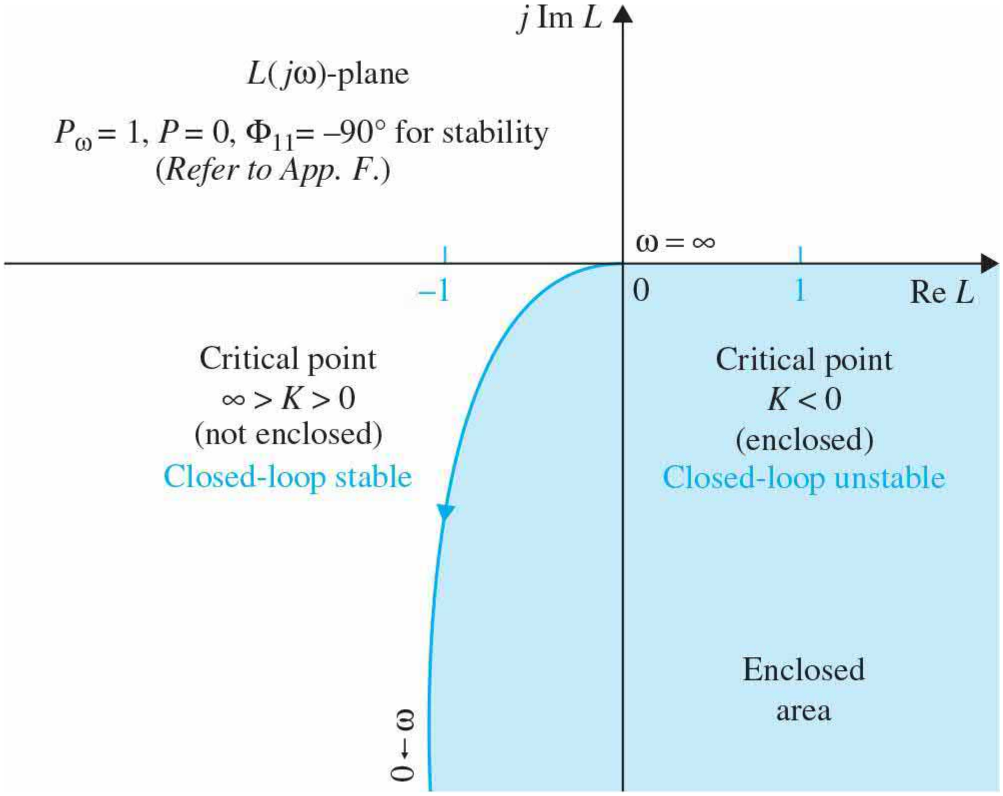


Figure 10-38 Nyquist plot of $L(s) = \dfrac{K}{s(1+T_1 s)}$.

$$\lim_{\omega \to \infty} \angle L(j\omega) = -(p+1)90° \tag{10-91}$$

$$\lim_{\omega \to 0} \angle L(j\omega) = -p \times 90° \tag{10-92}$$

$$\lim_{\omega \to \infty} |L(j\omega)| = 0 \tag{10-93}$$

$$\lim_{\omega \to 0} |L(j\omega)| = \infty \tag{10-94}$$

The following example illustrates the effects of adding multiple-order poles to $L(s)$.

EXAMPLE 10-8-1 Figure 10-39 shows the Nyquist plot of

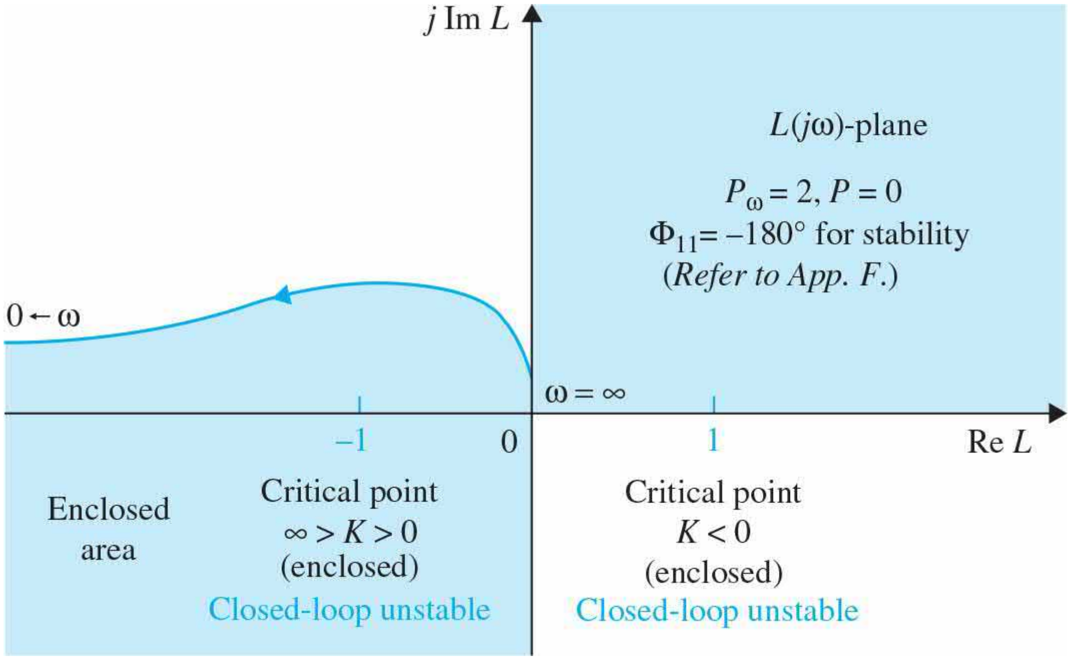


Figure 10-39 Nyquist plot of $L(s) = \dfrac{K}{s^2(1+T_1 s)}$.

$$L(s) = \frac{K}{s^2(1+T_1 s)} \tag{10-95}$$

and the critical points, with stability interpretations. Figure 10-40 illustrates the same for

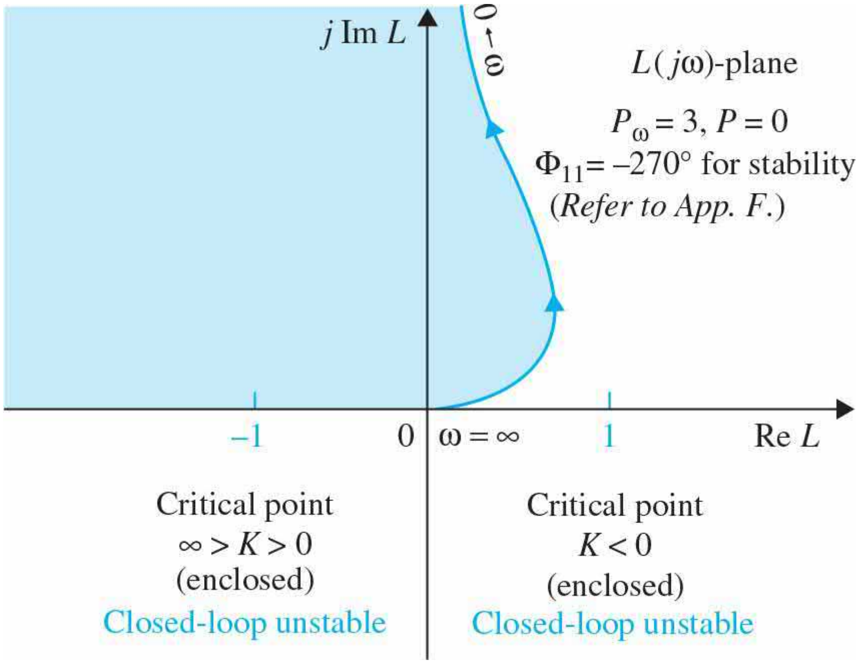


Figure 10-40 Nyquist plot of $L(s) = \dfrac{K}{s^3(1+T_1 s)}$.

$$L(s) = \frac{K}{s^3(1+T_1 s)} \tag{10-96}$$

The conclusion from these illustrations is that the addition of poles at $s = 0$ to a loop transfer function will affect the stability of the closed-loop system adversely. A system that has a loop transfer function with more than one pole at $s = 0$ (type 2 or higher) is likely to be unstable or difficult to stabilize.

10-8-2 Addition of Finite Nonzero Poles

When a pole at $s = -1/T_2 T_2 > 0$) is added to the function $L(s)$ of Eq. (10-89), we have

$$L(s) = \frac{K}{(1+T_1 s)(1+T_2 s)} \tag{10-97}$$

The Nyquist plot of $L(j\omega)$ at $\omega = 0$ is not affected by the addition of the pole, since

$$\lim_{\omega \to 0} L(j\omega) = K \tag{10-98}$$

Adding poles at s = 0 to a loop transfer function will reduce stability of the closed-loop system.

The value of $L(j\omega)$ at $\omega = \infty$ is

$$\lim_{\omega\to\infty} L(j\omega) = \lim_{\omega\to\infty} \frac{-K}{T_1 T_2 \omega^2} = 0\angle -180^\circ \tag{10-99}$$

Thus, the effect of adding a pole at $s = -1/T_2$ to the transfer function of Eq. (10-75) is to shift the phase of the Nyquist plot by -90° at $\omega = \infty$, as shown in Fig. 10-41. The figure also shows the Nyquist plot of

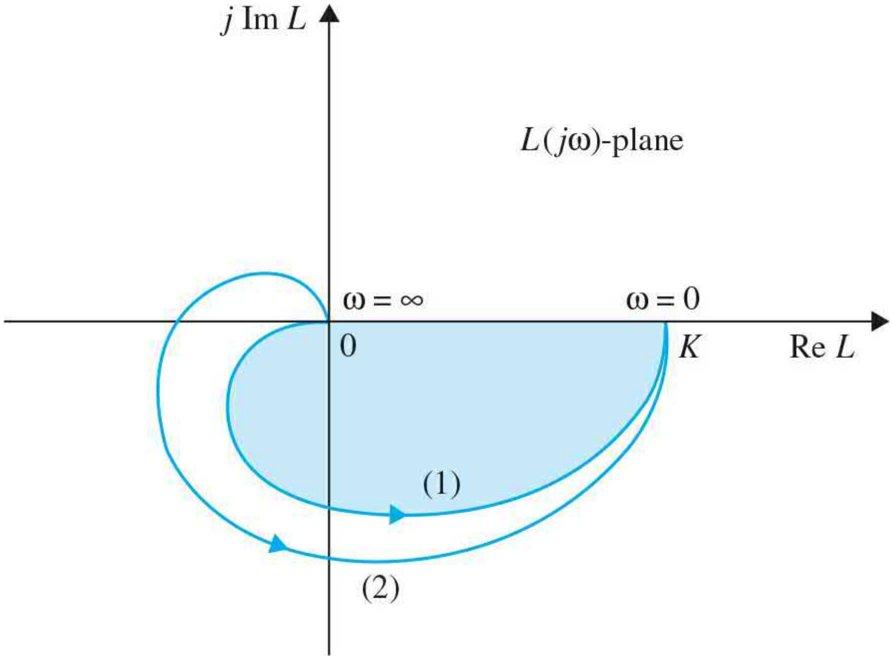


Figure 10-41 Nyquist plots. Curve (1): $L(s) = \frac{K}{(1+T_1 s)(1+T_2 s)}$. Curve (2): $L(s) = \frac{K}{(1+T_1 s)}(1+T_2 s)(1+T_3 s)$.

Adding nonzero poles to the loop transfer function also reduces stability of the closed-loop system.

$$L(s)=\frac{K}{(1+T_1s)(1+T_2s)(1+T_3s)} \tag{10-100}$$

where two nonzero poles have been added to the transfer function of Eq. (10-89) (T_1, T_2, T_3, > 0). In this case, the Nyquist plot at $\omega = \infty$ is rotated clockwise by another 90° from that of Eq. (10-97). *These examples show the adverse effects on closed-loop stability when poles are added to the loop transfer function*. The closed-loop systems with the loop transfer functions of Eqs. (10-89) and (10-97) are all stable as long as K is positive. The system represented by Eq. (10-100) is unstable if the intersect of the Nyquist plot on the negative real axis is to the left of the $(-1, j0)$ point when K is positive.

10-8-3 Addition of Zeros

> Adding zeros to the loop transfer function has the effect of stabilizing the closed-loop system.

It was demonstrated in Chap. 7 that adding zeros to the loop transfer function has the effect of reducing the overshoot and the general effect of stabilization. In terms of the Nyquist criterion, this stabilization effect is easily demonstrated, since the multiplication of the term $(1 + T_ds)$ to the loop transfer function increases the phase of $L(s)$ by 90° at $\omega = \infty$. The following example illustrates the effect on stability of adding a zero at $-1/T_d$ to a loop transfer function.

EXAMPLE 10-8-2 Consider that the loop transfer function of a closed-loop control system is

$$L(s)=\frac{K}{s(1+T_1s)(1+T_2s)} \tag{10-101}$$

It can be shown that the closed-loop system is stable for

$$0<K<\frac{T_1+T_2}{T_1T_2} \tag{10-102}$$

Suppose that a zero at $s = -1/T_d (T_d > 0)$ is added to the transfer function of Eq. (10-101); then,

$$L(s)=\frac{K(1+T_ds)}{s(1+T_1s)(1+T_2s)} \tag{10-103}$$

The Nyquist plots of the two transfer functions of Eqs. (10-101) and (10-103) are shown in Fig. 10-42. The effect of the zero in Eq. (10-103) is to add 90° to the phase of the $L(j\omega)$ in Eq. (10-101) at $\omega = \infty$ while not affecting the value at $\omega = 0$. The intersect on the negative real axis of the $L(jw)$-plane is moved from $-KT_1T_2/(T_1 + T_2)$ to $-K(T_1T_2 - T_dT_1 - T_dT_2)/(T_1 + T_2)$. Thus, the system with the loop transfer function in Eq. (10-103) is stable for

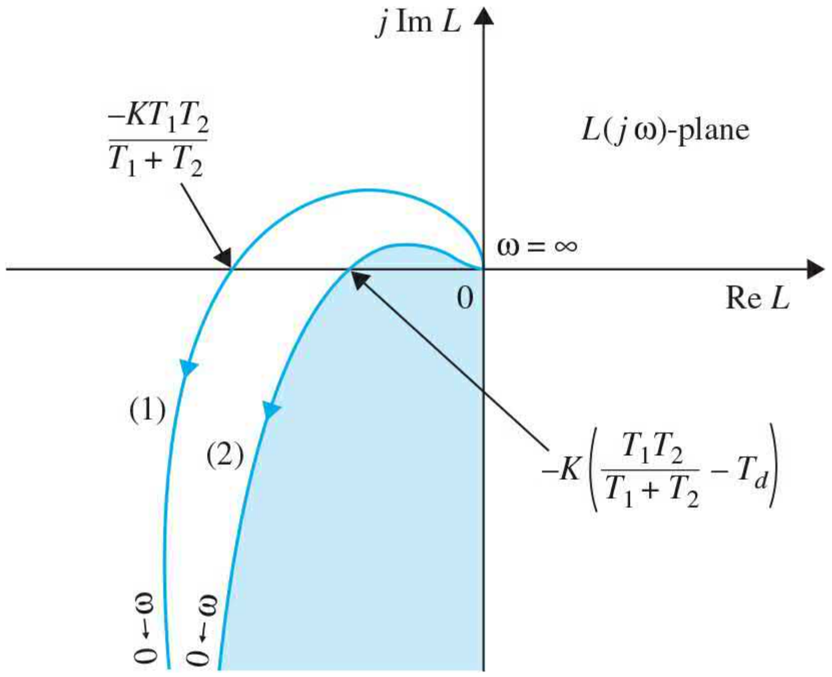


Figure 10-42 Nyquist plots. Curve (1): $L(s)=\frac{K}{s(1+T_1s)(1+T_2s)}$. Curve (2): $L(s)=\frac{K(1+T_ds)}{s(1+T_1s)(1+T_2s)};T_d<T_1;T_2$.

$$0<K<\frac{T_1+T_2}{T_1T_2-T_d(T_1+T_2)} \tag{10-104}$$

which, for positive T_d and K, has a higher upper bound than that of Eq. (10-102). ▲

10-9 RELATIVE STABILITY: GAIN MARGIN AND PHASE MARGIN

We have demonstrated in Sec. 10-2 the general relationship between the resonance peak *Mp* of the frequency response and the maximum overshoot of the time response. Comparisons and correlations between frequency-domain and time-domain parameters such as these are useful in the prediction of the performance of control systems. In general, we are interested not only in the absolute stability of a system but also how stable it is. The latter is often called **relative stability**. In the time domain, relative stability is measured by parameters such as the maximum overshoot and the damping ratio. In the frequency domain, the resonance peak *Mp* can be used to indicate relative stability. Another way of measuring relative stability in the frequency domain is by how close the Nyquist plot of $L(j\omega)$ is to the $(-1, j0)$ point.

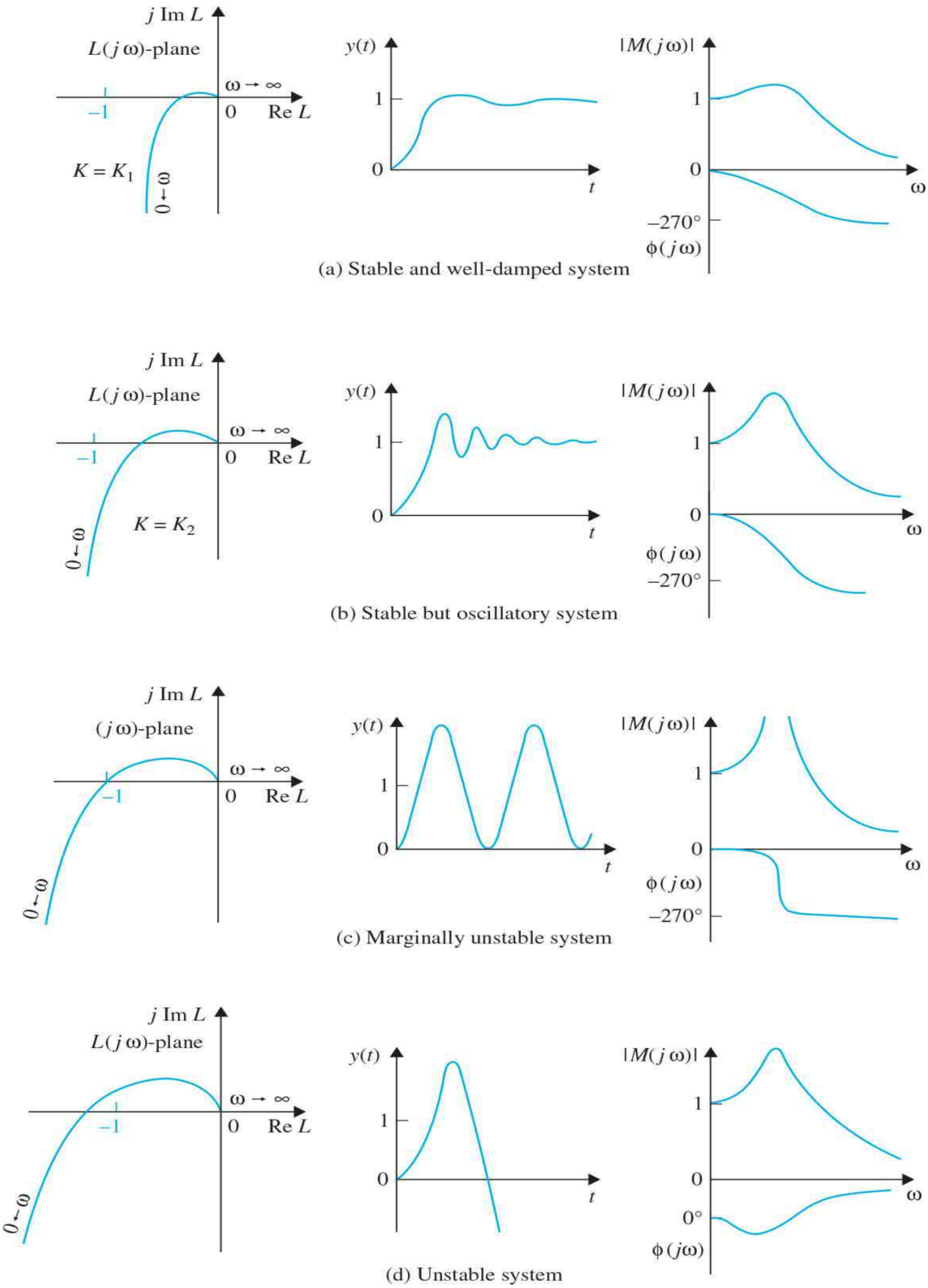
Relative stability is used to indicate how stable a system is.

To demonstrate the concept of relative stability in the frequency domain, the Nyquist plots and the corresponding step responses and frequency responses of a typical third-order system are shown in Fig. 10-43 for four different values of loop gain K. It is assumed that the function $L(j\omega)$ is of minimum-phase type, so that the enclosure of the $(-1, j0)$ point is sufficient for stability analysis. The four cases are evaluated as follows:

1. ***Figure 10-43a; the loop gain K is low*** The Nyquist plot of $L(j\omega)$ intersects the negative real axis at a point that is quite far to the right of the $(-1, j0)$ point. The corresponding step response is quite well damped, and the value of *Mr* of the frequency response is low.
2. ***Figure 10-43b; K is increased***. The intersect is moved closer to the $(-1, j0)$ point; the system is still stable because the critical point is not enclosed, but the step response has a larger maximum overshoot, and *Mr* is also larger.

3. *Figure 10-43c*; ***K is increased further***. The Nyquist plot now passes through the (−1, *j*0) point, and the system is marginally stable. The step response becomes oscillatory with constant amplitude, and *Mr* becomes infinite.

4. *Figure 10-43d*; ***K is relatively very large***. The Nyquist plot now encloses the (−1, *j*0) point, and the system is unstable. The step response becomes unbounded. The magnitude curve of $|M(j\omega)|$-versus-ω ceases to have any significance. In fact, for the unstable system, the value of *Mr* is still finite! In all the above analysis, the phase curve $\pi(j\omega)$ of the closed-loop frequency response also gives qualitative information about stability. Notice that the negative slope of the phase curve becomes steeper as the relative stability decreases. When the system is unstable, the slope beyond the resonant frequency becomes positive. In practice, the phase characteristics of the closed-loop system are seldom used for analysis and design purposes.



(a) Stable and well-damped system

(b) Stable but oscillatory system

(c) Marginally unstable system

(d) Unstable system

Figure 10-43 Correlation among Nyquist plots, step responses, and frequency responses.

10-9-1 Gain Margin

Gain margin (GM) is one of the most frequently used criteria for measuring relative stability of control systems. In the frequency domain, gain margin is used to indicate the closeness of the intersection of the negative real axis made by the Nyquist plot of $L(j\omega)$ to the $(-1, j0)$ point. Before giving the definition of gain margin, let us first define the **phase crossover** on the Nyquist plot and the **phase-crossover frequency**.

Gain margin is measured at the phase crossover.

Phase crossover. A phase-crossover on the $L(j\omega)$ plot is a point at which the plot intersects the negative real axis.

Phase-crossover frequency. The **phase-crossover frequency** ω_p is the frequency at the phase crossover, or where

$$\angle L(j\omega_p) = 180° \tag{10-105}$$

M_r ceases to have any meaning when the closed-loop system is unstable.

The Nyquist plot of a loop transfer function $L(j\omega)$ that is of minimum-phase type is shown in Fig. 10-44. The phase-crossover frequency is denoted as ω_p, and the magnitude of $L(j\omega)$ at $\omega = \omega_p$ is designated as $|L(j\omega_p)|$. Then, the gain margin of the closed-loop system that has $L(s)$ as its loop transfer function is defined as

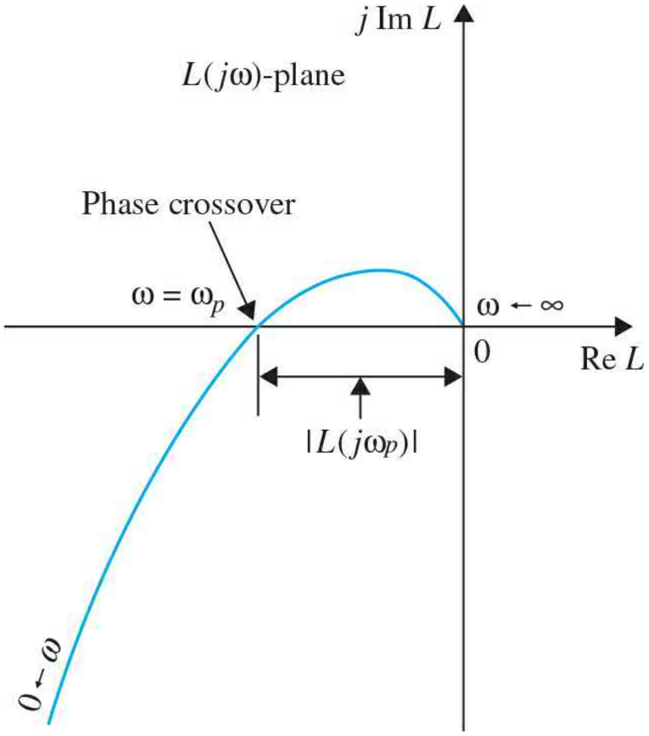


Figure 10-44 Definition of the gain margin in the polar coordinates.

$$\begin{aligned}\text{Gain margin} = \text{GM} &= 20\log_{10}\frac{1}{|L(j\omega_p)|} \\ &= -20\log_{10}|L(j\omega_p)|\,dB\end{aligned} \tag{10-106}$$

On the basis of this definition, we can draw the following conclusions about the gain margin of the system shown in Fig. 10-44, depending on the properties of the Nyquist plot.

1. The $L(j\omega)$ plot does not intersect the negative real axis (no finite nonzero phase crossover).

$$0 < |L(j\omega_p)| < 1 \quad \text{GM} > 0\text{ dB} \tag{10-108}$$

2. The $L(j\omega)$ plot intersects the negative real axis between (phase crossover lies between) 0 and the −1 point.

$$|L(j\omega_p)| = 1 \quad \text{GM} = 0\text{ dB} \tag{10-109}$$

3. The *L*(*jω*) plot passes through (phase crossover is at) the (−1, *j*0) point.

$$\left|L(j\omega_p)\right| > 1 \text{ GM} < 0 \text{ dB} \qquad (10\text{-}110)$$

Gain margin is the amount of gain in dB that can be added to the loop before the closed-loop system becomes unstable.

4. The *L*(*jω*) plot encloses (phase crossover is to the left of) the (−1, *j*0) point.

$$0 < \left|L(j\omega_p)\right| < 1 \quad \text{GM} > 0 \text{ dB} \qquad (10\text{-}108)$$

Based on the foregoing discussions, the physical significance of gain margin can be summarized as: **Gain margin is the amount of gain in decibels (dB) that can be added to the loop before the closed-loop system becomes unstable.**

- When the Nyquist plot does not intersect the negative real axis at any finite nonzero frequency, the gain margin is infinite in dB; this means that, theoretically, the value of the loop gain can be increased to infinity before instability occurs.
- When the Nyquist plot of *L*(*jω*) passes through the *L*(*j*ω) point, the gain margin is 0 dB, which implies that the loop gain can no longer be increased, as the system is at the margin of instability.
- When the phase-crossover is to the left of the (−1, *j*0) point, the phase margin is negative in dB, and the loop gain must be reduced by the gain margin to achieve stability.

10-9-2 Gain Margin of Nonminimum-Phase Systems

Care must be taken when attempting to extend gain margin as a measure of relative stability to systems with nonminimum-phase loop transfer functions. For such systems, a system may be stable even when the phase-crossover point is to the left of (−1, *j*0) and thus a negative gain margin may still correspond to a stable system. Nevertheless, the closeness of the phase-crossover to the (−1, *j*0) point still gives an indication of relative stability.

10-9-3 Phase Margin

- Phase margin is measured at the gain crossover.
- Phase margin is the amount of pure phase delay that can be added before the system becomes unstable.

The gain margin is only a one-dimensional representation of the relative stability of a closed-loop system. As the name implies, gain margin indicates system stability with respect to the variation in loop gain only. In principle, one would believe a system with a large gain margin should always be relatively more stable than one with a smaller gain margin. Unfortunately, gain margin alone is inadequate to indicate relative stability when system parameters other than the loop gain are subject to variation. For instance, the two systems represented by the $L(j\omega)$ plots in Fig. 10-45 apparently have the same gain margin. However, locus *A* actually corresponds to a more stable system than locus *B* because with any change in the system parameters that affect the phase of $L(j\omega)$, locus *B* may easily be altered to enclose the (−1, *j*0) point. Furthermore, we can show that the system *B* actually has a larger *Mr* than system *A*.

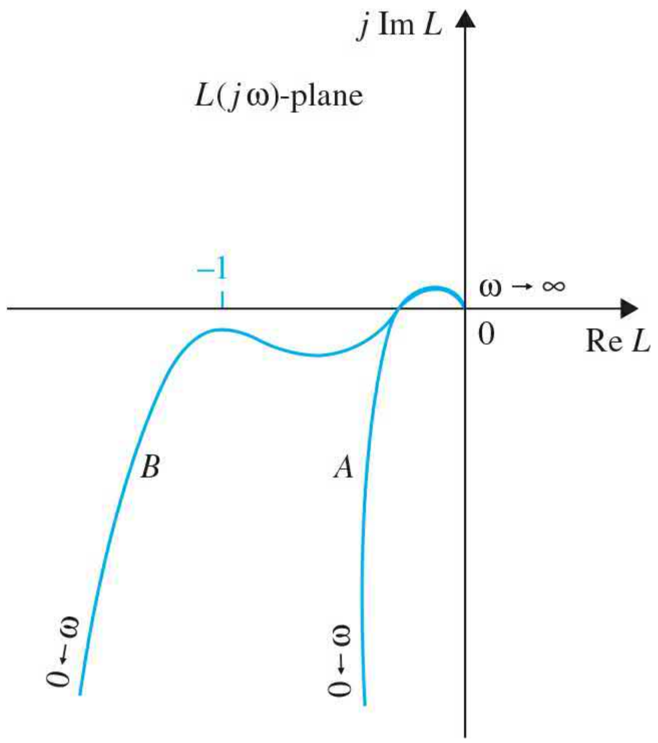


Figure 10-45 Nyquist plots showing systems with the same gain margin but different degrees of relative stability.

The definition of phase margin given here is for a system with a minimum-phase loop transfer function.

To include the effect of phase shift on stability, we introduce the **phase margin**, which requires that we first make the following definitions:

Gain crossover. The gain crossover is a point on the $L(j\omega)$ plot at which the magnitude of $L(j\omega)$ is equal to 1.

Gain-crossover frequency. The gain-crossover frequency, ω_s, is the frequency of $L(j\omega)$ at the gain crossover. Or where

$$\left|L(j\omega_g)\right| = 1 \tag{10-111}$$

The definition of phase margin is stated as: **Phase margin (PM) is defined as the angle in degrees through which the $L(j_v)$ plot must be rotated about the origin so that the gain crossover passes through the (−1,j0) point.**

Figure 10-46 shows the Nyquist plot of a typical minimum-phase $L(j\omega)$ plot, and the phase margin is shown as the angle between the line that passes through the gain crossover and the origin. In contrast to the gain margin, which is determined by loop gain, phase margin indicates the effect on system stability due to changes in system parameter, which theoretically alter the phase of $L(j\omega)$ by an equal amount at all frequencies. Phase margin is the amount of pure phase delay that can be added to the loop before the closed-loop system becomes unstable.

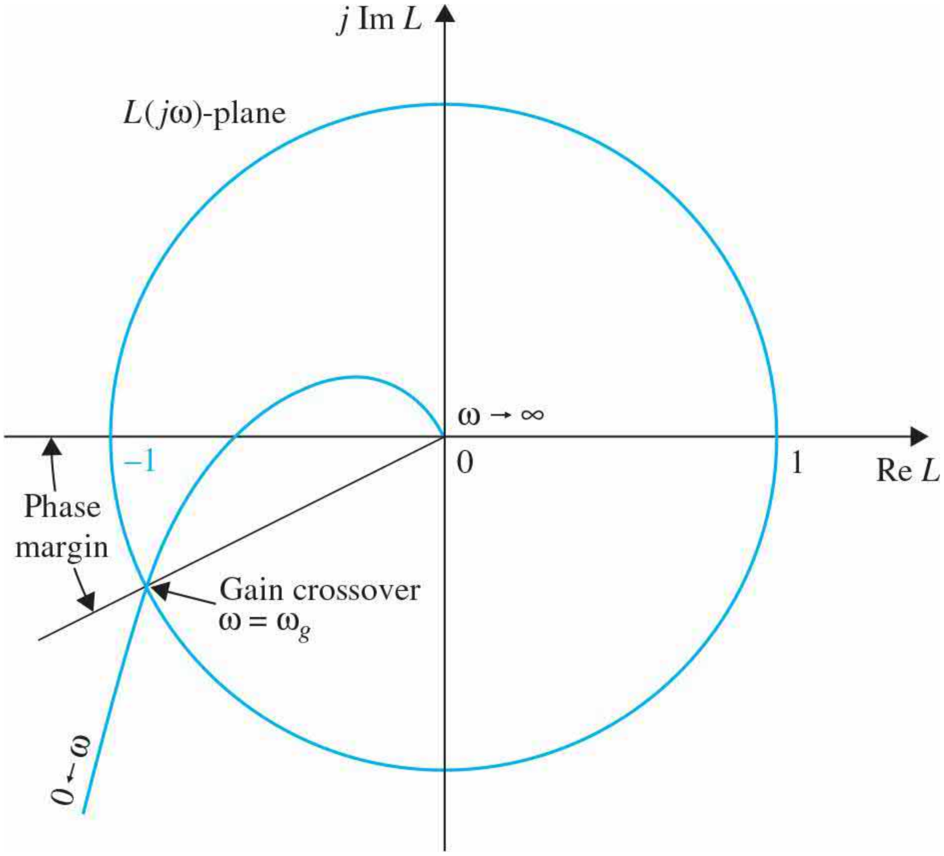


Figure 10-46 Phase margin defined in the $L(j\omega)$-plane.

When the system is of the minimum-phase type, the analytical expression of the phase margin, as seen from Fig. 10-46, can be expressed as

$$\text{phase margin (PM)} = \angle L(j\omega_g) - 180^\circ \tag{10-112}$$

where ω_g is the gain-crossover frequency.

Care should be taken when interpreting the phase margin from the Nyquist plot of a nonminimum-phase transfer function. When the loop transfer function is of the nonminimum-phase type, the gain crossover can occur in any quadrant of the $L(j\omega)$-plane, and the definition of phase margin given in Eq. (10-112) is no longer valid.

EXAMPLE 10-9-1 As an illustrative example on gain and phase margins, consider that the loop transfer function of a control system is

$$L(s)=\frac{2500}{s(s+5)(s+50)} \tag{10-113}$$

The Nyquist plot of $L(j\omega)$ is shown in Fig. 10-47. The following results are obtained from the Nyquist plot:

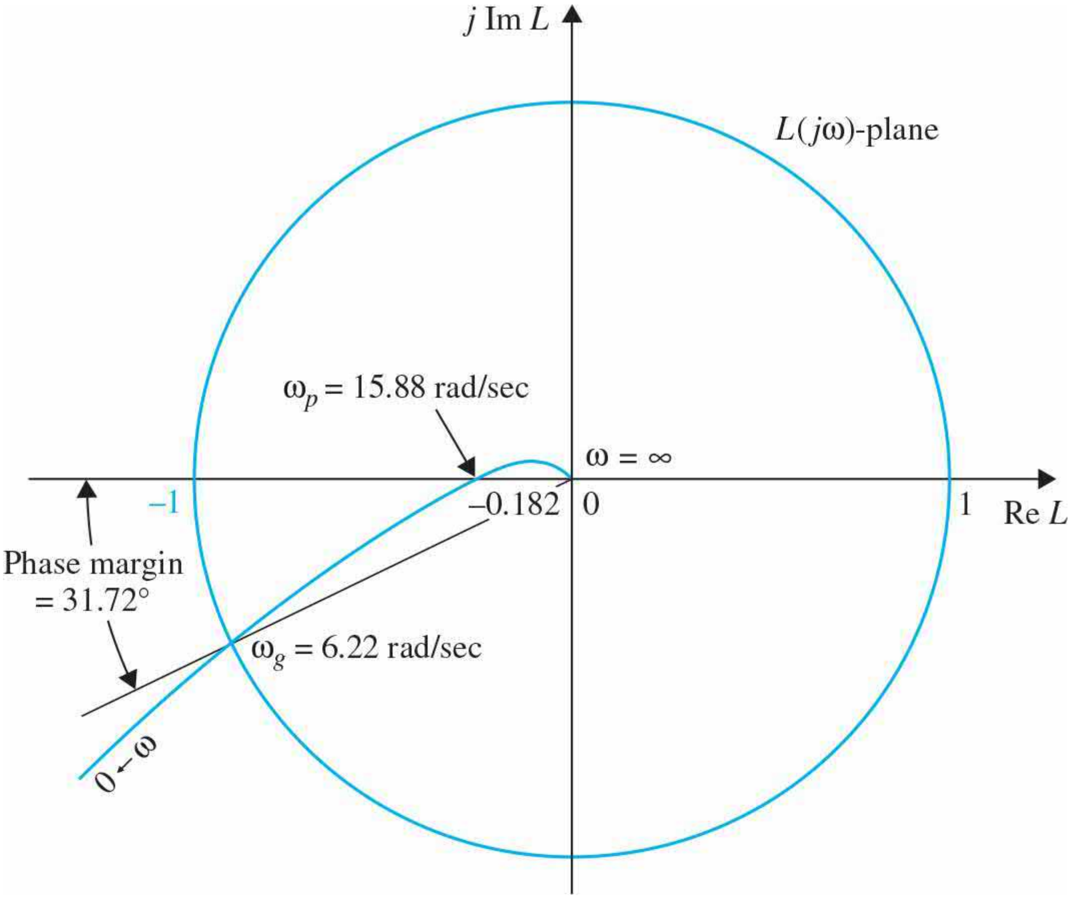


Figure 10-47 Nyquist plot of $L(s)=\frac{2500}{s(s+5)(s+50)}$.

$$\text{Gain cross over } \omega_g = 6.22 \text{ rad/s}$$

$$\text{Phase cross over } \omega_p = 15.88 \text{ rad/s}$$

The gain margin is measured at the phase crossover. The magnitude of $L(j\omega_p)$ is 0.182. Thus, the gain margin is obtained from Eq. (10-105):

$$\mathrm{GM} = 20 log_{10} \frac{1}{|L(j\omega_p)|} = 20 log_{10} \frac{1}{0.182} = 14.80 \text{ dB} \tag{10-114}$$

The phase margin is measured at the gain crossover. The phase of $L(j\omega_g)$ is 211.72°. Thus, the phase margin is obtained from Eq. (10-112):

$$\mathrm{PM} = \angle L(j\omega_g) - 180° = 211.72° - 180° = 31.72° \tag{10-115}$$

Before embarking on the Bode plot technique of stability study, it would be beneficial to summarize advantages and disadvantages of the Nyquist plot.

Advantages of the Nyquist plot

1. The Nyquist plot can be used for the study of stability of systems with nonminimum-phase transfer functions.
2. The stability analysis of a closed-loop system can be easily investigated by examining the Nyquist plot of the loop transfer function with reference to the (− 1, *j*0) point once the plot is made.

Disadvantage of the Nyquist plot

1. It is not so easy to carry out the design of the controller by referring to the Nyquist plot.

10-10 STABILITY ANALYSIS WITH THE BODE PLOT

The Bode plot of a transfer function described in App. G is a very useful graphical tool for the analysis and design of linear control systems in the frequency domain. Before the inception of computers, Bode plots were often called the “asymptotic plots” because the magnitude and phase curves can be sketched from their asymptotic properties without detailed plotting. Modern applications of the Bode plot for control systems should be identified with the following advantages and disadvantages:

Advantages of the Bode plot

1. In the absence of a computer, a Bode diagram can be sketched by approximating the magnitude and phase with straight line segments.
2. Gain crossover, phase crossover, gain margin, and phase margin are more easily determined on the Bode plot than from the Nyquist plot.
3. For design purposes, the effects of adding controllers and their parameters are more easily visualized on the Bode plot than on the Nyquist plot.

Disadvantage of the Bode plot

1. Absolute and relative stability of only minimum-phase systems can be determined from the Bode plot. For instance, there is no way of telling what the stability criterion is on the Bode plot.

Bode plots are useful only for stability studies of systems with minimum-phase loop transfer functions.

With reference to the definitions of gain margin and phase margin given in Figs. 10-44 and 10-46, respectively, the interpretation of these parameters from the Bode diagram is illustrated in Fig. 10-48 for a typical minimum-phase loop transfer function. The following observations can be made on system stability with respect to the properties of the Bode plot:

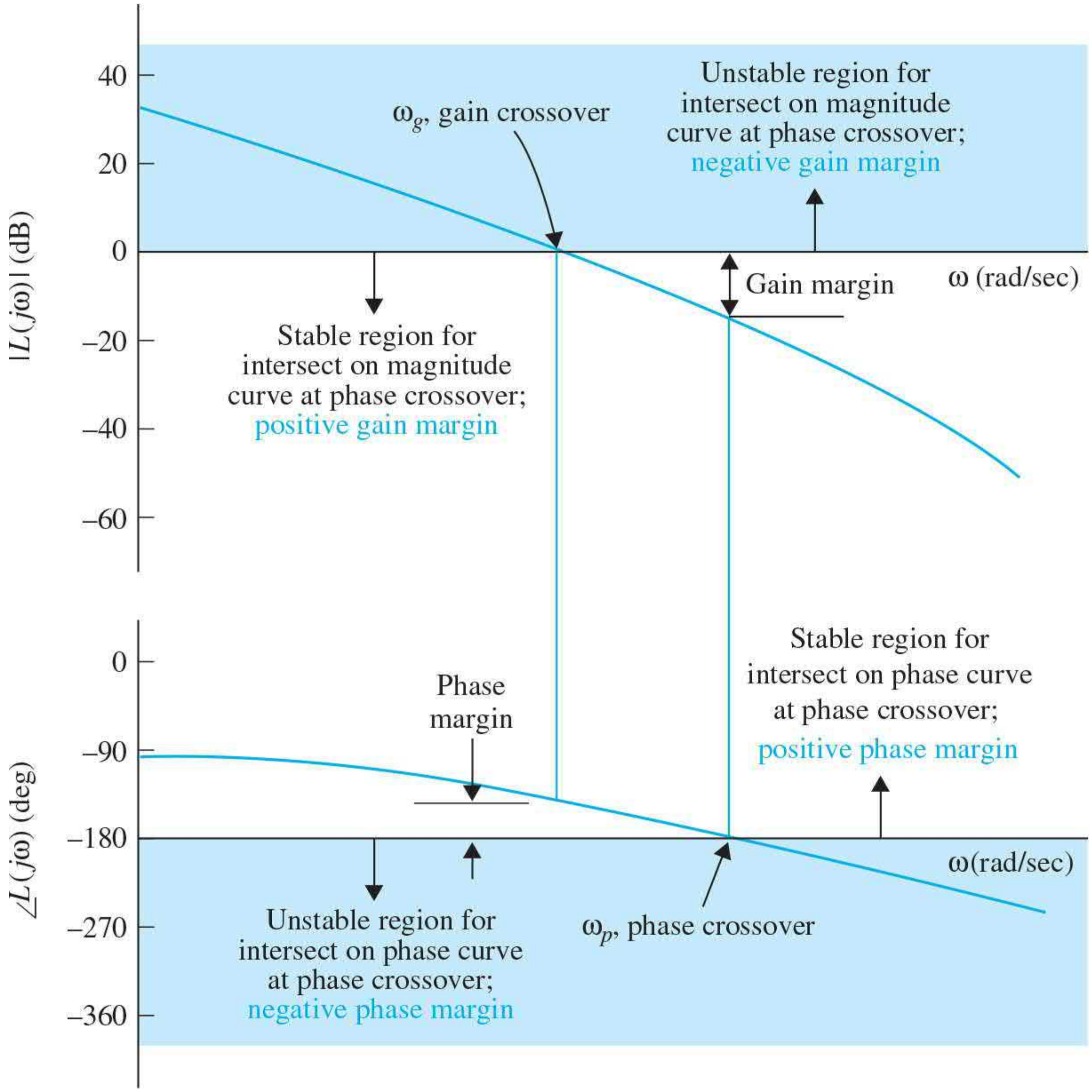


Figure 10-48 Determination of gain margin and phase margin on the Bode plot.

1. The gain margin is positive and the system is stable if the magnitude of $L(j\omega)$ at the phase crossover is negative in dB. That is, the gain margin is measured below the 0-dB axis. If the gain margin is measured above the 0-dB axis, the gain margin is negative, and the system is unstable.
2. The phase margin is positive and the system is stable if the phase of $L(j\omega)$ is greater than −180° at the gain crossover. That is, the phase

margin is measured above the −180° axis. If the phase margin is measured below the −180° axis, the phase margin is negative, and the system is unstable.

EXAMPLE 10-10-1 Consider the loop transfer function given in Eq. (10-113); the Bode plot of the function is drawn as shown in Fig. 10-49. The following results are observed easily from the magnitude and phase plots. The gain crossover is the point where the magnitude curve intersects the 0-dB axis.

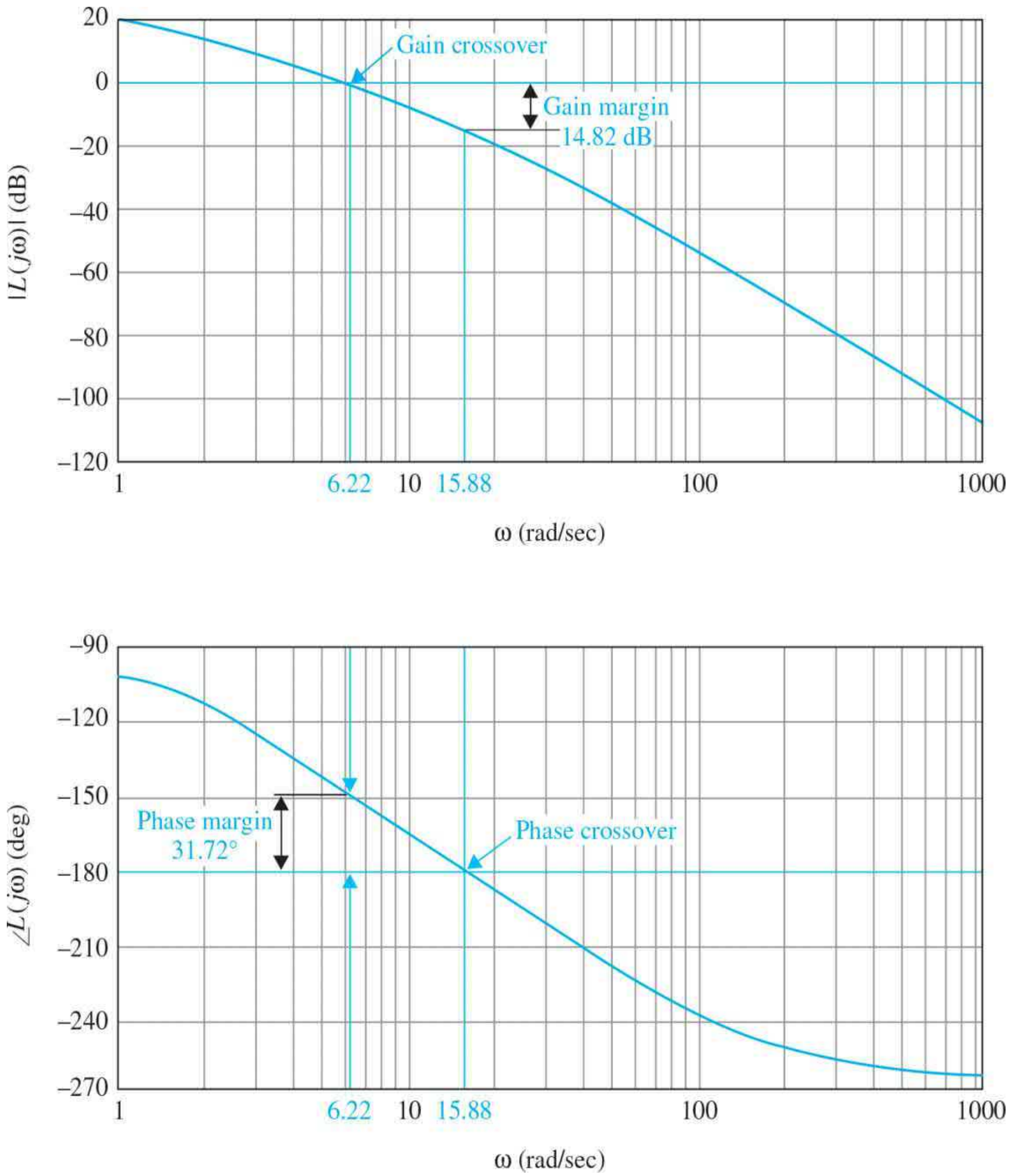


Figure 10-49 Bode plot of $L(s) = \dfrac{2500}{s(s+5)(s+50)}$.

The gain-crossover frequency ω_g is 6.22 rad/s. The phase margin is measured at the gain crossover. The phase margin is measured from the −180° axis and is 31.72°. Because the phase margin is measured above the −180° axis, the phase margin is positive, and the system is stable.

The phase crossover is the point where the phase curve intersects the −180° axis. The phase-crossover frequency is $\omega_p = 15.88$ rad/s. The gain margin is measured at the phase crossover and is 14.8 dB. Because the gain margin is measured below the 0-dB axis, the gain margin is positive, and the system is stable.

The reader should compare the Nyquist plot of Fig. 10-47 with the Bode plot of Fig. 10-49, and the interpretation of ω_g, ω_p, GM, and PM on these plots.

Toolbox 10-10-1

MATLAB code for Fig. 10-49.

```
G = zpk([],[0 -1 -1],2500)
margin(G)
grid
```

10-10-1 Bode Plots of Systems with Pure Time Delays

The stability analysis of a closed-loop system with a pure time delay in the loop was discussed in Sec. 10-4. This topic can also be conducted easily with the Bode plot. The next example illustrates the standard procedure.

EXAMPLE 10-10-2 Consider that the loop transfer function of a closed-loop system is

$$L(s) = \frac{Ke^{-T_d s}}{s(s+1)(s+2)} \tag{10-116}$$

Figure 10-50 shows the Bode plot of $L(j\omega)$ with $K = 1$ and $T_d = 0$. The following results are obtained:

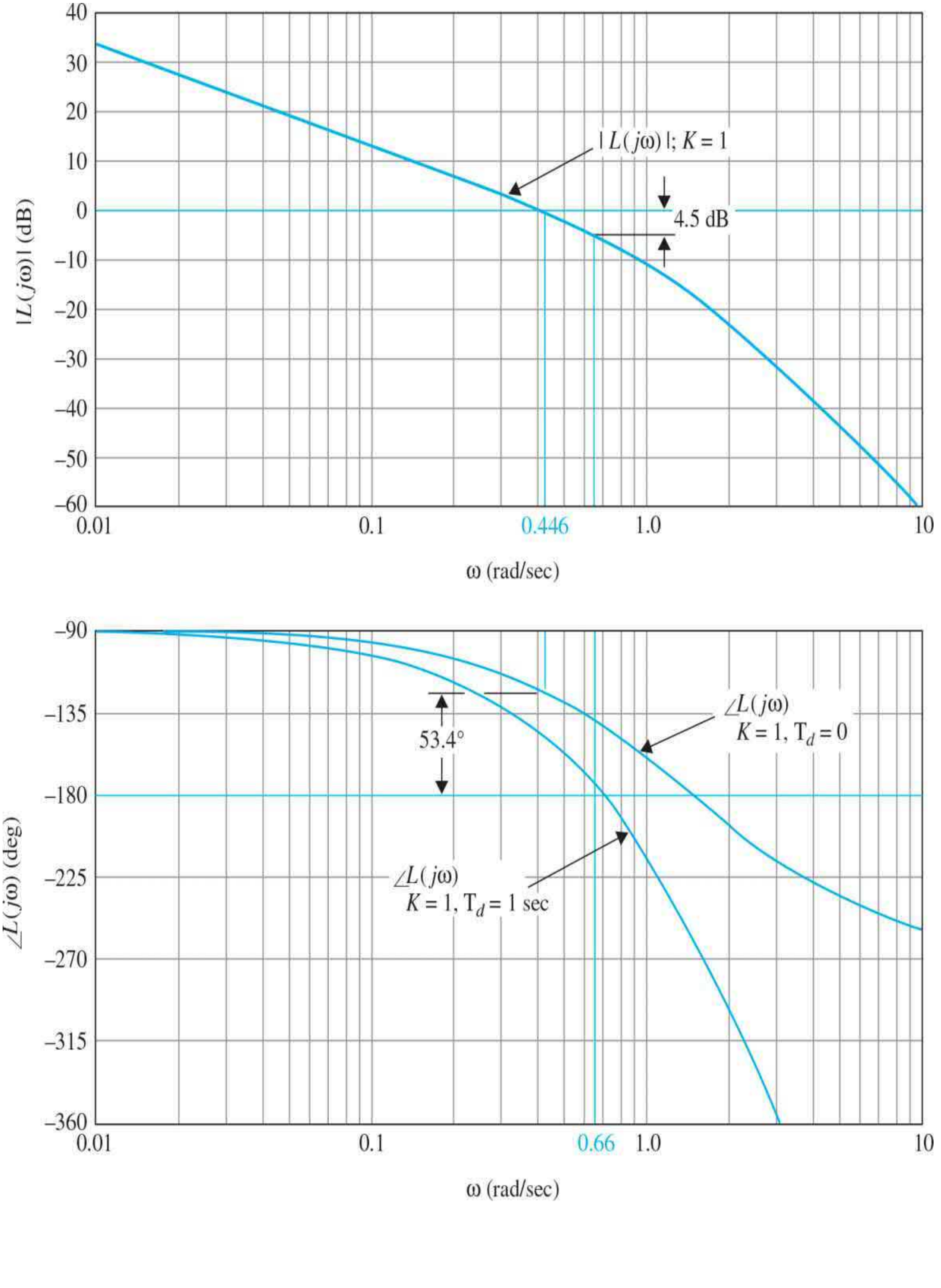

40
30
20
10
0
−10
−20
−30
−40
−50
−60
|L(jω)| (dB)
|L(jω)|; K = 1
4.5 dB
0.01
0.1
0.446
1.0
10
ω (rad/sec)
−90
−135
−180
−225
−270
−315
−360
∠L(jω) (deg)
53.4°
∠L(jω)
K = 1, T_d = 0
∠L(jω)
K = 1, T_d = 1 sec
0.01
0.1
0.66
1.0
10
ω (rad/sec)

Figure 10-50 Bode plot of $L(s) = \dfrac{Ke^{-T_d s}}{s(s+1)(s+2)}$.

Gain-cross over frequency = 0.446 rad/sec

Phase margin = 53.4°

Phase-cross over frequency = 1.416 rad/sec

Gain margin = 15.57 dB

Thus, the system with the present parameters is stable.

The effect of the pure time delay is to add a phase of $-T_d\,\omega$ radians to the phase curve while not affecting the magnitude curve. The adverse effect of the time delay on stability is apparent because the negative phase shift caused by the time delay increases rapidly with the increase in ω. To find the critical value of the time delay for stability, we set

$$T_d\omega_g = 53.4°\frac{\pi}{180°} = 0.932 \text{ radians} \tag{10-117}$$

Solving for T_d from the last equation, we get the critical value of T_d to be 2.09 s.

Continuing with the example, we set T_d arbitrarily at 1 s and find the critical value of K for stability. Figure 10-50 shows the Bode plot of $L(j\omega)$ with this new time delay. With K still equal to 1, the magnitude curve is unchanged. The phase curve droops with the increase in ω, and the following results are obtained:

$$\text{Phase-crossover frequency} = 0.66 \text{ rad/s}$$
$$\text{Gain margin} = 4.5 \text{ dB}$$

Thus, using the definition of gain margin of Eq. (10-106), the critical value of K for stability is $10^{4.5/20}$ = 1.68.

10-11 RELATIVE STABILITY RELATED TO THE SLOPE OF THE MAGNITUDE CURVE OF THE BODE PLOT

In addition to GM, PM, and M_p as relative stability measures, the slope of the magnitude curve of the Bode plot of the loop transfer function at the gain crossover also gives a qualitative indication on the relative stability of a closed-loop system. For example, in Fig. 10-49, if the loop gain of the system is decreased from the nominal value, the magnitude curve is shifted downward, while the phase curve is unchanged. This causes the gain-crossover frequency to be lower, and the slope of the magnitude curve at this frequency is less negative; the corresponding phase margin is increased. On the other hand, if the loop gain is increased, the gain-crossover frequency is increased, and the slope of the magnitude curve is more negative. This corresponds to a smaller phase margin, and the system is less stable. The reason behind these stability evaluations is quite simple. For a minimum-phase transfer function, the relation between its magnitude and phase is unique. Because the negative slope of the magnitude curve is a result of having more poles than zeros in the transfer function, the corresponding phase is also negative. In general, the steeper the slope of the magnitude curve, the more negative the phase. Thus, if the gain crossover is at a point where the slope of the magnitude curve is steep, it is likely that the phase margin will be small or negative.

10-11-1 Conditionally Stable System

The illustrative examples given thus far are uncomplicated in the sense that the slopes of the magnitude and phase curves are monotonically decreasing as ω increases. The following example illustrates a **conditionally stable system** that is capable of going through stable/unstable conditions as the loop gain varies.

EXAMPLE 10-11-1 Consider that the loop transfer function of a closed-loop system is

$$L(s)=\frac{100K(s+5)(s+40)}{s^3(s+100)(s+200)} \qquad (10\text{-}118)$$

The Bode plot of $L(j\omega)$ is shown in Fig. 10-51 for $K = 1$. The following results on the system stability are obtained:

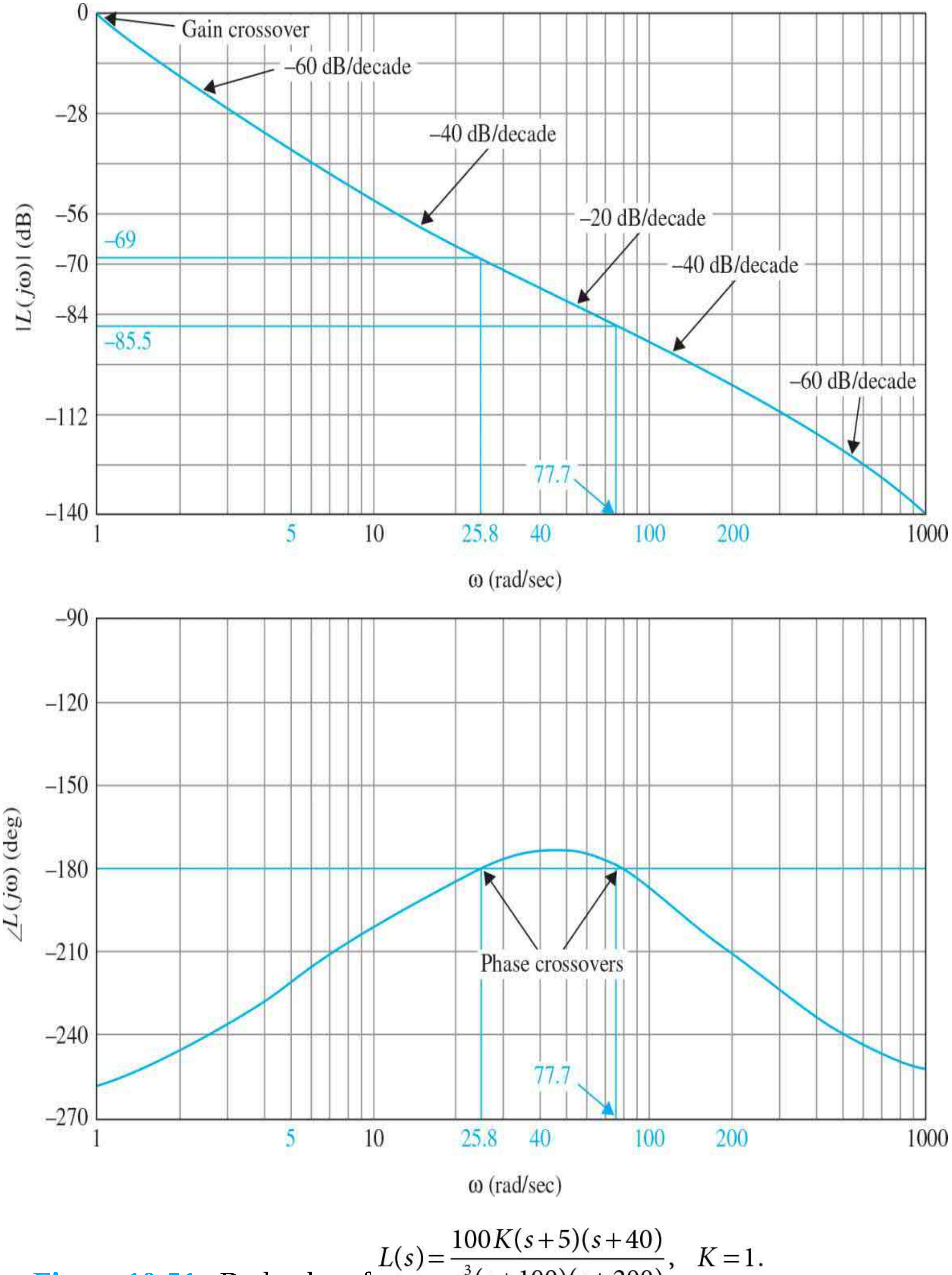


Figure 10-51 Bode plot of $L(s) = \frac{100K(s+5)(s+40)}{s^3(s+100)(s+200)}, \quad K = 1.$

$$\text{Gain-crossover frequency} = 1 \text{ rad/s}$$

$$\text{Phase margin} = -78°$$

There are two phase crossovers: one at 25.8 rad/s and the other at 77.7 rad/s. The phase characteristics between these two frequencies indicate that, if the gain crossover lies in this range, the system would be stable. From the magnitude curve, the range of *K* for stable operation is found to be between 69 and 85.5 dB. For values of *K* above and below this range, the phase of $L(j\omega)$ is less than −180°, and the system is unstable. This example serves as a good example of the relation between relative stability and the slope of the magnitude curve at the gain crossover. As observed from Fig. 10-51, at both very low and very high frequencies, the slope of the magnitude curve is −60 dB/decade; if the gain crossover falls in either one of these two regions, the phase margin is negative, and the system is unstable. In the two sections of the magnitude curve that have a slope of −40 dB/decade, the system is stable only if the gain crossover falls in about half of these regions, but even then the phase margin is small. If the gain crossover falls in the region in which the magnitude curve has a slope of −20 dB/decade, the system is stable.

Figure 10-52 shows the Nyquist plot of $L(j\omega)$. It is of interest to compare the results on stability derived from the Bode plot and the Nyquist plot. The root-locus diagram of the system is shown in Fig. 10-53. The root loci give a clear picture on the stability condition of the system with respect to *K*. The number of crossings of the root loci on the $j\omega$-axis of the *s*-plane equals the number of crossings of the phase curve of $L(j\omega)$ of the −180° axis of the Bode plot, and the number of crossings of the Nyquist plot of $L(j\omega)$ with the negative real axis. The reader should check the gain margins obtained from the Bode plot and the coordinates of the crossover points on the negative real axis of the Nyquist plot with the values of K at the $j\omega$-axis crossings on the root loci.

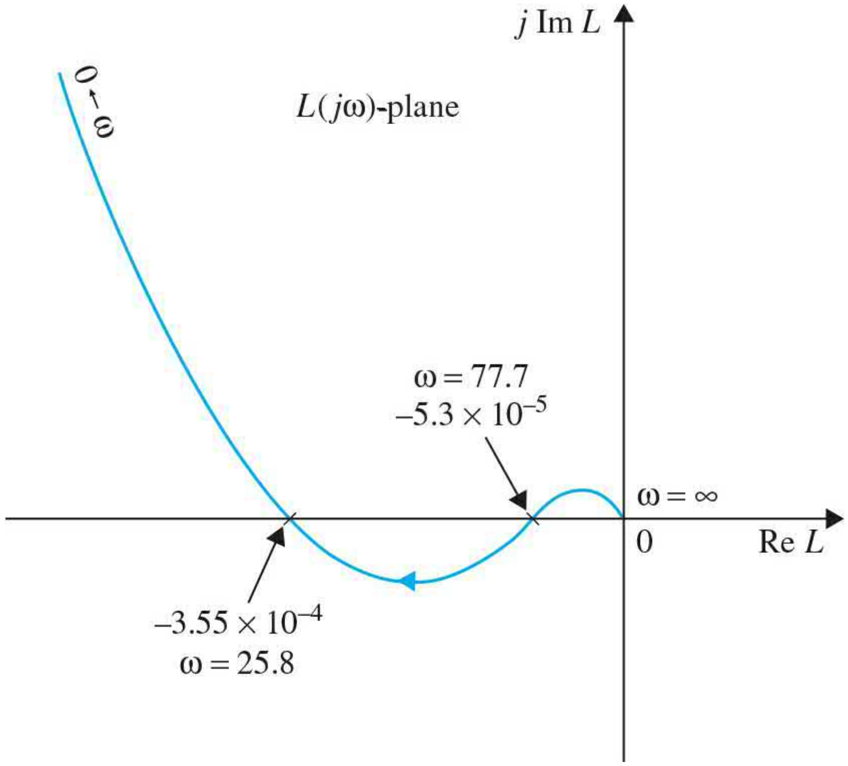


Figure 10-52 Nyquist plot of $L(s)=\frac{100K(s+5)(s+40)}{s^3(s+100)(s+200)}$, $K=1$. .

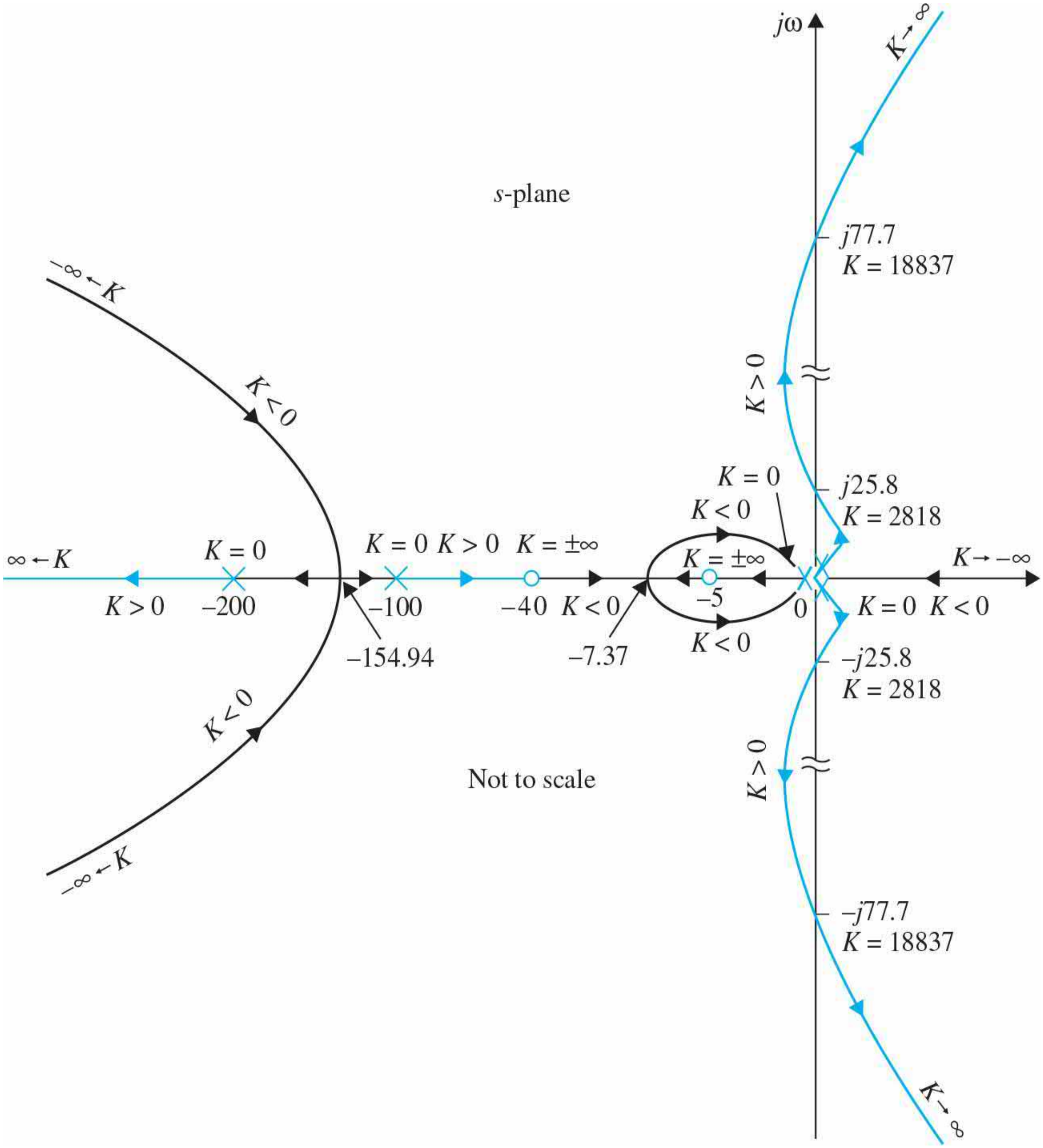


Figure 10-53 Root loci of $G(s) = \dfrac{100K(s+5)(s+40)}{s^3(s+100)(s+200)}$.

10-12 STABILITY ANALYSIS WITH THE MAGNITUDE-PHASE PLOT

The magnitude-phase plot described in App. G is another form of the frequency-domain plot that has certain advantages for analysis and design in the frequency domain. The magnitude-phase plot of a transfer function $L(j\omega)$ is done in $|L(j\omega)|$(dB) versus $\angle L(j\omega)$ (degrees). The magnitude-phase plot of the transfer function in Eq. (10-113) is constructed in Fig. 10-54 by use of the data from the Bode plot of Fig. 10-49. The gain and phase crossovers and the gain and phase margins are clearly indicated on the magnitude-phase plot of $L(j\omega)$.

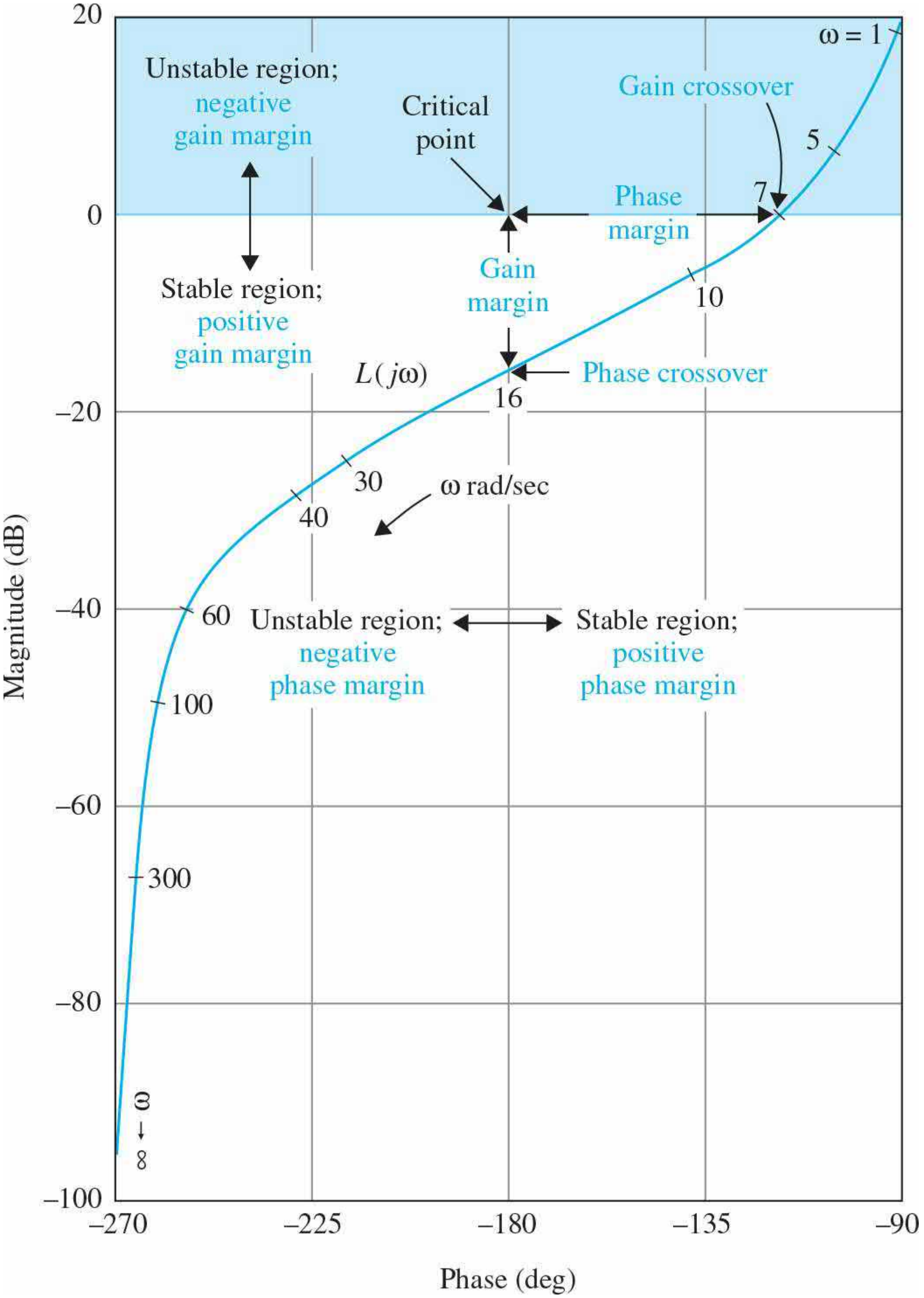


Figure 10-54 Gain-phase plot of $L(s) = \frac{10}{s(1+0.2s)(1+0.02s)}$.

- The critical point is the intersect of the 0-dB-axis and the −180°-axis.

- The phase crossover is where the locus intersects the −180°-axis.
- The gain crossover is where the locus intersects the 0-dB axis.
- The gain margin is the vertical distance in dB measured from the phase crossover to the critical point.
- The phase margin is the horizontal distance measured in degrees from the gain crossover to the critical point.

The regions in which the gain and phase crossovers should be located for stability are also indicated. Because the vertical axis for $|L(j\omega)|$ is in dB, when the loop gain of $L(j\omega)$ changes, the locus is simply shifted up and down along the vertical axis. Similarly, when a constant phase is added to $L(j\omega)$, the locus is shifted horizontally without distortion to the curve. If $L(j\omega)$ contains a pure time delay T_d, the effect of the time delay is to add a phase equal to $-\omega T_d \times 180°/\pi$ along the curve.

Another advantage of using the magnitude-phase plot is that, *for unity-feedback systems*, closed-loop system parameters such as M_r, ω_r, and BW can all be determined from the plot with the help of the constant-*M* loci. These closed-loop performance parameters are not represented on the Bode plot of the forward-path transfer function of a unity-feedback system.

10-13 CONSTANT-*M* LOCI IN THE MAGNITUDE-PHASE PLANE: THE NICHOLS CHART

It was pointed out earlier that, analytically, the resonant peak M_r and bandwidth BW are difficult to obtain for high-order systems, and the Bode plot provides information on the closed-loop system only in the form of gain margin and phase margin. It is necessary to develop a graphical method for the determination of M_r, ω_r, and BW using the forward-path transfer function $G(j\omega)$. As we shall see in the following development, the method is directly applicable only to unity-feedback systems, although with some modification it can also be applied to nonunity-feedback systems.

Consider that $G(s)$ is the forward-path transfer function of a unity-feedback system. The closed-loop transfer function is

$$M(s)=\frac{G(s)}{1+G(s)} \tag{10-119}$$

For sinusoidal steady state, we replace *s* with *jω* ; *G*(*s*) becomes

$$\begin{aligned} G(j\omega) &= \operatorname{Re}G(j\omega)+jImG(j\omega) \\ &= x+jy \end{aligned} \tag{10-120}$$

where, for simplicity, *x* denotes Re*G*(*jω*) and *y* denotes Im*G*(*jω*). The magnitude of the closed-loop transfer function is written

$$|M(j\omega)|=\left|\frac{G(j\omega)}{1+G(j\omega)}\right|=\frac{\sqrt{x^2+y^2}}{\sqrt{(1+x)^2+y^2}} \tag{10-121}$$

For simplicity of notation, let *M* denote |*M*(*jω*)|; then Eq. (10-121) leads to

$$M\sqrt{(1+x)^2+y^2}=\sqrt{x^2+y^2} \tag{10-122}$$

Squaring both sides of Eq. (10-122) gives

$$M^2[(1+x)^2+y^2]=x^2+y^2 \tag{10-123}$$

Rearranging Eq. (10-108) yields

$$(1-M^2)x^2+(1-M^2)y^2-2M^2x=M^2 \tag{10-124}$$

This equation is conditioned by dividing through by (1 − M^2) and adding the term [M^2/(1 − M^2)]^2 on both sides. We have

$$x^2+y^2-\frac{2M^2}{1-M^2}x+\left(\frac{M^2}{1-M^2}\right)^2=\frac{M^2}{1-M^2}+\left(\frac{M^2}{1-M^2}\right)^2 \tag{10-125}$$

which is finally simplified to

$$\left(x-\frac{M^2}{1-M^2}\right)^2+y^2=\left(\frac{M}{1-M^2}\right)^2 \qquad M\neq 1 \tag{10-126}$$

For a given value of *M*, Eq. (10-126) represents a circle with the center at

$$x = \operatorname{Re} G(j\omega) = \frac{M^2}{1 - M^2} \qquad y = 0 \tag{10-127}$$

The radius of the circle is

$$r = \left| \frac{M}{1 - M^2} \right| \tag{10-128}$$

- When the system is unstable, the constant-*M* loci and M_r no longer have any meaning.
- BW is the frequency where the *G*(*jω*) curve intersects the *M* = −3 dB locus of the Nichols chart.

When *M* takes on different values, Eq. (10-126) describes in the *G*(*jω*)-plane a family of circles that are called the **constant-*M* loci,** or the **constant-*M* circles**. Figure 10-55 illustrates a typical set of constant-*M* circles in the *G*(*jω*)-plane. These circles are symmetrical with respect to the *M* = 1 line and the real axis. The circles to the left of the *M* = 1 locus correspond to values of *M* greater than 1, and those to the right of the *M* = 1 line are for *M* less than 1. Equations (10-126) and (10-127) show that, when *M* becomes infinite, the circle degenerates to a point at (−1, *j*0). Graphically, the intersection of the *G*(*jω*) curve and the constant-*M* circle gives the value of *M* at the corresponding frequency on the *G*(*jω*) curve. If we want to keep the value of M_r less than a certain value, the *G*(*jω*) curve must not intersect the corresponding *M* circle at any point and at the same time must not enclose the (−1, *j*0) point. The constant-*M* circle with the smallest radius that is tangent to the *G*(*jω*) curve gives the value of M_r, and the resonant frequency ω_r is read off at the tangent point on the *G*(*jω*) curve.

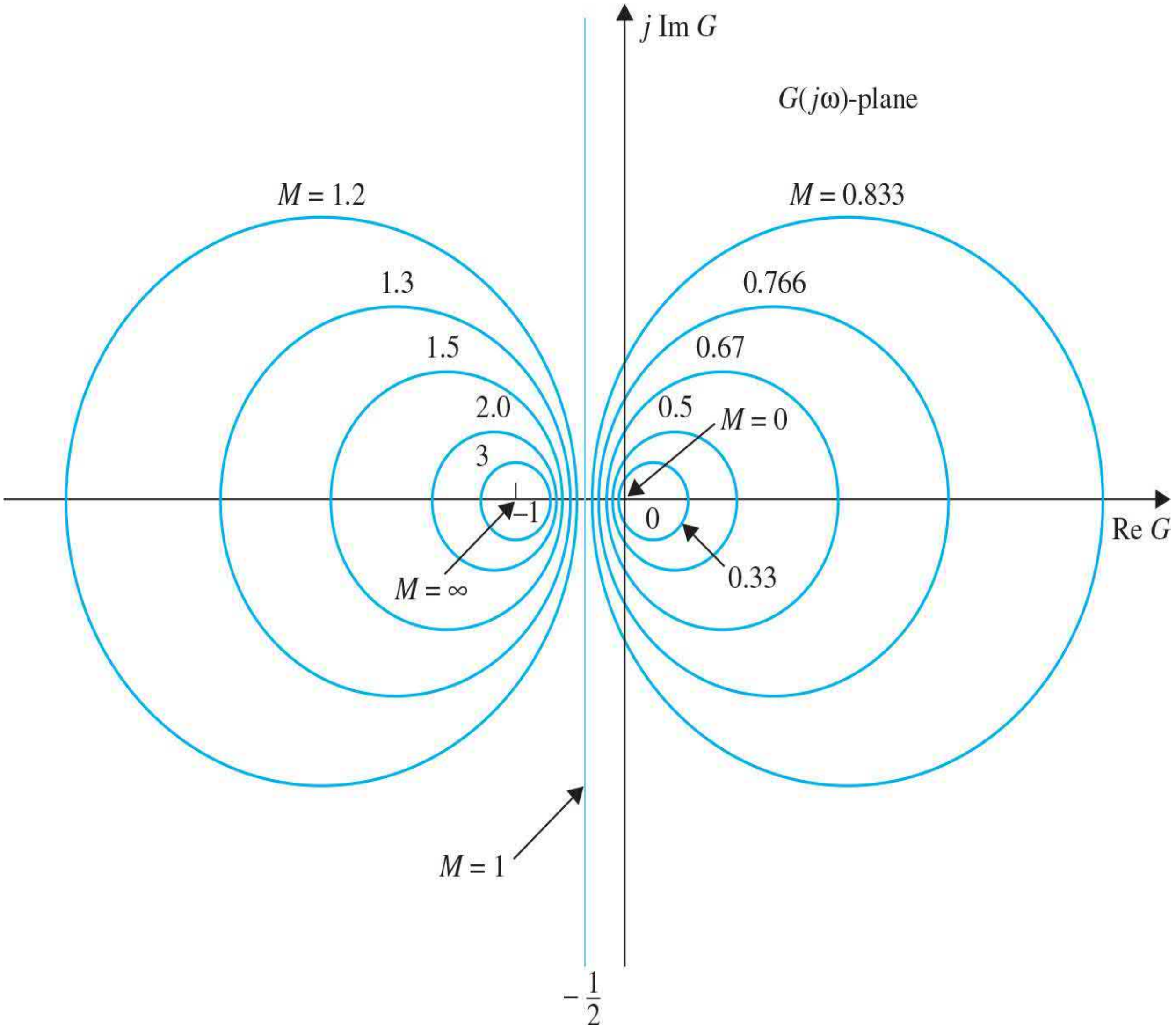
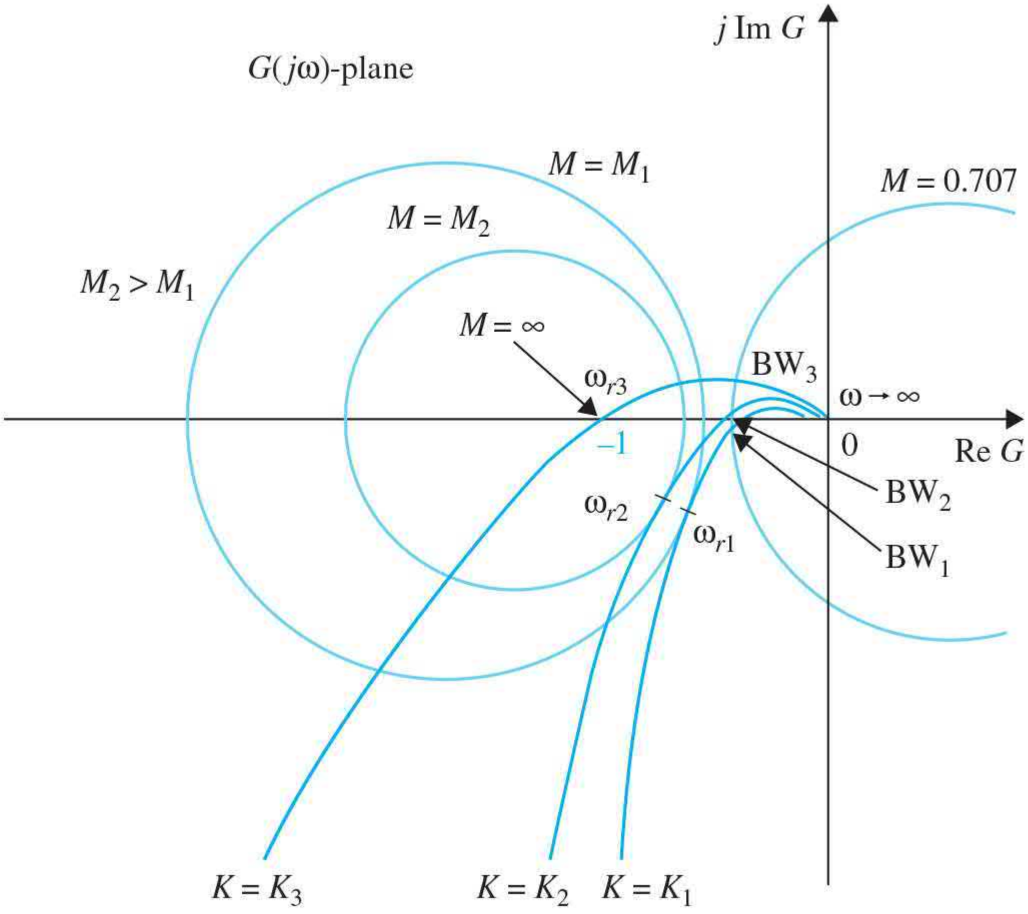


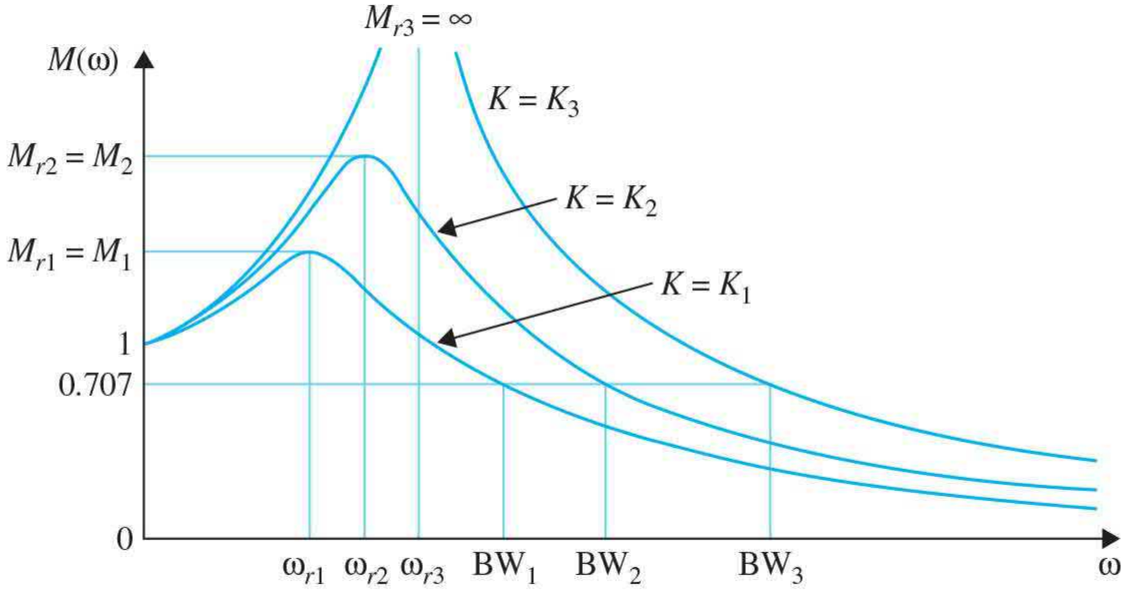
Figure 10-55 Constant-*M* circles in polar coordinates.

Figure 10-56*a* illustrates the Nyquist plot of $G(j\omega)$ for a unity-feedback control system together with several constant-*M* loci. For a given loop gain $K = K_1$, the intersects between the $G(j\omega)$ curve and the constant-*M* loci give the points on the $|M(j\omega)|$-versus-ω curve. The resonant peak M_r is found by locating the smallest circle that is tangent to the $G(j\omega)$ curve. The resonant frequency is found at the point of tangency and is designated as ω_{r1}. If the loop gain is increased to K_2, and if the system is still stable, a constant-*M* circle with a smaller radius that corresponds to a larger *M* is found tangent to the $G(j\omega)$ curve, and thus the resonant peak will be larger. The resonant frequency is shown to be ω_{r2}, which is closer to the phase-crossover frequency ω_p than ω_{r1}. When K is increased to K_3, so that the $G(j\omega)$ curve now passes through the $(-1, j0)$ point, the system is marginally stable, and M_r is infinite; ω_{p3} is now the same as the resonant frequency ω_r.



$G(j\omega)$-plane
$j \text{ Im } G$
$M = M_1$
$M = M_2$
$M = 0.707$
$M_2 > M_1$
$M = \infty$
ω_{r3}
BW_3
$\omega \to \infty$
-1
0
$\text{Re } G$
ω_{r2}
ω_{r1}
BW_2
BW_1
$K = K_3$
$K = K_2$
$K = K_1$

(a)



$M_{r3} = \infty$
$M(\omega)$
$K = K_3$
$M_{r2} = M_2$
$K = K_2$
$M_{r1} = M_1$
$K = K_1$
1
0.707
0
ω_{r1}
ω_{r2}
ω_{r3}
BW_1
BW_2
BW_3
ω

(b)

Figure 10-56 (*a*) Polar plots of $G(s)$ and constant-M loci. (*b*) Corresponding magnification curves.

When enough points of intersection between the $G(j\omega)$ curve and the constant-M loci are obtained, the magnification curves of $|M(j\omega)|$-versus-ω are plotted, as shown in Fig. 10-56*b*.

The bandwidth of the closed-loop system is found at the intersect of the $G(j\omega)$ curve and the $M = 0.707$ locus. For values of K beyond K_3, the system is unstable, and the constant-M loci and M_r no longer have any meaning.

A major disadvantage in working in the polar coordinates of the Nyquist plot of $G(j\omega)$ is that the curve no longer retains its original shape when a simple modification such as the change of the loop gain is made to the system. Frequently, in design situations, not only must the loop gain be altered, but a series controller may have to be added to the system. This requires the complete reconstruction of the Nyquist plot of the modified $G(j\omega)$. For design work involving M_r and BW as specifications, it is more convenient to work with the magnitude-phase plot of $G(j\omega)$, because when the loop gain is altered, the entire $G(j\omega)$ curve is shifted up or down vertically without distortion. When the phase properties of $G(j\omega)$ are changed independently, without affecting the gain, the magnitude-phase plot is affected only in the horizontal direction.

For that reason, the constant-M loci in the polar coordinates are plotted in magnitude-phase coordinates, and the resulting loci are called the **Nichols chart**. A typical Nichols chart of selected constant-M loci is shown in Fig. 10-57. Once the $G(j\omega)$ curve of the system is constructed in the Nichols chart, the intersects between the constant-M loci and the $G(j\omega)$ trajectory give the value of M at the corresponding frequencies of $G(j\omega)$. The resonant peak M_r is found by locating the smallest of the constant-M locus ($M \geq 1$) that is tangent to the $G(j\omega)$ curve from above. The resonant frequency is the frequency of $G(j\omega)$ at the point of tangency. *The bandwidth of the closed-loop system is the frequency at which the* $\mathrm{G}(\,\mathrm{j}\omega)$ *curve intersects the* $\mathrm{M} = 0.707$ *or* $\mathrm{M} = -3$*dB locus*.

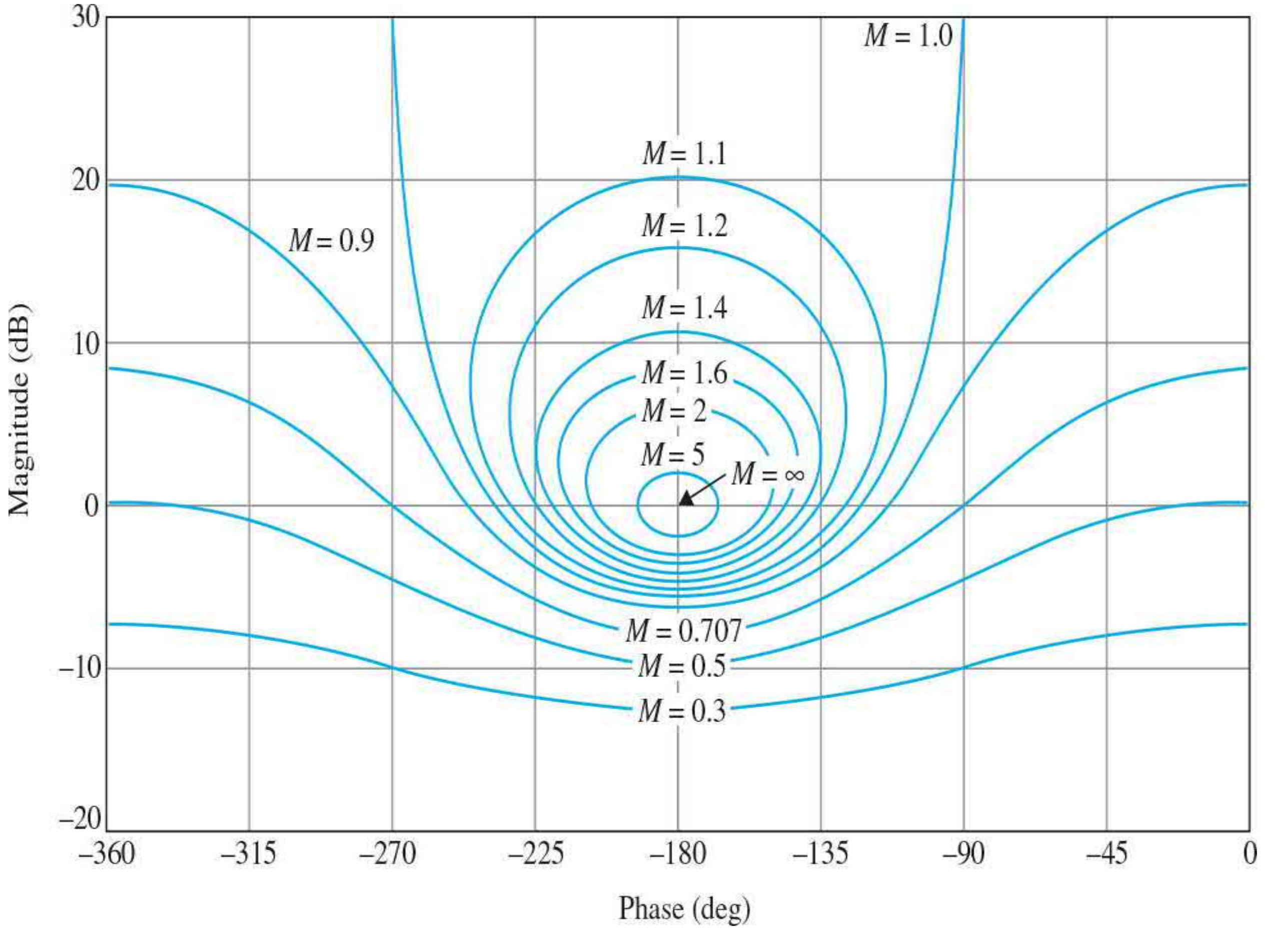


Figure 10-57 Nichols chart.

The following example illustrates the relationship among the analysis methods using the Bode plot and the Nichols chart.

EXAMPLE 10-13-1 Consider the position-control system of the control surfaces of the airplane analyzed in Sec. 7-9. The forward-path transfer function of the unity-feedback system is given by Eq. (7-169), and is repeated here:

$$G(s) = \frac{1.5\times 10^7\ K}{s(s+400.26)(s+3008)} \tag{10-129}$$

The Bode plots for $G(j\omega)$ are shown in Fig. 10-58 for K = 7.248, 14.5, 181.2, and 273.57. The gain and phase margins of the closed-loop system for these values of K are determined and shown on the Bode plots. The magnitude-phase plots of $G(j\omega)$ corresponding to the Bode plots are shown in Fig. 10-59. These magnitude-phase plots, together with the Nichols chart, give information on the resonant peak M_r, resonant frequency ω_r, and the bandwidth BW. The gain and phase margins are also clearly marked on the magnitude-phase plots. Figure 10-60 shows the closed-loop frequency responses. Table 10-3 summarizes the results of the frequency-domain analysis for the four different values of K together with the time-domain maximum overshoot Sec. 7-9.

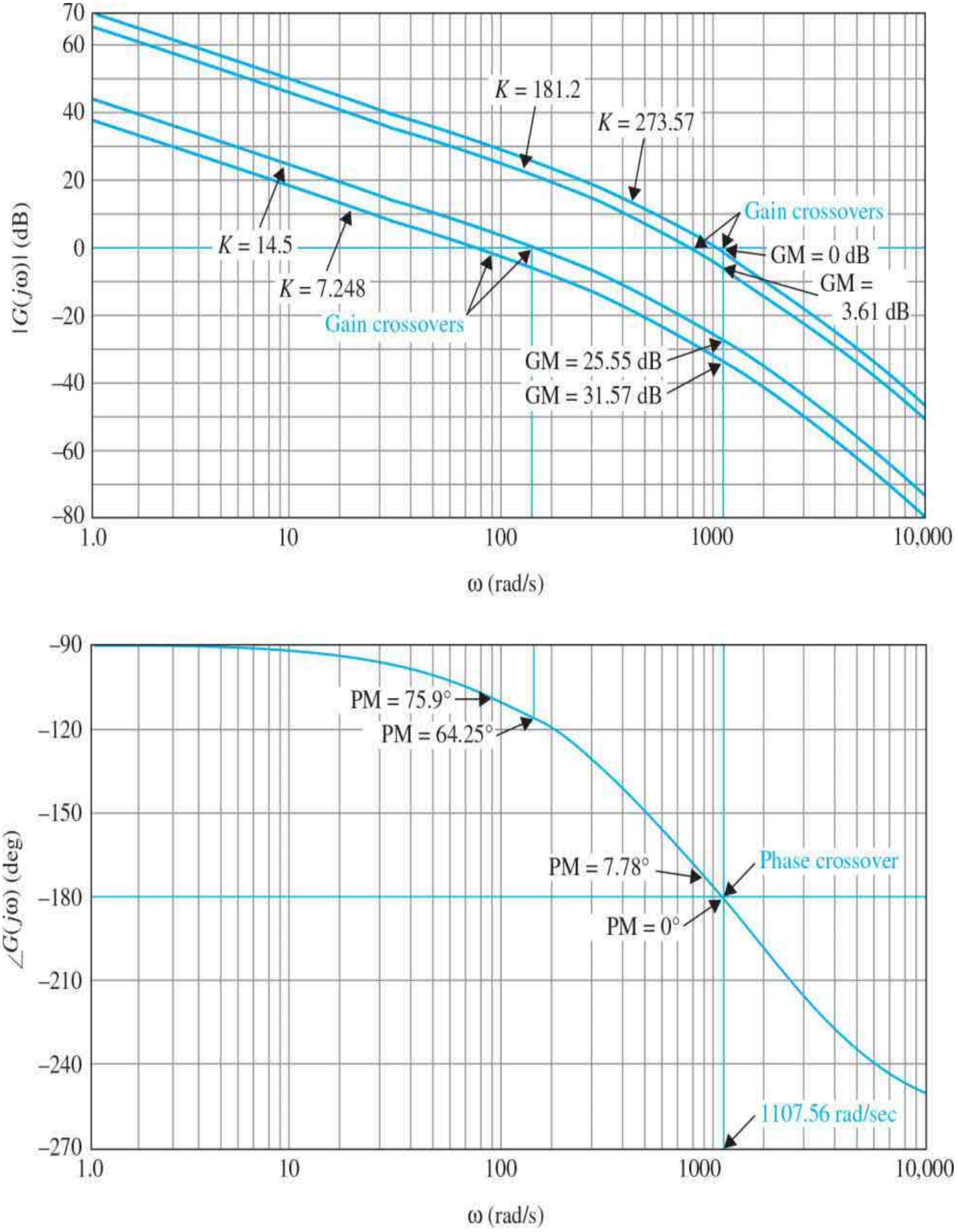


Figure 10-58 Bode diagrams of the system in Example 10-13-1.

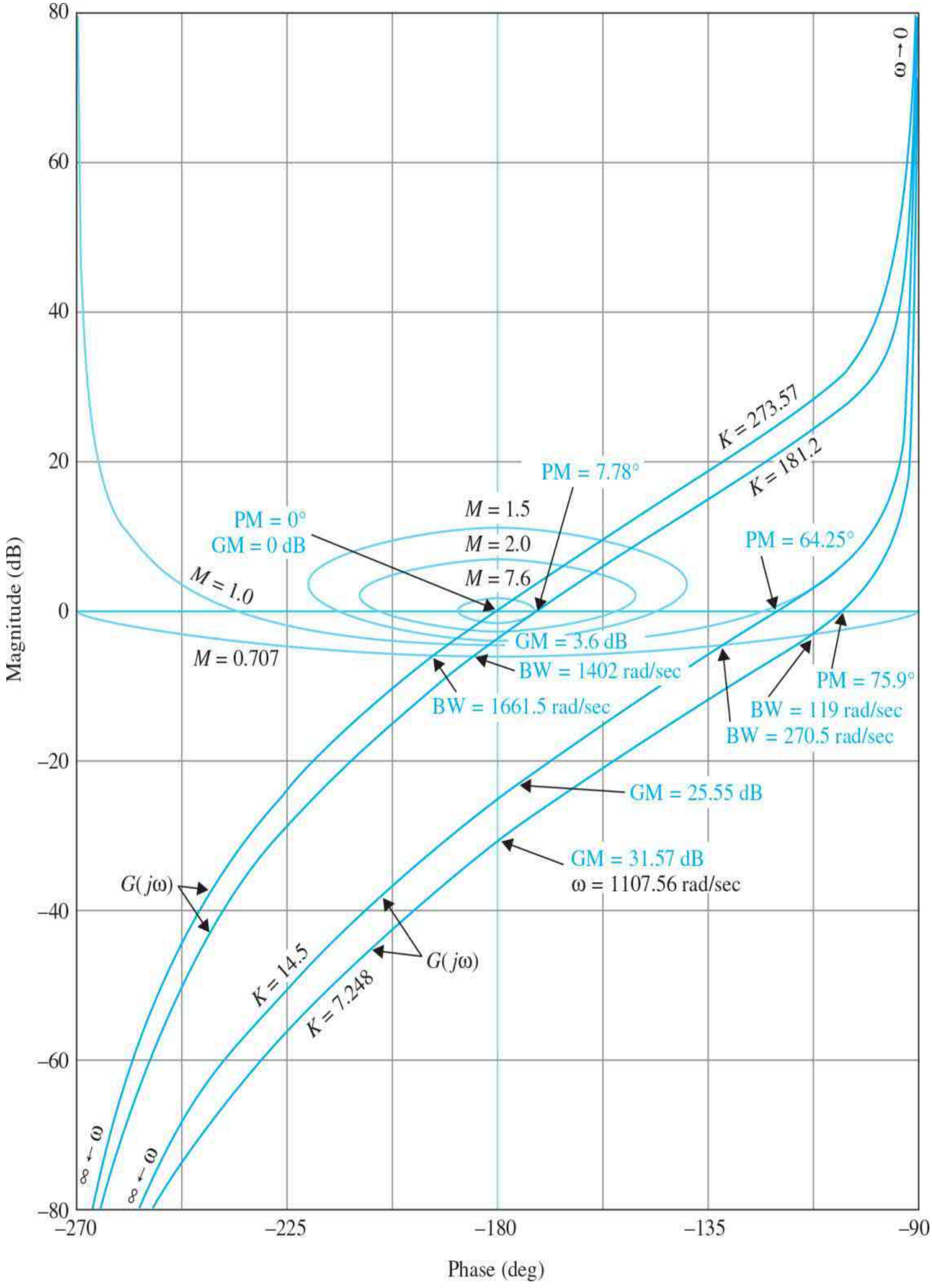

80
60
40
20
0
–20
–40
–60
–80
Magnitude (dB)
–270
–225
–180
–135
–90
Phase (deg)
ω → 0
K = 273.57
K = 181.2
PM = 7.78°
M = 1.5
PM = 0°
GM = 0 dB
M = 2.0
M = 7.6
PM = 64.25°
M = 1.0
GM = 3.6 dB
M = 0.707
BW = 1402 rad/sec
BW = 1661.5 rad/sec
PM = 75.9°
BW = 119 rad/sec
BW = 270.5 rad/sec
GM = 25.55 dB
GM = 31.57 dB
ω = 1107.56 rad/sec
G(jω)
K = 14.5
G(jω)
K = 7.248
ω → ∞
ω → ∞

Figure 10-59 Gain-phase plots and Nichols chart of the system in Example 10-13-1.

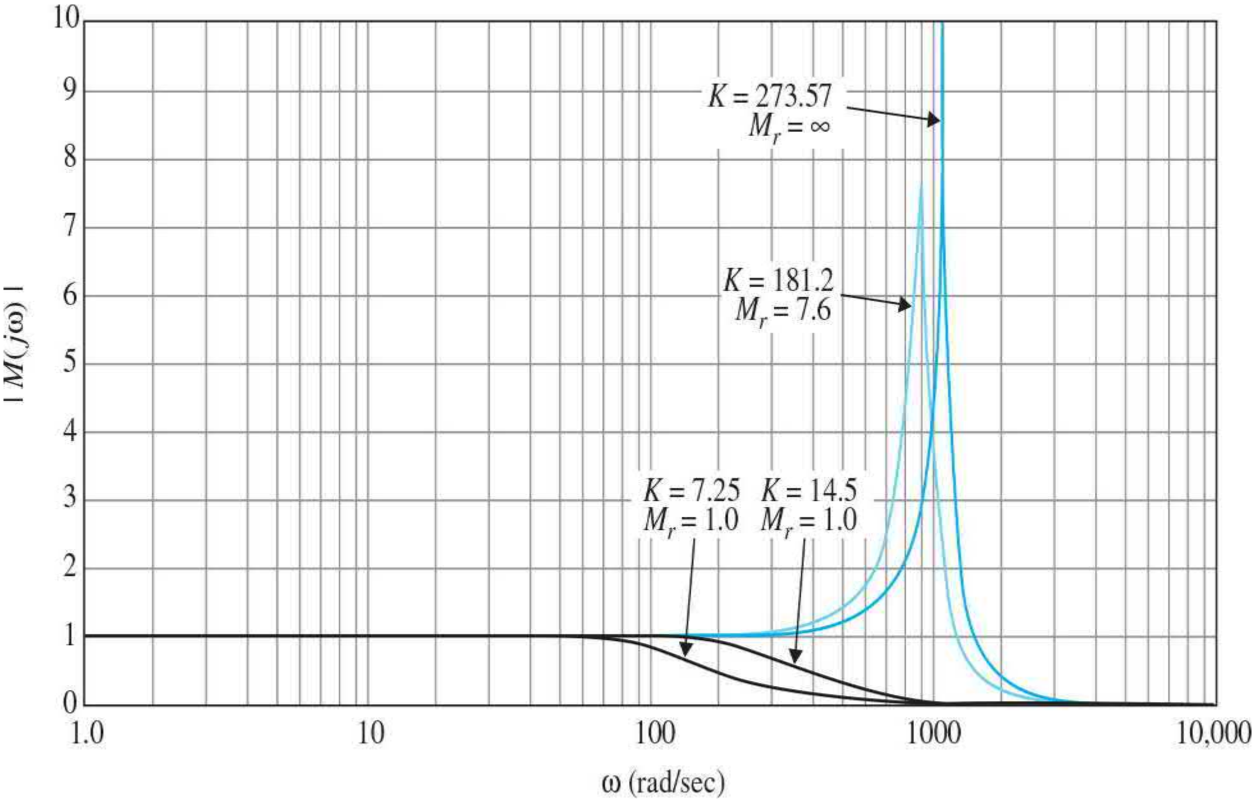


Figure 10-60 Closed-loop frequency response of the system in Example 10-13-1.

TABLE 10-3 Summary of Frequency-Domain Analysis

K	Maximum Overshoot (%)	M_r	ω_r (rad/s)	Gain Margin (dB)	Phase Margin (deg)	BW (rad/s)
7.25	0	1.0	1.0	31.57	75.9	119.0
14.5	4.3	1.0	43.33	5.55	64.25	270.5
181.2	15.2	7.6	900.00	3.61	7.78	1402.0
273.57	100.0	∞	1000.00	0	0	1661.5

10-14 NICHOLS CHART APPLIED TO

NONUNITY-FEEDBACK SYSTEMS

The constant-*M* loci and the Nichols chart presented in the preceding sections are limited to closed-loop systems with unity feedback whose transfer function is given by Eq. (10-119). When a system has nonunity feedback, the closed-loop transfer function of the system is expressed as

$$M(s)=\frac{G(s)}{1+G(s)H(s)} \tag{10-130}$$

where *H(s)* ≠ 1. The constant-*M* loci and the Nichols chart cannot be applied directly to obtain the closed-loop frequency response by plotting $G(j\omega)H(j\omega)$, since the numerator of $M(s)$ does not contain $H(j\omega)$.

By proper modification, the constant-*M* loci and Nichols chart can still be applied to a nonunity-feedback system. Let us consider the function

$$P(s)=H(s)M(s)=\frac{G(s)H(s)}{1+G(s)H(s)} \tag{10-131}$$

Apparently, Eq. (10-131) is of the same form as Eq. (10-119). The frequency response of $P(j\omega)$ can be determined by plotting the function $G(j\omega)H(j\omega)$ in the amplitude-phase coordinates along with the Nichols chart. Once this is done, the frequency-response information for $M(j\omega)$ is obtained as follows:

$$|M(j\omega)|=\frac{|P(j\omega)|}{|H(j\omega)|} \tag{10-132}$$

or, in terms of dB,

$$|M(j\omega)|(dB)=|P(j\omega)|(dB)-|H(j\omega)|(dB) \tag{10-133}$$

$$\phi_m(j\omega)=\angle M(j\omega)=\angle P(j\omega)-\angle H(j\omega) \tag{10-134}$$

10-15 SENSITIVITY STUDIES IN THE FREQUENCY DOMAIN

Sensitivity study is easily carried out in the frequency domain.

The advantage of using the frequency domain in linear control systems is that higher-order systems can be handled more easily than in the time domain. Furthermore, the sensitivity of the system with respect to parameter variations can be easily interpreted using frequency-domain plots. We shall show how the Nyquist plot and the Nichols chart can be utilized for the analysis and design of control systems based on sensitivity considerations.

Consider a linear control system with unity feedback described by the transfer function

$$M(s) = \frac{G(s)}{1+G(s)} \tag{10-135}$$

The sensitivity of $M(s)$ with respect to the loop gain K, which is a multiplying factor in $G(s)$, is defined as

$$S_G^M(s) = \frac{\dfrac{dM(s)}{M(s)}}{\dfrac{dG(s)}{G(s)}} = \frac{dM(s)}{dG(s)}\frac{G(s)}{M(s)} \tag{10-136}$$

Taking the derivative of $M(s)$ with respect to $G(s)$ and substituting the result into Eq. (10-136) and simplifying, we have

$$S_G^M(s) = \frac{1}{1+G(s)} = \frac{1/G(s)}{1+1/G(s)} \tag{10-137}$$

Clearly, the sensitivity function $S^M_G(s)$ is a function of the complex variable s. Figure 10-61 shows the magnitude plot of $S^M_G(s)$ when $G(s)$ is the transfer function given in Eq. (10-98).

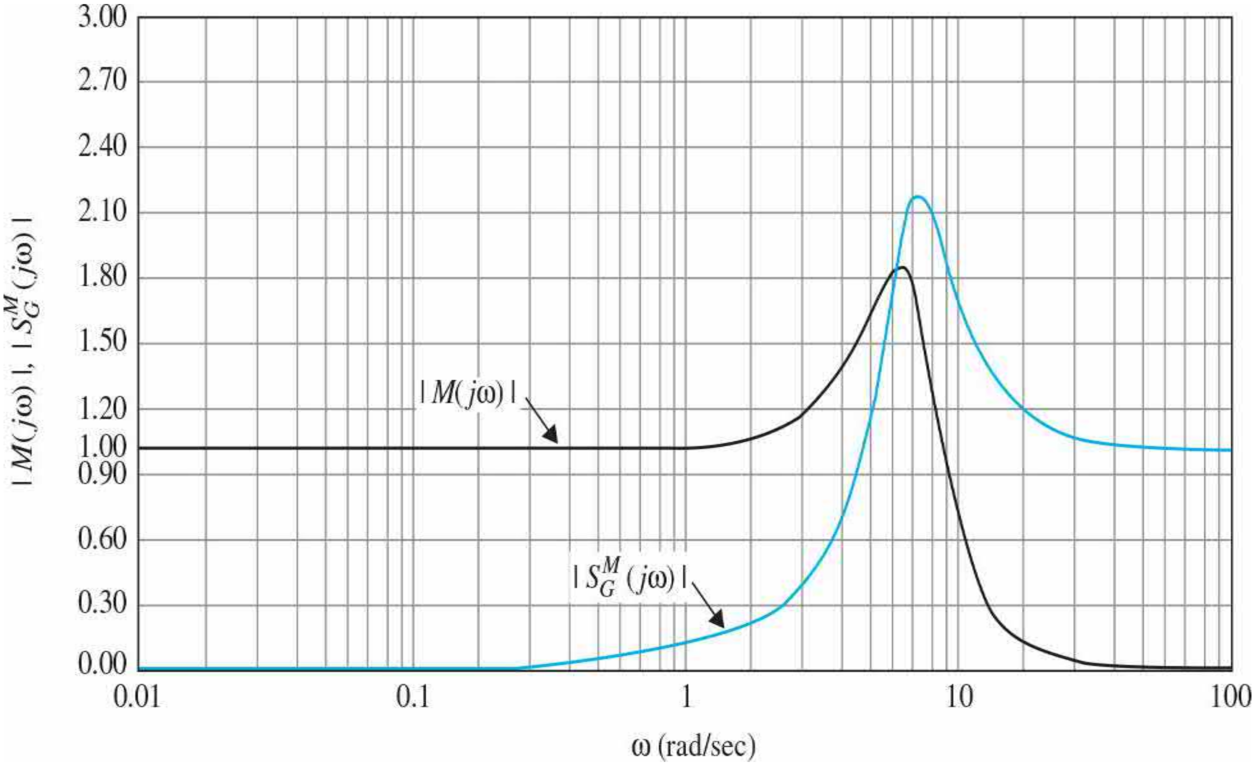


Figure 10-61 $|M(j\omega)|$ and $|S_G^M(j\omega)|$ versus ω for $G(s)=\dfrac{2500}{s(s+5)(s+2500)}$.

It is interesting to note that the sensitivity of the closed-loop system is inferior at frequencies greater than 4.8 rad/s to the open-loop system whose sensitivity to the variation of K is always unity. In general, it is desirable to formulate a design criterion on sensitivity in the following manner:

$$\left|S_G^M(j\omega)\right|=\frac{1}{|1+G(j\omega)|}=\frac{|1/G(j\omega)|}{|1+1/G(j\omega)|}\leq k \qquad (10\text{-}138)$$

where k is a positive real number. This sensitivity criterion is in addition to the regular performance criteria on the steady-state error and the relative stability.

Equation (10-138) is analogous to the magnitude of the closed-loop transfer function, $|M(j\omega)|$, given in Eq. (10-121), with $G(j\omega)$ replaced by $1/G(j\omega)$. Thus, the sensitivity function of Eq. (10-138) can be determined by plotting $1/G(j\omega)$ in the magnitude-phase coordinates with the Nichols chart. Figure 10-62 shows the magnitude-phase plots of $G(j\omega)$ and $1/G(j\omega)$ of Eq.

(10-113). Notice that $G(j\omega)$ is tangent to the $M = 1.8$ locus from below, which means that M_r of the closed-loop system is 1.8. The $1/G(j\omega)$ curve is tangent to the $M = 2.2$ curve from above and, according to Fig. 10-61, is the maximum value of $|S^M_G(s)|$. Equation (10-138) shows that, for low sensitivity, the loop gain of $G(j\omega)$ must be high, but it is known that, in general, high gain could cause instability. Thus, the designer is again challenged by the task of designing a system with both a high degree of stability and low sensitivity.

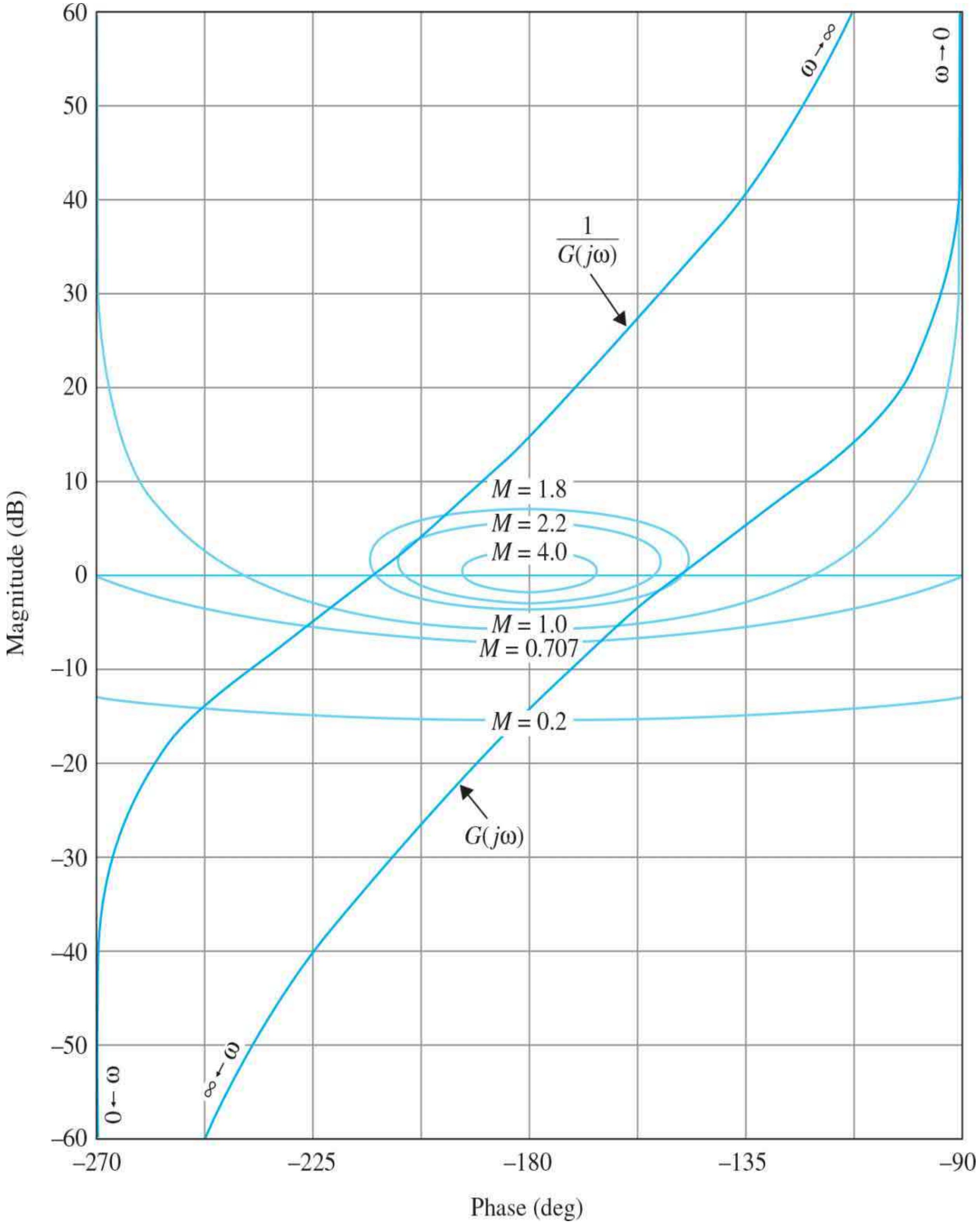


Figure 10-62 Magnitude-phase plots of $G(j\omega)$ and $1/G(j\omega)$ for

$$G(s)=\frac{2500}{s(s+5)(s+50)}.$$

The design of robust control systems (low sensitivity) with the frequency-domain methods is discussed in Chap. 11.

10-16 MATLAB TOOLS AND CASE STUDIES

Apart from the MATLAB toolboxes in this chapter, this chapter does not contain any software. In Chap. 11, we will introduce the MATLAB SISO design tool, which provides a GUI (graphical user interface) approach for the analysis of control engineering transfer functions.

10-17 SUMMARY

The chapter began by describing typical relationships between the open-loop and closed-loop frequency responses of linear systems. Performance specifications such as the resonance peak M_r, resonant frequency ω_r, and bandwidth BW were defined in the frequency domain. The relationships among these parameters of a second-order prototype system were derived analytically. The effects of adding simple poles and zeros to the loop transfer function on M_r and BW were discussed.

The Nyquist criterion for stability analysis of linear control systems was developed. The stability of a single-loop control system can be investigated by studying the behavior of the Nyquist plot of the loop transfer function $G(s)H(s)$ for $\omega = 0$ to $\omega = \infty$ with respect to the critical point. If $G(s)H(s)$ is a minimum-phase transfer function, the condition of stability is simplified so that the Nyquist plot will not enclose the critical point.

The relationship between the root loci and the Nyquist plot was described in Sec. 10-6. The discussion should add more perspective to the understanding of both subjects.

Relative stability was defined in terms of gain margin and phase margin. These quantities were defined in the polar coordinates as well as on the Bode diagram. The gain-phase plot allows the Nichols chart to be constructed for closed-loop analysis. The values of M_r and BW can be easily found by plotting the $G(j\omega)$ locus on the Nichols chart.

The stability of systems with pure time delay is analyzed by use of the Bode plot.

Sensitivity function $S^{M}_{G}(j\omega)$ was defined as a measure of the variation of $M(j\omega)$ due to variations in $G(j\omega)$. It was shown that the frequency-response plots of $G(j\omega)$ and $1/G(j\omega)$ can be readily used for sensitivity studies.

Finally, using the MATLAB toolboxes developed in this chapter, the reader may practice all the concepts discussed here.

REFERENCES

Nyquist Criterion of Continuous-Data Systems

1. H. Nyquist, “Regeneration Theory,” *Bell System. Tech. J.*, Vol. 11, pp. 126–147, Jan. 1932.
2. R. W. Brockett and J. L. Willems, “Frequency Domain Stability Criteria —Part II,” *IEEE Trans. Automatic Control*, Vol. AC-10, pp. 255–261, July 1965.
3. R. W. Brockett and J. L. Willems, “Frequency Domain Stability Criteria —Part II,” *IEEE Trans. Automatic Control*, Vol. AC-10, pp. 407–413, Oct. 1965.
4. T. R. Natesan, “A Supplement to the Note on the Generalized Nyquist Criterion,” *IEEE Trans. Automatic Control*, Vol. AC-12, pp. 215–216, April 1967.
5. K. S. Yeung, “A Reformulation of Nyquist’s Criterion,” *IEEE Trans. Educ*. Vol. E-28, pp. 59–60, Feb. 1985.

Sensitivity Function

6. A. Gelb, “Graphical Evaluation of the Sensitivity Function Using the Nichols Chart,” **IRE Trans. Automatic Control**, Vol. AC-7, pp. 57–58, Jul. 1962.

PROBLEMS

10-1. The forward-path transfer function of a unity-feedback control system is

$$G(s) = \frac{K}{s(s+6.54)}$$

Analytically, find the resonance peak M_r, resonant frequency ω_r, and bandwidth BW of the closed-loop system for the following values of K:

(a) $K = 5$

(b) $K = 21.38$

(c) $K = 100$

Use the formulas for the second-order prototype system given in the text.

10-2. Use MATLAB to verify your answer to Prob. 10-1.

10-3. The transfer function of a system is

$$G(s) = \frac{s + \frac{1}{A_1}}{s + \frac{1}{A_2}}$$

Determine when the system is a lead-network or lag-network.

10-4. Use MATLAB to solve the following problems. Do not attempt to obtain the solutions analytically. The forward-path transfer functions of unity-feedback control systems are given in the following equations. Find the resonance peak M_r, resonant frequency ω_r, and bandwidth BW of the closed-loop systems. (Reminder: Make certain that the system is stable.)

(a) $G(s)=\dfrac{5}{s(1+0.5s)(1+0.1s)}$

(b) $G(s)=\dfrac{10}{s(1+0.5s)(1+0.1s)}$

(c) $G(s)=\dfrac{500}{(s+1.2)(s+4)(s+10)}$

(d) $G(s)=\dfrac{10(s+1)}{s(s+2)(s+10)}$

(e) $G(s)=\dfrac{0.5}{s(s^2+s+1)}$

(f) $G(s)=\dfrac{100e^{-s}}{s(s^2+10s+50)}$

(g) $G(s)=\dfrac{100e^{-s}}{s(s^2+10s+100)}$

(h) $G(s)=\dfrac{10(s+5)}{s(s^2+5s+5)}$

10-5. The specifications on a second-order unity-feedback control system with the closed-loop transfer function

$$M(s)=\frac{Y(s)}{R(s)}=\frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$$

are that the maximum overshoot must not exceed 10 percent and the rise time must be less than 0.1 s. Find the corresponding limiting values of M_r and BW analytically.

10-6. Repeat Prob. 10-5 for maximum overshoot less than or equal to 20 percent and $t_r \leq 0.2$ s.

10-7. Repeat Prob. 10-5 for maximum overshoot less than or equal to 30 percent and $K = 10$.

10-8. Consider the forward-path of a unity-feedback control system given by

$$G(s)=\frac{0.5K}{s(0.25s^2+0.375s+1)}$$

(a) Analytically find K such that the closed-loop bandwidth is about 1.5 rad/s (0.24 Hz).

(b) Use MATLAB to verify your answer to part (a).

10-9. Repeat Prob. 10-8 with a resonance peak of 2.2.

10-10. The closed-loop frequency response $|M(j\omega)|$-versus-frequency of a second-order prototype system is shown in Fig 10P-10. Sketch the corresponding unit-step response of the system; indicate the values of the maximum overshoot, peak time, and the steady-state error due to a unit-step input.

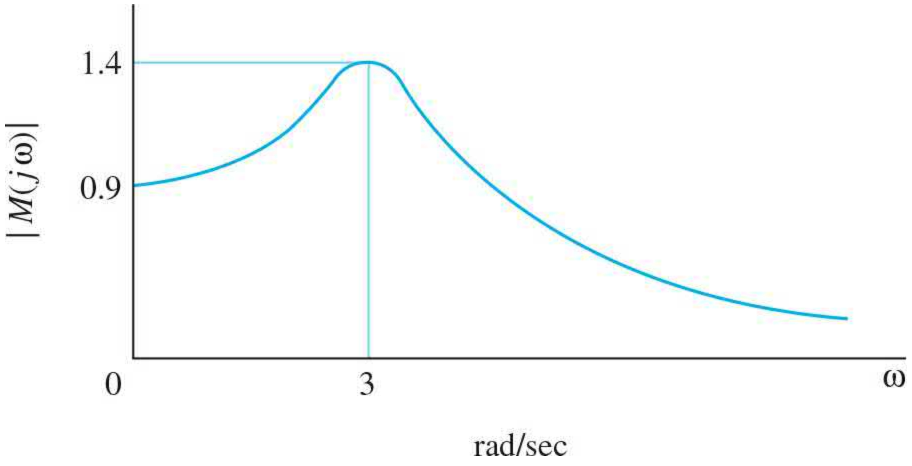


Figure 10P-10

10-11. The forward-path transfer function of a system with an integral control $H(s)=\dfrac{K}{s}$ is

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

(a) Find K when the closed-loop resonance peak is 1.4.

(b) Determine the frequency at resonance, overshoot for step input, phase margin, and closed-loop BW according to the result of part (a).

10-12. The forward-path transfer function of a unity-feedback control system is

$$G(s)=\frac{1+Ts}{2s(s^2+s+1)}$$

Use MATLAB to find the values of BW and M_r of the closed-loop system for $T = 0.05$, 1, 2, 3, 4, and 5.

10-13. The forward-path transfer function of a unity-feedback control system is

$$G(s)=\frac{1}{2s(s^2+s+1)(1+Ts)}$$

Use MATLAB to find the values of BW and M_r of the closed-loop system for T = 0, 0.5, 1, 2, 3, 4, and 5. Use MATLAB to find the solutions.

10-14. If a loop transfer function of a system is given by

$$G(s)H(s)=\frac{0.5K}{0.25s^3+0.375s^2+s+0.5k}$$

(a) Use the second-order approximation to find the BW and the damping ratio.

(b) If BW = 1.5 rad/s, find K and the damping ratio.

(c) Use MATLAB to verify your answer to part (b).

10-15. The loop transfer functions $L(s)$ of single-feedback-loop systems are given below. Sketch the Nyquist plot of $L(j\omega)$ for $\omega = 0$ to $\omega = \infty$. Determine the stability of the closed-loop system. If the system is unstable, find the number of poles of the closed-loop transfer function that are in the right-half s-plane. Solve for the intersect of $L(j\omega)$ on the negative real axis of the $L(j\omega)$-plane analytically. You may construct the Nyquist plot of $L(j\omega)$ using MATLAB.

(a) $L(s)=\frac{20}{s(1+0.1s)(1+0.5s)}$

(b) $L(s)=\frac{10}{s(1+0.1s)(1+0.5s)}$

(c) $L(s)=\frac{100(1+s)}{s(1+0.1s)(1+0.2s)(1+0.5s)}$

(d) $L(s)=\frac{10}{s^2(1+0.2s)(1+0.5s)}$

(e) $L(s)=\frac{3(s+2)}{s(s^3+3s+1)}$

(f) $L(s)=\frac{0.1}{s(s+1)(s^2+s+1)}$

(g) $L(s)=\frac{100}{s(s+1)(s^2+2)}$

(h) $L(s)=\frac{10(s+10)}{s(s+1)(s+100)}$

10-16. The loop transfer functions of single-feedback-loop control systems are given in the following equations. Apply the Nyquist criterion and determine the values of K for the system to be stable. Sketch the Nyquist plot of $L(j\omega)$ with $K = 1$ for $\omega = 0$ to $\omega = \infty$. You may use a computer program to plot the Nyquist plots.

(a) $L(s)=\dfrac{K}{s(s+2)(s+10)}$

(b) $L(s)=\dfrac{K(s+1)}{s(s+2)(s+5)(s+15)}$

(c) $L(s)=\dfrac{K}{s^2(s+2)(s+10)}$

(d) $L(s)=\dfrac{K}{(s+5)(s+2)^2}$

(e) $L(s)=\dfrac{K(s+5)(s+1)}{(s+50)(s+2)^3}$

10-17. The forward-path transfer function of a unity-feedback control system is

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

Determine by means of the Nyquist criterion, the range of $K(-\infty < K < \infty)$ for the closed-loop system to be stable. Sketch the Nyquist plot of $G(j\omega)$ for $\omega = 0$ to $\omega = \infty$.

(a) $n = 2$

(b) $n = 3$

(c) $n = 4$

10-18. Sketch the Nyquist plot for the controlled system shown in Fig. 10P-18.

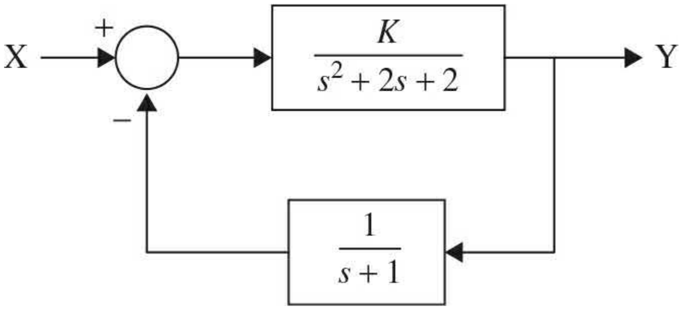


Figure 10P-18

Determine by means of the Nyquist criterion, the range of $K(-\infty < K < \infty)$

for the closed-loop system to be stable.

10-19. The characteristic equation of a linear control system is given in the following equation.

$$s(s^3+2s^2+s+1)+K(s^2+s+1)=0$$

(a) Apply the Nyquist criterion to determine the values of K for system stability.

(b) Check the answers by means of the Routh-Hurwitz criterion.

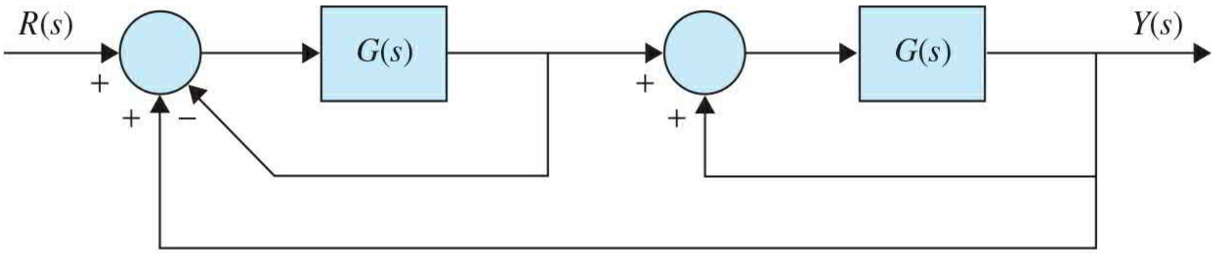
10-20. Repeat Prob. 10-19 for $s^3 + 3s^2 + 3s + 1 + K = 0$.

10-21. The forward-path transfer function of a unity-feedback control system with a PD (proportional-derivative) controller is

$$G(s)=\frac{10(K_P+K_D s)}{s^2}$$

Select the value of K_P so that the parabolic-error constant K_a is 100. Find the equivalent forward-path transfer function $G_{eq}(s)$ for $\omega = 0$ to $\omega = \infty$. Determine the range of K_D for stability by the Nyquist criterion.

10-22. The block diagram of a feedback control system is shown in Fig. 10P-22.



$$G(s)=\frac{K}{(s+4)(s+5)}$$

Figure 10P-22

(a) Apply the Nyquist criterion to determine the range of K for stability.

(b) Check the answer obtained in part (a) with the Routh-Hurwitz criterion.

10-23. The forward-path transfer function of the liquid-level control system shown in Prob. 2-36 is

$$G(s)=\frac{K_a K_i n K_I N}{s(R_a Js+K_i K_b)(As+K_o)}$$

The following system parameters are given: $K_a = 50$, $K_i = 10$, $K_t = 50$, $J = 0.006$, $K_b = 0.0706$, $n = 0.01$, and $R_a = 10$. The values of A, N, and K_o are variable.

(a) For $A = 50$ and $K_o = 100$, sketch the Nyquist plot of $G(j\omega)$ for ω = 0 to ∞ to ∞ with N as a variable parameter. Find the maximum integer value of N so that the closed-loop system is stable.

(b) Let $N = 10$ and $K_o = 100$. Sketch the Nyquist plot of an equivalent transfer function $G_{eq}(j\omega)$ that has A as a multiplying factor. Find the critical value of K_o for stability.

(c) For $A = 50$ and $N = 10$, sketch the Nyquist plot of an equivalent transfer function $G_{eq}(j\omega)$ that has K_o as a multiplying factor. Find the critical value of K_o for stability.

10-24. The block diagram of a dc-motor control system is shown in Fig. 10P-24. Determine the range of K for stability using the Nyquist criterion when K_t has the following values:

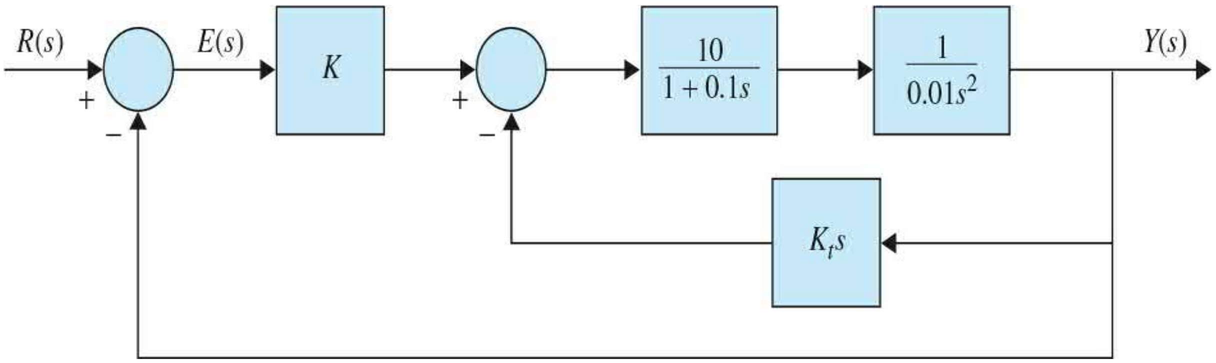


Figure 10P-24

(a) $K_t = 0$

(b) $K_t = 0.01$

(c) $K_t = 0.1$

10-25. For the system shown in Fig. 10P-24, let $K = 10$. Find the range of

K_t for stability with the Nyquist criterion.

10-26. Figure 10P-26 shows the block diagram of a servomotor.

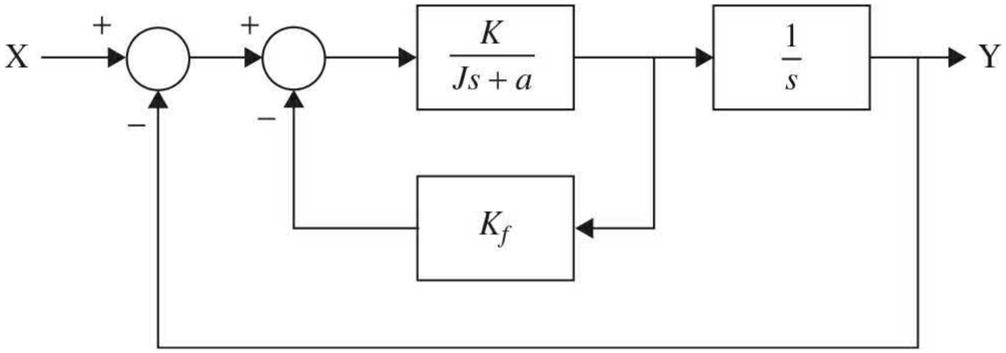


Figure 10P-26

Assume $J = 1$ kg·m² and $B = 1$ N·m/rad/s. Determine the range of K for stability using the Nyquist criterion when K_f has the following values:

(a) $K_f = 0$

(b) $K_f = 0.1$

(c) $K_f = 0.2$

10-27. For the system shown in Fig. 10P-26, let $K = 10$. Find the range of K_f for stability with the Nyquist criterion.

10-28. For the controlled system shown in Fig. 10P-28, draw the Nyquist plot and apply the Nyquist criterion to determine the range of K for stability and determine the number of roots in the right-half s-plane for the values of K, where the system is unstable.

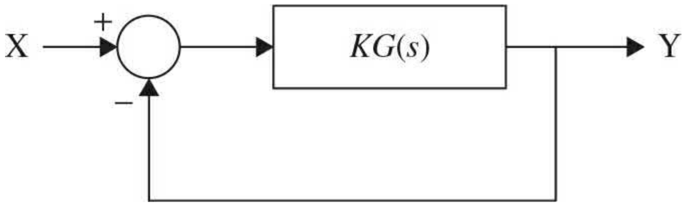


Figure 10P-28

(a) $G(s) = \dfrac{s+1}{(s-1)^2}$

(b) $G(s)=\frac{s-1}{(s+1)^2}$

10-29. The steel-rolling control system shown in Fig. 4P-18 has the forward-path transfer function

$$G(s)=\frac{100Ke^{-T_d s}}{s(s^2+10s+100)}$$

(a) When K = 1, determine the maximum time delay T_d in seconds for the closed-loop system to be stable.

(b) When the time delay T_d is 1 s, find the maximum value of K for system stability.

10-30. Repeat Prob. 10-29 with the following conditions.

(a) When K = 0.1, determine the maximum time delay T_d in seconds for the closed-loop system to be stable.

(b) When the time delay T_d is 0.1 s, find the maximum value of K for system stability.

10-31. The open-loop transfer function of a system is given by

$$G(s)H(s)=\frac{K}{s(\tau_1 s+1)(\tau_2 s+1)}$$

Study the stability of the system for the following:

(a) K is small.

(b) K is large.

10-32. The system schematic shown in Fig. 10P-32 is devised to control the concentration of a chemical solution by mixing water and concentrated solution in appropriate proportions. The transfer function of the system components between the amplifier output e_a (V) and the valve position x (in) is

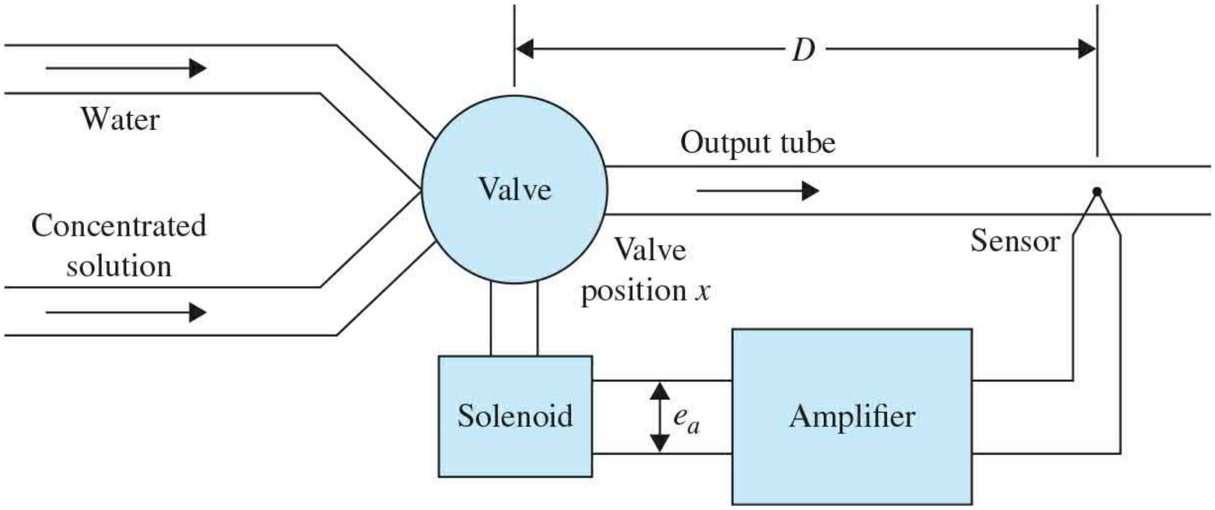


Figure 10P-32

$$\frac{X(s)}{E_a(s)} = \frac{K}{s^2 + 10s + 100}$$

When the sensor is viewing pure water, the amplifier output voltage e_a is zero; when it is viewing concentrated solution, e_a = 10 V; and 0.1 in of the valve motion changes the output concentration from zero to maximum. The valve ports can be assumed to be shaped so that the output concentration varies linearly with the valve position. The output tube has a cross-sectional area of 0.1 in^2, and the rate of flow is 10^3 in/s regardless of the valve position. To make sure the sensor views a homogeneous solution, it is desirable to place it at some distance D in from the valve.

(a) Derive the loop transfer function of the system.

(b) When K = 10, find the maximum distance D (in) so that the system is stable. Use the Nyquist stability criterion.

(c) Let D = 10 in. Find the maximum value of K for system stability.

10-33. For the mixing system described in Prob. 10-32, the following system parameters are given:

When the sensor is viewing pure water, the amplifier output voltage e_s = 0 V; when it is viewing concentrated solution, e_a = 1 V; and 0.1 in of the valve motion changes the output concentration from zero to maximum. The rest of the system characteristics are the same as in Prob. 10-32. Repeat the three parts of Prob. 10-32.

10-34. Figure 10P-34 shows the block diagram of a controlled system.

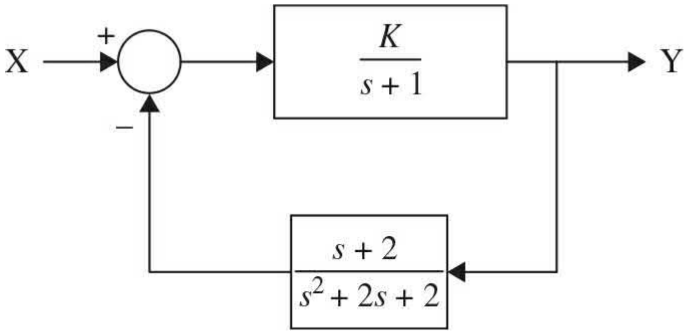


Figure 10P-34

(a) Draw the Nyquist plot and apply the Nyquist criterion to determine the range of *K* for stability.

(b) Determine the number of roots in the right-half *s*-plane for the values of *K*, where the system is unstable.

(c) Use Routh's criterion to determine the range of *K* for stability.

10-35. The forward-path transfer function of a unity-feedback control system is

$$G(s) = \frac{1000}{s(s^2 + 105s + 600)}$$

(a) Find the values of M_r, ω_r, and BW of the closed-loop system.

(b) Find the parameters of the second-order system with the open-loop transfer function

$$G_L(s) = \frac{\omega_n^2}{s(s + 2\zeta\omega_n)}$$

that will give the same values for M_r and ω_r as the third-order system. Compare the values of BW of the two systems.

10-36. Sketch or plot the Bode diagrams of the forward-path transfer functions given in Prob. 10-4. Find the gain margin, gain-crossover frequency, phase margin, and the phase-crossover frequency for each system.

10-37. The transfer function of a system is given by

$$G(s)H(s)=\frac{25(s+1)}{s(s+2)(s^2+2s+16)}$$

Use MATLAB to plot the Bode diagrams of the system and find the phase margin and gain margin of the system.

10-38. Use MATLAB to plot the Bode diagrams of the system shown in Fig. 10P-34, where $K = 1$, and determine the stable range of K by using phase margin and gain margin.

10-39. The forward-path transfer functions of unity-feedback control systems are given in the following equations. Plot the Bode diagram of $G(j\omega)/K$, and do the following: (1) Find the value of K so that the gain margin of the system is 20 dB. (2) Find the value of K so that the phase margin of the system is 45°.

(a) $G(s)=\dfrac{K}{s(1+0.1s)(1+0.5s)}$

(b) $G(s)=\dfrac{K(s+1)}{s(1+0.1s)(1+0.2s)(1+0.5s)}$

(c) $G(s)=\dfrac{K}{(s+3)^3}$

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

(e) $G(s)=\dfrac{Ke^{-s}}{s(1+0.1s+0.01s^2)}$

(f) $G(s)=\dfrac{K(1+0.5s)}{s(s^2+s+1)}$

10-40. The forward-path transfer functions of unity-feedback control systems are given in the following equations. Plot $G(j\omega)/K$ in the gain-phase coordinates of the Nichols chart, and do the following: (1) Find the value of K so that the gain margin of the system is 10 dB. (2) Find the value of K so that the phase margin of the system is 45°. (3) Find the value of K so that M_r 1.2.

(a) $G(s)=\dfrac{10K}{s(1+0.1s)(1+0.5s)}$

(b) $G(s)=\dfrac{5K(s+1)}{s(1+0.1s)(1+0.2s)(1+0.5s)}$

(c) $G(s)=\dfrac{10K}{s(1+0.1s+0.01s^2)}$

(d) $G(s)=\dfrac{10Ke^{-s}}{s(1+0.1s+0.01s^2)}$

10-41. The forward-path of a unity-feedback system is given by

$$G(s)H(s) = \frac{K(s+1)(s+2)}{s^2(s+3)(s^2+2s+25)}$$

(a) Plot the Bode diagram.

(b) Plot the root locus.

(c) Find the gain and frequency where instability occurs.

(d) Find the gain at the phase margin of 20°.

(e) Find the gain margin when the phase margin is 20°.

10-42. The Bode diagram of the forward-path transfer function of a unity-feedback control system is obtained experimentally (as shown in Fig. 10P-42 when the forward gain *K* is set at its nominal value.

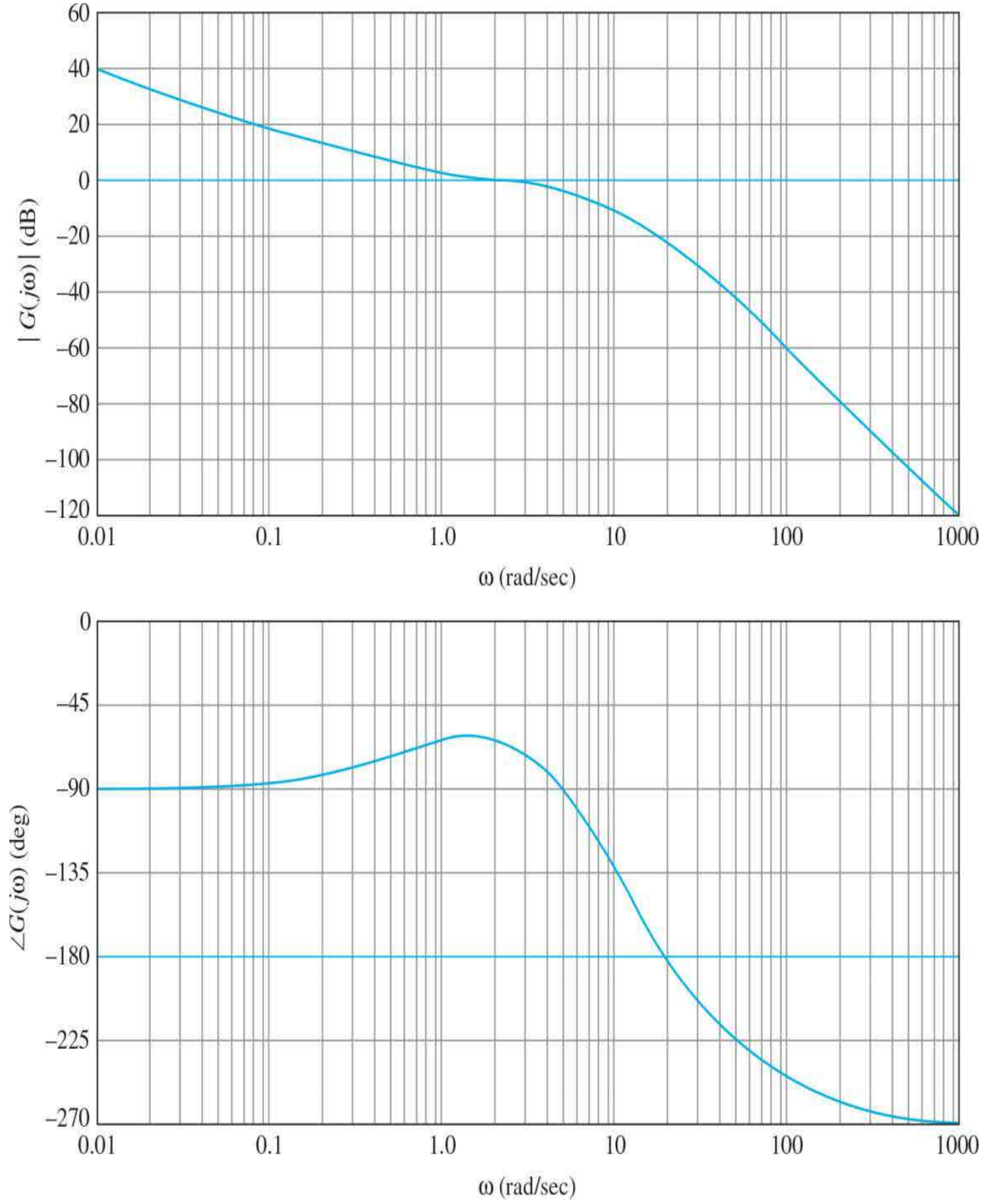


Figure 10P-42

(a) Find the gain and phase margins of the system from the diagram as best you can read them. Find the gain- and phase-crossover frequencies.

(b) Repeat part (a) if the gain is doubled from its nominal value.

(c) Repeat part (a) if the gain is 10 times its nominal value.

(d) Find out how much the gain must be changed from its nominal value if the gain margin is 40 dB.

(e) Find out how much the loop gain must be changed from its nominal value if the phase margin is 45°.

(f) Find the steady-state error of the system if the reference input to the system is a unit-step function.

(g) The forward path now has a pure time delay of T_ds, so that the forward-path transfer function is multiplied by $e^{-T_d s}$. Find the gain margin and the phase margin for $T_d = 0.1$ s. The gain is set at nominal.

(h) With the gain set at nominal, find the maximum time delay T_d the system can tolerate without going into instability.

10-43. Repeat Prob. 10-42 using Fig. 10P-42 for the following parts.

(a) Find the gain and phase margins if the gain is four times its nominal value. Find the gain- and phase-crossover frequencies.

(b) Find out how much the gain must be changed from its nominal value if the gain margin is 20 dB.

(c) Find the marginal value of the forward-path gain for system stability.

(d) Find out how much the gain must be changed from its nominal value if the phase margin is 60°.

(e) Find the steady-state error of the system if the reference input is a unit-step function and the gain is twice its nominal value.

(f) Find the steady-state error of the system if the reference input is a unit-step function and the gain is 20 times its nominal value.

(g) The system now has a pure time delay so that the forward-path transfer function is multiplied by $e^{-T_d s}$. Find the gain and phase margins when $T_d = 0.1$ s. The gain is set at its nominal value.

(h) With the gain set at 10 times its nominal, find the maximum time delay T_d the system can tolerate without going into instability.

10-44. The forward-path transfer function of a unity-feedback control system is

$$G(s)H(s)=\frac{80e^{-0.1s}}{s(s+4)(s+10)}$$

(a) Draw the Nyquist plot of the system.

(b) Plot the Bode diagram of the system.

(c) Find the phase margin and gain margin of the system.

10-45. The forward-path transfer function of a unity-feedback control system is

$$G(s)=\frac{K(1+0.2s)(1+0.1s)}{s^2(1+s)(1+0.01s)^2}$$

(a) Construct the Bode and Nyquist plots of $G(j\omega)/K$ and determine the range of K for system stability.

(b) Construct the root loci of the system for $K \geq 0$. Determine the values of K and ω at the points, where the root loci cross the $j\omega$-axis, using the information found from the Bode plot.

10-46. Repeat Prob. 10-45 for the following transfer function:

$$G(s)=\frac{K(s+1.5)(s+2)}{s^2(s^2+2s+2)}$$

10-47. Repeat Prob. 10-45 for the following transfer function:

$$G(s)H(s)=\frac{16000(s+1)(s+5)}{s(s+0.1)(s+8)(s+20)(s+50)}$$

10-48. The forward-path transfer function of the dc-motor control system described in Fig. 4P-11 is

$$G(s)=\frac{6.087\times 10^8 K}{s(s^3+423.42s^2+2.6667\times 10^6 s+4.2342\times 10^8)}$$

Plot the Bode diagram of $G(j\omega)$ with $K = 1$, and determine the gain margin and phase margin of the system. Find the critical value of K for stability.

10-49. The transfer function between the output position $\Theta_L(s)$ and the motor current $I_a(s)$ of the robot arm modeled in Fig. 6P-20 is

$$G_p(s) = \frac{\Theta_L(s)}{I_a(s)} = \frac{K_i(Bs+K)}{\Delta_o}$$

where

$$\Delta_o(s) = s\{J_L J_m s^3 + [J_L(B_m + B) + J_m(B_L + B)]s^2 + [B_L B_m + (B_L + B_m)B + (J_m + J_L)K]s + K(B_L + B_m)\}$$

The arm is controlled by a closed-loop system. The system parameters are

$$K_a = 65,\ K = 100,\ K_i = 0.4,\ B = 0.2,\ J_m = 0.2,\ B_L = 0.01,\ J_L = 0.6, \text{ and } B_m = 0.25.$$

(a) Derive the forward-path transfer function $G(s) = \Theta_L(s)/E(s)$.

(b) Draw the Bode diagram of $G(j\omega)$. Find the gain and phase margins of the system.

(c) Draw $|M(j\omega)|$ versus ω, where $M(s)$ is the closed-loop transfer function. Find M_r, ω_r, and BW.

10-50. For the ball-and-beam system described in Prob. 4-11 and shown in Fig. 10P-50, assume the following:

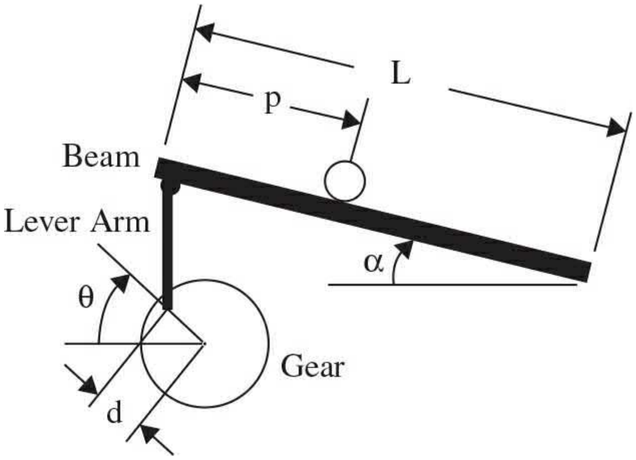
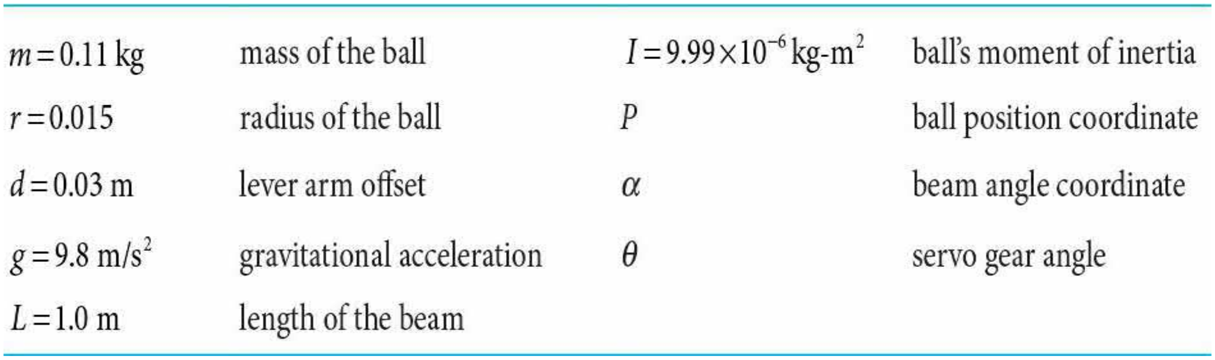


Figure 10P-50



$m = 0.11$ kg	mass of the ball	$I = 9.99 \times 10^{-6}$ kg-m^2	ball's moment of inertia
$r = 0.015$	radius of the ball	P	ball position coordinate
$d = 0.03$ m	lever arm offset	α	beam angle coordinate
$g = 9.8$ m/s^2	gravitational acceleration	θ	servo gear angle
$L = 1.0$ m	length of the beam		

If the system is controlled by a proportional controller in a unity-feedback control system,

(a) Find the transfer function from the gear angle (θ) to the ball position (P).

(b) Find the closed-loop transfer function.

(c) Find the range of K for stability.

(d) Plot the Bode diagram for the system for $K = 1$, and find the gain and phase margins of the system.

(e) Draw $|M(j\omega)|$ versus ω, where $M(s)$ is the closed-loop transfer function. Find M_r, ω_r, and BW.

10-51. The gain-phase plot of the forward-path transfer function of $G(j\omega)/K$ of a unity-feedback control system is shown in Fig. 10P-51. Find the following performance characteristics of the system.

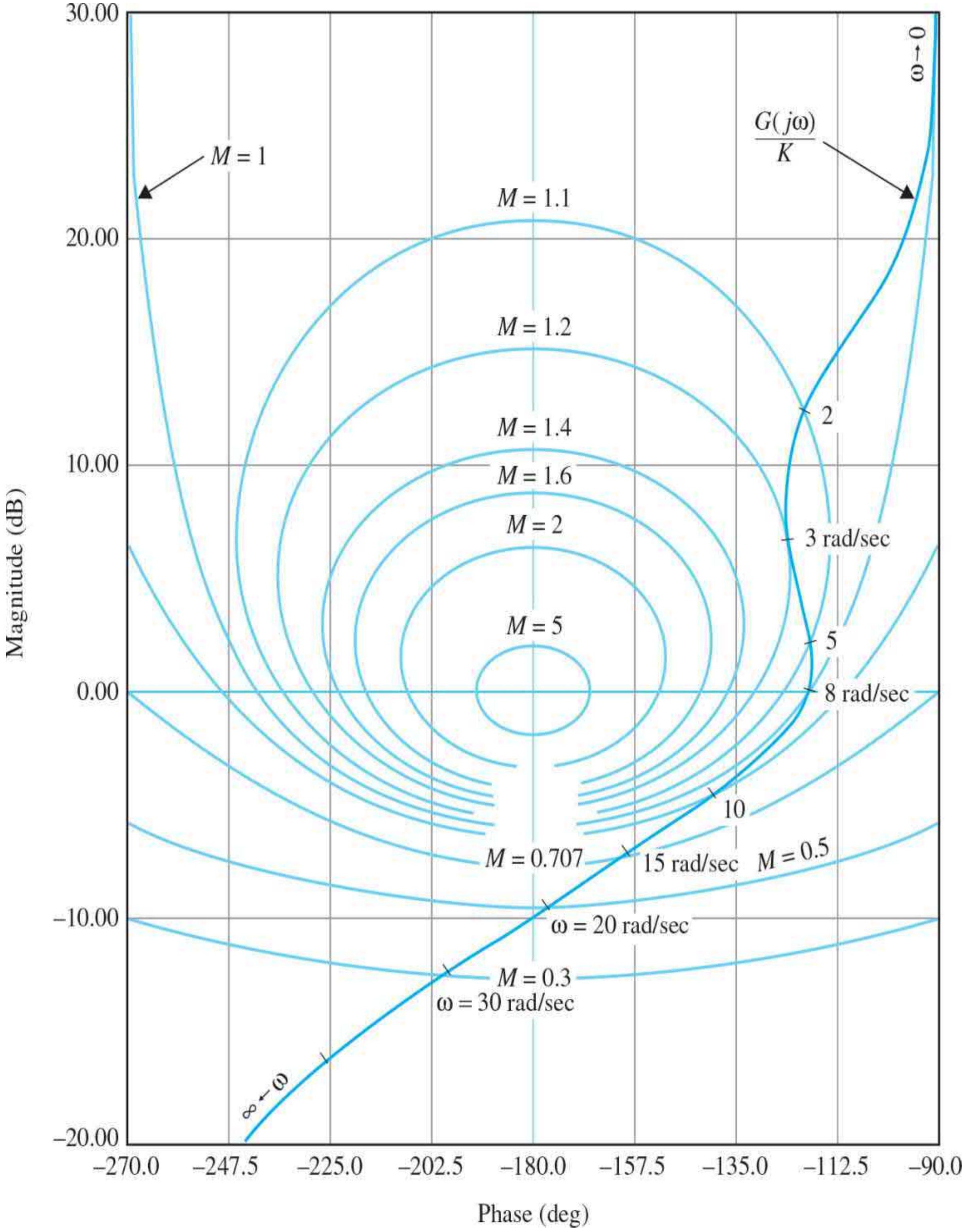


Figure 10P-51

(a) Gain-crossover frequency (rad/s), when $K = 1$.

(b) Phase-crossover frequency (rad/s) when $K = 1$.

(c) Gain margin (dB) when $K = 1$.

(d) Phase margin (deg) when $K = 1$.

(e) Resonance peak Mr when $K = 1$.

(f) Resonant frequency ω_r (rad/s) when $K = 1$.

(g) BW of the closed-loop system when $K = 1$.

(h) The value of K so that the gain margin is 20 dB.

(i) The value of K so that the system is marginally stable. Find the frequency of sustained oscillation in rad/s.

(j) Steady-state error when the reference input is a unit-step function.

10-52. Repeat parts (a) through (g) of Prob. 10-51, when $K = 10$. Repeat part (h) for gain margin = 40 dB.

10-53. For the system in Prob. 10-44, plot the Nichols chart and find magnitudes and phase angles of the closed-loop frequency response. Then plot the Bode diagram of the closed-loop system.

10-54. Use ACSYS or MATLAB to analyze the frequency response of the following unity-feedback control systems. Plot the Bode diagrams, polar plots, and gain-phase plots, and compute the phase margin, gain margin, M_r, and BW.

(a) $$G(s) = \frac{1+0.1s}{s(s+1)(1+0.01s)}$$

(b) $$G(s) = \frac{0.5(s+1)}{s(1+0.2s)(1+s+0.5s^2)}$$

(c) $$G(s) = \frac{(s+1)}{s(1+0.2s)(1+0.5s)}$$

(d) $$G(s) = \frac{1}{s(1+s)(1+0.5s)}$$

(e) $$G(s) = \frac{50}{s(s+1)(1+0.5s^2)}$$

(f) $$G(s) = \frac{(1+0.1s)e^{-0.1s}}{s(s+1)(1+0.01s)}$$

(g) $$G(s) = \frac{10e^{-0.1s}}{s^2+2s+2}$$

10-55. For the gain-phase plot of $G(j\omega)/K$ shown in Fig. 10P-51, the system now has a pure time delay of T_d in the forward path, so that the forward-path transfer function becomes $G(s)e^{-T_d s}$.

(a) With K = 1, find Td so that the phase margin is 40°.

(b) With K = 1, find the maximum value of Td so that the system will remain stable.

10-56. Repeat Prob. 10-55 with K = 10.

10-57. Repeat Prob. 10-55 so that the gain margin is 5 dB when K = 1.

10-58. The block diagram of a furnace-control system is shown in Fig. 10P-58. The transfer function of the process is

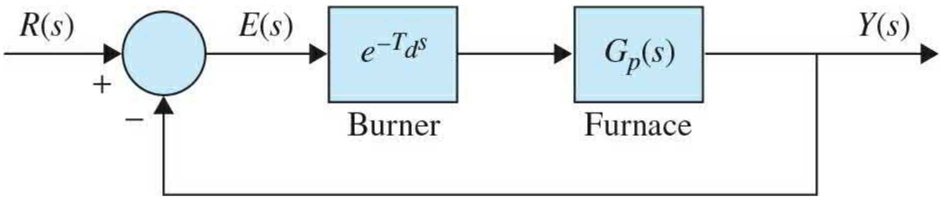


Figure 10P-58

$$G_p(s)=\frac{1}{(1+10s)(1+25s)}$$

The time delay T_d is 2 s.

There are many ways of approximating e^{-T_ds} by a rational function. One way is to approximate the exponential function by a Maclaurin series; that is,

$$e^{-T_ds}1\cong T_ds+\frac{T_d^2s^2}{2}$$

or

$$e^{-T_ds}\cong\frac{1}{1+T_ds+T_d^2s^2/2}$$

where only three terms of the series are used. Apparently, the approximations are not valid when the magnitude of T_{ds} is large.

A better approximation is to use the Pade approximation, which is given in the following for a two-term approximation:

$$e^{-T_ds}\cong\frac{1-T_ds/2}{1+T_ds/2}$$

This approximation of the transfer function contains a zero in the right-half *s*-plane so that the step response of the approximating system may exhibit a small negative undershoot near $t = 0$.

(a) Plot the Bode diagram of $G(s) = Y(s)/E(s)$, and find the gain-crossover and phase-crossover frequencies. Find the gain margin and the phase margin.

(b) Approximate the time delay by

$$e^{-T_d s} \frac{1}{1+T_d s+T_d s^2/2}$$

and repeat part (a). Comment on the accuracy of the approximation. What is the maximum frequency below which the polynomial approximation is accurate?

(c) Repeat part (b) for approximating the time delay term by

$$e^{-T_d s} \cong \frac{1-T_d s/2}{1+T_d s/2}$$

10-59. Repeat Prob. 10-58 with $T_d = 1$ s.

10-60. Plot the $S^M_G(j\omega)$|-versus-ω plot for the system described in Prob. 10-49 for $K = 1$. Find the frequency at which the sensitivity is maximum and the value of the maximum sensitivity.

10-61. Figure 10P-61 shows the pitch controller system for an aircraft, as described in Sec. 7-9.

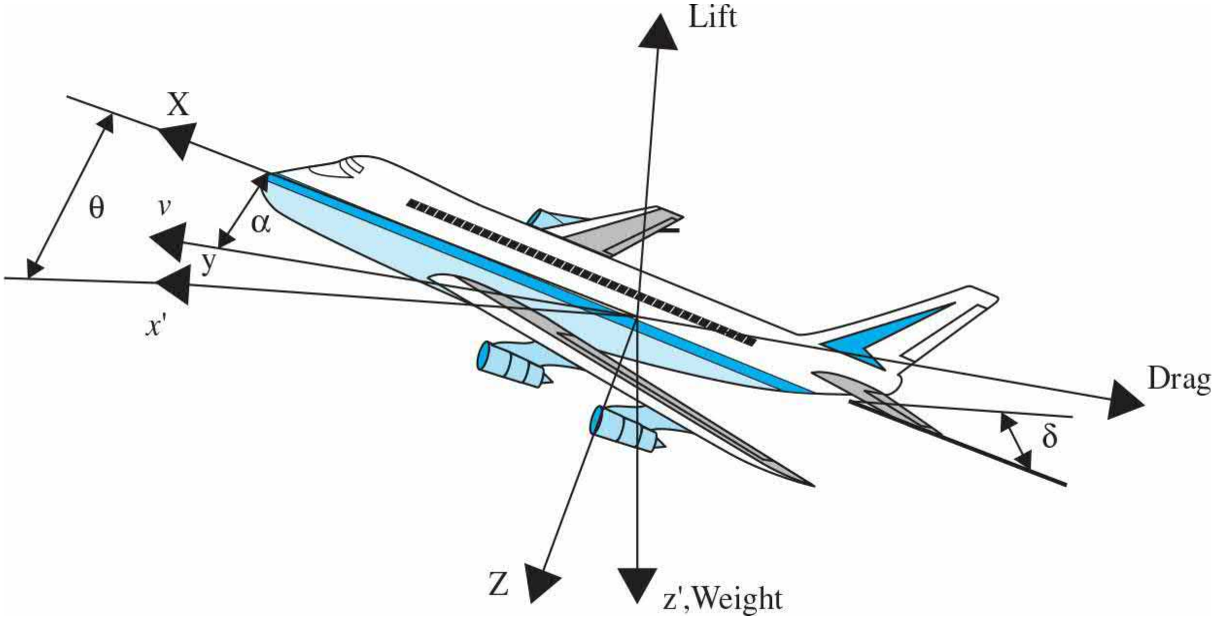


Figure 10P-61

If the system is controlled by a proportional controller in a unity-feedback control system,

(a) Find the transfer function between pitch angle and elevator deflection angle.

(b) Find the closed-loop transfer function.

(c) Find the range of K for stability.

(d) Plot the Bode diagram for the system for $K = 1$, and find the gain and phase margins of the system.

(e) Draw $|M(j\omega)|$ versus ω, where $M(s)$ is the closed-loop transfer function. Find M_r, ω_r, and BW.

[1] In the absence of delay, i.e., $T_d = 0$, Eq. (10-52) takes the form:

CHAPTER 11

Design of Control Systems

11-1 INTRODUCTION

We now utilize all the foundations and analyses that we have provided in the preceding chapters in the ultimate goal of design of control systems. Starting with the controlled process such as that shown by the block diagram in Fig. 11-1, control system design involves the following three steps:

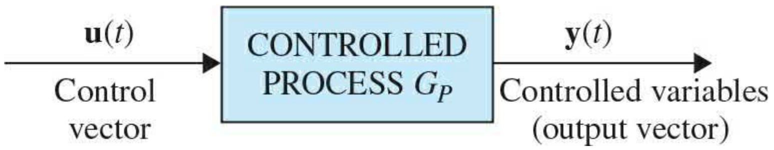


Figure 11-1 Controlled process.

1. Use design specifications to determine what the system should do and how to do.
2. Determine the controller or compensator configuration, relative to how it is connected to the controlled process.
3. Determine the parameter values of the controller to achieve the design goals.

These design tasks are explored further in the following sections.

Learning Outcomes

After successful completion of this chapter, you will be able to

1. Design simple control systems using time-domain and frequency-domain approaches.
2. Incorporate various controllers, including proportional, derivative, integral, lead, and lag, into your control system for simple processes.

3. Use MATLAB to investigate the time- and frequency-domain performance of the control systems.
4. Use the MATLAB SISO design tool to expedite the design process.

11-1-1 Design Specifications

As discussed in Chap. 7, we use design specifications to describe the expected performance of a system for a given input. These specifications are unique to individual applications and often include specifications about **relative stability**, **steady-state accuracy** (**error**), **transient-response characteristics**, and **frequency-response characteristics**. In some applications, there may be additional specifications on **sensitivity to parameter variations**, that is, **robustness**, or **disturbance rejection**.

The design of linear control systems can be carried out in either the time domain or the frequency domain. For instance, **steady-state accuracy** is often specified with respect to a step input, a ramp input, or a parabolic input, and the design to meet a certain requirement is more conveniently carried out in the time domain. Other specifications, such as **maximum overshoot, rise time**, and **settling time,** are all defined for a unit-step input and, therefore, are used specifically for time-domain design. We have learned that relative stability is also measured in terms of **gain margin**, **phase margin**, and $\boldsymbol{M}_r$. These are typical frequency-domain specifications, which should be used in conjunction with such tools as the Bode plot, polar plot, gain-phase plot, and Nichols chart.

We have shown that, for a second-order prototype system, there are simple analytical relationships between some of these time-domain and frequency-domain specifications. However, for higher-order systems, correlations between time-domain and frequency-domain specifications are difficult to establish. As pointed out earlier, the analysis and design of control systems is pretty much an exercise of selecting from several alternative methods for solving the same problem.

Thus, the choice of whether the design should be conducted in the time domain or the frequency domain depends often on the preference of the designer. We should be quick to point out, however, that in most cases, time-domain specifications such as maximum overshoot, rise time, and settling time are usually used as the final measure of system performance. To an inexperienced designer, it is difficult to comprehend the physical connection between frequency-domain specifications such as gain and phase margins and resonance peak to actual system performance. For instance, does a gain margin of 20 dB guarantee a maximum overshoot of less than 10 percent? To a designer it makes more sense to specify, for example, that the maximum overshoot should be less than 5 percent and a settling time less than 0.01 s. It is less obvious what, for example, a phase margin of 60° and an M_r of less than 1.1 may bring in system performance. The following outline will hopefully clarify and explain the choices and reasons for using time-domain versus frequency-domain specifications.

1. Historically, the design of linear control systems was developed with a wealth of graphical tools such as the Bode plot, Nyquist plot, gain-phase plot, and Nichols chart, which are all carried out in the frequency domain. The advantage of these tools is that they can all be sketched by following approximation methods without detailed plotting. Therefore, the designer can carry out designs using frequency-domain specifications such as **gain margin, phase margin**, $\boldsymbol{M_r}$, and the like. High-order systems do not generally pose any particular problem. For certain types of controllers, design procedures in the frequency domain are available to reduce the trial-and-error effort to a minimum.
2. Design in the time domain using such performance specifications as **rise time**, **delay time**, **settling time**, **maximum overshoot**, and the like is possible *analytically* only for second-order systems or for systems that can be approximated by second-order systems. General design procedures using time-domain specifications are difficult to establish for systems with an order higher than the second.

The development and availability of high-powered and user-friendly computer software, such as MATLAB, is rapidly changing the practice of control system design, which until recently had been dictated by historical development. Now with MATLAB, the designer can go through a large number of design runs using the time-domain specifications within a matter of minutes. This diminishes considerably the historical edge of the frequency-domain design, which is based on the convenience of performing graphical design manually.

Throughout this chapter, we have incorporated small MATLAB toolboxes to help your understanding of the examples, and, at the end of the chapter, we introduce the MATLAB SISO design tool that would enhance your ability to design controllers using the root-locus and frequency-domain approaches.

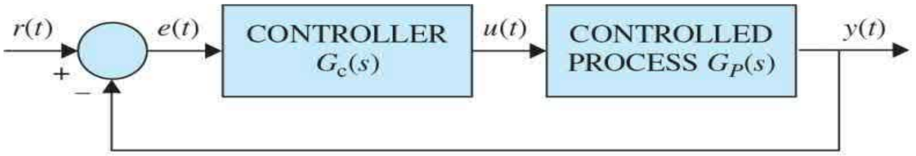
Finally, it is generally difficult (except for an experienced designer) to select a meaningful set of frequency-domain specifications that will correspond to the desired time-domain performance requirements. For example, specifying a phase margin of 60° would be meaningless unless we know that it corresponds to a certain maximum overshoot. As it turns out, to control maximum overshoot, usually one has to specify at least phase margin and M_r. Eventually, establishing an intelligent set of frequency-domain specifications becomes a trial-and-error process that precedes the actual design, which often is also a trial-and-error effort. However, frequency-domain methods are still valuable in interpreting noise rejection and sensitivity properties of the system, and, most important, they offer another perspective to the design process. Therefore, in this chapter the design techniques in the time domain and the frequency domain are treated **side by side**, so that the methods can be easily compared.

11-1-2 Controller Configurations

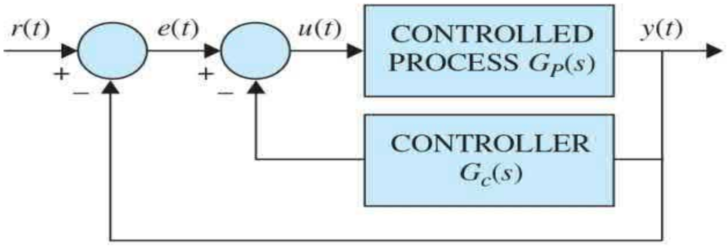
In general, the dynamics of a linear controlled process can be represented by the block diagram shown in Fig. 11-1. The design objective is to have the controlled variables, represented by the output vector $\mathbf{y}(t)$, behave in certain desirable ways. The problem essentially involves the determination of the control signal $\mathbf{u}(t)$ over the prescribed time interval so that the design objectives are all satisfied.

Most of the conventional design methods in control systems rely on the so-called **fixed-configuration design** in that the designer at the outset decides the basic configuration of the overall designed system and decides where the controller is to be positioned relative to the controlled process. The problem then involves the design of the elements of the controller. Because most control efforts involve the modification or compensation of the system-performance characteristics, the general design using fixed configuration is also called **compensation**.

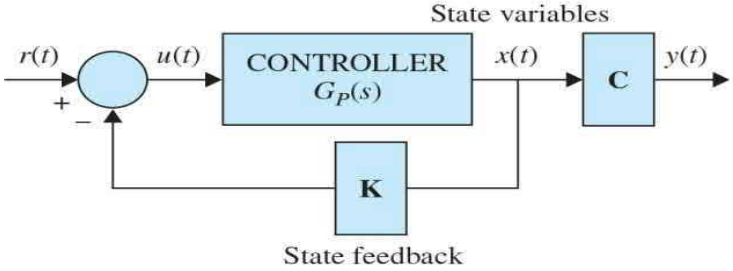
Figure 11-2 illustrates several commonly used system configurations with controller compensation. These are described briefly as follows:


$r(t)$
$+$
$-$
$e(t)$
CONTROLLER $G_c(s)$
$u(t)$
CONTROLLED PROCESS $G_P(s)$
$y(t)$

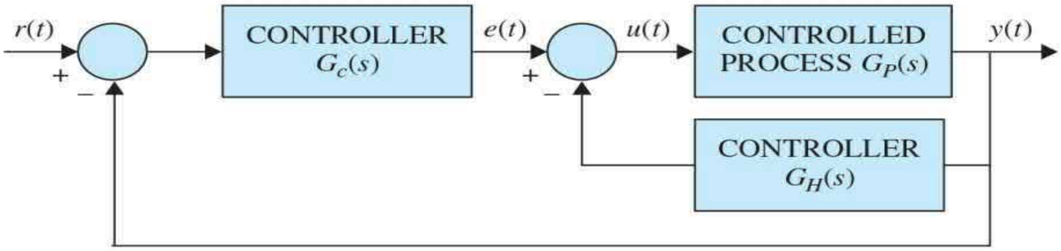
(a)


$r(t)$
$+$
$-$
$e(t)$
$+$
$-$
$u(t)$
CONTROLLED PROCESS $G_P(s)$
$y(t)$
CONTROLLER $G_c(s)$

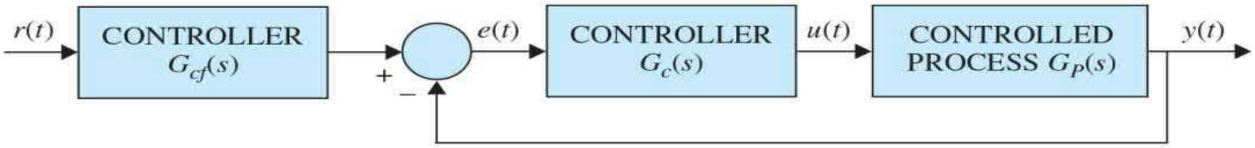
(b)


State variables
$r(t)$
$+$
$-$
$u(t)$
CONTROLLER $G_P(s)$
$x(t)$
C
$y(t)$
K
State feedback

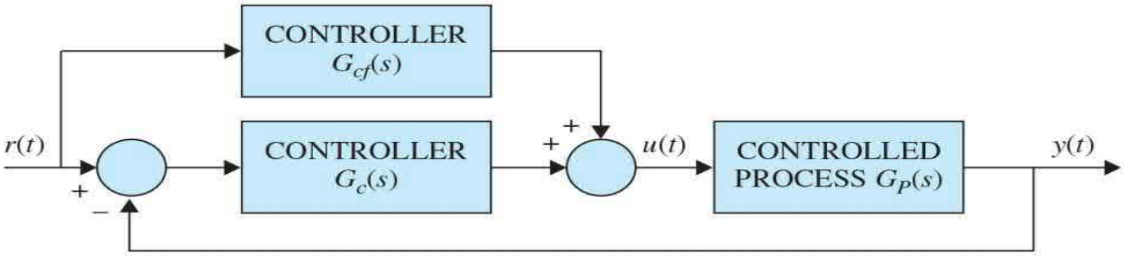
(c)


$r(t)$
$+$
$-$
CONTROLLER $G_c(s)$
$e(t)$
$+$
$-$
$u(t)$
CONTROLLED PROCESS $G_P(s)$
$y(t)$
CONTROLLER $G_H(s)$

(d)


$r(t)$
CONTROLLER $G_{cf}(s)$
$+$
$-$
$e(t)$
CONTROLLER $G_c(s)$
$u(t)$
CONTROLLED PROCESS $G_P(s)$
$y(t)$

(e)


CONTROLLER $G_{cf}(s)$
$r(t)$
$+$
$-$
CONTROLLER $G_c(s)$
$+$
$+$
$u(t)$
CONTROLLED PROCESS $G_P(s)$
$y(t)$

(f)

Figure 11-2 Various controller configurations in control-system compensation. (*a*) Series or cascade compensation. (*b*) Feedback compensation. (*c*) State-feedback control. (*d*) Series-feedback compensation (two degrees of freedom). (*e*) Forward compensation with series compensation (two degrees of freedom). (*f*) Feedforward compensation (two degrees of freedom).

- ***Series (cascade) compensation.*** Figure 11-2*a* shows the most commonly used system configuration with the controller placed in series with the controlled process, and the configuration is referred to as **series** or **cascade compensation**.
- ***Feedback compensation.*** In Fig. 11-2*b*, the controller is placed in the minor feedback path, and the scheme is called **feedback compensation**.
- ***State-feedback compensation.*** Figure 11-2*c* shows a system that generates the control signal by feeding back the state variables through constant real gains, and the scheme is known as **state feedback**. The problem with state-feedback control is that, for high-order systems, the large number of state variables involved would require a large number of transducers to sense the state variables for feedback. Thus, the actual implementation of the state-feedback control scheme may be costly or impractical. Even for low-order systems, often not all the state variables are directly accessible, and an **observer** or **estimator** may be necessary to create the estimated state variables from measurements of the output variables.
- ***Series-feedback compensation.*** Figure 11-2*d* shows the series-feedback compensation for which a series controller and a feedback controller are used.
- ***Feedforward compensation.*** Figure 11-2*e* and *f* shows the so-called **feedforward compensation**. In Fig. 11-2*e*, the feedforward controller $G_{cf}(s)$ is placed in series with the closed-loop system, which has a controller $G_c(s)$ in the forward path.

In Fig. 11-2*f*, the feedforward controller $G_{cf}(s)$ is placed in parallel with the forward path. The key to the feedforward compensation is that the controller $G_{cf}(s)$ is not in the loop of the system, so it does not affect the roots of the characteristic equation of the original system. The poles and zeros of $G_{cf}(s)$ may be selected to add or cancel the poles and zeros of the closed-loop system transfer function.

The compensation schemes shown in Fig. 11-2*a* to *c* all have one degree of freedom in that there is only one controller in each system, even though the controller may have more than one parameter that can be varied. The disadvantage with a one-degree-of-freedom controller is that the performance criteria that can be realized are limited. For example, if a system is to be designed to achieve a certain amount of relative stability, it may have poor sensitivity to parameter variations. Or if the roots of the characteristic equation are selected to provide a certain amount of relative damping, the maximum overshoot of the step response may still be excessive because of the zeros of the closed-loop transfer function. The compensation schemes shown in Fig. 11-2*d* to *f* all have two degrees of freedom.

One of the commonly used controllers in the compensation schemes just described is a PID controller, which applies a signal to the process that is proportional to the actuating signal in addition to adding integral and derivative of the actuating signal. Because these signal components are easily realized and visualized in the time domain, PID controllers are commonly designed using time-domain methods. In addition to the PID-type controllers, lead, lag, lead-lag, and notch controllers are also frequently used. The names of these controllers come from properties of their respective frequency-domain characteristics. As a result, these controllers are often designed using frequency-domain concepts. Despite these design tendencies, however, all control system designs will benefit by viewing the resulting design from both time- and frequency-domain viewpoints. Thus, both methods will be used extensively in this chapter.

PID controllers are the most common controllers used in industrial applications.

It should be pointed out that these compensation schemes are by no means exhaustive. The details of these compensation schemes will be discussed in later sections of this chapter. Although the systems illustrated in Fig. 11-2 are all for continuous-data control, the same configurations can be applied to discrete-data control, in which case the controllers are all digital, with the necessary interfacings and signal converters.

11-1-3 Fundamental Principles of Design

After a controller configuration is chosen, the designer must choose a controller type that, with proper selection of its element values, will satisfy all the design specifications. The types of controllers available for control-system design are bounded only by one's imagination. Engineering practice usually dictates that one choose the simplest controller that meets all the design specifications. In most cases, the more complex a controller is, the more it costs, the less reliable it is, and the more difficult it is to design. Choosing a specific controller for a specific application is often based on the designer's past experience and sometimes intuition, and it entails as much *art* as it does *science*. As a novice, you may initially find it difficult to make intelligent choices of controllers with confidence. By understanding that confidence comes only through experience, this chapter provides guided experiences that illustrate the basic elements of control system designs.

After a controller is chosen, the next task is to choose controller parameter values. These parameter values are typically the coefficients of one or more transfer functions making up the controller. The basic design approach is to use the analysis tools discussed in the previous chapters to determine how individual parameter values influence the design specifications and, finally, system performance. Based on this information, controller parameters are selected so that all design specifications are met. While this process is sometimes straightforward, more often than not it involves many design iterations since controller parameters usually interact with each other and influence design specifications in conflicting ways. For example, a particular parameter value may be chosen so that the maximum overshoot is satisfied, but in the process of varying another parameter value in an attempt to meet the rise-time requirement, the maximum overshoot specification may no longer be met! Clearly, the more design specifications there are and the more controller parameters there are, the more complicated the design process becomes.

In carrying out the design either in the time domain or the frequency domain, it is important to establish some basic guidelines or design rules. Keep in mind that time-domain design usually relies heavily on the *s*-plane and the root loci. Frequency-domain design is based on manipulating the gain and phase of the loop transfer function so that the specifications are met.

In general, it is useful to summarize the time-domain and frequency-domain characteristics so that they can be used as guidelines for design purposes.

1. Complex-conjugate poles of the closed-loop transfer function lead to a step response that is underdamped. If all system poles are real, the step response is overdamped. However, zeros of the closed-loop transfer function may cause overshoot even if the system is overdamped.
2. The response of a system is dominated by those poles closest to the origin in the *s*-plane. Transients due to those poles farther to the left decay faster.
3. The farther to the left in the *s*-plane the system's dominant poles are, the faster the system will respond and the greater its bandwidth will be.
4. The farther to the left in the *s*-plane the system's dominant poles are, the more expensive it will be and the larger its internal signals will be. While this can be justified analytically, it is obvious that striking a nail harder with a hammer drives the nail in faster but requires more energy per strike. Similarly, a sports car can accelerate faster, but it uses more fuel than an average car.
5. When a pole and zero of a system transfer function nearly cancel each other, the portion of the system response associated with the pole will have a small magnitude.
6. Time-domain and frequency-domain specifications are loosely associated with each other. Rise time and bandwidth are inversely proportional. Larger phase margin, larger gain margin, and lower M_r will improve damping.

11-2 DESIGN WITH THE PD CONTROLLER

In most examples of control systems we have discussed thus far, the controller has been typically a simple amplifier with a constant gain *K*. This type of control action is formally known as **proportional control** because the control signal at the output of the controller is simply related to the input of the controller by a proportional constant.

Intuitively, one should also be able to use the derivative or integral of the input signal, in addition to the proportional operation. Therefore, we can consider a more general continuous-data controller to be one that contains such components as adders (addition or subtraction), amplifiers, attenuators, differentiators, and integrators. The designer’s task is to determine which of these components should be used, in what proportion, and how they are connected. For example, one of the best-known controllers used in practice is the PID controller, where the letters stand for **proportional**, **integral**, and **derivative**. The integral and derivative components of the PID controller have individual performance implications, and their applications require an understanding of the basics of these elements. To gain an understanding of this controller, we consider just the PD portion of the controller first.

Figure 11-3 shows the block diagram of a feedback control system that arbitrarily has a second-order prototype process with the transfer function

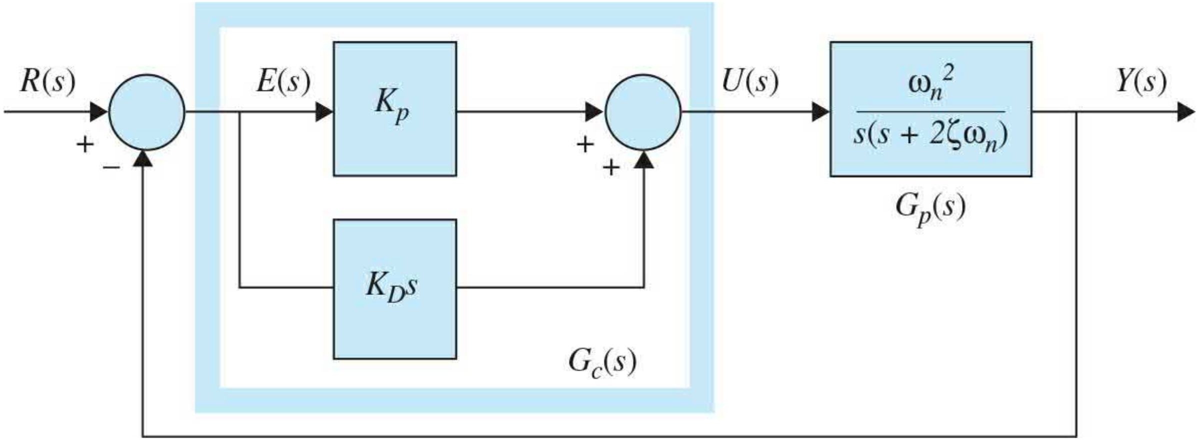


Figure 11-3 Control system with PD controller.

$$G_p(s) = \frac{\omega_n^2}{s(s+2\zeta\omega_n)} \tag{11-1}$$

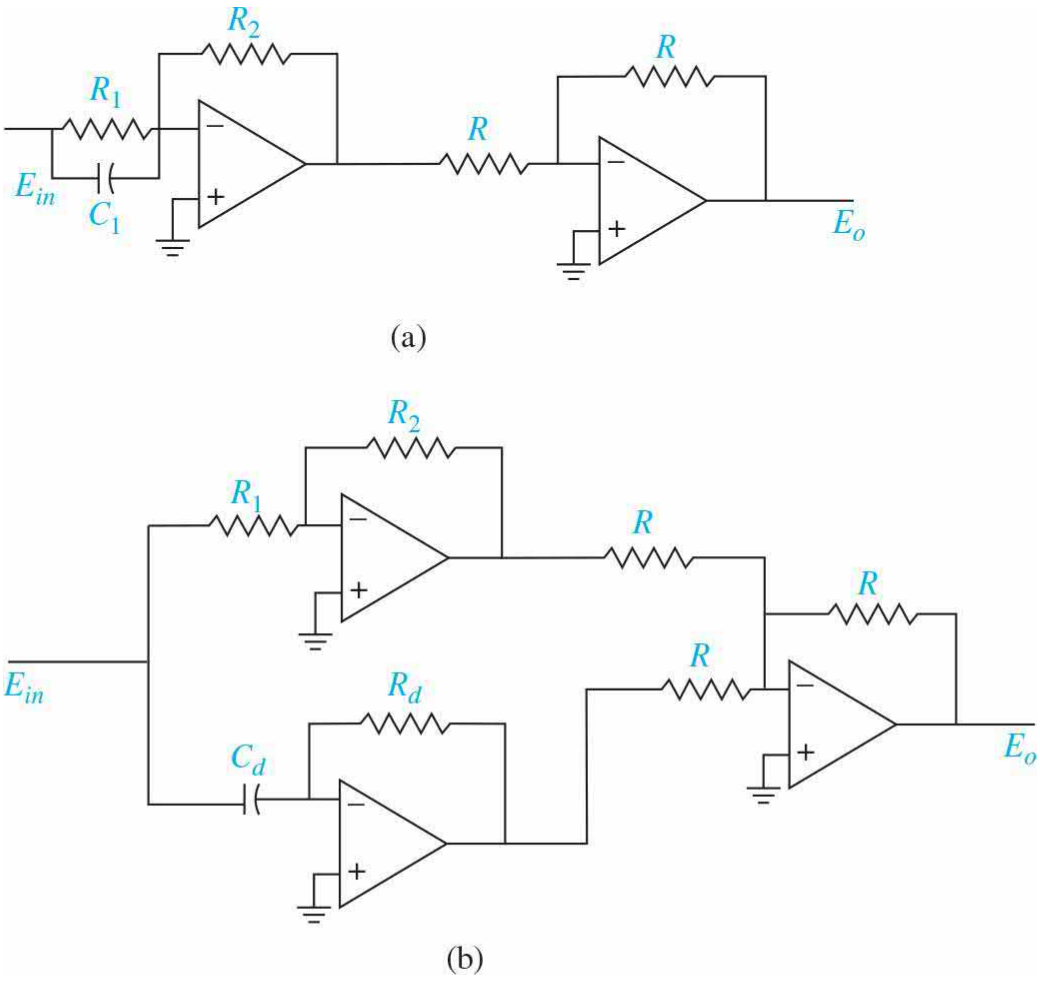
The series controller is a proportional-derivative (PD) type with the transfer function

$$G_c(s) = K_P + K_D s \tag{11-2}$$

Thus, the control signal applied to the process is

$$u(t) = K_P e(t) + K_D \frac{de(t)}{dt} \tag{11-3}$$

where K_P and K_D are the proportional and derivative constants, respectively. Using the components given in Table 6-1, two electronic-circuit realizations of the PD controller are shown in Fig. 11-4. The transfer function of the circuit in Fig. 11-4*a* is



(a)

(b)

Figure 11-4 Op-amp circuit realization of the PD controller.

$$\frac{E_o(s)}{E_{\text{in}}(s)} = \frac{R_2}{R_1} + R_2 C_1 s \tag{11-4}$$

Comparing Eq. (11-2) with Eq. (11-4), we have

$$K_P = R_2/R_1 \quad K_D = R_2 C_1 \tag{11-5}$$

The transfer function of the circuit in Fig. 11-4*b* is

$$\frac{E_o(s)}{E_{\text{in}}(s)} = \frac{R_2}{R_1} + R_d C_d s \qquad (11\text{-}6)$$

Comparing Eq. (11-2) with Eq. (11-6), we have

$$K_P = R_2/R_1 \quad K_D = R_d C_d \qquad (11\text{-}7)$$

The advantage with the circuit in Fig. 11-4*a* is that only two op-amps are used. However, the circuit does not allow the independent selection of K_P and K_D because they are commonly dependent on R_2. An important concern of the PD controller is that, if the value of K_D is large, a large capacitor C_1 would be required. The circuit in Fig. 11-4*b* allows K_P and K_D to be independently controlled. A large K_D can be compensated by choosing a large value for R_d, thus resulting in a realistic value for C_d. Although the scope of this text does not include all the practical issues involved in controller transfer function implementation, these issues are of the utmost importance in practice.

The forward-path transfer function of the compensated system is

$$G(s) = \frac{Y(s)}{E(s)} = G_c(s)G_p(s) = \frac{\omega_n^2(K_P + K_D s)}{s(s + 2\zeta\omega_n)} \qquad (11\text{-}8)$$

which shows that the PD control is equivalent to adding a simple zero at s = $-K_p/K_D$ to the forward-path transfer function.

11-2-1 Time-Domain Interpretation of PD Control

> PD control adds a simple zero at s = $-K_p/K_D$ to the forward-path transfer function.

The effect of the PD control on the transient response of a control system can be investigated by referring to the time responses shown in Fig. 11-5. Let us assume that the unit-step response of a stable system with only proportional control is as shown in Fig. 11-5*a*, which has a relatively high maximum overshoot and is rather oscillatory. The corresponding error signal, which is the difference between the unit-step input and the output $y(t)$ and its time derivative $de(t)/dt$ are shown in Figs. 9.5*b* and *c*, respectively. The overshoot and oscillation characteristics are also reflected in $e(t)$ and $de(t)/dt$. For the sake of illustration, we assume that the system contains a motor of some kind with its torque proportional to $e(t)$. The performance of the system with proportional control is analyzed as follows:

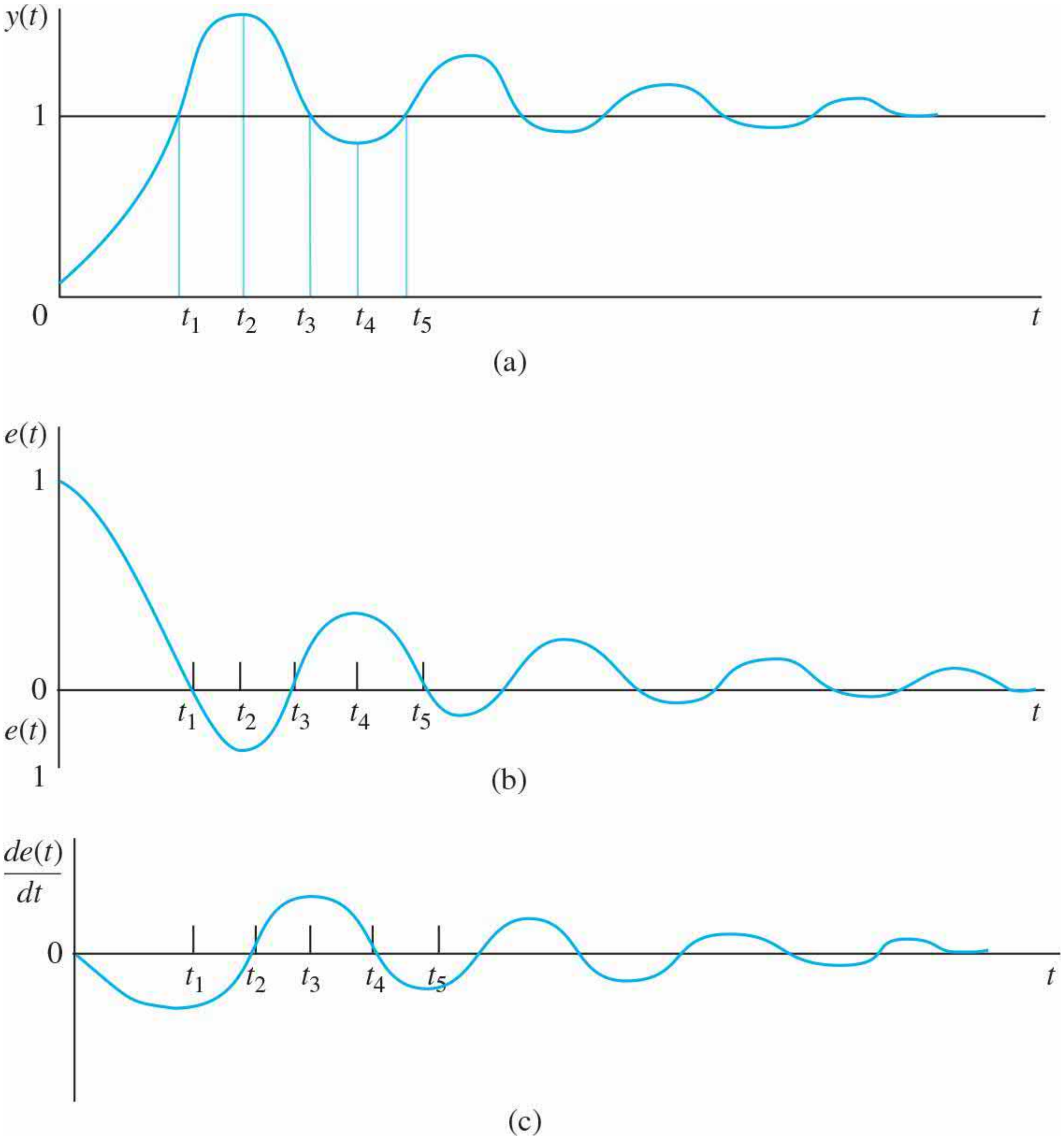


Figure 11-5 Waveforms of $y(t)$, $e(t)$, and $de(t)/dt$, showing the effect of derivative control. (*a*) Unit-step response. (*b*) Error signal. (*c*) Time rate of

change of the error signal.

1. During the time interval $0<t<t_1$: The error signal $e(t)$ is positive. The motor torque is positive and rising rapidly. The large overshoot and subsequent oscillations in the output $y(t)$ are due to the excessive amount of torque developed by the motor and the lack of damping during this time interval.
2. During the time interval $t_1<t<t_3$: The error signal $e(t)$ is negative, and the corresponding motor torque is negative. This negative torque tends to slow down the output acceleration and eventually causes the direction of the output $y(t)$ to reverse and undershoot.
3. During the time interval $t_3<t<t_5$: The motor torque is again positive, thus tending to reduce the undershoot in the response caused by the negative torque in the previous time interval. Because the system is assumed to be stable, the error amplitude is reduced with each oscillation, and the output eventually settles to its final value.

Considering the above analysis of the system time response, we can say that the contributing factors to the high overshoot are as follows:

1. The positive correcting torque in the interval $0<t<t_1$ is too large.
2. The retarding torque in the time interval $t_1<t<t_2$ is inadequate.

Therefore, to reduce the overshoot in the step response, without significantly increasing the rise time, a logical approach would be to

1. Decrease the amount of positive correcting torque during $0<t<t_1$.
2. Increase the retarding torque during $t_1<t<t_2$.

Similarly, during the time interval, $t_2<t<t_4$, the negative corrective torque in $t_2<t<t_3$ should be reduced, and the retarding torque during $t_3<t<t_4$, which is now in the positive direction, should be increased to improve the undershoot of $y(t)$.

The PD control described by Eq. (11-2) gives precisely the compensation effect required. Because the control signal of the PD control is given by Eq. (11-3), Fig. 11-5*c* shows the following effects provided by the PD controller:

1. For $0<t<t_1$, $de(t)/dt$ is negative; this will reduce the original torque

developed due to $e(t)$ alone.

2. For $t_1<t<t_2$, both $e(t)$ and $de(t)/dt$ are negative, which means that the negative retarding torque developed will be greater than that with only proportional control.
3. For $t_2<t<t_3$, $e(t)$ and $de(t)/dt$ have opposite signs. Thus, the negative torque that originally contributes to the undershoot is reduced also.

Therefore, all these effects will result in smaller overshoots and undershoots in $y(t)$.

PD is essentially an anticipatory control.

Another way of looking at the derivative control is that since $de(t)/dt$ represents the slope of $e(t)$, the PD control is essentially an *anticipatory* control. That is, by knowing the slope, the controller can anticipate direction of the error and use it to better control the process. Normally, in linear systems, if the slope of $e(t)$ or $y(t)$ due to a step input is large, a high overshoot will subsequently occur. The derivative control measures the instantaneous slope of $e(t)$, predicts the large overshoot ahead of time, and makes a proper corrective effort before the excessive overshoot actually occurs.

Intuitively, derivative control affects the steady-state error of a system only if the steady-state error varies with time. If the steady-state error of a system is constant with respect to time, the time derivative of this error is zero, and the derivative portion of the controller provides no input to the process. But if the steady-state error increases with time, a torque is again developed in proportion to $de(t)/dt$, which reduces the magnitude of the error. Equation (11-8) also clearly shows that the PD control does not alter the system type that governs the steady-state error of a unity-feedback system.

Derivative or PD control will have an effect on a steady-state error only if the error varies with time.

11-2-2 Frequency-Domain Interpretation of PD Control

For frequency-domain design, the transfer function of the PD controller is written as

$$G_c(s) = K_P + K_D s = K_P\left(1 + \frac{K_D}{K_P}s\right) \tag{11-9}$$

so that it is more easily interpreted on the Bode plot. The Bode plot of Eq. (11-9) is shown in Fig. 11-6 with K_p=1. In general, the proportional-control gain K_P can be combined with a series gain of the system, so that the zero-frequency gain of the PD controller can be regarded as unity. The high-pass filter characteristics of the PD controller are clearly shown by the Bode plot in Fig. 11-6. The phase-lead property may be utilized to improve the phase margin of a control system. Unfortunately, the magnitude characteristics of the PD controller push the gain-crossover frequency to a higher value. Thus, *the design principle of the PD controller involves the placing of the corner frequency of the controller,* $\omega = K_p/K_D$, *such that an effective improvement of the phase margin is realized at the new gain-crossover frequency*. For a given system, there is a range of values of K_P/K_D that is optimal for improving the damping of the system. Another practical consideration in selecting the values of K_P and K_D is in the physical implementation of the PD controller. Other apparent effects of the PD control in the frequency domain are that, due to its high-pass characteristics, in most cases it will increase the BW of the system and reduce the rise time of the step response. The practical disadvantage of the PD controller is that the differentiator portion is a high-pass filter, which usually accentuates any high-frequency noise that enters at the input.

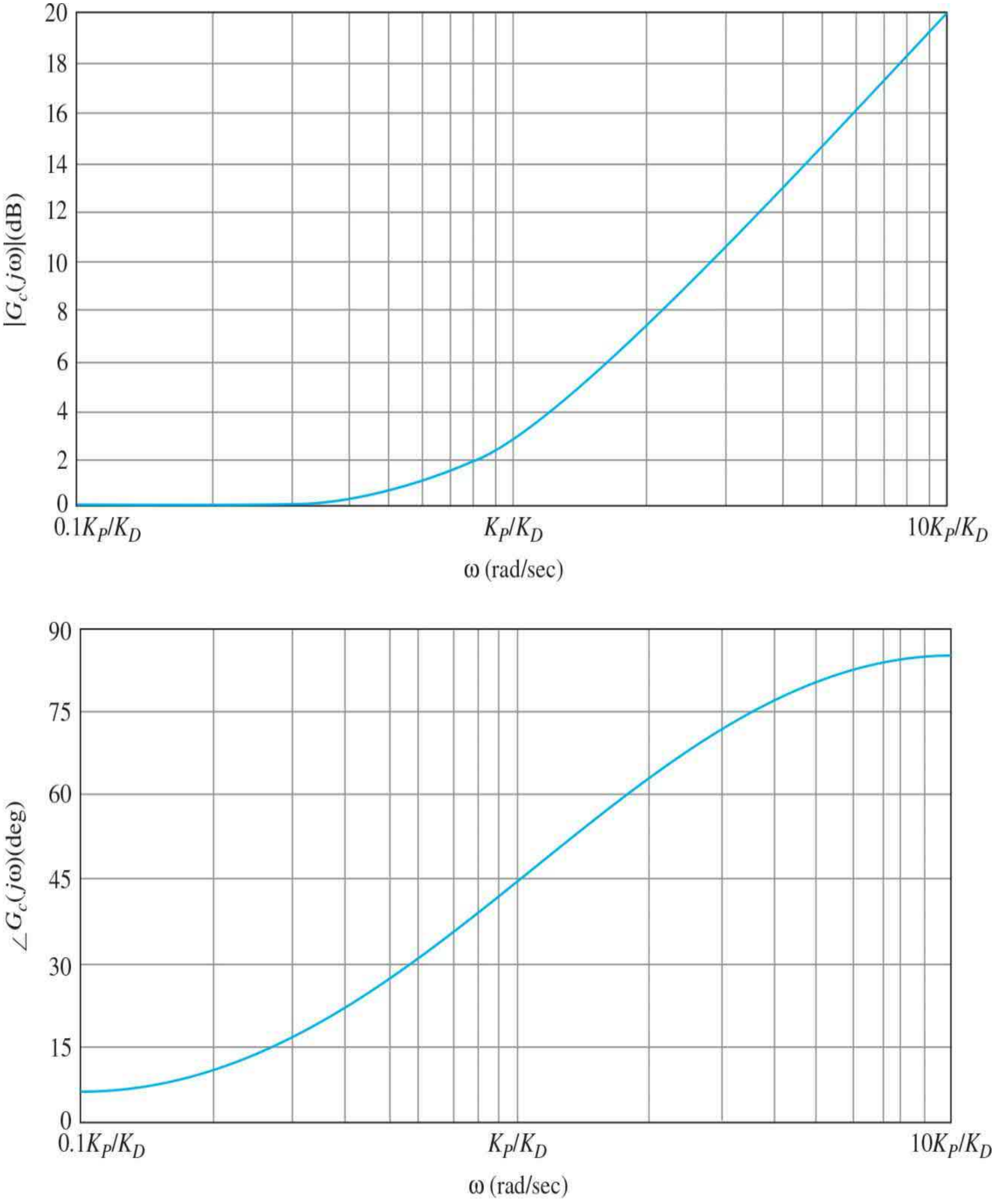


Figure 11-6 Bode diagram of $1+\dfrac{K_D s}{K_P},\ K_P = 1.$

The PD controller is a high-pass filter.

The PD controller has the disadvantage that it accentuates high-frequency noise.

The PD controller will generally increase the BW and reduce the rise time of the step response.

11-2-3 Summary of Effects of PD Control

Though it is not effective with lightly damped or initially unstable systems, a properly designed PD controller can affect the performance of a control system in the following ways:

1. Improving damping and reducing maximum overshoot.
2. Reducing rise time and settling time.
3. Increasing BW.
4. Improving GM, PM, and M_r.
5. Possibly accentuating noise at higher frequencies.
6. Possibly requiring a relatively large capacitor in circuit implementation.

The following example illustrates the effects of the PD controller on the time-domain and frequency-domain responses of a second-order system.

EXAMPLE 11-2-1[1] Let us reconsider the second-order model of the aircraft attitude control system shown in Fig. 7-52. The forward-path transfer function of the system is given in Eq. (7-161) and is repeated here:

Control of a DC Motor: Small Time-Constant Model

$$G(s) = \frac{4500K}{s(s+361.2)} \tag{11-10}$$

Let us set the performance specifications as follows:

Steady-state error due to unit-ramp input ≤ 0.000443

Maximum overshoot ≤5%

Rise time t_r≤0.005 s

2% settling time t_s≤0.005 s

To satisfy the maximum value of the specified steady-state error requirement, *K* should be set at 181.17. However, with this value of *K*, the damping ratio of the system is 0.2, and the maximum overshoot is 52.7 percent, as shown by the unit-step response in Fig. 7-54. Let us consider using a PD controller in the forward path of the system so that the damping and the maximum overshoot of the system are improved while maintaining the steady-state error due to the unit-ramp input at 0.000443.

Time-Domain Design With the PD controller of Eq. (11-9) and *K* = 181.17, the forward-path transfer function of the system becomes

$$G(s)=\frac{\Theta_y(s)}{\Theta_e(s)}=\frac{815{,}265(K_P+K_D s)}{s(s+361.2)} \tag{11-11}$$

The closed-loop transfer function is

$$\frac{\Theta_y(s)}{\Theta_r(s)}=\frac{815{,}265K_D\left(s+\dfrac{K_P}{K_D}\right)}{s^2+(361.2+815{,}265K_D)s+815{,}265K_P} \tag{11-12}$$

Equation (11-12) shows that the effects of the PD controller are as follows:

1. Adding a zero at s = $-K_p/K_D$ to the closed-loop transfer function
2. Increasing the *damping term*, which is the coefficient of the *s* term in the denominator, from 361.2 to 361.2+815,265K_D
3. No impact on steady-state response

From Eq. (11-12), the following observations are made:

The steady-state error due to a unit-step input is e_{ss} = 0.

The ramp-error constant is

$$K_v=\lim_{s\to 0} sG(s)=\frac{815{,}265K_P}{361.2}=2257.1K_P \tag{11-13}$$

The steady-state error due to a unit-ramp input is $e_{ss} = 1/K_v = 0.000443/K_p$. .

Also, from Eq. (11-12), the characteristic equation is written as

$$s^2 + (361.2 + 815{,}265K_D)s + 815{,}265K_P = 0 \tag{11-14}$$

which clearly shows the positive effect of K_D on damping. We should quickly point out that Eq. (11-11) no longer represents a prototype second-order system, since the transient response is also affected by the zero of the transfer function at s = $-K_p/K_D$. In order to design a PD controller, we first approximate the system as a prototype second-order system. That is, from the discussions surrounding Fig. 11-5, and earlier in Sec. 7-7-5, if we select a small K_D relative to K_p, the controller zero at s = $-K_p/K_D$ can be assumed to have a small effect on the time response, in comparison to the dominant poles of the system. Hence upon neglecting the zero, if we use a prototype second-order transfer function, from Eq. (11-14)

$$\text{maximum overshoot} = 0.05 = e^{-\pi\zeta/\sqrt{1-\zeta^2}} \tag{11-15}$$

provides the desired damping ratio for a 5 percent overshoot. Hence, $\zeta = 0.69$. Using the 2-percent settling time formula, for a 0.005 s settling time,

$$\text{2-percent settling time: } t_s = 0.005 \cong \frac{4.0}{\zeta\omega_n} \quad 0 < \zeta < 0.9 \tag{11-16}$$

the desired value of natural frequency is $\omega_n = 1{,}159.2$ rad/s. As a result,

$$K_P = \frac{(1159.2)^2}{815{,}265} = 1.648231 \tag{11-17}$$

$$\zeta = \frac{361.2 + 815{,}265K_D}{(2)(1159.2)} = 0.156 + 351.6K_D \tag{11-18}$$

or

$$K_D = 0.001519 \tag{11-19}$$

Note that from Eq. (11-13), the value of K_P in Eq. (11-17) automatically satisfies the steady-state error due to unit-ramp input ≤0.000443. With these values, the poles and the zero of the system are at

$$
\begin{aligned}
s_{1,2} &= -\frac{(361.2+815,265K_D)}{2} \\
&\quad \pm j\sqrt{\frac{(361.2+815,265K_D)^2}{4} - 815,265(K_P)} \\
&= -800 \pm j838.9
\end{aligned}
\tag{11-20}
$$

Or,

$$
s = -K_P/K_D = -1085 \tag{11-21}
$$

Recall from Eq. (8-19) in Chap. 8 that for the desired poles of the closed loop system to be on the root locus, the poles must meet the angle criterion. In our case, as shown in Fig. 11-7,

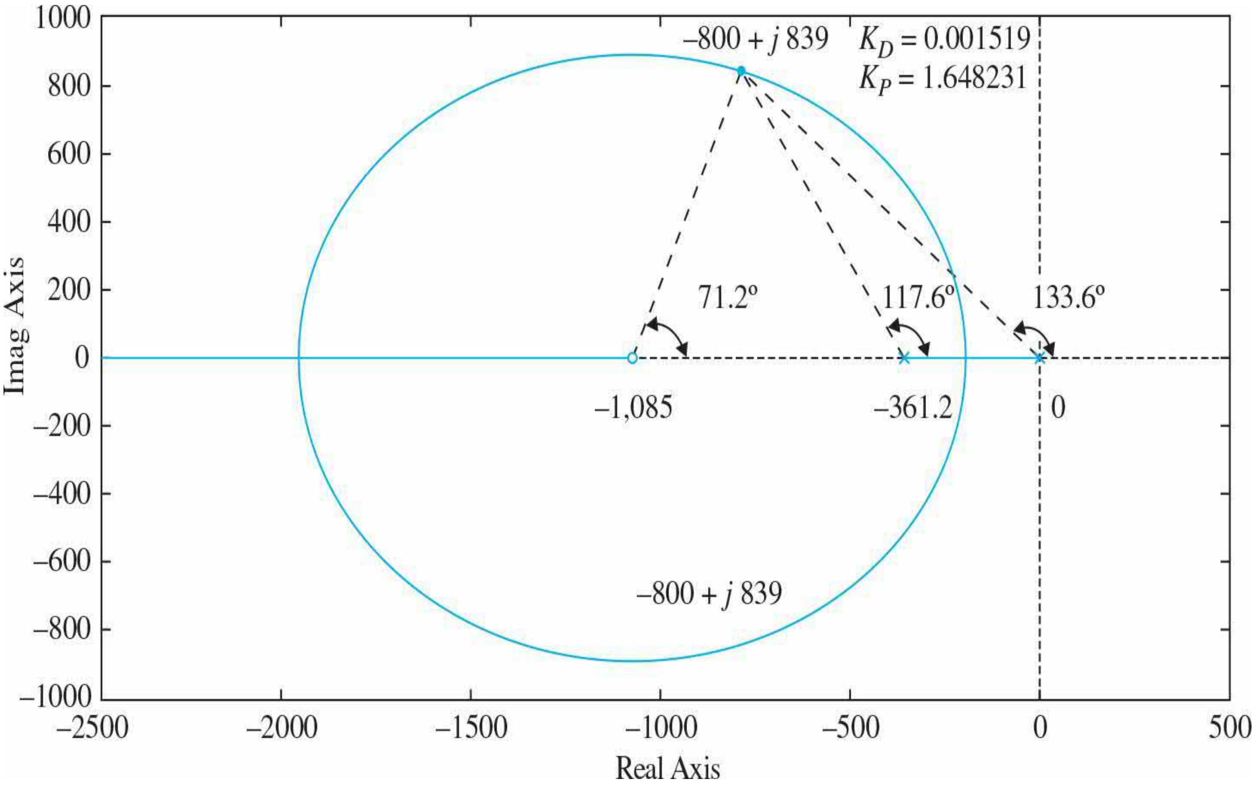


Figure 11-7 Root locus of Eq. (11-12) for the controller zero fixed at $s = -K_P/K_D = -1085$.

$$
\angle(s+361.2)+\angle s-\angle(s+1,085)=117.6^\circ+133.6^\circ-71.2^\circ=180^\circ \tag{11-22}
$$

the closed loop poles and zero from will obviously meet the root-locus angle criterion.

Using Toolbox 11-2-1, we obtain the time response for a unit-step input based on the PD controller parameter values from Eqs. (11-17) and (11-19). The system response, shown in Fig. 11-8, meets the rise time and settling time criteria while the maximum overshoot is well above the desired 5 percent. This is because of the influence of the controller zero, see Sec. 7-8. In order to arrive at the desired response, we must move the poles of the system, along the root locus, to a new location while exploring the time response behavior. The easiest strategy is to substitute the fixed zero value s = -K_p/K_D =-1085 into Eq. (11-14) and solve for the closed loop poles as K_D increases. Fixing the zero of the controller has the advantage of reducing the number of unknown controller parameters from two to one. As a result the revised characteristic equation of the system is

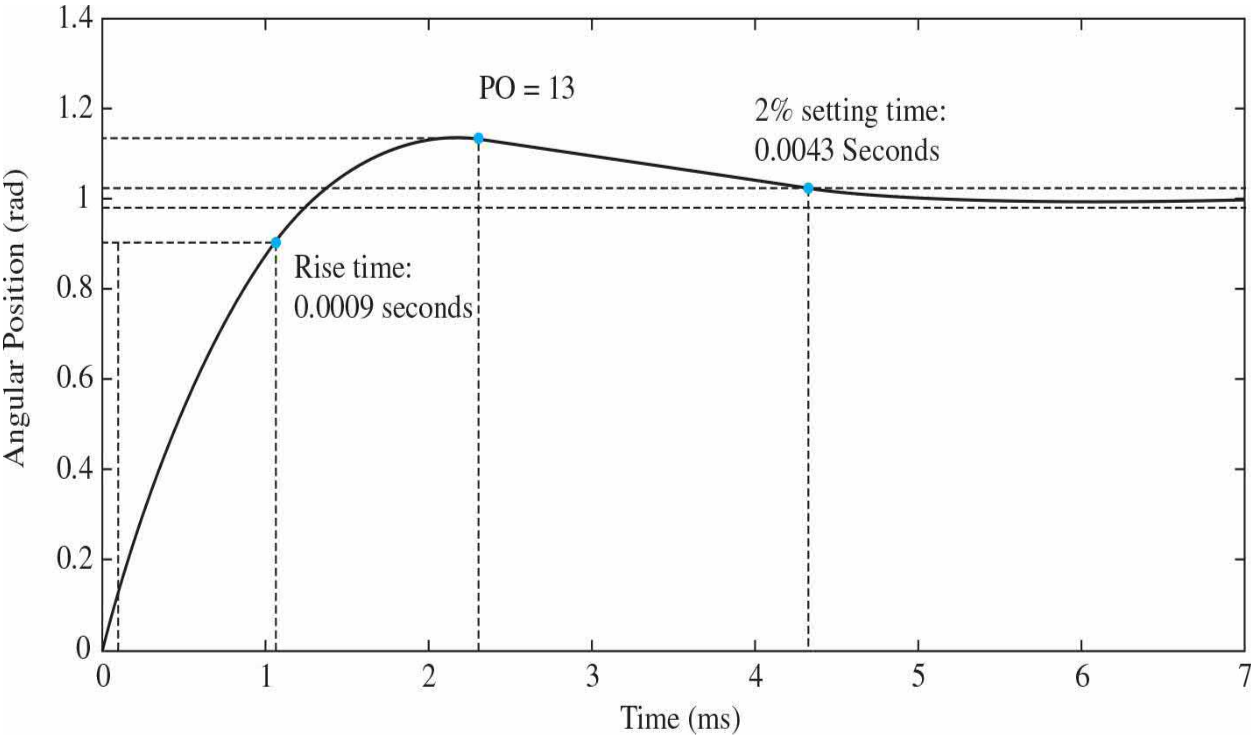


Figure 11-8 Unit-step response of Eq. (11-12) for the controller zero fixed at s = -K_p/K_D = -1085 and poles at $s_{1,2}$ =-800;±j838.9.

Always check whether the dc motor can provide the required torque to achieve the desired response. You must operate the motor below its stall torque limits.

$$s^2+(361.2+815,265K_D)s+815,265(1085K_D)=0 \tag{11-23}$$

Solving for the poles of the system in Eq. (11-14), we have

$$s_{1,2}=-\frac{(361.2+815,265K_D)}{2} \pm j\sqrt{\frac{(361.2+815,265K_D)^2}{4}-815,265(1085K_D)} \tag{11-24}$$

The root-locus diagram in Fig. 11-9 is obtained by varying K_D in Eq. (11-24). The desired response of the system is attained for K_D = 0.01105 and K_P = 13.1285, as shown in Fig. 11-10. From the root locus in Fig. 11-9, the poles in Eq. (11-24) are located at s_1 = -1200 and s_2 = -8170. Notice, the two poles are expected to exhibit a highly overdamped response. However, the controller zero dominance contributes to a nonoscillatory overshoot. The attributes of the unit-step responses of the system are shown in Table 11-1.

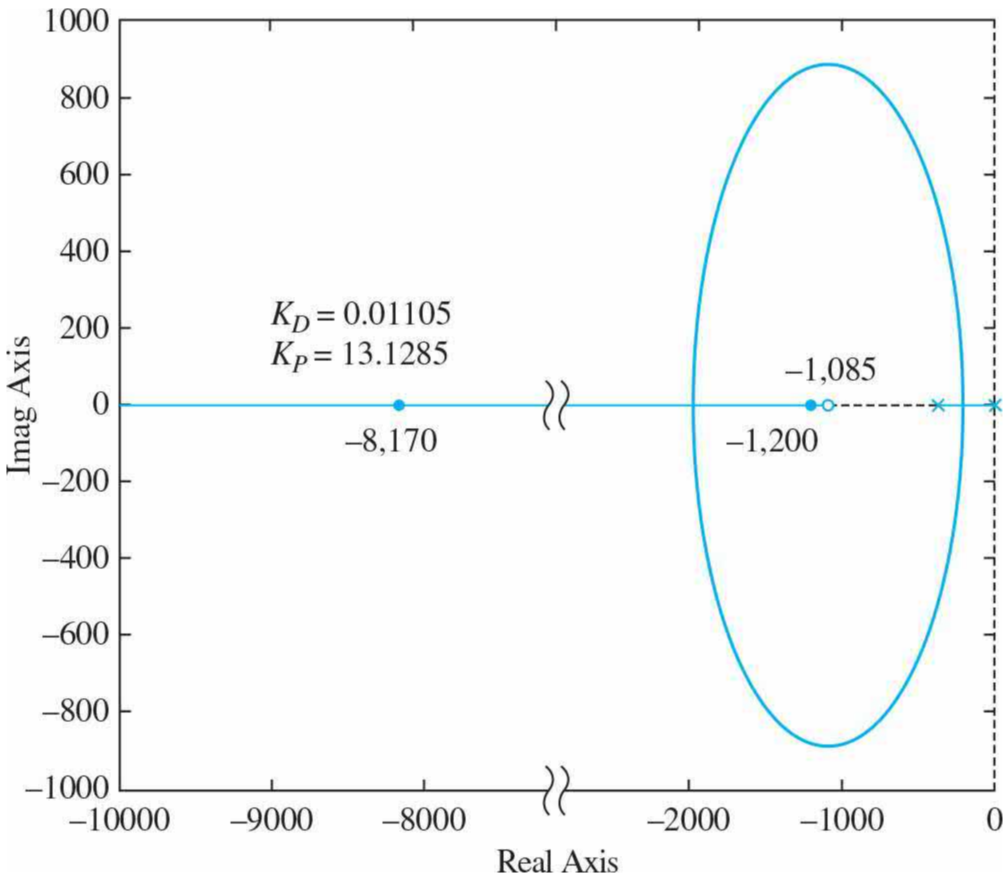


Figure 11-9 Root locus of Eq. (11-12) for the controller zero fixed at $s = -K_P/K_D = -1085$. showing desired response poles for $K_D = 0.01105$ and $K_P = 13.1285$.

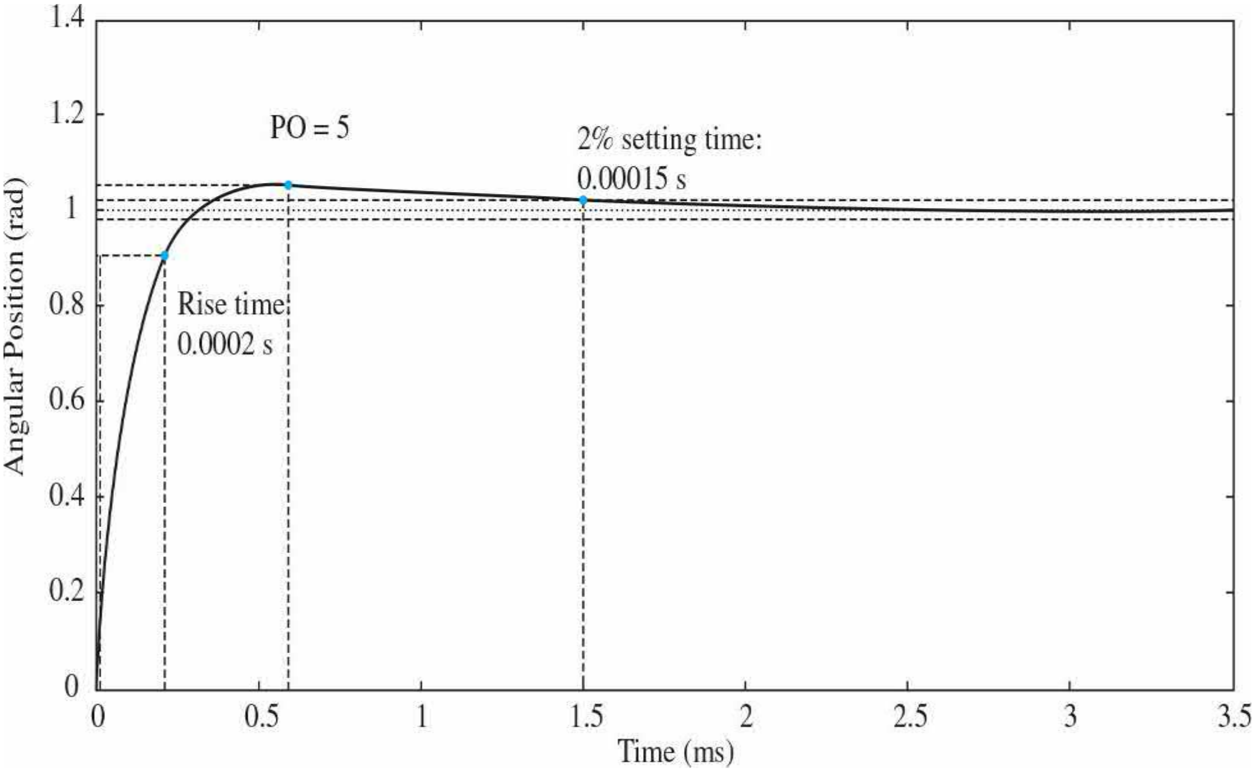


Figure 11-10 Desired unit-step response of Eq. (11-12) for $K_D = 0.01105$ and $K_P = 13.1285$.

Table 11-1 Attributes of the Unit-Step Responses of the System in Example 11-2-1 with PD Controller Using a Fixed Zero at $s = -K_P/K_D = -1085$

K_D	t_r (s)	t_s (s)	Maximum Overshoot (%)
0	0.00125	0.0151	52.2
0.000152	0.0009	0.0043	13
0.01105	0.0002	0.00015	5

In practice, you must always check the required motor torque to achieve this response. You can do this by finding the motor torque transfer function and find its time response; see also the discussions in App. D regarding actuator current saturation. Remember, if the required torque is higher than the motor stall torque, your motor will not be able to provide the desired response. If your controller parameters require a higher torque than what the motor can provide, consider moving the controller zero to the right (as much as you can) and repeat the process to find the desired response. For the zero at *s*= - K_P/K_D = -565, a desired response may be obtained when K_P = 1 and K_D = 0.00177; see the alternative design approach using the root contours approach for the step response, and check Example 11-10-1 for the MATLAB SISO design approach.

Toolbox 11-2-1

Root loci of Eq. (11-11) shown in Fig. 11-7 are obtained by the following sequence of MATLAB functions:

```
num = [815265 815265*1085];
den = [1 361.2 0];
rlocus(num,den)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 11-8 is obtained by the following sequence of MATLAB functions:

```
KD=0.001519;KP=1.648231;
num =[815265*KD 815265*KP];
den = [1 361.2+815265*KD 815265*KP];
step(num,den)
```

Figure 11-10 is obtained by the following sequence of MATLAB functions:

```
KD=0.0121;KP=13.1285;
KP/KD % zero location
num =[815265*KD 815265*KP];
den = [1 361.2+815265*KD 815265*KP];
step(num,den)
```

Alternative Time-Domain Design Approach Using Root Contours

Considering the characteristic equation in Eq. (11-14), we can set K_P=1, which is acceptable from the steady-state error requirement. The damping ratio of the system is

$$\zeta = \frac{361.2 + 815{,}265K_D}{1805.84} = 0.2 + 451.46K_D \tag{11-25}$$

It turns out that, for this second-order system, as the value of K_D increases, the zero will move very close to the origin and effectively cancels the pole of $G(s)$ at s = 0. Thus, as K_D increases, the transfer function in Eq. (11-11) approaches that of a first-order system with the pole at s = -361.2, and the closed-loop system will not have any overshoot. In general, for higher-order systems, however, the zero at s = $-K_p/K_D$ may increase the overshoot when K_D becomes very large.

We can apply the root-contour method to the characteristic equation in Eq. (11-14) to examine the effect of varying K_P and K_D. First, by setting K_D to zero, Eq. (11-14) becomes

$$s^2 + 361.2s + 815{,}265K_P = 0 \tag{11-26}$$

The root loci of the last equation as K_P varies between 0 and ∞ are shown in Fig. 11-11. You can use Toolbox 11-2-2 to draw the root locus. According to the discussions in Chap. 9, when $K_D \neq 0$, the characteristic equation in Eq. (11-14) is conditioned as:

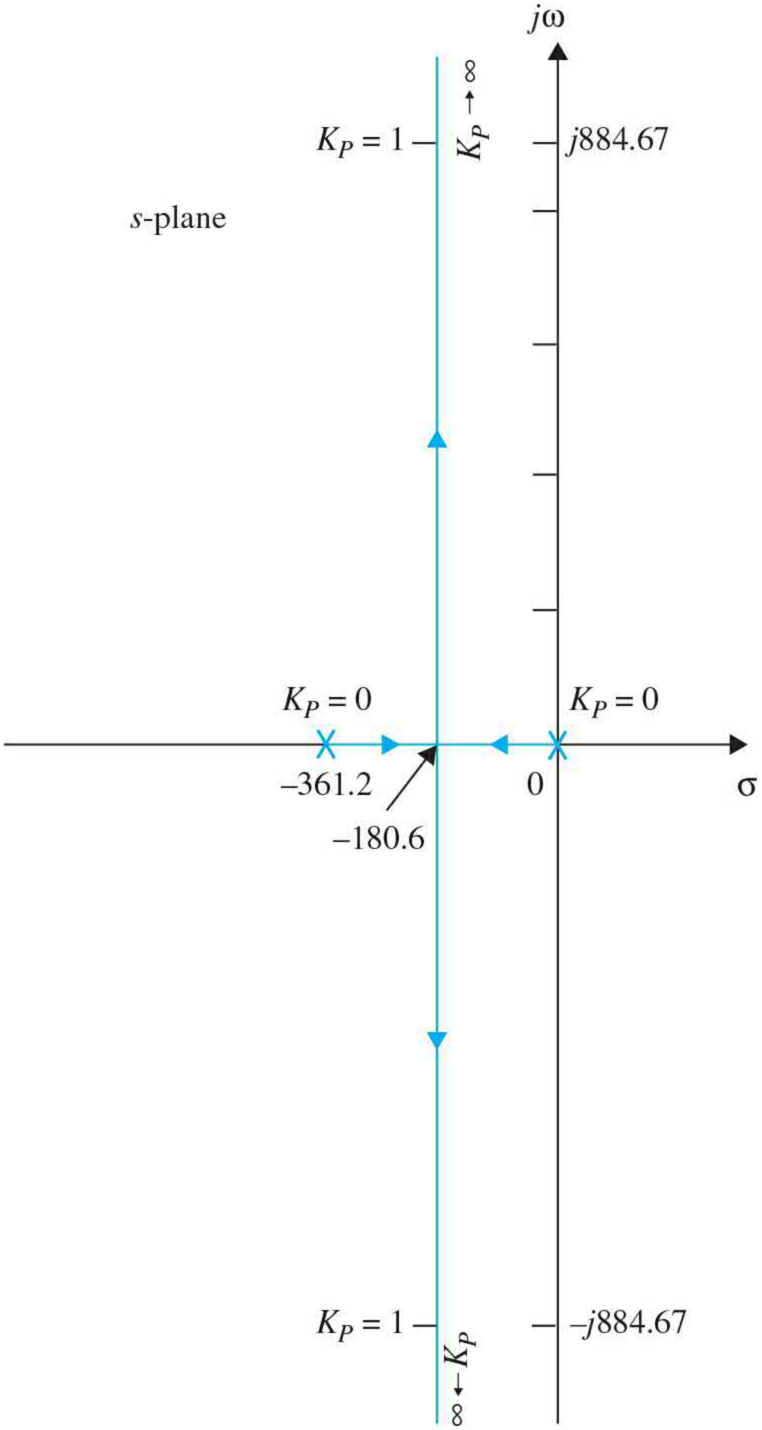


Figure 11-11 Root loci of Eq. (11-26).

$$1+G_{\text{eq}}(s)=1+\frac{815{,}265K_D s}{s^2+361.2s+815{,}265K_P}=0 \tag{11-27}$$

The root contours of Eq. (11-14) with K_P = and K_D varying are constructed based on the pole-zero configuration of $G_{eq}(s)$. Alternatively, the root contours may also be constructed by solving for the pole values of characteristic equation of the system shown in Eq. (11-20). Using either approach, the root contours are plotted by fixing the K_P value and varying K_D, as shown in Fig. 11-12 for K_P is 0.25 and $K_P = 1$.

When K_P is 0.25 and $K_D = 0$, the two characteristic equation roots are at –180.6 + j413.76 and –180.6 – j413.76. As K_D increases in value, the root contours show the improved damping due to the PD controller. Note that this value of K_P is not acceptable as far as the steady-state requirements are concerned.

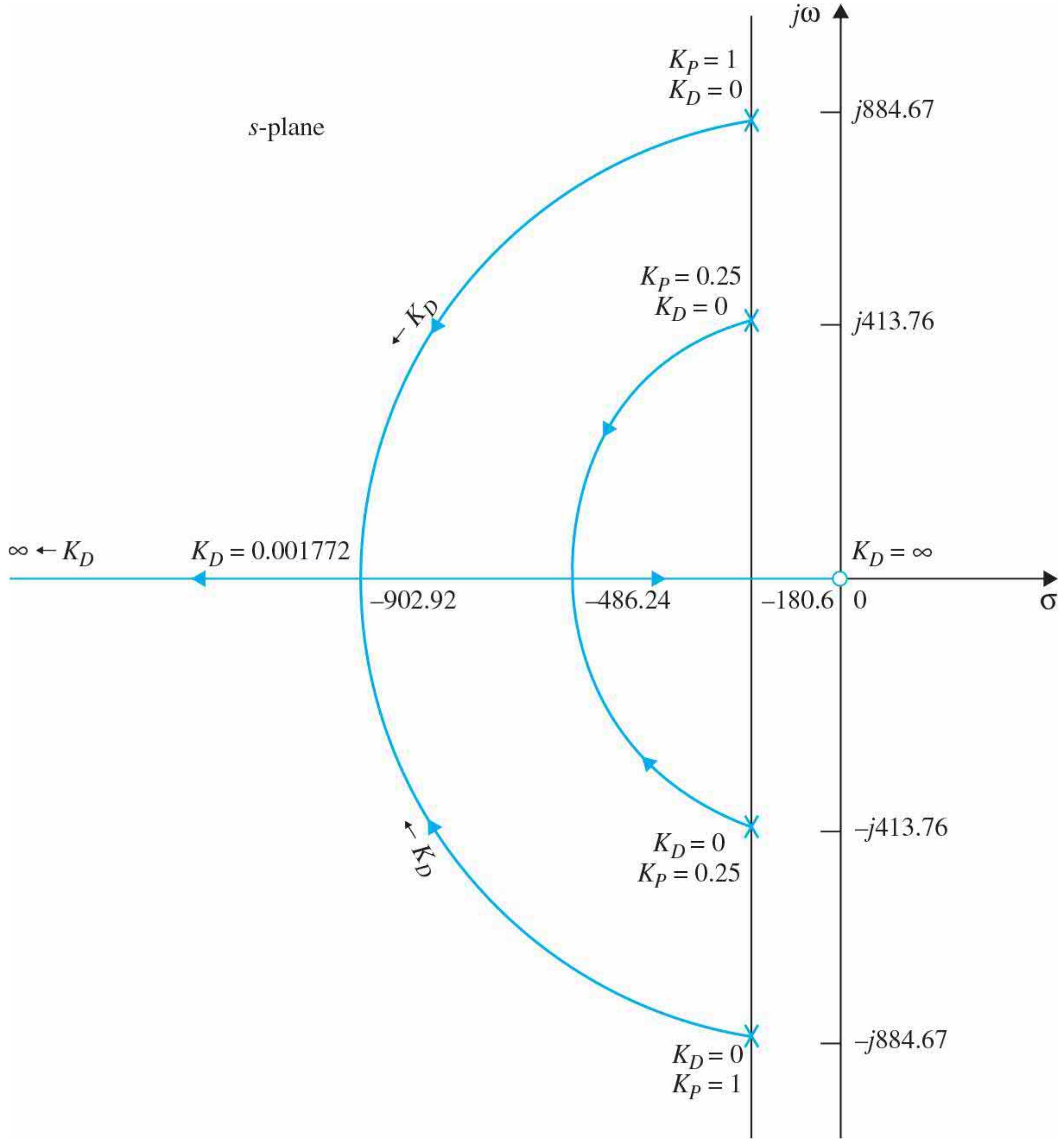


Figure 11-12 Root loci of Eq. (11-14) when K_P= 0.25 and 1.0; K_D varies.

We see that, when K_P = 1 and K_D = 0, the characteristic equation roots are at –180.6 + j884.67 and –180.6 – j884.67, and the damping ratio of the closed-loop system is 0.2. When the value of K_D is increased, the two characteristic equation roots move toward the real axis along a circular arc. When K_D is increased to 0.00177, the roots are real and equal at -902.92, and the damping is critical. When K_D is increased beyond 0.00177, the two roots become real and unequal at -900.065 and -905.78, and the system is overdamped, but expect the effect of controller zero to cause a nonoscillatory response with an overshoot. Figure 11-13 shows the unit-step responses of the closed-loop system without PD control and with $K_P = 1$ and K_D. With the PD control, the maximum overshoot is 4.2 percent. In the present case, although K_D is chosen for critical damping, the overshoot is due to the dominant zero at s = $-K_p/K_D$ = -565 of the closed-loop transfer function.

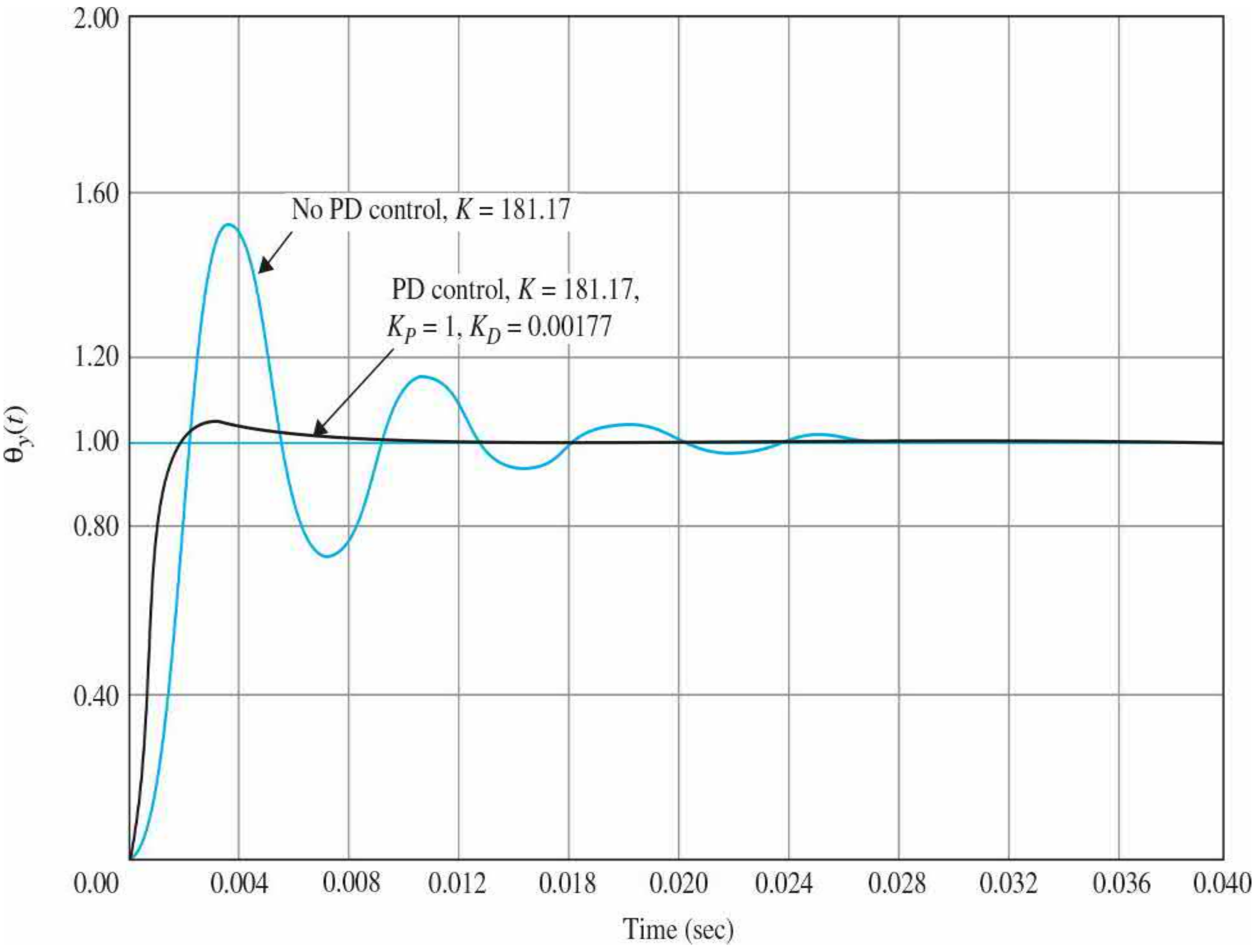
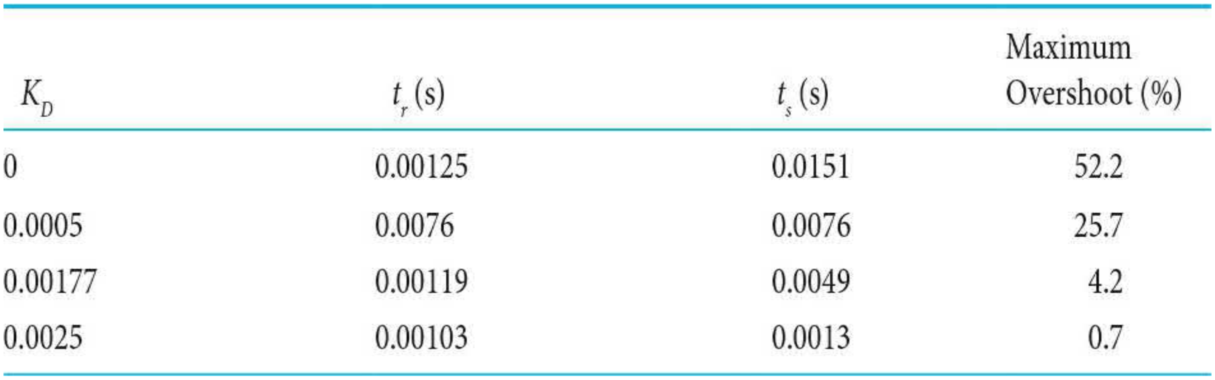


Figure 11-13 Unit-step response of the attitude control system in 7-52 with and without PD controller.

Table 11-2 gives the results on maximum overshoot, rise time, and settling time for $K_P = 1$ and $K_D = 0$, 0.0005, 0.00177, and 0.0025. The results in Table 11-2 show that the performance requirements are all satisfied with $K_D \geq 0.00177$. The performance specifications here are closer to desired values and should place a smaller burden on the motor than in the previous design approach. Please keep in mind that K_D should only be large enough to satisfy the performance requirements. Large K_D corresponds to large BW, which may cause high-frequency noise problems, and there is also the concern of the capacitor value in the op-amp-circuit implementation. Finally, as discussed earlier, do not forget to check if your motor can provide the required torque for the desired response.

Table 11-2 Attributes of the Unit-Step Responses of the System in Example 11-2-1 with PD Controller Using the Root-Contour Approach



K_D	t_r (s)	t_s (s)	Maximum Overshoot (%)
0	0.00125	0.0151	52.2
0.0005	0.0076	0.0076	25.7
0.00177	0.00119	0.0049	4.2
0.0025	0.00103	0.0013	0.7

The general conclusion is that the PD controller decreases the maximum overshoot, the rise time, and the settling time.

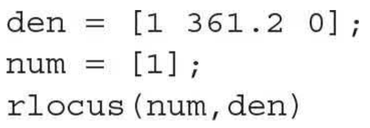
Another analytic way of studying the effects of the parameters K_P and K_D is to evaluate the performance characteristics in the parameter plane of K_P and K_D. From the characteristic equation of Eq. (11-14), we have

$$\zeta = \frac{0.2 + 451.46K_D}{\sqrt{K_P}} \tag{11-28}$$

Applying the stability requirement to Eq. (11-14), we find that, for system stability, $K_P > 0$ and $K_D > -0.000443$

Toolbox 11-2-2

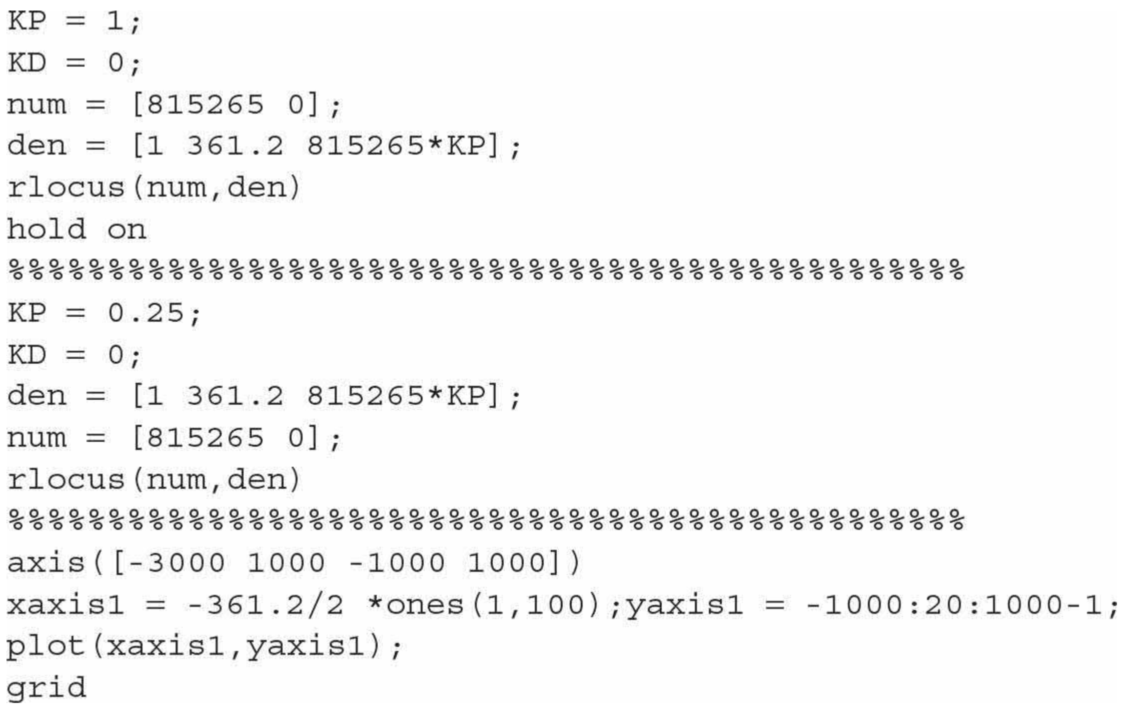
Root loci of Eq. (11-26) shown in Fig. 11-11 are obtained by the following sequence of MATLAB functions:



```
den = [1 361.2 0];
num = [1];
rlocus(num,den)
```

Toolbox 11-2-3

Root contours of Eq. (11-14) shown in Fig. 11-12 are obtained by the following sequence of MATLAB functions:



```
KP = 1;
KD = 0;
num = [815265 0];
den = [1 361.2 815265*KP];
rlocus(num,den)
hold on
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
KP = 0.25;
KD = 0;
den = [1 361.2 815265*KP];
num = [815265 0];
rlocus(num,den)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
axis([-3000 1000 -1000 1000])
xaxis1 = -361.2/2 *ones(1,100);yaxis1 = -1000:20:1000-1;
plot(xaxis1,yaxis1);
grid
```

The boundaries of stability in the K_P-versus-K_D parameter plane are shown in Fig. 11-14. The constant-damping-ratio trajectory is described by Eq. (11-28) and is a parabola. Figure 11-14 illustrates the constant-ζ trajectories for $\zeta = 0.5$, 0.707, and 1.0. The ramp-error constant K_v is given by Eq. (11-13), which describes a horizontal line in the parameter plane, as shown in Fig. 11-10. The figure gives a clear picture as to how the values of K_P and K_D affect the various performance criteria of the system. For instance, if K_v is set at 2257.1, which corresponds to $K_P = 1$, the constant-ζ loci show that the damping is increased monotonically with the increase in K_D. The intersection between the constant-K_v locus and the constant-ζ locus gives the value of K_D

for the desired K_v and ζ.

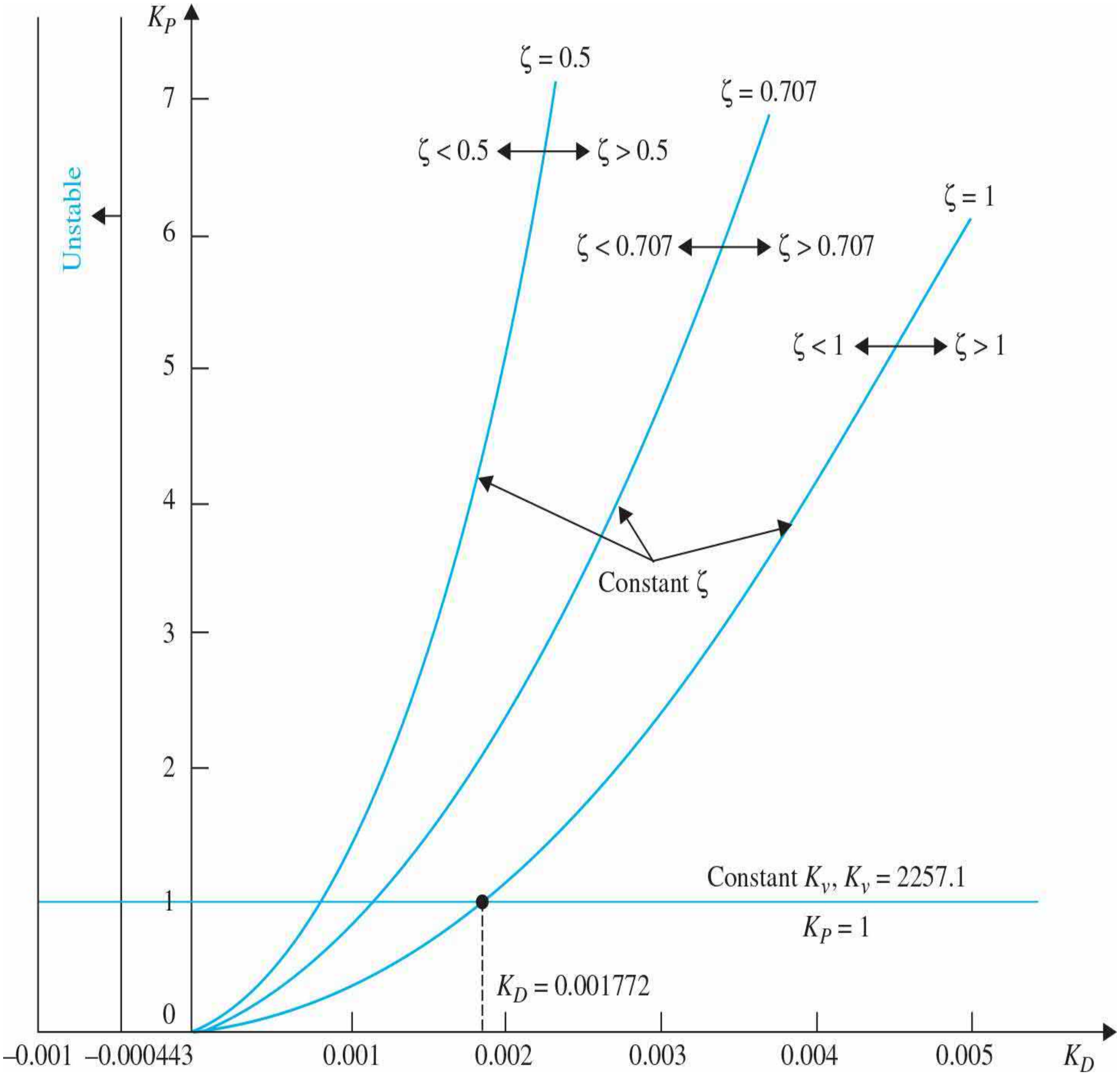


Figure 11-14 K_P-versus-K_D parameter plane for the attitude control system with a PD controller.

Frequency-Domain Design

Now let us carry out the design of the PD controller in the frequency domain. Figure 11-15 shows the Bode plot of $G(s)$ in Eq. (11-11) with $K_P = 1$ and $K_D = 0$. The phase margin of the uncompensated system is 22.68°, and the resonant peak M_r is 2.522. These values correspond to a lightly damped system. Let us give the following performance criteria:

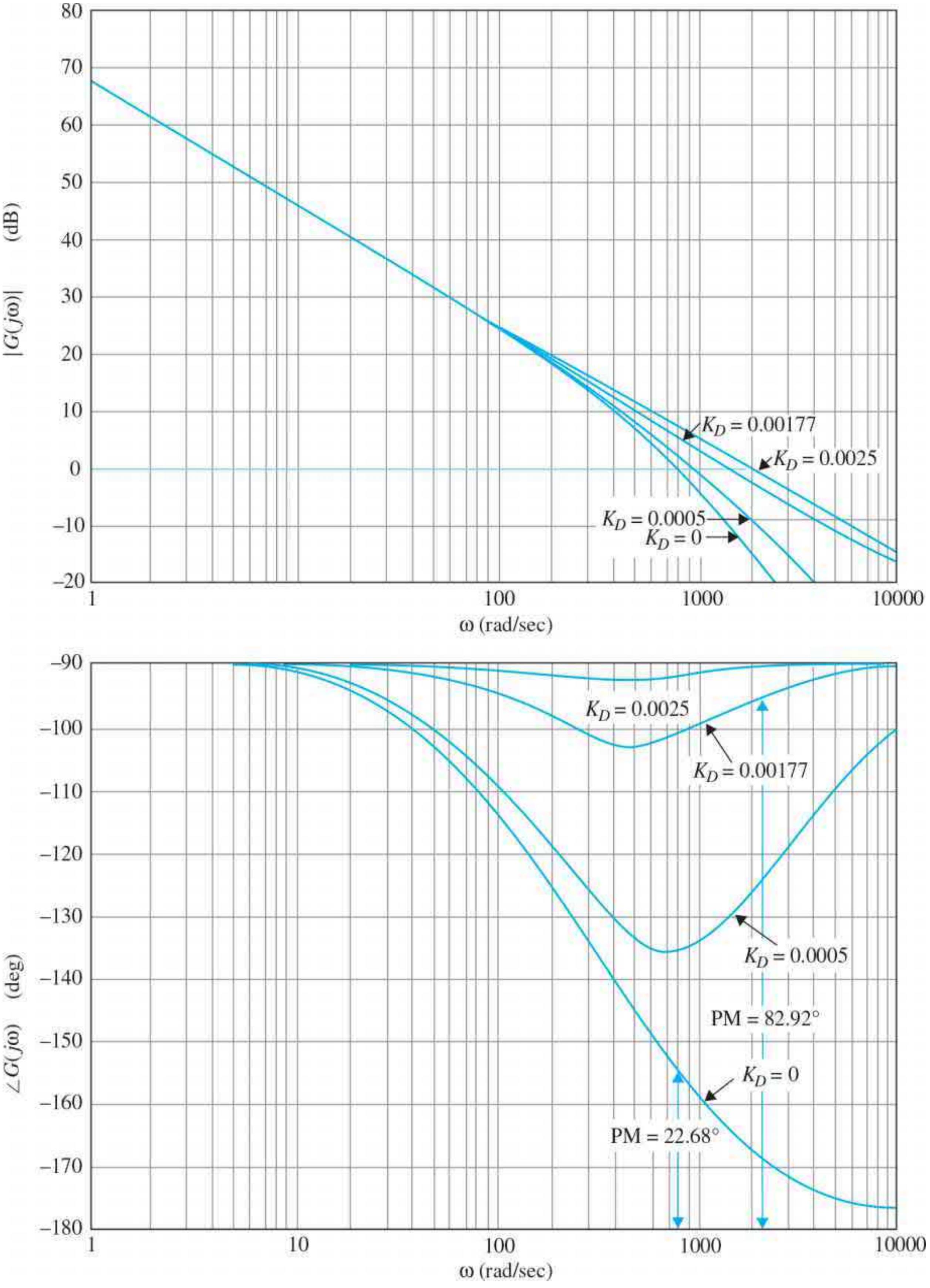
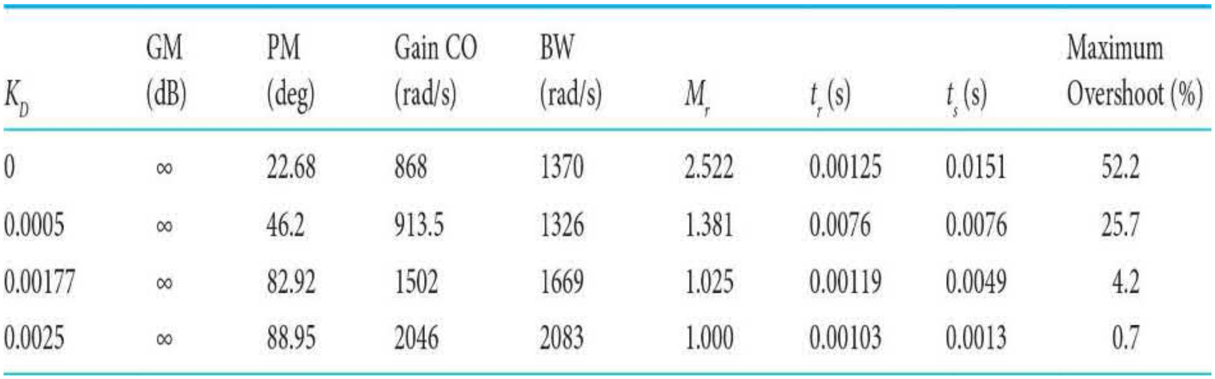

80
70
60
50
40
30
20
10
0
−10
−20
$|G(j\omega)|$ (dB)
1
100
1000
10000
ω (rad/sec)
$K_D = 0.00177$
$K_D = 0.0025$
$K_D = 0.0005$
$K_D = 0$
−90
−100
−110
−120
−130
−140
−150
−160
−170
−180
$\angle G(j\omega)$ (deg)
1
10
100
1000
10000
ω (rad/sec)
$K_D = 0.0025$
$K_D = 0.00177$
$K_D = 0.0005$
PM = 82.92°
$K_D = 0$
PM = 22.68°

Figure 11-15 Bode plot of $G(s)=\dfrac{815,265(1+K_D s)}{s(s+361.2)}$.

Steady-state error due to a unit-ramp input ≤0.00443
Phasemargin ≥80°
Resonant peak M_r≤1.05
BW ≤2000 rad/s

The Bode plots of $G(s)$ for K_P = 1 and K_D = 0, 0.005, 0.00177, and 0.0025 are shown in Fig. 11-15. The performance measures in the frequency domain for the compensated system with these controller parameters are tabulated in Table 11-3, along with the time-domain attributes for comparison. The Bode plots as well as the performance data were generated by using Toolbox 11-2-4.

Table 11-3 Frequency-Domain Characteristics of the System in Example 11-2-1 with PD Controller



K_D	GM (dB)	PM (deg)	Gain CO (rad/s)	BW (rad/s)	M_r	t_r (s)	t_s (s)	Maximum Overshoot (%)
0	∞	22.68	868	1370	2.522	0.00125	0.0151	52.2
0.0005	∞	46.2	913.5	1326	1.381	0.0076	0.0076	25.7
0.00177	∞	82.92	1502	1669	1.025	0.00119	0.0049	4.2
0.0025	∞	88.95	2046	2083	1.000	0.00103	0.0013	0.7

The results in Table 11-3 show that the gain margin is always infinite, and thus the relative stability is measured by the phase margin. This is one example where the gain margin is not an effective measure of the relative stability of the system. When K_D = 0.00177, which corresponds to critical damping, the phase margin is 82.92°, the resonant peak M_r is 1.025, and BW is 1669 rad/s. The performance requirements in the frequency domain are all satisfied. Other effects of the PD control are that the BW and the gain-crossover frequency are increased. The phase-crossover frequency is always infinite in this case.

Toolbox 11-2-4

Figure 11-67 is obtained by the following sequence of MATLAB functions:

```
KD = [0 0.0005 0.0025 0.00177];
for i = 1:length(KD)
num = [815265*KD(i) 815265];
den =[1 361.2 0];
bode(tf(num,den));
hold on;
end
axis([1 10000 -180 -90]);
grid
```

EXAMPLE 11-2-2[2] Consider the third-order aircraft attitude control system discussed in Sec. 7-9, with the forward-path transfer function given in Eq. (7-169),

Control of a DC Motor: Electrical Time Constant Not Neglected

$$G(s)=\frac{1.5\times10^{7}K}{s(s^{2}+3408.3s+1,204,000)} \tag{11-29}$$

The same set of time-domain specifications given in Example 11-2-1 is to be used. It was shown in Sec. 7-9 that, when *k* = 181.17, the maximum overshoot of the system is 78.88 percent.

Let us attempt to meet the transient performance requirements by use of a PD controller with the transfer function given in Eq. (11-2). The forward-path transfer function of the system with the PD controller and *k* = 181.17 is

$$G(s)=\frac{2.718\times10^{9}(K_P+K_D s)}{s(s^{2}+3408.3s+1,204,000)} \tag{11-30}$$

Note that because the system transfer function is third order, it may become unstable for a choice of controller parameters.

If a system is unstable, the PD control may not be effective in improving the stability of the system.

Time-Domain Design Setting K_P = 1 arbitrarily, the characteristic equation of the closed-loop system is written as

$$s^3 + 3408.3s^2 + (1{,}204{,}000 + 2.718\times 10^9 K_D)s + 2.718\times 10^9 = 0 \tag{11-31}$$

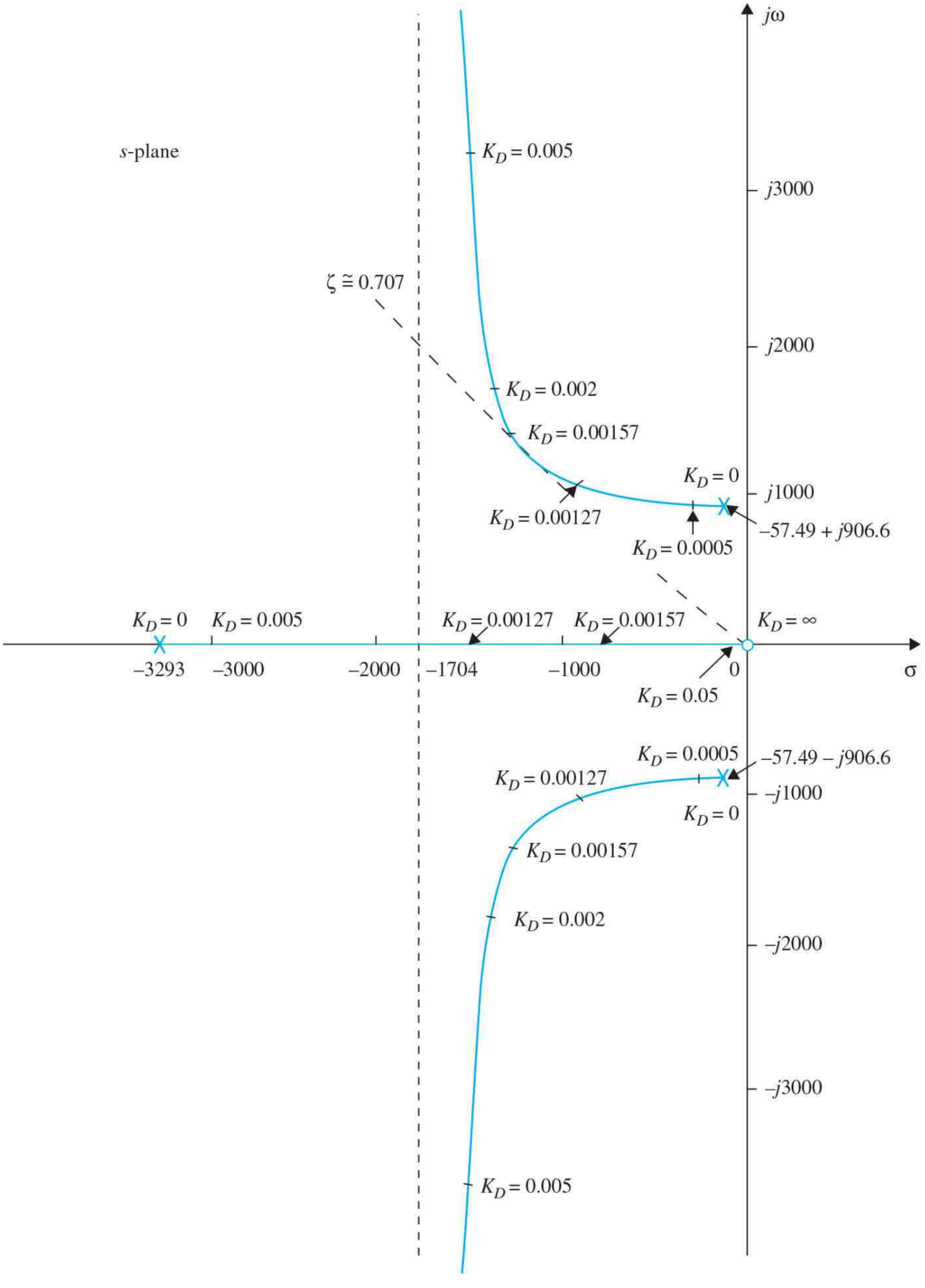
To apply the root-contour method, we condition Eq. (11-31) as

$$1 + G_{\text{eq}}(s) = 1 + \frac{2.718\times 10^9 K_D s}{s^3 + 3408.2s^2 + 1{,}204{,}000s + 2.718\times 10^9} = 0 \tag{11-32}$$

where

$$G_{\text{eq}}(s) = \frac{2.718\times 10^9 K_D s}{(s+3293.3)(s+57.49+j906.6)(s+57.49-j906.6)} \tag{11-33}$$

The root contours of Eq. (11-31) are plotted as shown in Fig. 11-16, based on the pole-zero configuration of $G_{eq}(s)$. The root contours of Fig. 11-16 reveal the effectiveness of the PD controller for the improvement on the relative stability of the system. Notice that, as the value of K_D increases, one root of the characteristic equation moves from –3293.3 toward the origin, while the two complex roots start out toward the left and eventually approach the vertical asymptotes that intersect at *s*= -1704. The immediate assessment of the situation is that, if the value of K_D is too large, *the two complex roots will actually have reduced damping while increasing the natural frequency of the system*. It appears that the ideal location for the two complex characteristic equation roots, from the standpoint of relative stability, is near the bend of the root contour, where the relative damping ratio is approximately 0.707. The root contours of Fig. 11-16 clearly show that, if the original system has low damping or is unstable, the zero introduced by the PD controller may not be able to add sufficient damping or even stabilize the system.



jω
s-plane
$K_D = 0.005$
j3000
ζ ≅ 0.707
j2000
$K_D = 0.002$
$K_D = 0.00157$
$K_D = 0$
j1000
$K_D = 0.00127$
−57.49 + j906.6
$K_D = 0.0005$
$K_D = 0$
$K_D = 0.005$
$K_D = 0.00127$
$K_D = 0.00157$
$K_D = \infty$
−3293
−3000
−2000
−1704
−1000
0
σ
$K_D = 0.05$
$K_D = 0.0005$
−57.49 − j906.6
$K_D = 0.00127$
−j1000
$K_D = 0$
$K_D = 0.00157$
$K_D = 0.002$
−j2000
−j3000
$K_D = 0.005$

Figure 11-16 Root contours of $s^3 + 3408.3s^2 + (1{,}204{,}000 + 2.718\times10^9 K_D)s + 2.718\times10^9 = 0.$

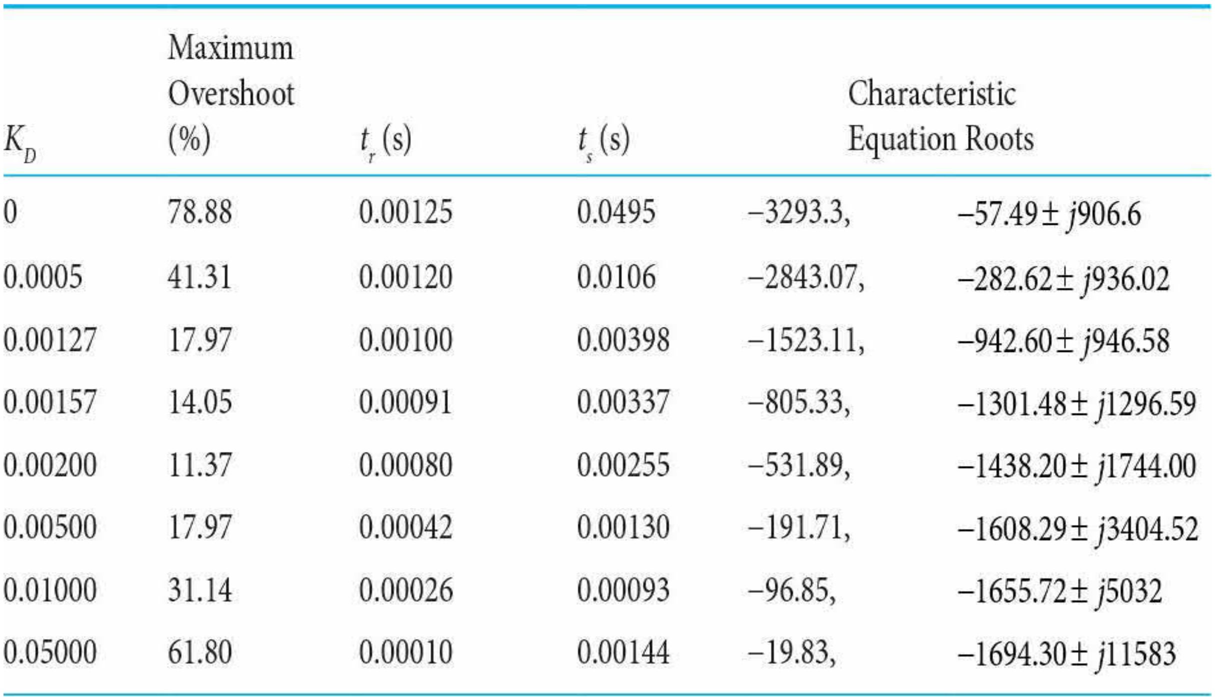
Toolbox 11-2-5

Root contours of Eq. (11-232) shown in Fig. 11-83 are obtained by the following sequence of MATLAB functions:

```
kd=0.005;
num = [2.718*10^9*kd 0];
den = [1 3408.2 1204000 2.718*10^9];
rlocus(num,den)
```

Table 11-4 gives the results of maximum overshoot, rise time, settling time, and the roots of the characteristic equation as functions of the parameter K_D. The following conclusions are drawn on the effects of the PD controller on the third-order system.

Table 11-4 Time-Domain Attributes of the Third-Order System in Example 11-2-2 with PD Controller



K_D	Maximum Overshoot (%)	t_r (s)	t_s (s)	Characteristic Equation Roots	
0	78.88	0.00125	0.0495	−3293.3,	$-57.49 \pm j906.6$
0.0005	41.31	0.00120	0.0106	−2843.07,	$-282.62 \pm j936.02$
0.00127	17.97	0.00100	0.00398	−1523.11,	$-942.60 \pm j946.58$
0.00157	14.05	0.00091	0.00337	−805.33,	$-1301.48 \pm j1296.59$
0.00200	11.37	0.00080	0.00255	−531.89,	$-1438.20 \pm j1744.00$
0.00500	17.97	0.00042	0.00130	−191.71,	$-1608.29 \pm j3404.52$
0.01000	31.14	0.00026	0.00093	−96.85,	$-1655.72 \pm j5032$
0.05000	61.80	0.00010	0.00144	−19.83,	$-1694.30 \pm j11583$

1. The minimum value of the maximum overshoot, 11.37 percent, occurs when K_D is approximately 0.002.
2. Rise time is improved (reduced) with the increase of K_D.
3. Too high a value of K_D will actually increase the maximum overshoot and the settling time substantially. The latter is because the damping is reduced as K_D is increased indefinitely.

Figure 11-17 shows the unit-step responses of the system with the PD controller for several values of K_D. The conclusion is that, while the PD control does improve the damping of the system, it does not meet the maximum-overshoot requirement.

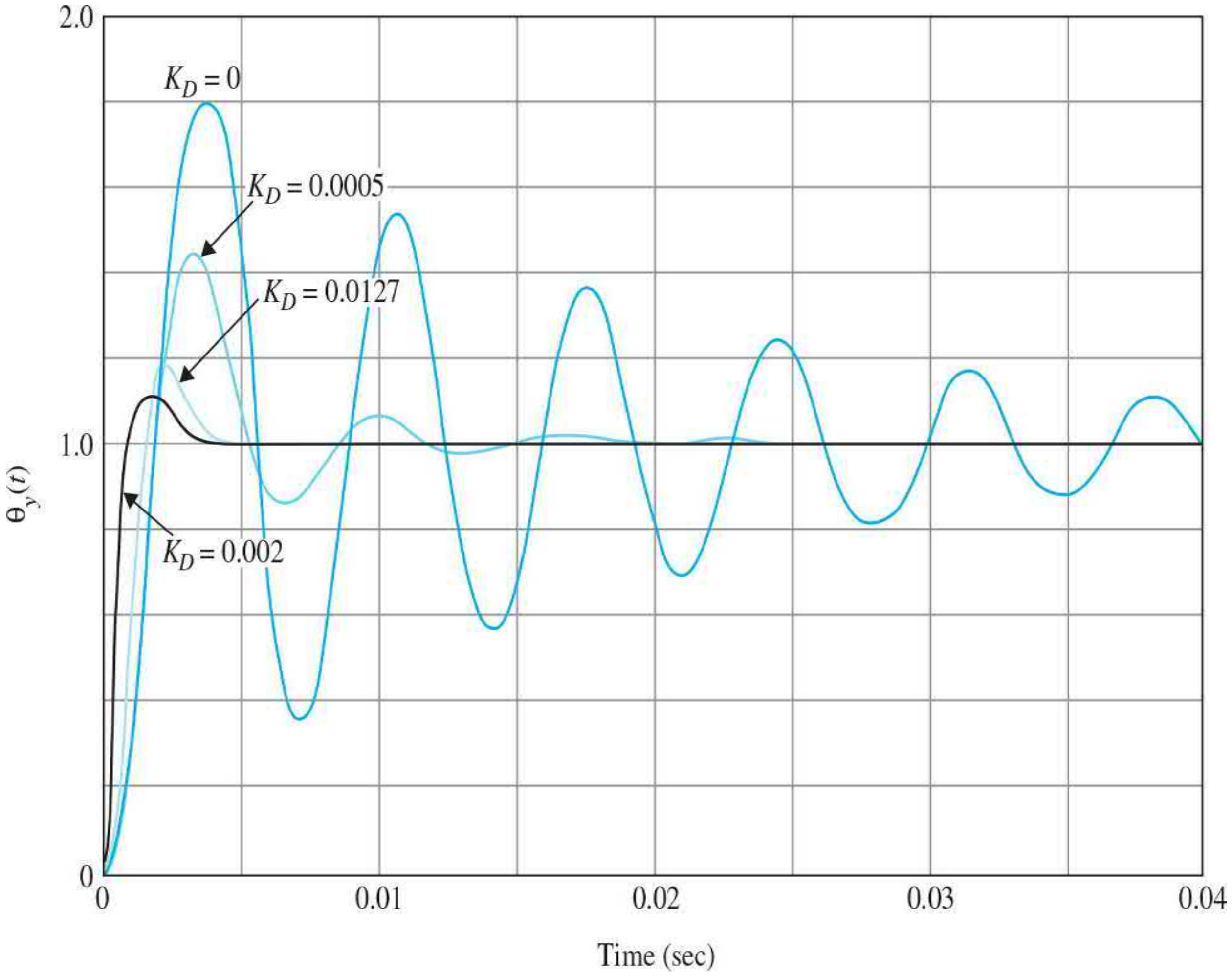


Figure 11-17 Unit-step responses of the system in Example 11-2-2 with PD controller.

Frequency-Domain Design The Bode plot of Eq. (11-30) is used to conduct the frequency-domain design of the PD controller. Figure 11-18 shows the Bode plot for $K_P = 1$ and $K_D = 0$. The following performance data are obtained for the uncompensated system:

Gain margin = 3.6 dB

Phase margin = 7.77°

Resonant peak $M_r = 7.62$

Bandwidth BW = 1408.83 rad/s

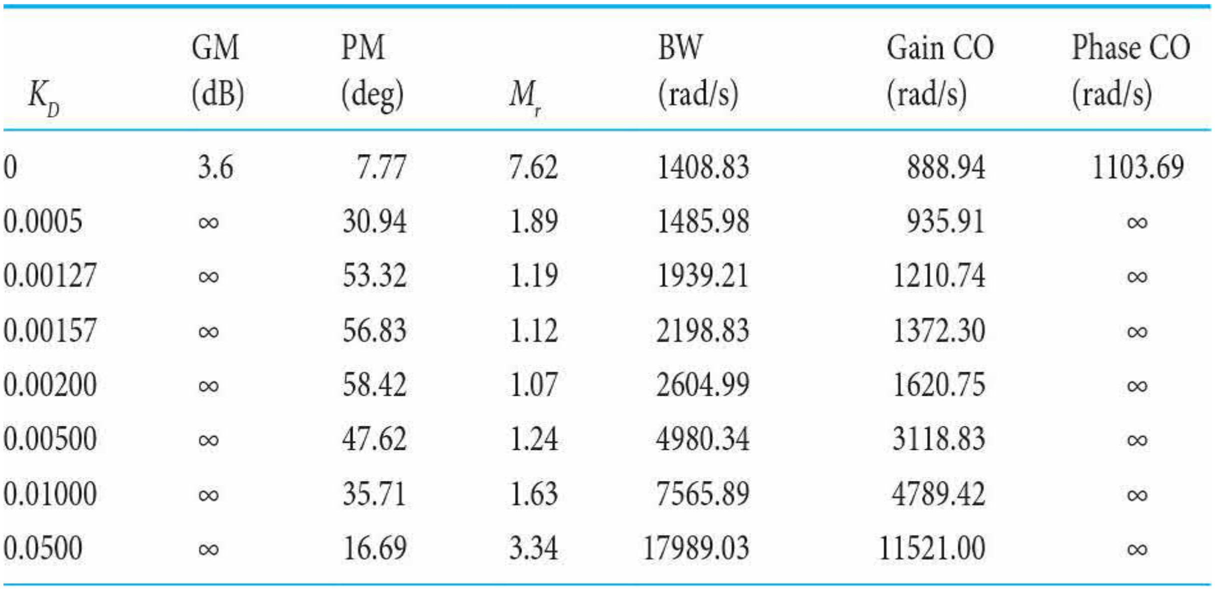
Gain crossover (GCO) = 888.94 rad/s

Phase crossover (PCO) = 1103.69 rad/s

Let us use the same set of frequency-domain performance requirements listed in Example 11-2-1. The logical way to approach this problem is to first examine how much additional phase is needed to realize a phase margin of 80°. Because the uncompensated system with the gain set to meet the steady-state requirement is only 7.77°, the PD controller must provide an additional phase of 72.23°. This additional phase must be placed at the gain crossover of the compensated system in order to realize a PM of 80°. Referring to the Bode plot of the PD controller in Fig. 11-6, we see that the additional phase is always accompanied by a gain in the magnitude curve. As a result, the gain crossover of the compensated system will be pushed to a higher frequency at which the phase of the uncompensated system would correspond to an even smaller PM. Thus, we may run into the problem of diminishing returns. This symptom is parallel to the situation illustrated by the root-contour plot in Fig. 11-16, in which case the larger K_D would simply push the roots to a higher frequency, and the damping would actually be decreased. The frequency-domain performance data of the compensated system with the values of K_D used in Table 11-4 are obtained from the Bode plots for each case, and the results are shown in Table 11-5. The Bode plots of some of these cases are shown in Fig. 11-18. Notice that the gain margin becomes infinite when the PD controller is added, and the phase margin becomes the dominant measure of relative stability. This is because the phase curve of the PD-compensated system stays above the –180°-axis, and the phase crossover is at infinity.

Table 11-5 Frequency-Domain Characteristics of the Third-Order

System in Example 11-2-2 with PD Controller



K_D	GM (dB)	PM (deg)	M_r	BW (rad/s)	Gain CO (rad/s)	Phase CO (rad/s)
0	3.6	7.77	7.62	1408.83	888.94	1103.69
0.0005	∞	30.94	1.89	1485.98	935.91	∞
0.00127	∞	53.32	1.19	1939.21	1210.74	∞
0.00157	∞	56.83	1.12	2198.83	1372.30	∞
0.00200	∞	58.42	1.07	2604.99	1620.75	∞
0.00500	∞	47.62	1.24	4980.34	3118.83	∞
0.01000	∞	35.71	1.63	7565.89	4789.42	∞
0.0500	∞	16.69	3.34	17989.03	11521.00	∞

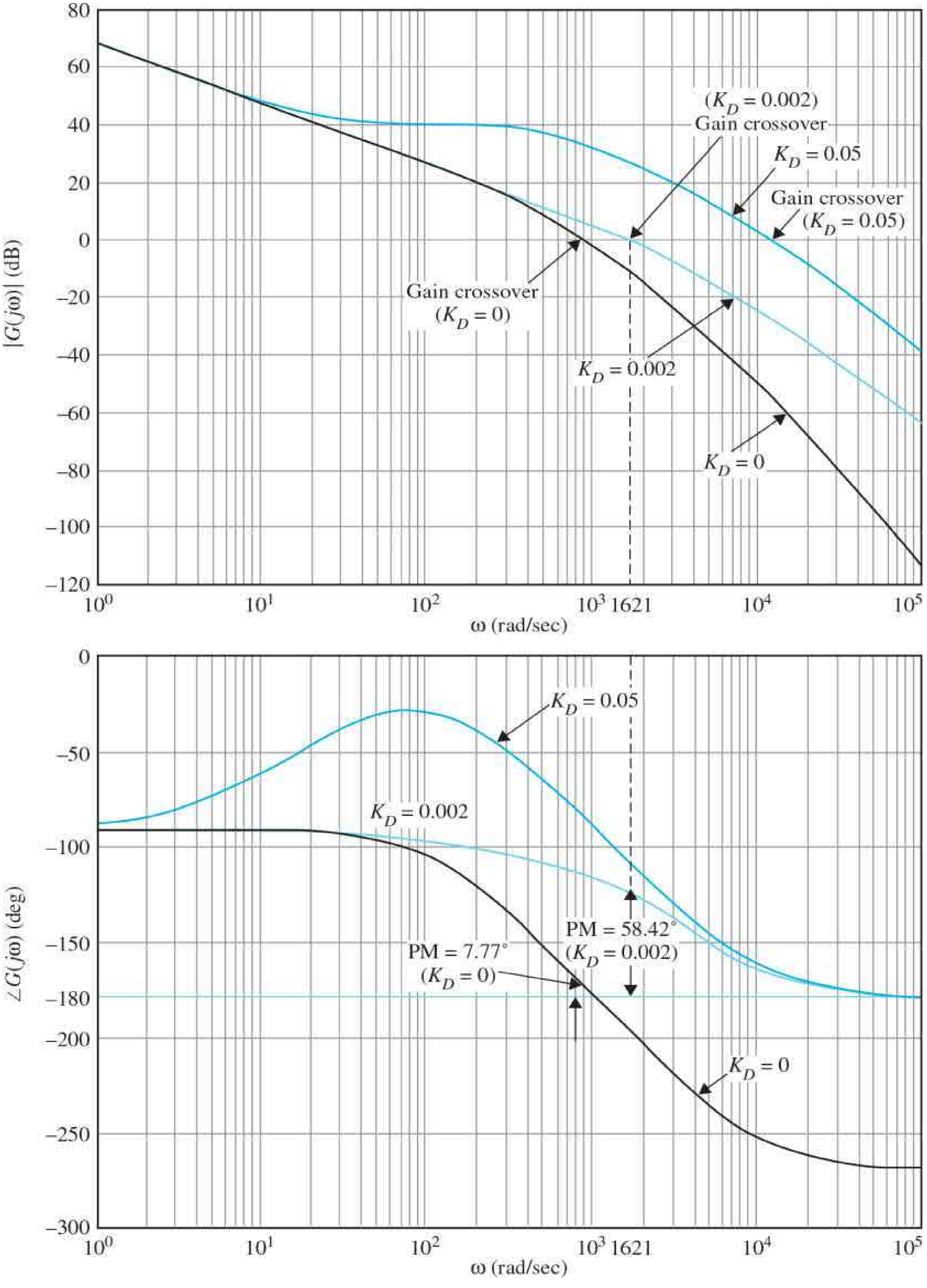
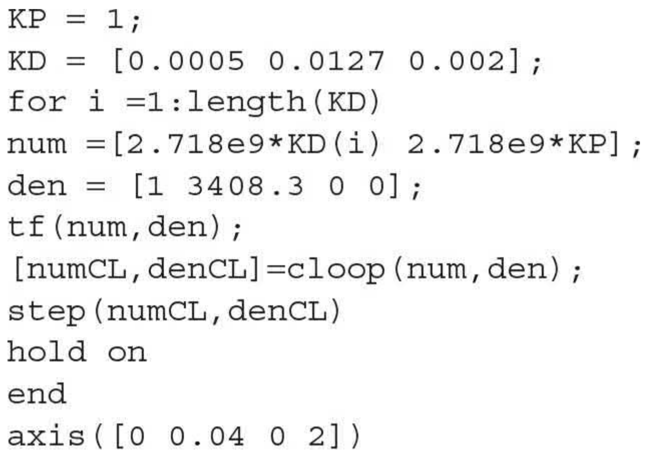

80
60
40
20
0
−20
−40
−60
−80
−100
−120
$|G(j\omega)|$ (dB)
$(K_D = 0.002)$
Gain crossover
$K_D = 0.05$
Gain crossover
$(K_D = 0.05)$
Gain crossover
$(K_D = 0)$
$K_D = 0.002$
$K_D = 0$
10^0
10^1
10^2
10^3 1621
10^4
10^5
ω (rad/sec)
0
−50
−100
−150
−180
−200
−250
−300
$\angle G(j\omega)$ (deg)
$K_D = 0.05$
$K_D = 0.002$
PM = 58.42°
$(K_D = 0.002)$
PM = 7.77°
$(K_D = 0)$
$K_D = 0$
10^0
10^1
10^2
10^3 1621
10^4
10^5
ω (rad/sec)

Figure 11-18 Bode diagram of $G(s)$ of the system in Example 11-2-2 with PD controller.

Toolbox 11-2-6

Bode plot of Fig. 11-15 is obtained by the following sequence of MATLAB functions:



```
KP = 1;
KD = [0.0005 0.0127 0.002];
for i =1:length(KD)
num =[2.718e9*KD(i) 2.718e9*KP];
den = [1 3408.3 0 0];
tf(num,den);
[numCL,denCL]=cloop(num,den);
step(numCL,denCL)
hold on
end
axis([0 0.04 0 2])
```

When K_D = 0.002, the phase margin is at a maximum of 58.42°, and M_r is also minimum at 1.07, which happens to agree with the optimal value obtained in the time-domain design summarized in Table 11-4. When the value of K_D is increased beyond 0.002, the phase margin decreases, which agrees with the findings from the time-domain design that large values of K_D actually decreases damping. However, the BW and the gain crossover increase continuously with the increase in K_D. The frequency-domain design again shows that the PD control falls short in meeting the performance requirements imposed on the system. Just as in the time-domain design, we have demonstrated that if the original system has very low damping, or is unstable, PD control may not be effective in improving the stability of the system. Another situation under which PD control may be ineffective is if the slope of the phase curve near the gain-crossover frequency is steep, in which case the rapid decrease of the phase margin due to the increase of the gain crossover from the added gain of the PD controller may render the additional phase ineffective.

Toolbox 11-2-7

Bode diagram of G(s) in Example 11-2-2 in Fig. 11-18 is obtained by the following sequence of MATLAB functions:

```
KD = [0,0.002,0.05];
KP=1;
for i = 1:length(KD)
num =[2.718e9*KD(i) 2.718e9*KP];
den = [1 3408.3 1204000 0];
bode(num,den);
hold on;
end
```

11-3 DESIGN WITH THE PI CONTROLLER

We see from Sec. 11-2 that the PD controller can improve the damping and rise time of a control system at the expense of higher bandwidth and resonant frequency, and the steady-state error is not affected unless it varies with time, which is typically not the case for step-function inputs. Thus, the PD controller may not fulfill the compensation objectives in many situations.

The integral part of the PID controller produces a signal that is proportional to the time integral of the input of the controller. Figure 11-19 illustrates the block diagram of a prototype second-order system with a series PI controller. The transfer function of the PI controller is

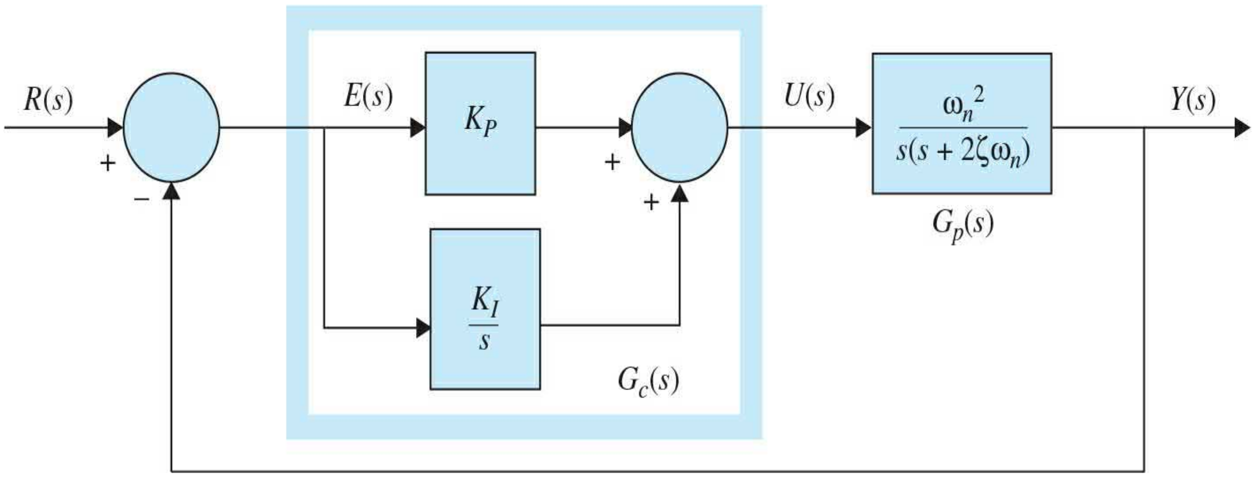
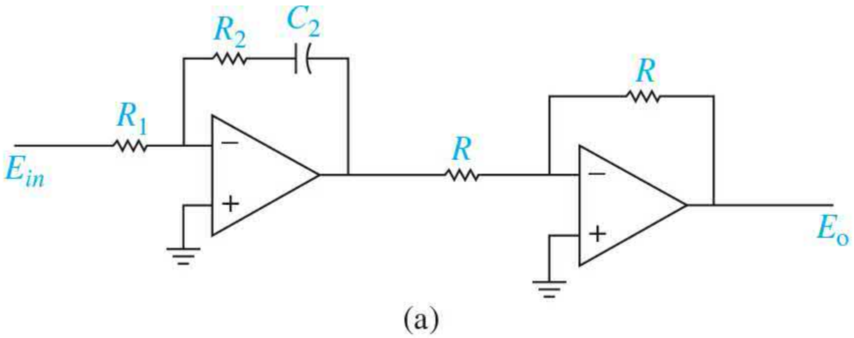


Figure 11-19 Control system with PI controller.

$$G_c(s) = K_P + \frac{K_I}{s} \tag{11-34}$$

Using the circuit elements given in Table 6-1, two op-amp-circuit realizations of Eq. (11-34) are shown in Fig. 11-20. The transfer function of the two-op-amp circuit in Fig. 11-20*a* is



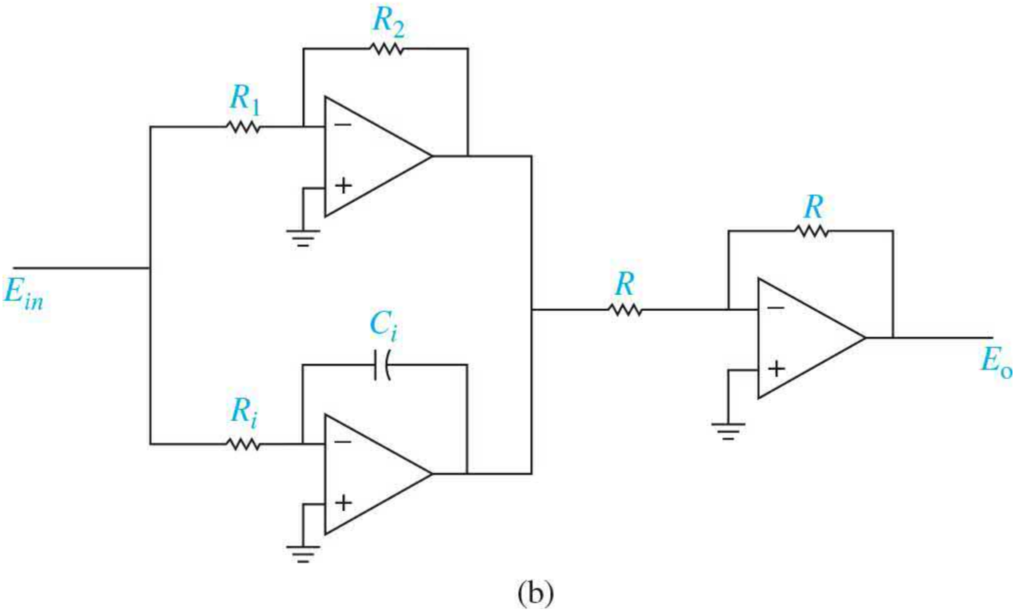


Figure 11-20 Op-amp-circuit realization of the PI controller, $G_c(s) = K_P + \frac{K_I}{s}$. (*a*) Two-op-amp circuit. (*b*) Three-op-amp circuit.

$$G_c(s) = \frac{E_o(s)}{E_{\text{in}}(s)} = \frac{R_2}{R_1} + \frac{R_2}{R_1 C_2 s} \tag{11-35}$$

Comparing Eq. (11-34) with Eq. (11-35), we have

$$K_P = \frac{R_2}{R_1} \quad K_I = \frac{R_2}{R_1 C_2} \tag{11-36}$$

The transfer function of the three-op-amp circuit in Fig. 11-20*b* is

$$G_c(s) = \frac{E_o(s)}{E_{\text{in}}(s)} = \frac{R_2}{R_1} + \frac{1}{R_i C_i s} \tag{11-37}$$

Thus, the parameters of the PI controller are related to the circuit parameters as

$$K_P = \frac{R_2}{R_1} \quad K_I = \frac{1}{R_i C_i} \tag{11-38}$$

The advantage with the circuit in Fig. 11-20*b* is that the values of K_P and K_I are independently related to the circuit parameters. However, in either circuit, K_I is inversely proportional to the value of the capacitor. Unfortunately, effective PI-control designs usually result in small values of K_I, and thus we must again watch out for unrealistically large capacitor values.

The forward-path transfer function of the compensated system is

$$G(s) = G_c(s)G_p(s) = \frac{\omega_n^2(K_P s + K_I)}{s^2(s + 2\zeta\omega_n)} \tag{11-39}$$

Clearly, the immediate effects of the PI controller are as follows:

1. Adding a zero at s = $-K_I/K_P$ to the forward-path transfer function.
2. Adding a pole at $s = 0$ to the forward-path transfer function. This means that the system type is increased by 1 to a type 2 system. Thus, the steady-state error of the original system is improved by one order; that is, if the steady-state error to a given input is constant, the PI control reduces it to zero (provided that the compensated system remains stable).

The system in Fig. 11-19, with the forward-path transfer function in Eq.

(11-39), will now have a zero steady-state error when the reference input is a ramp function. However, because the system is now of the third order, *it may be less stable* than the original second-order system or even become *unstable* if the parameters K_P and K_I are not properly chosen.

In the case of a type 1 system with a PD control, the value of K_P is important because the ramp-error constant K_v is directly proportional to K_P, and thus the magnitude of the steady-state error is inversely proportional to K_P when the input is a ramp. On the other hand, if K_P is too large, the system may become unstable. Similarly, for a type 0 system, the steady-state error due to a step input will be inversely proportional to K_P.

When a type 1 system is converted to type 2 by the PI controller, K_P no longer affects the steady-state error, and the latter is always zero for a stable system with a ramp input. The problem is then to choose the proper combination of K_P and K_I so that the transient response is satisfactory.

11-3-1 Time-Domain Interpretation and Design of PI Control

The pole-zero configuration of the PI controller in Eq. (11-34) is shown in Fig. 11-21. At first glance, it may seem that PI control will improve the steady-state error at the expense of stability. However, we can show that, if the location of the zero of $G_c(s)$ is selected properly, both the damping and the steady-state error can be improved. Because the PI controller is essentially a low-pass filter, the compensated system usually will have a slower rise time and longer settling time. *A viable method of designing the PI control is to select the zero at* $s = -K_I/K_P$ *so that it is relatively close to the origin and away from the most significant poles of the process; the values of* K_P *and* K_I *should be relatively small.*

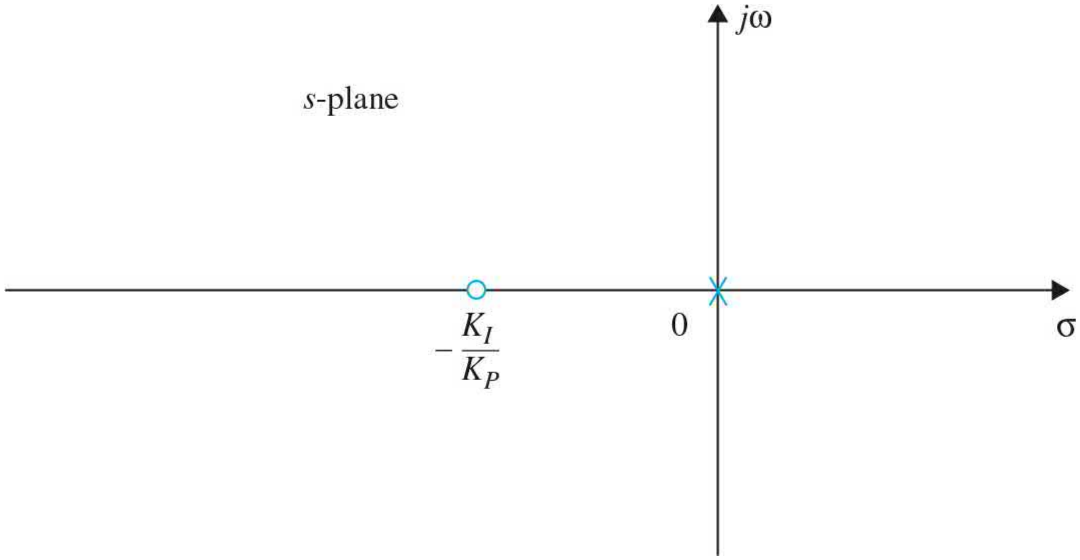


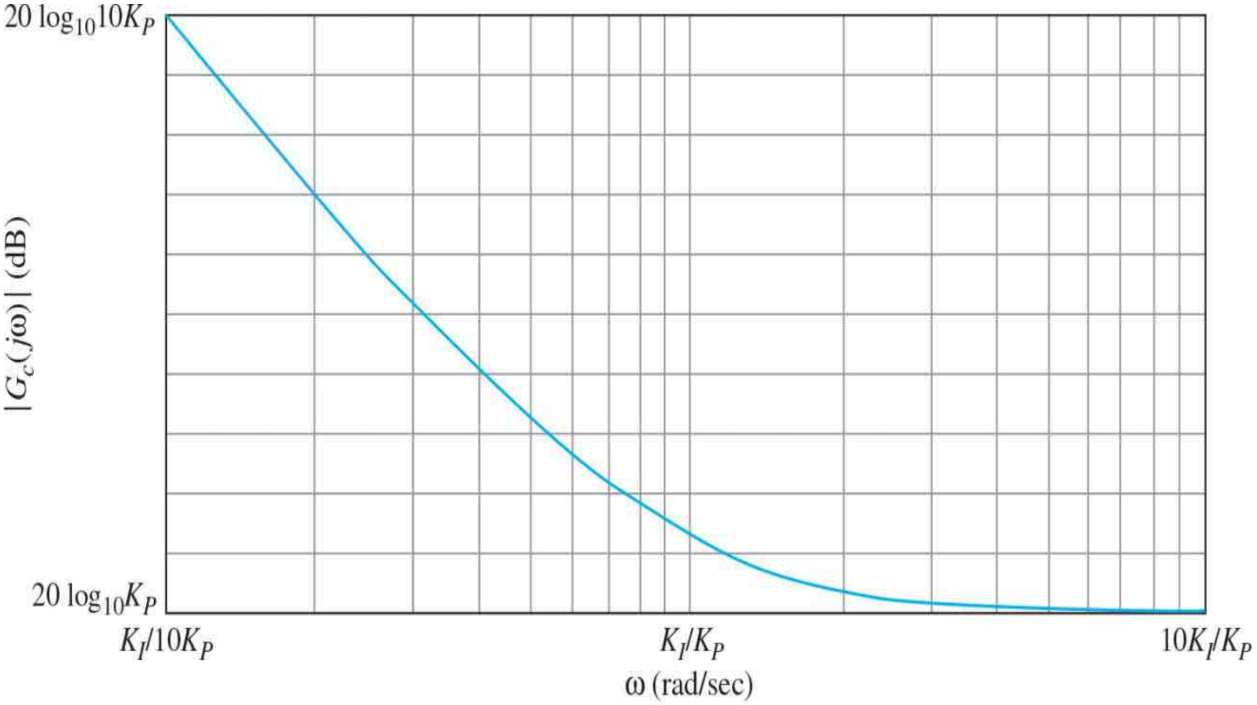
Figure 11-21 Pole-zero configuration of a PI controller.

11-3-2 Frequency-Domain Interpretation and Design of PI Control

For frequency-domain design, the transfer function of the PI controller is written as

$$G_c(s) = K_P + \frac{K_I}{s} = \frac{K_I\left(1 + \frac{K_P}{K_I}s\right)}{s} \tag{11-40}$$

The Bode plot of $G_c(j\omega)$ is shown in Fig. 11-22. Notice that the magnitude of $G_c(j\omega)$ at $\omega = \infty$ is 20 $\log_{10} K_P$ dB, which represents an attenuation if the value of K_P is less than 1. This attenuation may be utilized to improve the stability of the system. The phase of $G_c(j\omega)$ is always negative, which is detrimental to stability. Thus, we should place the corner frequency of the controller, $\omega = K_I/K_P$, as far to the left as the bandwidth requirement allows, so the phase-lag properties of $G_c(j\omega)$ do not degrade the achieved phase margin of the system.


$20\log_{10}10K_P$
$|G_c(j\omega)|$ (dB)
$20\log_{10}K_P$
$K_I/10K_P$
K_I/K_P
$10K_I/K_P$
ω (rad/sec)

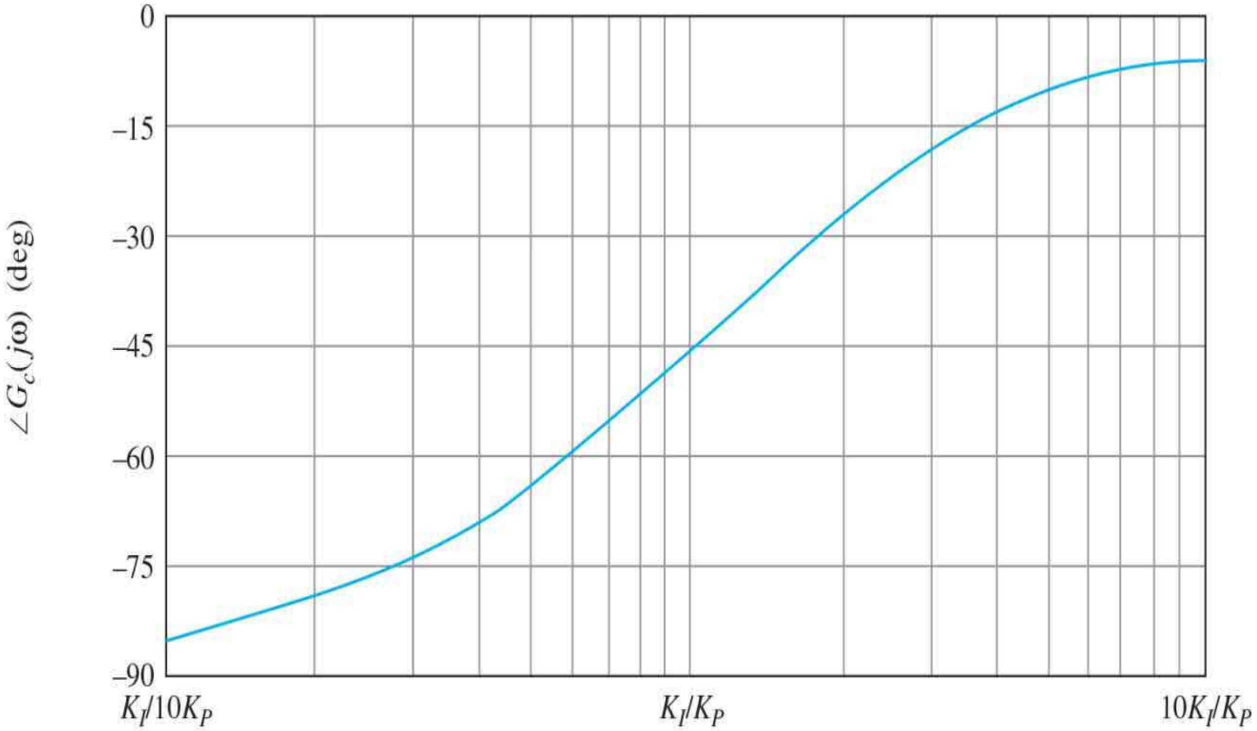

0
–15
–30
–45
–60
–75
–90
$\angle G_c(j\omega)$ (deg)
$K_I/10K_P$
K_I/K_P
$10K_I/K_P$
ω (rad/sec)

Figure 11-22 Bode diagram of the PI controller. $G_c(s) = K_P + \frac{K_I}{s}$.

The frequency-domain design procedure for the PI control to realize a given phase margin is outlined as follows:

1. The Bode plot of the forward-path transfer function $G_p(s)$ of the uncompensated system is made with the loop gain set according to the steady-state performance requirement.
2. The phase margin and the gain margin of the uncompensated system are determined from the Bode plot. For a specified phase margin requirement, the new gain-crossover frequency ω'_g corresponding to this phase margin is found on the Bode plot. The magnitude plot of the compensated transfer function must pass through the 0-dB axis at this new gain-crossover frequency in order to realize the desired phase margin.

As a general guideline, K_I/K_P should correspond to a frequency that is at least one decade, sometimes as much as two decades, below ω'_g.

3. To bring the magnitude curve of the uncompensated transfer function down to 0 dB at the new gain-crossover frequency ω'_g, the PI controller must provide the amount of attenuation equal to the gain of the magnitude curve at the new gain-crossover frequency. In other words, set

$$\left|G_P(j\omega'_g)\right|_{\text{dB}} = -20\log_{10}K_P\,\text{dB} \quad K_P < 1 \tag{11-41}$$

from which we have

$$K_P = 10^{-\left|G_P(j\omega'_g)\right|_{\text{dB}}/20} \quad K_P < 1 \tag{11-42}$$

Once the value of K_P is determined, it is necessary only to select the proper value of K_I to complete the design. Up to this point, we have assumed that, although the gain-crossover frequency is altered by attenuating the magnitude of $G_c(j\omega)$ at ω'_g, the original phase is not affected by the PI controller. This is not possible, however, since, as shown in Fig. 11-22, the attenuation property of the PI controller is accompanied with a phase lag that is detrimental to the phase margin. It is apparent that, if the corner frequency $\omega = K_I/K_P$ is placed far below ω'_g, the phase lag of the PI controller will have a negligible effect on the phase of the compensated system near ω'_g. On the other hand, the value of K_I/K_P should not be too small or the bandwidth of the system will be too low, causing the rise time and settling time to be too large. As a general guideline, K_I/K_P should correspond to a frequency that is at least one decade, sometimes as much as two decades, below ω'_g. That is, we set

$$\frac{K_I}{K_P} = \frac{\omega'_g}{10} \text{ rad/s} \tag{11-43}$$

Within the general guideline, the selection of the value of K_I/K_P is pretty much at the discretion of the designer, who should be mindful of its effect on BW and its practical implementation by an op-amp circuit.

4. The Bode plot of the compensated system is investigated to see if the performance specifications are all met.
5. The values of K_I and K_P are substituted in Eq. (11-40) to give the desired transfer function of the PI controller.

If the controlled process $G_P(s)$ is type 0, the value of K_I may be selected based on the ramp-error-constant requirement, and then there would only be one parameter, K_P, to determine. By computing the phase margin, gain margin, M_r, and BW of the closed-loop system with a range of values of K_P, the best value for K_P can be easily selected.

Based on the preceding discussions, we can summarize the advantages and disadvantages of a properly designed PI controller as the following:

1. Improving damping and reducing maximum overshoot.
2. Increasing rise time.
3. Decreasing BW.
4. Improving gain margin, phase margin, and M_r.
5. Filtering out high-frequency noise.

It should be noted that in the PI controller design process, selection of a proper combination of K_I and K_P, so that the capacitor in the circuit implementation of the controller is not excessively large, is more difficult than in the case of the PD controller.

The following examples will illustrate how the PI control is designed and what its effects are.

EXAMPLE 11-3-1 Control of a DC Motor: Small Time-Constant Model

Consider the second-order attitude-control system discussed in Example 11-2-1. Applying the PI controller of Eq. (11-34), the forward-path transfer function of the system becomes

$$G(s) = G_c(s)G_P(s) = \frac{4500KK_P(s + K_I/K_P)}{s^2(s + 361.2)} \tag{11-44}$$

Time-Domain Design Let the time-domain performance requirements be

Steady-state error due to parabolic input $t^2u_s(t)/2 \leq 0.2$

Maximum overshoot ≤5%

Rise time $t_r \leq 0.01$ s

Settling time $t_s \leq 0.02$ s

The significance of the requirement on the steady-state error due to a parabolic input is that it indirectly places a minimum requirement on the speed of the transient response.

The parabolic-error constant is

$$\begin{aligned} K_a &= \lim_{s\to 0} s^2G(s) = \lim_{s\to 0} s^2 \frac{4500KK_P(s + K_I/K_P)}{s^2(s + 361.2)} \\ &= \frac{4500KK_I}{361.2} = 12.46KK_I \end{aligned} \tag{11-45}$$

The steady-state error due to the parabolic input $t^2u_s(t)/2$ is

$$e_{ss} = \frac{1}{K_a} = \frac{0.08026}{KK_I} (\leq 0.2) \tag{11-46}$$

Let us set K = 181.17, simply because this was the value used in Example 11-2-1. As it appears, to satisfy a given steady-state error requirement for a parabolic input, the larger the K, the smaller K_I can be. Substituting K = 181.17 in Eq. (11-46) and solving K_I for the minimum steady-state error requirement of 0.2, we get the minimum value of K_I to be 0.002215. If necessary, the value of K can be adjusted later.

With K = 181.17, the characteristic equation of the closed-loop system is

$$s^3 + 361.2s^2 + 815{,}265K_P s + 815{,}265K_I = 0 \tag{11-47}$$

Check Stability. Applying Routh's test to Eq. (11-47) yields the result that the system is stable for $0<K_I/K_P$. This means that the zero of $G(s)$ at s = $-K_I/K_P$ cannot be placed too far to the left in the left-half s-plane, or the system will be unstable. For the present case, the most significant pole of $G_P(S)$, besides the pole at s = 0, is at −361.2. Thus, K_I/K_P should be chosen so that the following condition is satisfied:

$$\frac{K_I}{K_P} << 361.2 \tag{11-48}$$

Let us place the zero at $-K_I/K_P$ relatively close to the origin. The root loci of Eq. (11-47) with K_1/K_P = 10 are shown in Fig. 11-23. Notice that, other than the small loop around the zero at s = -10, these root loci for the most part are very similar to those shown in Fig. 11-11, which are for Eq. (11-26). Let us assume that we wish to have a relative damping ratio of 0.707. In this case, given the similarity of the two root loci, Eq. (11-44) can be approximated by

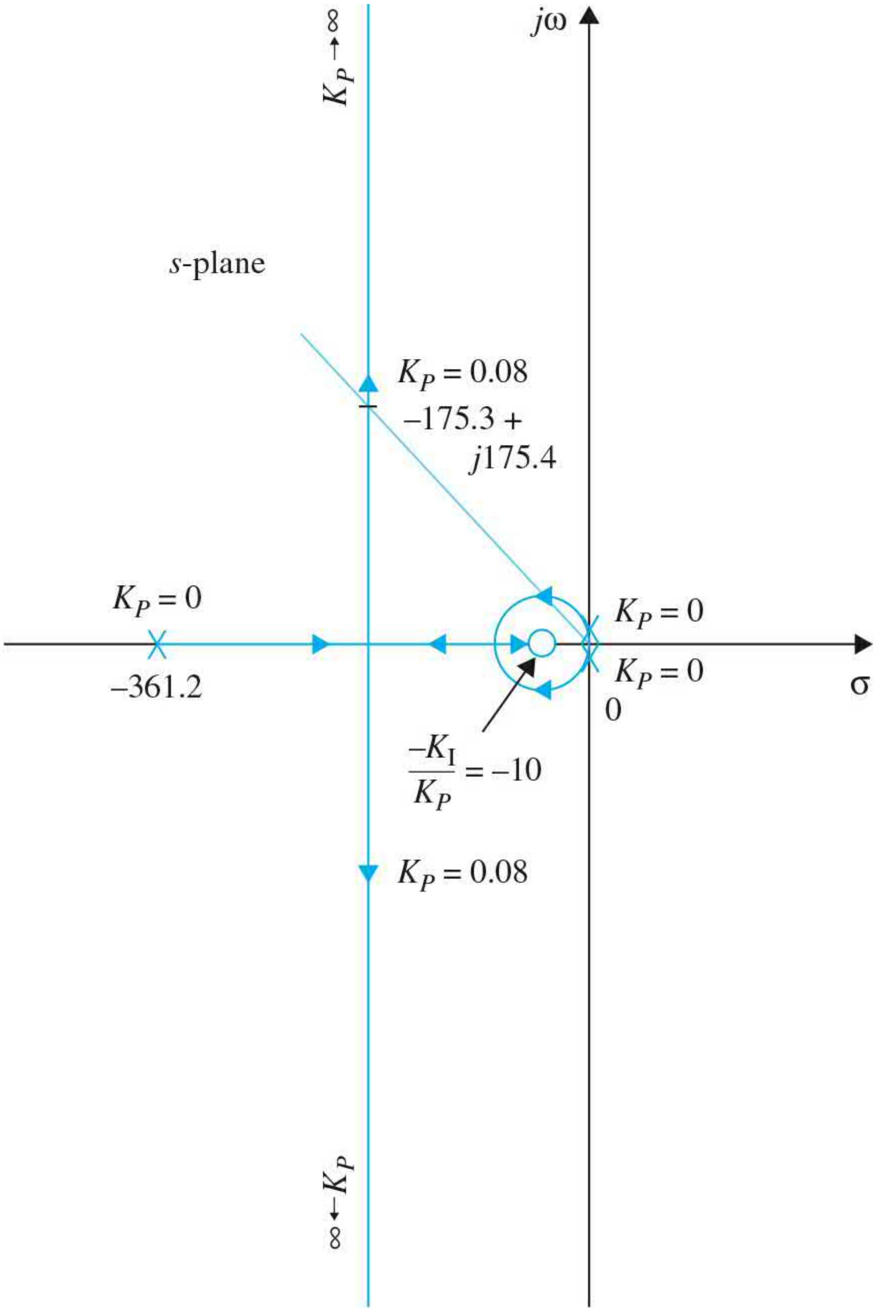


Figure 11-23 Root loci of Eq. (11-37) with $K_I/K_P = 10$; K_P varies.

$$G(s) \cong \frac{815{,}265 K_P}{s(s+361.2)} \tag{11-49}$$

where the term K_I/K_P in the numerator is neglected. From Eq. (11-49), the required value of K_P for this damping ratio is 0.08—compared to the prototype second-order system equivalent. This should also be true for the third-order system with the PI controller if the value of K_I/K_P satisfies Eq. (11-48). Thus, with $K_P = 0.08$, $K_I = 0.8$, the root-locus diagram in Fig. 11-23 shows that the relative damping ratio of the two complex roots is approximately 0.707, and the three characteristic equation roots are at $s_1 = -10.605$, and $s_{2,3} =-175.3\pm j175.4$.

In fact, we can show that, as long as K_P=0.08 and the value of K_I is chosen such that Eq. (11-48) is satisfied, the relative damping ratio of the complex roots will be very close to 0.707. For example, let us select $K_I/K_P = 5$; the three characteristic equation roots are at

$$s = -5.145, -178.03 + j178.03, \text{ and } -178.03 - j178.03$$

and the relative damping ratio is still 0.707. Although the real pole of the closed-loop transfer function is moved, it is close enough to the zero at s = -K_I/K_P so that the transient due to the real pole is negligible. As another example, when $K^P = 0.08$ and K_I, = 0.08 the closed-loop transfer function of the compensated system is

$$\frac{\Theta_y(s)}{\Theta_r(s)} = \frac{65{,}221.2(s+5)}{(s+5.145)(s+178.03+j178.03)(s+178.03-j178.03)} \tag{11-50}$$

Because the pole at s = 5.145 is very close to the zero at s = -5, the transient response due to this pole is negligible, and the system dynamics are essentially dominated by the two complex poles.

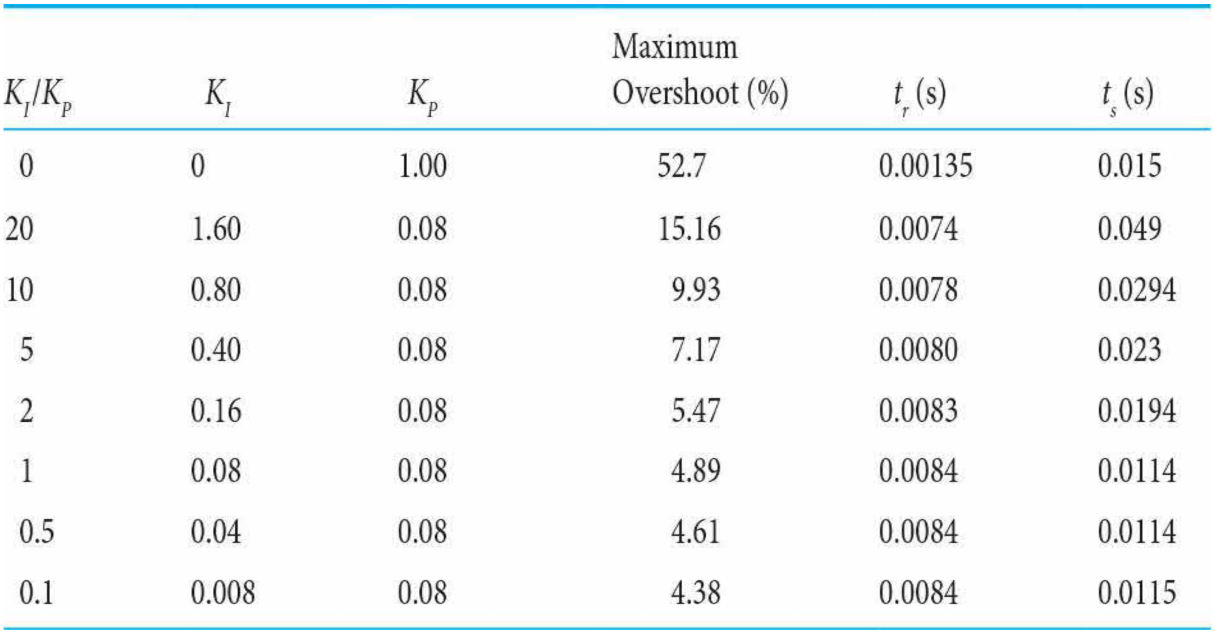
Toolbox 11-3-1

Root loci of Eq. (11-47) in Fig. 11-23 are obtained by the following sequence of MATLAB functions:

```
KP = 0.000001; % start with a very small KP
KI=10*KP;
num = [KP KI];
den = [1 361.2 815265*KP 815265*KI];
G=tf(num,den)
rlocus(G)
```

Table 11-6 gives the attributes of the unit-step responses of the system with PI control for various values of K_I/K_P, with $K_P = 0.08$, which corresponds to a relative damping ratio of 0.707.

Table 11-6 Attributes of the Unit-Step Responses of the System in Example 11-3-1 with PI Controller



K_I/K_P	K_I	K_P	Maximum Overshoot (%)	t_r (s)	t_s (s)
0	0	1.00	52.7	0.00135	0.015
20	1.60	0.08	15.16	0.0074	0.049
10	0.80	0.08	9.93	0.0078	0.0294
5	0.40	0.08	7.17	0.0080	0.023
2	0.16	0.08	5.47	0.0083	0.0194
1	0.08	0.08	4.89	0.0084	0.0114
0.5	0.04	0.08	4.61	0.0084	0.0114
0.1	0.008	0.08	4.38	0.0084	0.0115

The results in Table 11-6 verify the fact that PI control reduces the overshoot but at the expense of longer rise time. For $K_P = 0.08$, the settling times in Table 11-6 actually show a sharp reduction, which is misleading. This is because the settling times for these cases are measured at the points where the response enters the band between 0.95 and 1.00, since the maximum overshoots are less than 5 percent.

The maximum overshoot of the system can still be reduced further than those shown in Table 11-6 by using smaller values of K_P than 0.08. However, the rise time and settling time will be excessive. For example, with $K_P = 0.04$ and $K_I = 0.04$, the maximum overshoot is 1.1 percent, but the rise time is increased to 0.0182 s, and the settling time is 0.024 s.

For the system considered, improvement on the maximum overshoot slows the response down, for K_I less than 0.08, unless K_P is also reduced. As mentioned earlier, the value of the capacitor C_2, in Fig. 11-20, is inversely proportional to K_I. **Thus, for practical reasons, there is a lower limit on the**

value of K_I.

Figure 11-24 shows the unit-step responses of the attitude-control system with PI control, with K^P = 0.08 and several values of K_p. The unit-step response of the same system with the PD controller designed in Example 11-2-1, with K^P = 1 and K_D = 0.00177, is also plotted in the same figure as a comparison. You can obtain these results by modifying Toolbox 11-2-1.

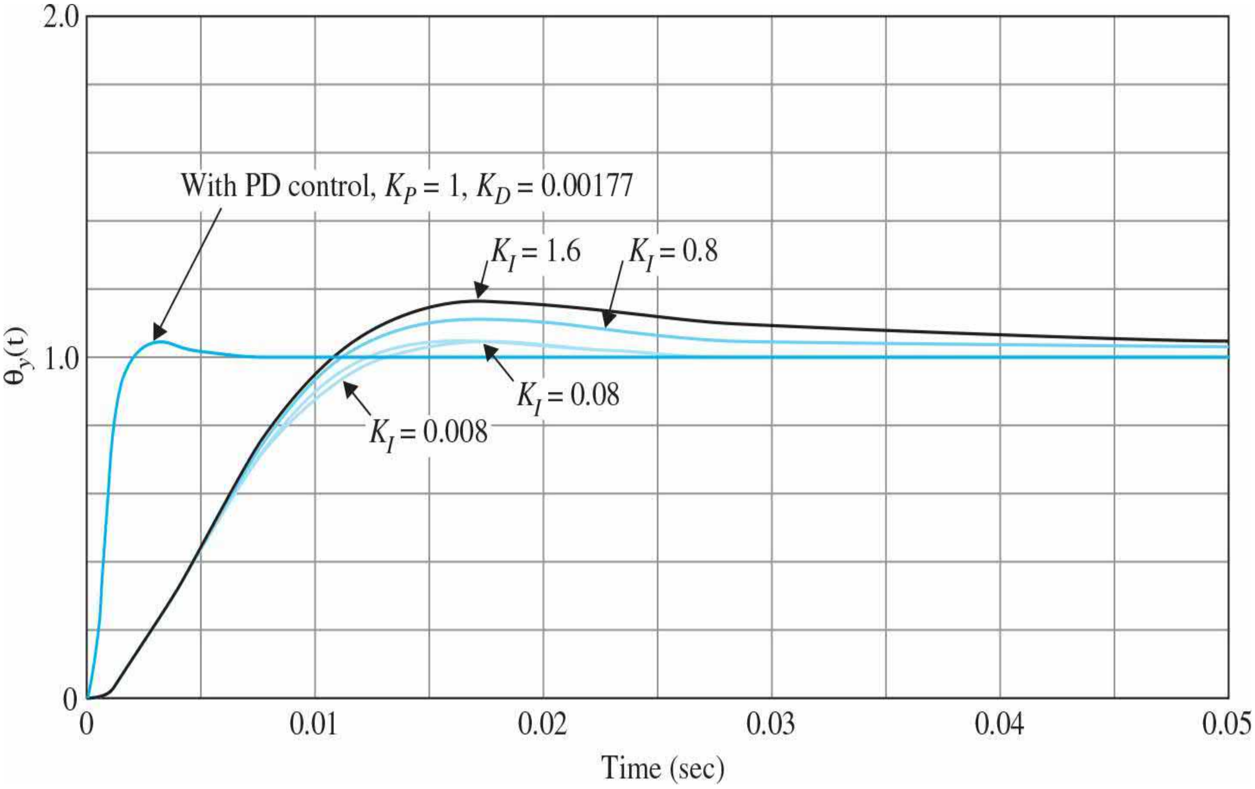
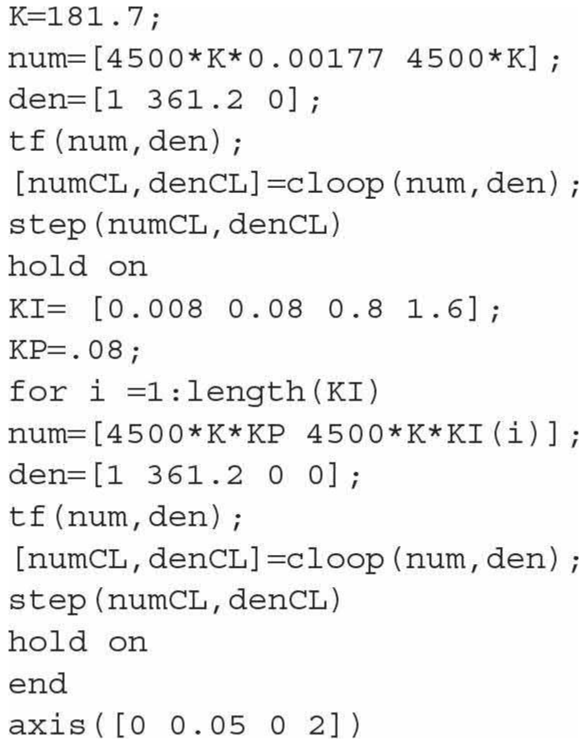


Figure 11-24 Unit-step responses of the system in Example 11-3-1 with PI control, compared to unit-step response of the system in Example 11-2-1 with a PD controller.

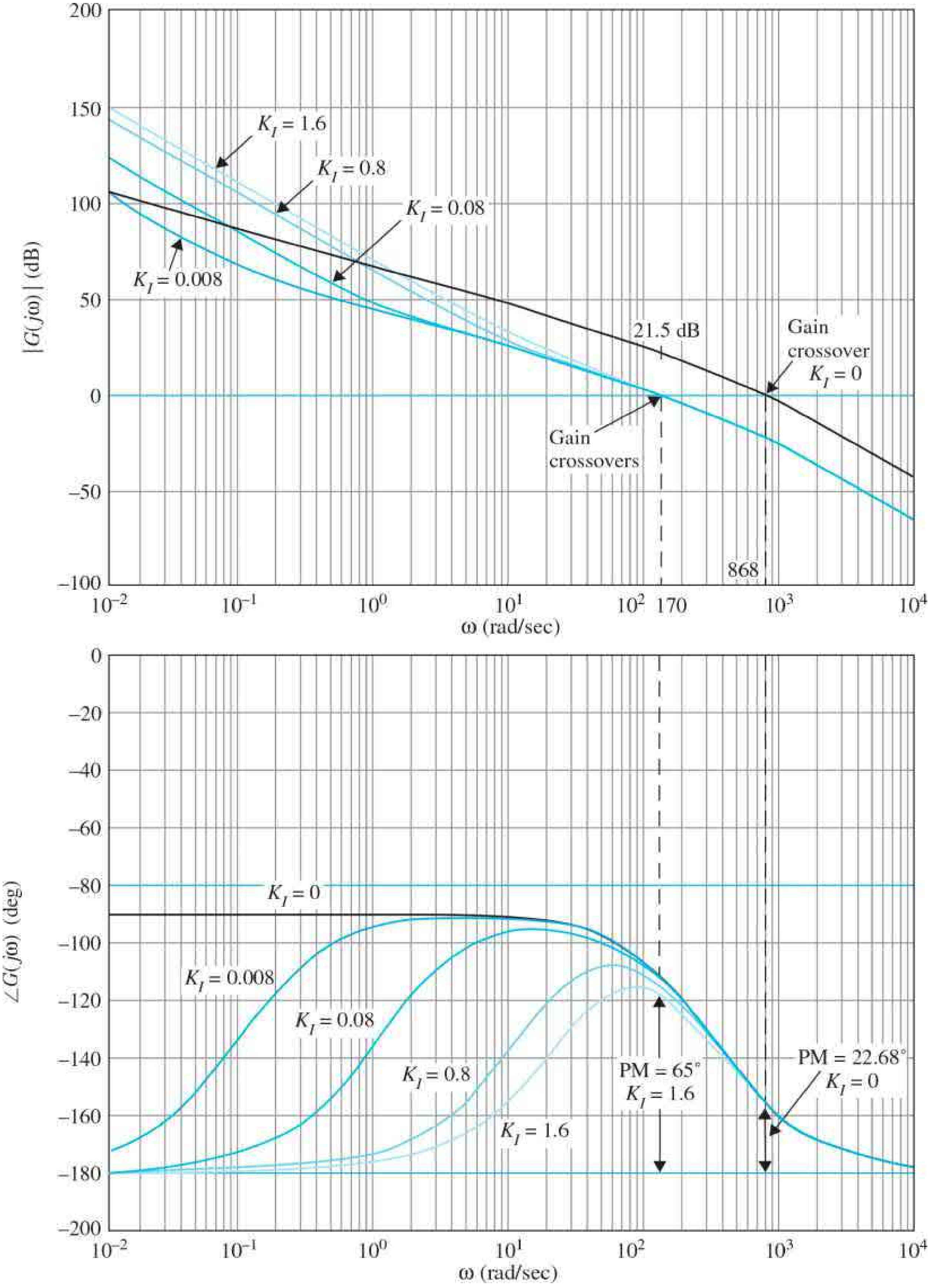
Toolbox 11-3-2

Figure 11-24 is obtained by the following sequence of MATLAB functions:



```
K=181.7;
num=[4500*K*0.00177 4500*K];
den=[1 361.2 0];
tf(num,den);
[numCL,denCL]=cloop(num,den);
step(numCL,denCL)
hold on
KI= [0.008 0.08 0.8 1.6];
KP=.08;
for i =1:length(KI)
num=[4500*K*KP 4500*K*KI(i)];
den=[1 361.2 0 0];
tf(num,den);
[numCL,denCL]=cloop(num,den);
step(numCL,denCL)
hold on
end
axis([0 0.05 0 2])
```

Frequency-Domain Design The forward-path transfer function of the uncompensated system is obtained by setting $K_P = 1$ and $K_P = 0$ in the $G(s)$ in Eq. (11-44), and the Bode plot is shown in Fig. 11-25. The phase margin is 22.68°, and the gain-crossover frequency is 868 rad/s.

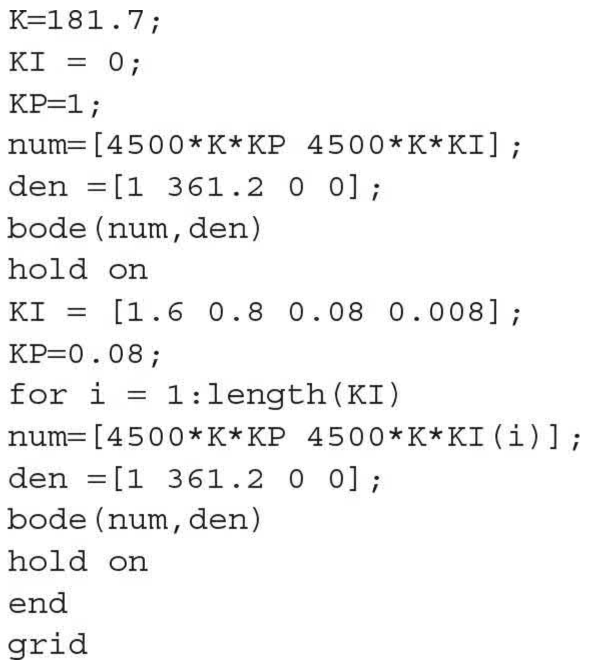


200
150
100
50
0
−50
−100
$|G(j\omega)|$ (dB)
$K_I = 1.6$
$K_I = 0.8$
$K_I = 0.08$
$K_I = 0.008$
21.5 dB
Gain crossover $K_I = 0$
Gain crossovers
868
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
170
10^{3}
10^{4}
ω (rad/sec)
0
−20
−40
−60
−80
−100
−120
−140
−160
−180
−200
$\angle G(j\omega)$ (deg)
$K_I = 0$
$K_I = 0.008$
$K_I = 0.08$
$K_I = 0.8$
$K_I = 1.6$
PM = 65° $K_I = 1.6$
PM = 22.68° $K_I = 0$
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
ω (rad/sec)

Figure 11-25 Bode plots of the control system in Example 11-3-1 with PI controller. $G(s)=\dfrac{815{,}265K_P(s+K_I/K_P)}{s^2(s+361.2)}$.

Toolbox 11-3-3

Bode plots of the control system in Example 11-3-1—Fig. 11-25 is obtained by the following sequence of MATLAB functions:



```
K=181.7;
KI = 0;
KP=1;
num=[4500*K*KP 4500*K*KI];
den =[1 361.2 0 0];
bode(num,den)
hold on
KI = [1.6 0.8 0.08 0.008];
KP=0.08;
for i = 1:length(KI)
num=[4500*K*KP 4500*K*KI(i)];
den =[1 361.2 0 0];
bode(num,den)
hold on
end
grid
```

Let us specify that the required phase margin should be at least 65°, and this is to be achieved with the PI controller of Eq. (11-40). Following the procedure outlined earlier in Eqs. (11-41) through (11-43) on the design of the PI controller, we conduct the following steps:

1. Look for the new gain-crossover frequency $\boldsymbol{\omega}'_g$ at which the phase margin of 65° is realized. From Fig. 11-25, $\boldsymbol{\omega}'_g$ is found to be 170 rad/s. The magnitude of $G(j\omega)$ at this frequency is 21.5 dB. Thus, the PI controller should provide an attenuation of -21.5 dB at $\boldsymbol{\omega}'_g$ = 170 rad/s Substituting $|G(j\boldsymbol{\omega}'_g)|$ into Eq. (11-42), and solving for K_P, we get

$$K_P = 10^{-|G(j\omega'_g)|_{\text{dB}}/20} = 10^{-21.5/20} = 0.084 \qquad (11\text{-}51)$$

Notice that, in the time-domain design conducted earlier, K_P was selected to be 0.08 so that the relative damping ratio of the complex characteristic equation roots will be approximately 0.707. (Note that in this case, for the sake of comparison with the time domain response, we have cheated a little by selecting the desired phase margin to be PM = 65°.)

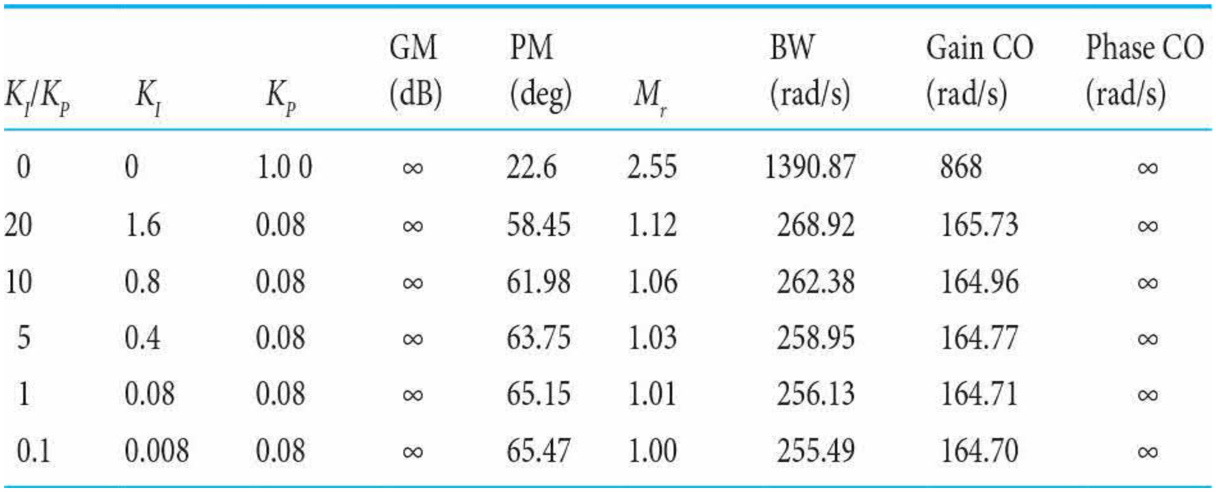
2. Let us choose K_P = 0.08, so that we can compare the design results of the frequency domain with those of the time-domain design obtained earlier. Equation (11-43) gives the general guideline of finding K_I once K_P is determined. Thus,

$$K_I = \frac{\omega'_g K_P}{10} = \frac{170 \times 0.08}{10} = 1.36 \qquad (11\text{-}52)$$

As pointed out earlier, the value of K_I is not rigid, as long as the ratio K_I/K_P is sufficiently smaller than the magnitude of the pole of $G(s)$ at −361.2. As it turns out, the value of K_I given by Eq. (11-52) is not sufficiently small for this system.

The Bode plots of the forward-path transfer function with K_P = 0.08 and K_I = 0, 0.008, 0.008, 0.08, 0.8, and 1.6 are shown in Fig. 11-25. Table 11-7 shows the frequency-domain properties of the uncompensated system and the compensated system with various values of K_I. Notice that, for values of K_I/K_P that are sufficiently small, the phase margin, M_r, BW, and gain-crossover (CO) frequency all vary a little.

Table 11-7 Frequency-Domain Performance Data of the System in Example 11-3-1 with PI Controller



K_I/K_P	K_I	K_P	GM (dB)	PM (deg)	M_r	BW (rad/s)	Gain CO (rad/s)	Phase CO (rad/s)
0	0	1.0 0	∞	22.6	2.55	1390.87	868	∞
20	1.6	0.08	∞	58.45	1.12	268.92	165.73	∞
10	0.8	0.08	∞	61.98	1.06	262.38	164.96	∞
5	0.4	0.08	∞	63.75	1.03	258.95	164.77	∞
1	0.08	0.08	∞	65.15	1.01	256.13	164.71	∞
0.1	0.008	0.08	∞	65.47	1.00	255.49	164.70	∞

It should be noted that the phase margin of the system can be improved further by reducing the value of K_P below 0.08. However, the bandwidth of the system will be further reduced. For example, for K_P = 0.04 and K_I = 0.04, the phase margin is increased to 75.7°, and M_r = 1.01, but BW is reduced to 117.3 rad/s. ▲

EXAMPLE 11-3-2 Control of a DC Motor: Electrical Time Constant Not Neglected

Now let us consider using the PI control for the third-order attitude control system described by Eq. (11-19). First, the time-domain design is carried out as follows.

Time-Domain Design Let the time-domain specifications be as follows:

Steady-state error due to the parabolic input $t^2u_s(t)/2 \leq 0.2$

Maximum overshoot ≤5%

Rise time $t_r \leq 0.01$ s

Settling time $t_s \leq 0.02$ s

These are identical to the specifications given for the second-order system in Example 11-3-1.

Applying the **PI** controller of Eq. (11-24), the forward-path transfer function of the system becomes

$$G(s)=G_c(s)G_p(s)=\frac{1.5\times10^7 KK_P(s+K_I/K_P)}{s^2(s^2+3408.3s+1{,}204{,}000)}$$

$$=\frac{1.5\times10^7 KK_P(s+K_I/K_P)}{s^2(s+400.26)(s+3008)} \tag{11-53}$$

We can show that the steady-state error of the system due to the parabolic input is again given by Eq. (11-46), and, arbitrarily setting $K = 181.17$, the minimum value of K_I is 0.002215.

The characteristic equation of the closed-loop system with $K = 181.17$ is

$$s^4+3408.3s^3+1{,}204{,}000s^2+2.718\times10^9 K_P s+2.718\times10^9 K_I=0 \tag{11-54}$$

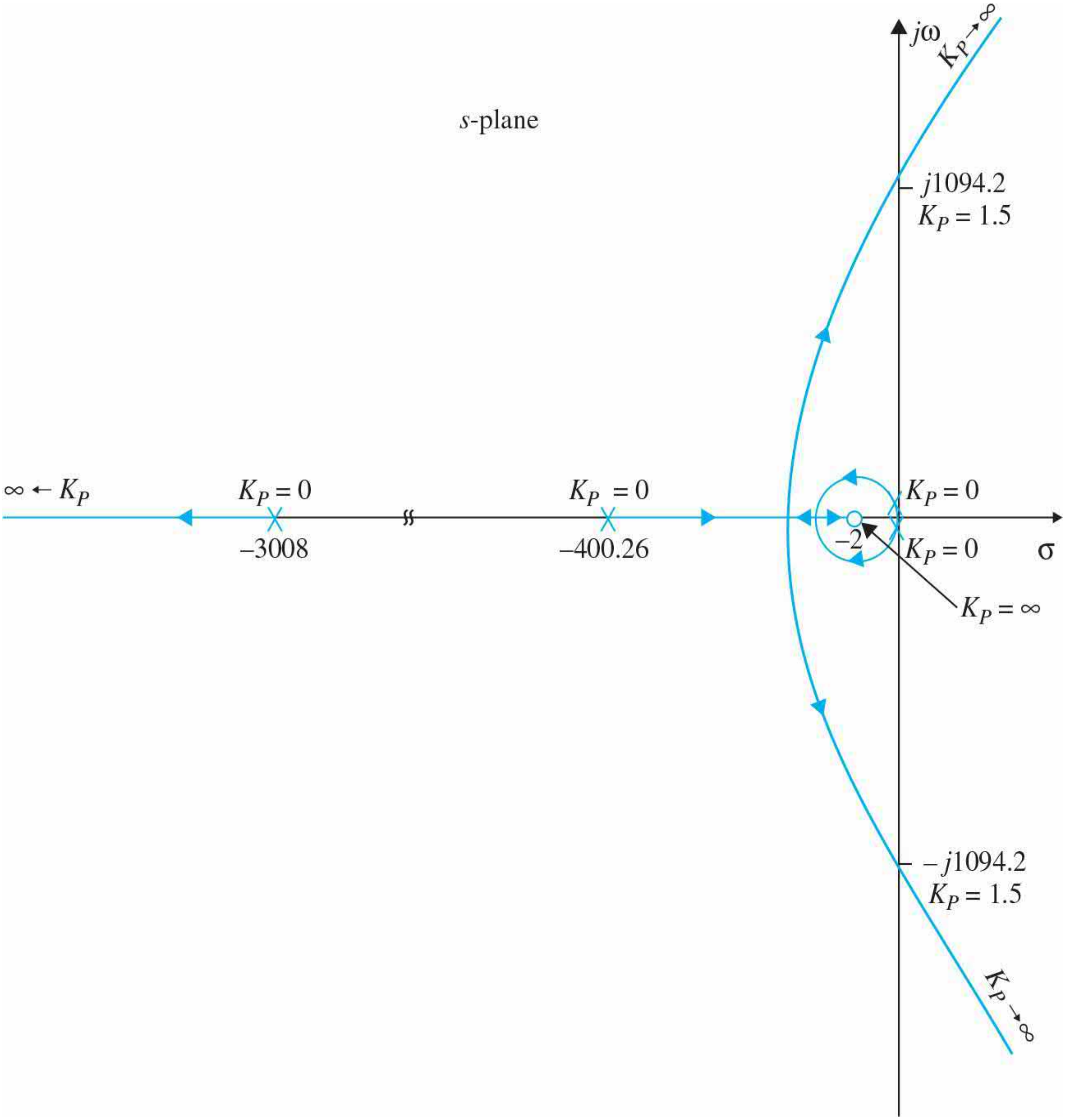
The Routh's tabulation of the last equation is performed as follows:

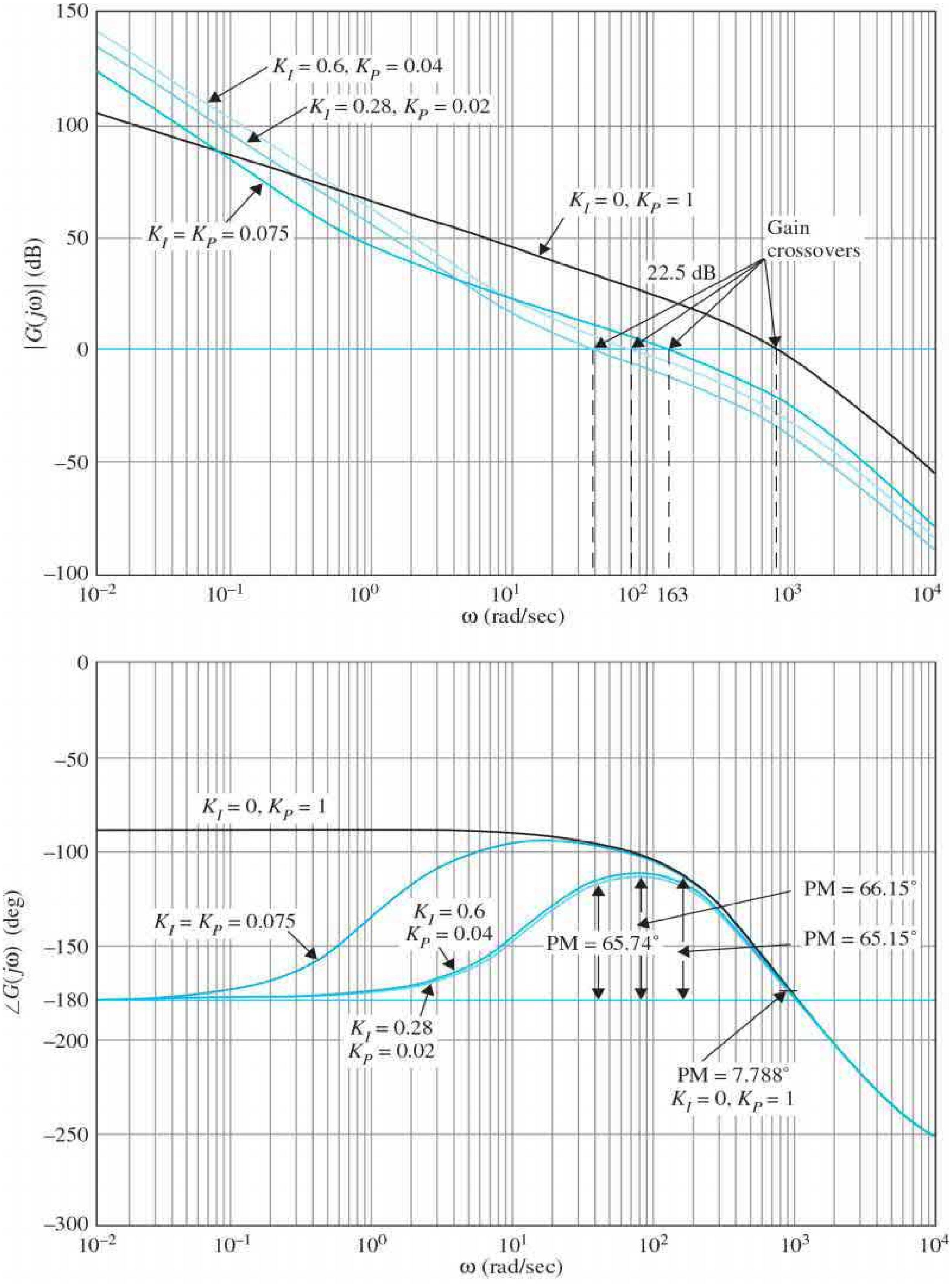
s^4	1	$1{,}204{,}000$	$2.718\times10^9 K_I$
s^3	3408.3	$2.718\times10^9 K_P$	0
s^2	$1{,}204{,}000-797465K_P$	$2.718\times10^9 K_I$	0
s^1	$\dfrac{1{,}204{,}000K_P-797465K_P^2-3408.3K_I}{1{,}204{,}000-797465K_P}$	0	
s^0	$2.718\times10^9 K_I$	0	

The stability requirements are

$$\begin{gathered} K_I>0 \\ K_P<1.5098 \\ K_I<353.255K_P-233.98K_P^2 \end{gathered} \tag{11-55}$$

The design of the PI controller calls for the selection of a small value for K_I/K_P, relative to the nearest pole of $G(s)$ to the origin, which is at −400.26. The root loci of Eq. (11-54) are plotted using the pole–zero configuration of Eq. (11-53). Figure 11-26*a* shows the root loci as K_P varies for $K_I/K_P = 2$. The root loci near the origin due to the pole and zero of the PI controller again form a small loop, and the root loci at a distance away from the origin will be very similar to those of the uncompensated system, which are shown in Fig. 7-53. By selecting the value of K_P properly along the root loci, it may be possible to satisfy the performance specifications given above. To minimize the rise time and settling time, we should select K_P so that the dominant roots are complex conjugate. Table 11-8 gives the performance attributes of several combinations of K_I/K_P and K_P. Notice that, although several combinations of these parameters correspond to systems that satisfy the performance specifications, the one with $K_P = 0.075$ and $K_I = 0.075$ gives the best rise and settling times among those shown.

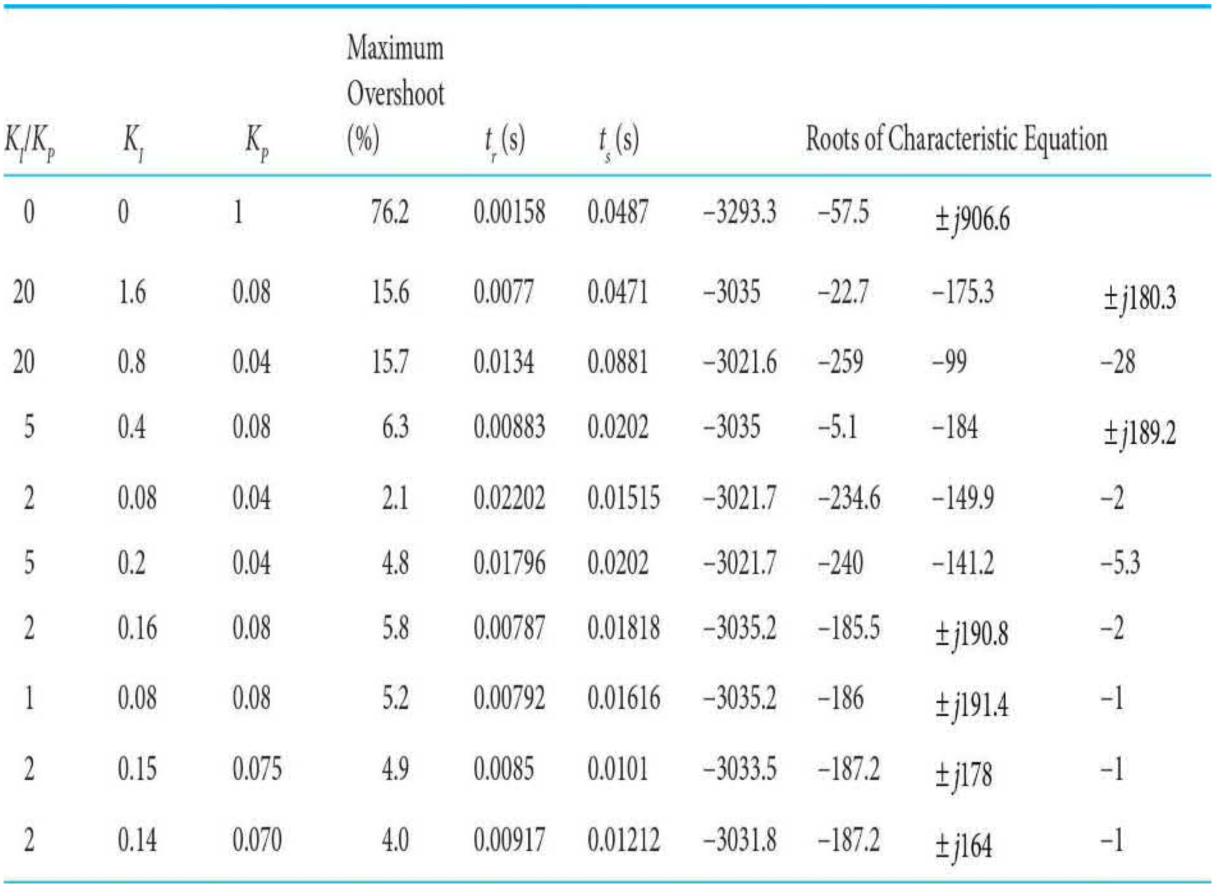



150
100
50
0
−50
−100
$|G(j\omega)|$ (dB)
$K_I = 0.6,\ K_P = 0.04$
$K_I = 0.28,\ K_P = 0.02$
$K_I = K_P = 0.075$
$K_I = 0,\ K_P = 1$
Gain
crossovers
22.5 dB
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
163
10^{3}
10^{4}
ω (rad/sec)
0
−50
−100
−150
−180
−200
−250
−300
$\angle G(j\omega)$ (deg)
$K_I = 0,\ K_P = 1$
PM = 66.15°
$K_I = K_P = 0.075$
$K_I = 0.6$
$K_P = 0.04$
PM = 65.74°
PM = 65.15°
$K_I = 0.28$
$K_P = 0.02$
PM = 7.788°
$K_I = 0,\ K_P = 1$
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
ω (rad/sec)

(b)

Figure 11-26 (a) Root loci of the control system in Example 11-3-2 with PI controller $K_I/K_P = 2, 0 \le K_P < \infty$. (b) Bode plots of the control system in Example 11-3-2 with PI control.

Table 11-8 Attributes of the Unit-Step Responses of the System in Example 11-3-2 with PI Controller



K_I/K_P	K_I	K_P	Maximum Overshoot (%)	t_r (s)	t_s (s)	Roots of Characteristic Equation			
0	0	1	76.2	0.00158	0.0487	−3293.3	−57.5	$\pm j906.6$	
20	1.6	0.08	15.6	0.0077	0.0471	−3035	−22.7	−175.3	$\pm j180.3$
20	0.8	0.04	15.7	0.0134	0.0881	−3021.6	−259	−99	−28
5	0.4	0.08	6.3	0.00883	0.0202	−3035	−5.1	−184	$\pm j189.2$
2	0.08	0.04	2.1	0.02202	0.01515	−3021.7	−234.6	−149.9	−2
5	0.2	0.04	4.8	0.01796	0.0202	−3021.7	−240	−141.2	−5.3
2	0.16	0.08	5.8	0.00787	0.01818	−3035.2	−185.5	$\pm j190.8$	−2
1	0.08	0.08	5.2	0.00792	0.01616	−3035.2	−186	$\pm j191.4$	−1
2	0.15	0.075	4.9	0.0085	0.0101	−3033.5	−187.2	$\pm j178$	−1
2	0.14	0.070	4.0	0.00917	0.01212	−3031.8	−187.2	$\pm j164$	−1

Frequency-Domain Design The Bode plot of Eq. (11-53) for K = 181.17, K_P = 1, and K_I = 0 is shown in Fig. 11-26*b*. The performance data of the uncompensated system are as follows:

Gainmargin = 3.578 dB

Phasemargin = 7.788°

M_r. = 6.572

BW = 1378rad/s

Let us require that the compensated system has a phase margin of at least 65°, and this is to be achieved with the PI controller of Eq. (11-40). Following the procedure outlined in Eqs. (11-41) through (11-43) on the design of the PI controller, we carry out the following steps.

1. Look for the new gain-crossover frequency $\boldsymbol{\omega}'_g$ at which the phase margin of 65° is realized. From Fig. 11-20, $\boldsymbol{\omega}'_g$ is found to be 163 rad/s, and the magnitude of $G(j\omega)$ at this frequency is 22.5 dB. Thus, the PI controller should provide an attenuation of −22.5 dB at $\boldsymbol{\omega}'_g$ = 163 rad/s. Substituting $|G(j\omega'_g)|$ into Eq. (11-42), and solving for K_P, we get

$$K_P = 10^{-|G(j\omega'_g)|_{\text{dB}}/20} = 10^{-22.5/20} = 0.075 \tag{11-56}$$

This is exactly the same result that was selected for the time-domain design that resulted in a system with a maximum overshoot of 4.9 percent when K_I = 0.15, or K_I/K_2 = 2.

2. The suggested value of K_I is found from Eq. (11-43):

$$K_I = \frac{\omega'_g K_P}{10} = \frac{163 \times 0.075}{10} = 1.222 \tag{11-57}$$

Toolbox 11-3-4

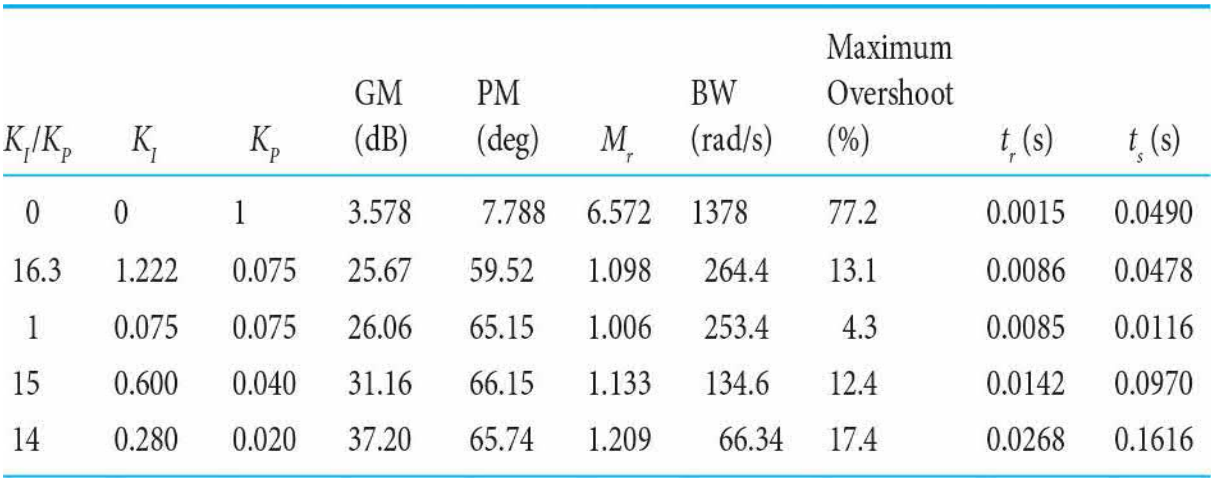
Bode plots of the control system in Fig. 11-26b are obtained by the following sequence of MATLAB functions:

```
KI = [1.222 0.6 0.28 0.075 0];
KP = [0.075 0.04 0.02 0.075 1];
for i = 1:length(KI)
num = [1.5e7*181.17*KP(i) 1.5e7*181.17*KI(i)];
den =[1 3408.3 1204000 0 0];
bode(num,den)
hold on
end
grid
axis([0.01 10000 -270 0]);
```

Thus, K_I/K_P. However, the phase margin of the system with these design parameters is only 59.52.

To realize the desired PM of 65°, we can reduce the value of K_P or K_I. Table 11-9 gives the results of several designs with various combinations of K_P and K_I. Notice that the last three designs in the table all satisfy the PM requirements. However, the design ramifications show the following:

Table 11-9 Performance Summary of the System in Example 11-3-2 with PI Controller



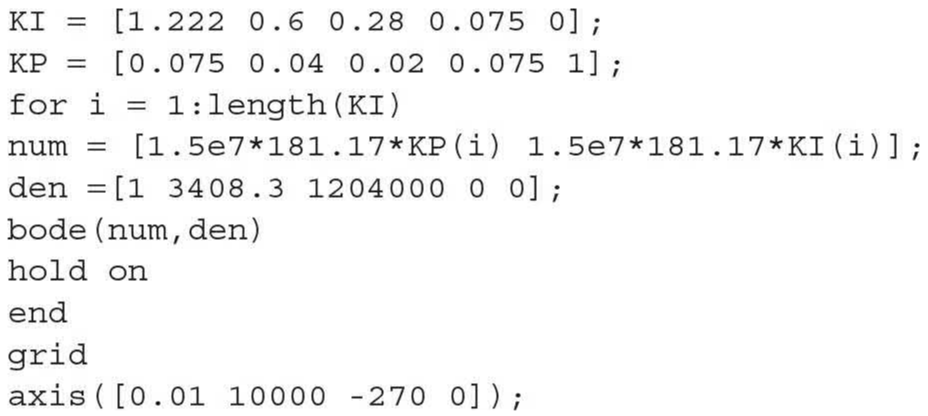
K_I/K_P	K_I	K_P	GM (dB)	PM (deg)	M_r	BW (rad/s)	Maximum Overshoot (%)	t_r (s)	t_s (s)
0	0	1	3.578	7.788	6.572	1378	77.2	0.0015	0.0490
16.3	1.222	0.075	25.67	59.52	1.098	264.4	13.1	0.0086	0.0478
1	0.075	0.075	26.06	65.15	1.006	253.4	4.3	0.0085	0.0116
15	0.600	0.040	31.16	66.15	1.133	134.6	12.4	0.0142	0.0970
14	0.280	0.020	37.20	65.74	1.209	66.34	17.4	0.0268	0.1616

Reducing K_P would reduce BW and increase M_r.

Reducing K_I would increase the capacitor value in the implementing circuit.

Toolbox 11-3-5

Figure 11-27 is obtained by the following sequence of MATLAB functions:



```
KI = [1.222 0.6 0.28 0.075 0];
KP = [0.075 0.04 0.02 0.075 1];
for i = 1:length(KI)
num = [1.5e7*181.17*KP(i) 1.5e7*181.17*KI(i)];
den =[1 3408.3 1204000 0 0];
bode(num,den)
hold on
end
grid
axis([0.01 10000 -270 0]);
```

In fact, only the $K_I = K_P = 0.075$ case gives the best all-around performance in both the frequency domain and the time domain. In attempting to increase K_I, the maximum overshoot becomes excessive. This is one example that shows the inadequacy of specifying phase margin only. The purpose of this example is to bring out the properties of the PI controller and the important considerations in its design. No details are explored further.

Figure 11-27 shows the unit-step responses of the uncompensated system and several systems with PI control. ▲

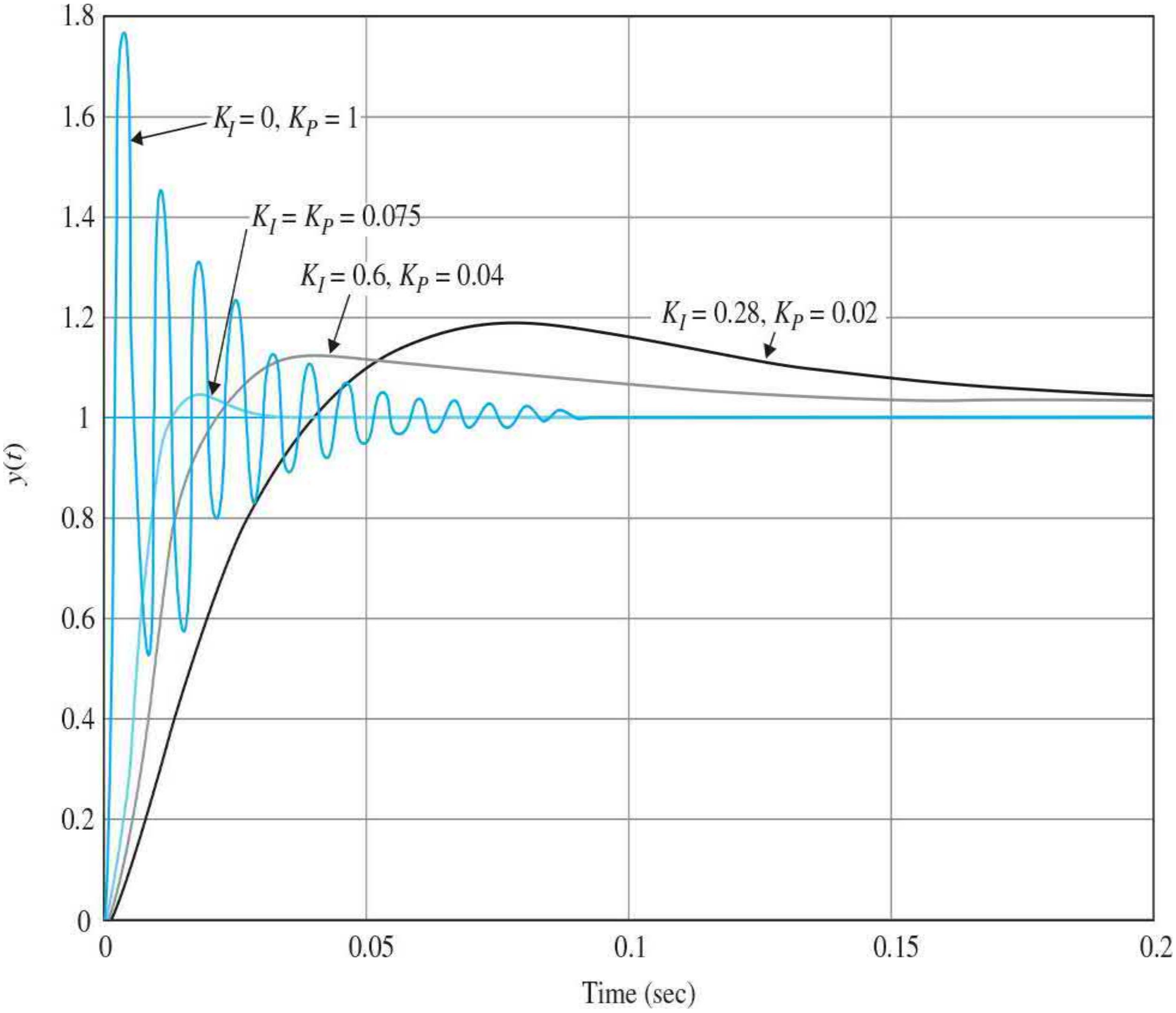


Figure 11-27 Unit-step response of system with PI controller in Example 11-3-2.

11-4 DESIGN WITH THE PID CONTROLLER

PD controllers enhance damping of a system, but the steady-state response is not affected.

PI controllers improve the relative stability and improve the steady-state error at the same time, but the rise time is increased.

PID controllers combine the features of PD and PI controllers.

From the preceding discussions, we see that the PD controller could add damping to a system, but the steady-state response is not affected. The PI controller could improve the relative stability and improve the steady-state error at the same time, but the rise time is increased. This leads to the motivation of using a PID controller so that the best features of each of the PI and PD controllers are utilized. We can outline the following procedure for the design of the PID controller.

1. Consider that the PID controller consists of a PI portion connected in cascade with a PD portion. The transfer function of the PID controller is written as

$$G_c(s) = K_P + K_D s + \frac{K_I}{s} = (K_{P1} + K_{D1}s)\left(K_{P2} + \frac{K_{I2}}{s}\right) \tag{11-58}$$

The proportional constant of the PD portion is set to unity, $K_{P1} = 1$, since we need only three parameters in the PID controller. Equating both sides of Eq. (11-58), we have

$$K_P = K_{P2} + K_{D1}K_{I2} \tag{11-59}$$

$$K_D = K_{D1}K_{P2} \tag{11-60}$$

$$K_I = K_{I2} \tag{11-61}$$

2. Consider that the PD portion only is in effect. Select the value of K_{D1} so that a portion of the desired relative stability is achieved. In the time domain, this relative stability may be measured by the maximum overshoot, and in the frequency domain it is the phase margin.
3. Select the parameters K_{I2} and K_{P2} so that the total requirement on relative stability is satisfied.

Note that setting $K_{PI} = 1$ is in line with our PD controller design in Examples 11-2-1 and 11-2-2. The following example illustrates how the PID controller is designed in the time domain and the frequency domain.

EXAMPLE 11-4-1 Control of a DC Motor: Small Time-Constant Model

Consider the third-order attitude control system represented by the forward-path transfer function given in Eq. (11-29). With K = 181.17, the transfer function is

$$G_p(s) = \frac{2.718 \times 10^9}{s(s+400.26)(s+3008)} \tag{11-62}$$

Time-Domain Design Let the time-domain performance specifications be as follows:

Steady-state error due to a ramp input $t^2u_s(t)/2 \le 0.2$

Maximum overshoot ≤5%

Rise time t_r ≤0.005 s

Settling time t_s ≤0.0005 s

We realize from the previous examples that these requirements cannot be fulfilled by either the PI or PD control acting alone. Let us apply the PD control with the transfer function (1+K_{DI}s). The forward-path transfer function becomes

$$G(s) = \frac{2.718 \times 10^9 (1+K_{D1}s)}{s(s+400.26)(s+3008)} \tag{11-63}$$

Table 11-4 shows that the best PD controller that can be obtained from the maximum overshoot standpoint is with K_{PI} = 1 and K_{PI} = 0.002, and the maximum overshoot is 11.37 percent. The rise time and settling time are well within the required values. Next, we add the PI controller, and the forward-path transfer function becomes

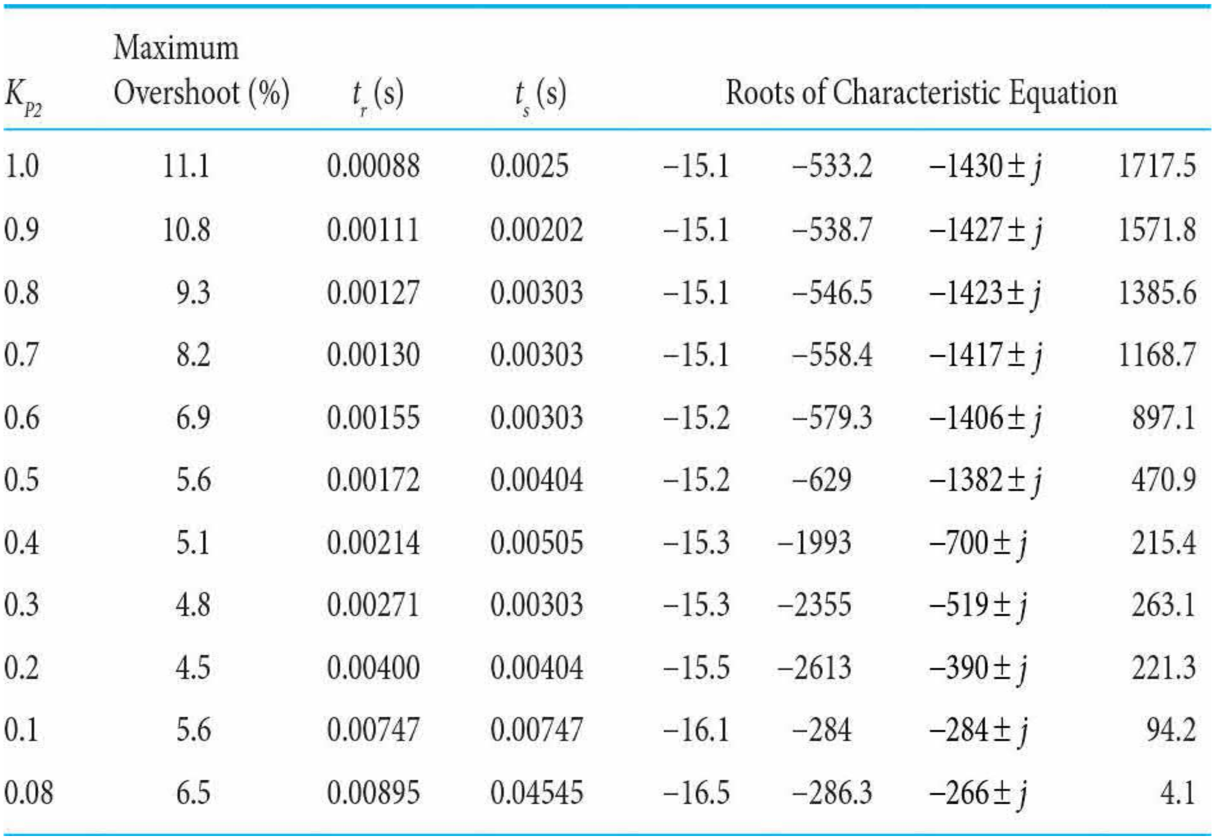
$$G(s) = \frac{5.436 \times 10^6 K_{P2}(s+500)(s+K_{I2}/K_{P2})}{s^2(s+400.26)(s+3008)} \tag{11-64}$$

Following the guideline of choosing a relatively small value for K_{I2}/K_{P2} (see Examples 11-3-1 and 11-3-2), we let K_{I2}/K_{P2} =15. Equation (11-64) becomes

$$G(s) = \frac{5.436 \times 10^6 K_{P2}(s+500)(s+15)}{s^2(s+400.26)(s+3008)} \tag{11-65}$$

Table 11-10 gives the time-domain performance characteristics along with the roots of the characteristic equation for various values of K_{P2}. Apparently, the optimal value of K_{P2} is in the neighborhood of between 0.2 and 0.4.

Table 11-10 Time-Domain Performance Characteristics of Third-Order Attitude Control System with PID Controller Designed in Example 11-4-1



K_{P2}	Maximum Overshoot (%)	t_r (s)	t_s (s)	Roots of Characteristic Equation			
1.0	11.1	0.00088	0.0025	−15.1	−533.2	−1430 ± j	1717.5
0.9	10.8	0.00111	0.00202	−15.1	−538.7	−1427 ± j	1571.8
0.8	9.3	0.00127	0.00303	−15.1	−546.5	−1423 ± j	1385.6
0.7	8.2	0.00130	0.00303	−15.1	−558.4	−1417 ± j	1168.7
0.6	6.9	0.00155	0.00303	−15.2	−579.3	−1406 ± j	897.1
0.5	5.6	0.00172	0.00404	−15.2	−629	−1382 ± j	470.9
0.4	5.1	0.00214	0.00505	−15.3	−1993	−700 ± j	215.4
0.3	4.8	0.00271	0.00303	−15.3	−2355	−519 ± j	263.1
0.2	4.5	0.00400	0.00404	−15.5	−2613	−390 ± j	221.3
0.1	5.6	0.00747	0.00747	−16.1	−284	−284 ± j	94.2
0.08	6.5	0.00895	0.04545	−16.5	−286.3	−266 ± j	4.1

Selecting K_{P2}=0.3, and with K_{D1}=0.002 and K_{I2} = 15K_{P2}=4.5, the following results are obtained for the parameters of the PID controller using Eqs. (11-59) through (11-61):

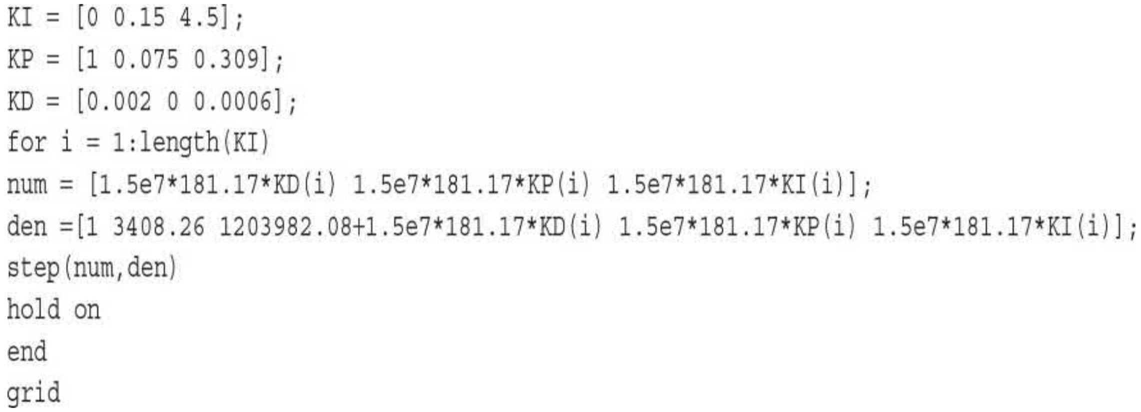
$$
\begin{aligned}
K_I &= K_{I2} = 4.5 \\
K_P &= K_{P2} + K_{D1}K_{I2} = 0.3 + 0.002 \times 4.5 = 0.309 \qquad (11\text{-}66) \\
K_D &= K_{D1}K_{P2} = 0.002 \times 0.3 = 0.0006
\end{aligned}
$$

Notice that the PID design resulted in a smaller K_D and a larger K_I, which correspond to smaller capacitors in the implementing circuit.

Figure 11-28 shows the unit-step responses of the system with the PID controller, as well as those with PD and PI controls designed in Examples 11-2-2 and 11-3-2, respectively. Notice that the PID control, when designed properly, captures the advantages of both the PD and the PI controls.

Toolbox 11-4-1

Figure 11-28 is obtained by the following sequence of MATLAB functions:



```
KI = [0 0.15 4.5];
KP = [1 0.075 0.309];
KD = [0.002 0 0.0006];
for i = 1:length(KI)
num = [1.5e7*181.17*KD(i) 1.5e7*181.17*KP(i) 1.5e7*181.17*KI(i)];
den =[1 3408.26 1203982.08+1.5e7*181.17*KD(i) 1.5e7*181.17*KP(i) 1.5e7*181.17*KI(i)];
step(num,den)
hold on
end
grid
```

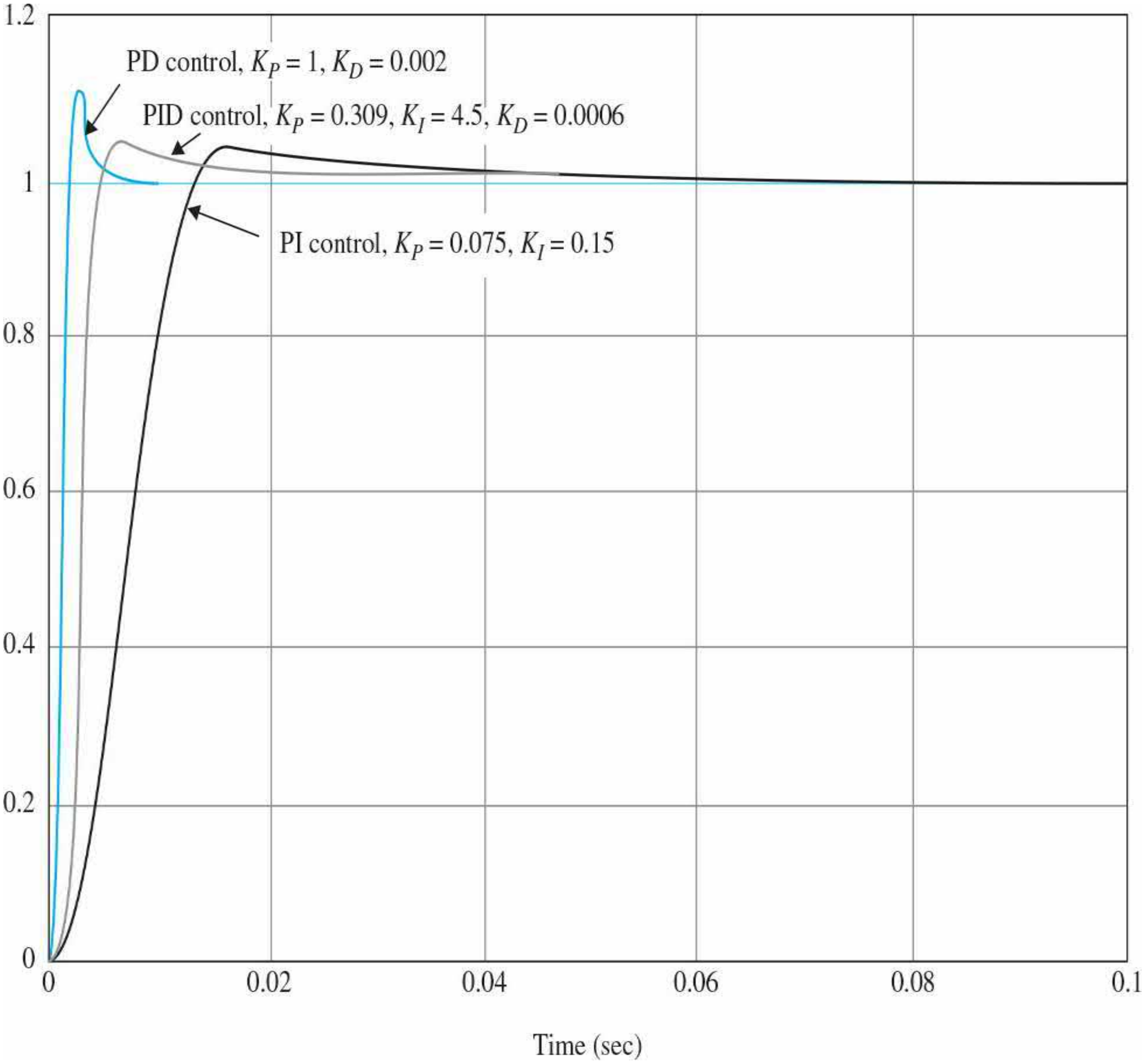


Figure 11-28 Step responses of the system in Example 11-4-1 with PD, PI, and PID controllers.

Frequency-Domain Design The PD control of the third-order attitude control systems was already carried out in Example 11-2-2, and the results were tabulated in Table 11-4. When $K_P = 1$ and $K_D = 0.002$, the maximum overshoot is 11.37 percent, but this is the best that the PD control could offer. Using this PD controller, the forward-path transfer function of the system is

$$G(s) = \frac{2.718 \times 10^9 (1 + 0.002s)}{s(s + 400.26)(s + 3008)} \qquad (11\text{-}67)$$

and its Bode plot is shown in Fig. 11-29. Let us estimate that the following set of frequency-domain criteria corresponds to the time-domain specifications given in this problem.

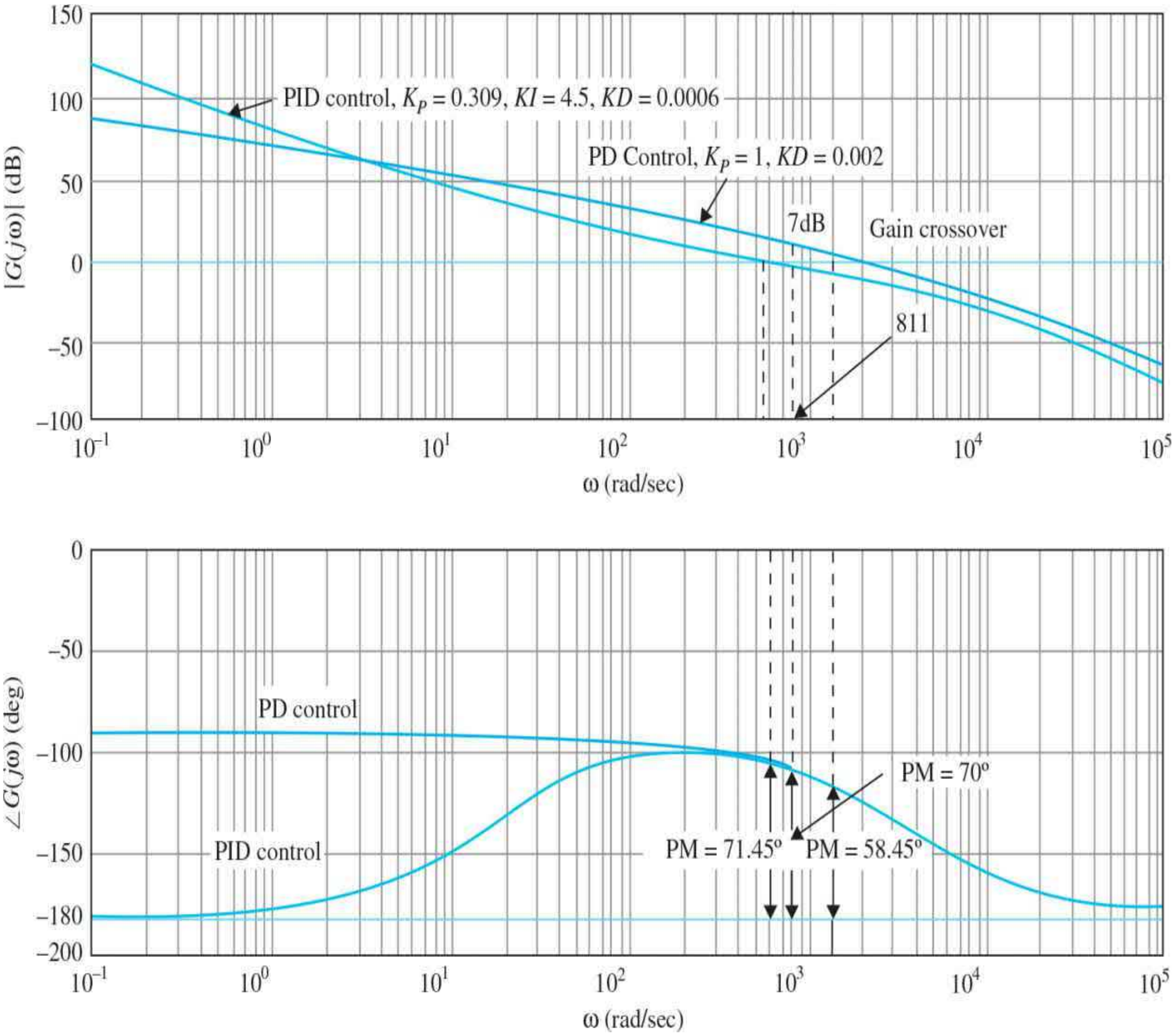


Figure 11-29 Bode plot of the system in Example 11-4-1 with PD and PID controllers.

Phase margin ≥ 70°

M_r ≤1.1

BW ≥1.000 rad/s

From the Bode diagram in Fig. 11-29, we see that, to achieve a phase margin of 70°, the new phase-crossover frequency should be $\boldsymbol{\omega}'_g$=811, at which the magnitude of $G(j\omega)$ is 7 dB. Thus, using Eq. (11-42), the value of K_{P2} is calculated to be

$$K_{P2} = 10^{-7/20} = 0.45 \tag{11-68}$$

Notice that the desirable range of K_{P2} found from the time-domain design with K_{I2}/K_{P2}=15 is from 0.2 to 0.4. The result given in Eq. (11-68) is slightly out of the range. Table 11-11 shows the frequency-domain performance results with K_D=0.002, K_{I2}/K_{P2}=15, and several values of K_{P2} starting with 0.45. It is interesting to note that, as K_{P2} continues to decrease, the phase margin increases monotonically, but below K_{P2}=0.2, the maximum overshoot actually increases. In this case, the phase margin results are misleading, but the resonant peak M_r is a more accurate indication of this. ▲

Table 11-11 Frequency-Domain Performance of System in Example 11-4-1 with PID Controller

K_{P2}	K_{I2}	GM (dB)	PM (deg)	M_r	BW (rad/s)	t_r (s)	t_s (s)	Maximum Overshoot (%)
1.00	0	∞	58.45	1.07	2607	0.0008	0.00255	11.37
0.45	6.75	∞	68.5	1.03	1180	0.0019	0.0040	5.6
0.40	6.00	∞	69.3	1.027	1061	0.0021	0.0050	5.0
0.30	4.50	∞	71.45	1.024	1024	0.0027	0.00303	4.8
0.20	3.00	∞	73.88	1.031	528.8	0.0040	0.00404	4.5
0.10	1.5	∞	76.91	1.054	269.5	0.0076	0.0303	5.6

11-5 DESIGN WITH PHASE-LEAD AND PHASE-LAG CONTROLLERS

The high-pass filter is often referred to as a **phase-lead controller**, because positive phase is introduced to the system over some frequency range.

The low-pass filter is also known as a **phase-lag controller**, because the corresponding phase introduced is negative.

The PID controller and its components in the form of PD and PI controls represent simple forms of controllers that utilize derivative and integration operations in the compensation of control systems. In general, we can regard the design of controllers of control systems as a filter design problem; then there are a large number of possible schemes. From the filtering standpoint, the PD controller is a high-pass filter, the PI controller is a low-pass filter, and the PID controller is a band-pass or band-attenuate filter, depending on the values of the controller parameters. In this section, we introduce the high-pass filter that is often referred to as a **phase-lead controller** because positive phase is introduced to the system over some frequency range. The low-pass filter is also known as a **phase-lag controller** because the corresponding phase introduced is negative. Both cases can be represented by the circuit diagram shown in Fig. 11-30.

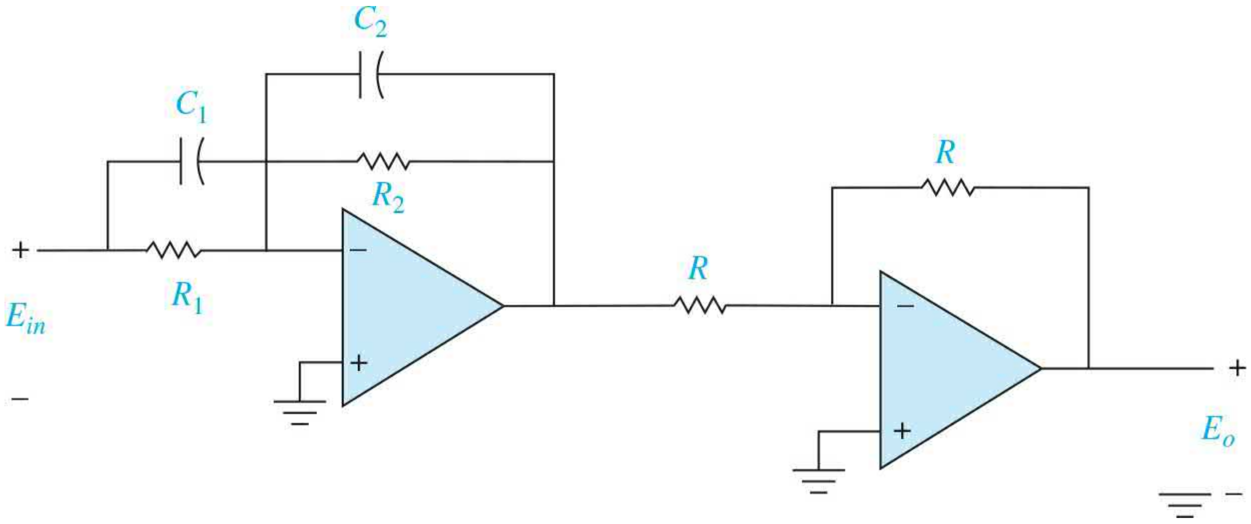


Figure 11-30 Op-amp circuit implementation of $G(s) = K_c \frac{s + z_1}{s + P_1}$.

The transfer function of a simple lead or lag controller is expressed as

PD controller is a high-pass filter.
PI controller is a low-pass filter.

PID controller is a band-pass or band-attenuate filter.

$$G_c(s) = K_c \frac{s+z_1}{s+p_1} \tag{11-69}$$

where the controller is high-pass or phase-lead if $p_1 > z_1$, and low-pass or phase-lag if $p_1 < z_1$.

The op-amp circuit implementation of Eq. (11-69) is given in Table 6-1*g* of Chap. 6 and is repeated in Fig. 11-30 with an inverting amplifier. The transfer function of the circuit is

$$G_c(s) = \frac{E_o(s)}{E_{\text{in}}(s)} = \frac{C_1}{C_2} \frac{s + \dfrac{1}{R_1C_1}}{s + \dfrac{1}{R_2C_2}} \tag{11-70}$$

Comparing the last two equations, we have

$$\begin{aligned} K_c &= C_1/C_2 \\ z_1 &= 1/R_1C_1 \\ p_1 &= 1/R_2C_2 \end{aligned} \tag{11-71}$$

We can reduce the number of design parameters from four to three by setting C = C_1 = C_2. Then Eq. (11-70) is written as

$$\begin{aligned} G_c(s) &= \frac{R_2}{R_1}\left(\frac{1+R_1Cs}{1+R_2Cs}\right) \\ &= \frac{1}{a}\left(\frac{1+aTs}{1+Ts}\right) \end{aligned} \tag{11-72}$$

where

$$a = \frac{R_1}{R_2} \tag{11-73}$$

$$T = R_2 C \qquad (11\text{-}74)$$

11-5-1 Time-Domain Interpretation and Design of Phase-Lead Control

In this section, we consider that Eqs. (11-70) and (11-72) represent a phase-lead controller ($z_1 < P_1$ or a > 1). In order that the phase-lead controller will not degrade the steady-state error, the factor *a* in Eq. (11-72) should be absorbed by the forward-path gain *K*. Then, for design purposes, $G_c(s)$ can be written as

$$G_c(s) = \frac{1 + aTs}{1 + Ts} \quad (a > 1) \qquad (11\text{-}75)$$

The pole-zero configuration of Eq. (11-75) is shown in Fig. 11-31. Based on the discussions given in Chap. 7 on the effects of adding a pole-zero pair (with the zero closer to the origin) to the forward-path transfer function, the phase-lead controller can improve the stability of the closed-loop system if its parameters are chosen properly. The design of phase-lead control is essentially that of placing the pole and zero of $G_c(s)$ so that the design specifications are satisfied. The root-contour method can be used to indicate the proper ranges of the parameters. The following guidelines can be made with regard to the selection of the parameters *a* and *T*.

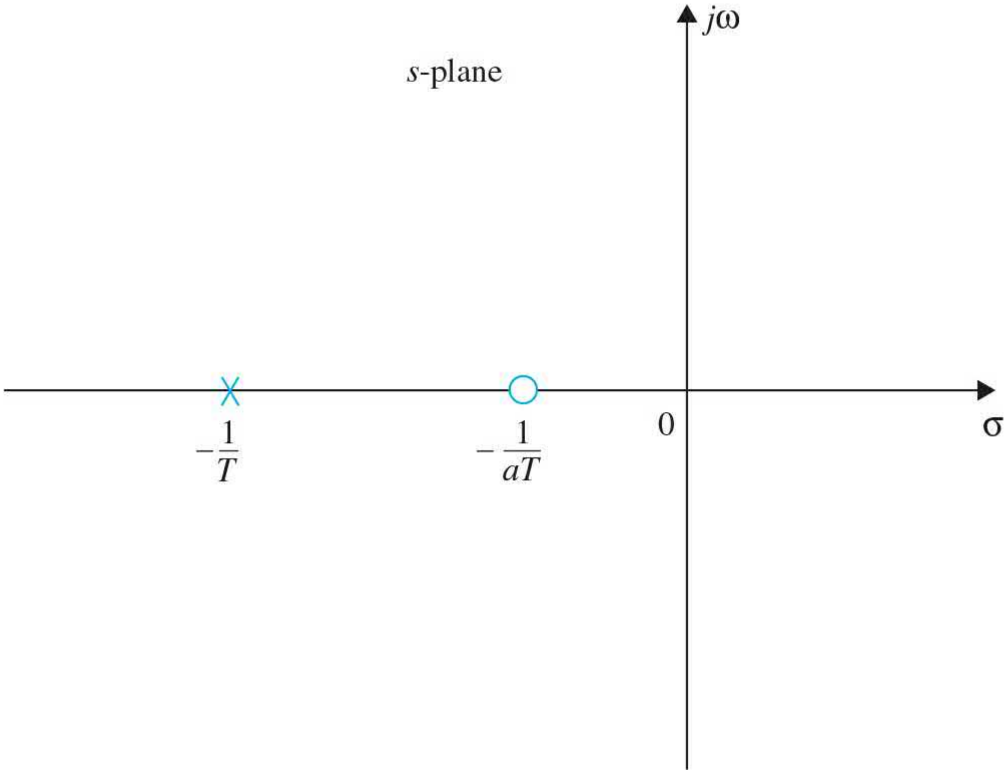


Figure 11-31 Pole-zero configuration of the phase-lead controller.

1. Moving the zero $-1/aT$ toward the origin should improve rise time and settling time. If the zero is moved too close to the origin, the maximum overshoot may again increase, because $-1/aT$ also appears as a zero of the closed-loop transfer function.
2. Moving the pole at $-1/T$ farther away from the zero and the origin should reduce the maximum overshoot, but if the value of T is too small, rise time and settling time will again increase.

We can make the following general statements with respect to the **effects of phase-lead control** on the time-domain performance of a control system:

1. When used properly, it can increase damping of the system.
2. It improves rise time and settling time.
3. In the form of Eq. (11-75), phase-lead control does not affect the steady-state error because $G_c(0) = 1$.

11-5-2 Frequency-Domain Interpretation and Design of Phase-Lead Control

The Bode plot of the phase-lead controller of Eq. (11-75) is shown in Fig. 11-32. The two corner frequencies are at $\omega = 1/aT$ and $\omega = 1/T$. The maximum value of the phase, Φ_m, and the frequency at which it occurs, ω_m, are derived as follows. Because ω_m is the geometric mean of the two corner frequencies, we write

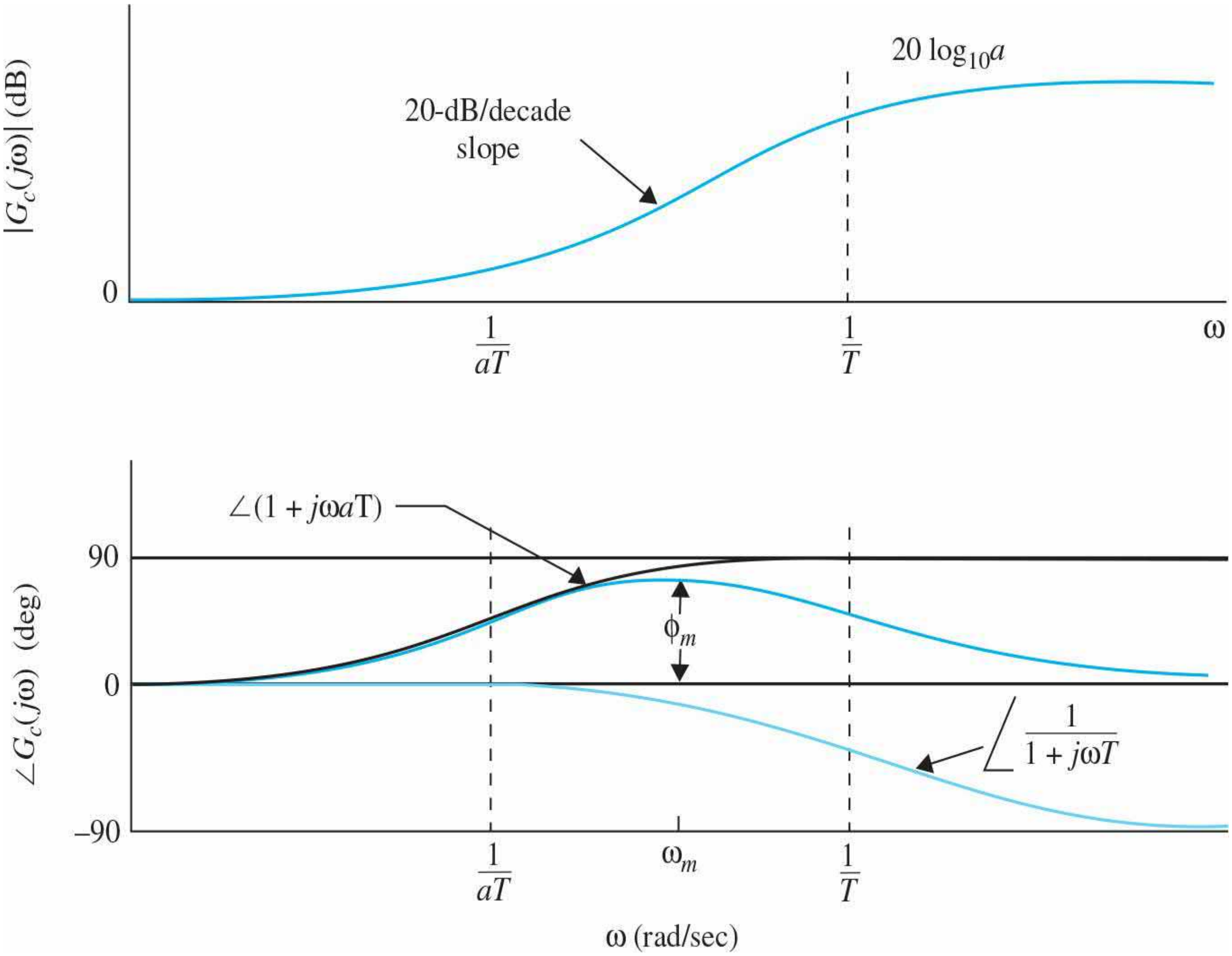


Figure 11-32 Bode plot of phase-lead controller $G_c(s) = a\dfrac{s+1/aT}{s+1/T} \quad a>1$.

$$\log_{10}\omega_m = \frac{1}{2}\left(\log_{10}\frac{1}{aT} + \log_{10}\frac{1}{T}\right) \tag{11-76}$$

Thus,

$$\omega_m = \frac{1}{\sqrt{a}T} \tag{11-77}$$

To determine the maximum phase Φ_m, the phase of $G_c(j\omega)$ is written

$$\angle G_c(j\omega) = \phi(j\omega) = \tan^{-1}\omega aT - \tan^{-1}\omega T \tag{11-78}$$

from which we get

$$\tan\phi(j\omega) = \frac{\omega aT - \omega T}{1 + (\omega aT)(\omega T)} \tag{11-79}$$

Substituting Eq. (11-77) into Eq. (11-79), we have

$$\tan\phi_m = \frac{a-1}{2\sqrt{a}} \tag{11-80}$$

or

$$\sin\phi_m = \frac{a-1}{a+1} \tag{11-81}$$

Thus, by knowing Φ_m, the value of a is determined from

$$a = \frac{1+\sin\phi_m}{1-\sin\phi_m} \tag{11-82}$$

The relationship between the phase Φ_m and a and the general properties of the Bode plot of the phase-lead controller provide an advantage of designing in the frequency domain. The difficulty is, of course, in the correlation between the time-domain and frequency-domain specifications. The general outline of phase-lead controller design in the frequency domain is given as follows. It is assumed that the design specifications simply include steady-state error and phase-margin requirements.

1. The Bode diagram of the uncompensated process $G_p(j\omega)$ is constructed with the gain constant K set according to the steady-state error requirement. The value of K has to be adjusted upward once the

value of a is determined.

2. The phase margin and the gain margin of the uncompensated system are determined, and the additional amount of phase lead needed to realize the phase margin is determined. From the additional phase lead required, the desired value of Φ_m is estimated accordingly, and the value of a is calculated from Eq. (11-82).

3. Once a is determined, it is necessary only to determine the value of T, and the design is in principle completed. This is accomplished by placing the corner frequencies of the phase-lead controller, $1/aT$ and $1/T$, such that Φ_m is located at the new gain-crossover frequency ω'_g, so the phase margin of the compensated system is benefited by Φ_m. It is known that the high-frequency gain of the phase-lead controller is 20 $\log_{10} a$ dB. Thus, to have the new gain crossover at ω_m, which is the geometric mean of $1/aT$ and $1/T$, we need to place ω_m at the frequency where the magnitude of the uncompensated $G_p(j\omega)$ is $10\log_{10}$ so that adding the controller gain of 10 $\log_{10} a$dB to this makes the magnitude curve go through 0 dB at ω_m.

4. The Bode diagram of the forward-path transfer function of the compensated system is investigated to check that all performance specifications are met; if not, a new value of ω_m must be chosen and the steps repeated.

5. If the design specifications are all satisfied, the transfer function of the phase-lead controller is established from the values of a and T.

If the design specifications also include M_r and/or BW, then these must be checked using either the Nichols chart or the output data from a computer program.

We use the following example to illustrate the design of the phase-lead controller in the time domain and frequency domain.

EXAMPLE 11-5-1 The block diagram of the sun-seeker control system described in Example 6-5-1 is again shown in Fig. 11-33. The system may be mounted on a space vehicle to track the sun with high accuracy. The variable θ_r represents the reference angle of the solar ray, and θ_0 denotes the vehicle axis. The objective of the sun-seeker system is to maintain the error ω between ω_r and ω_o near zero. The parameters of the system, for small motor time constant, are as follows:

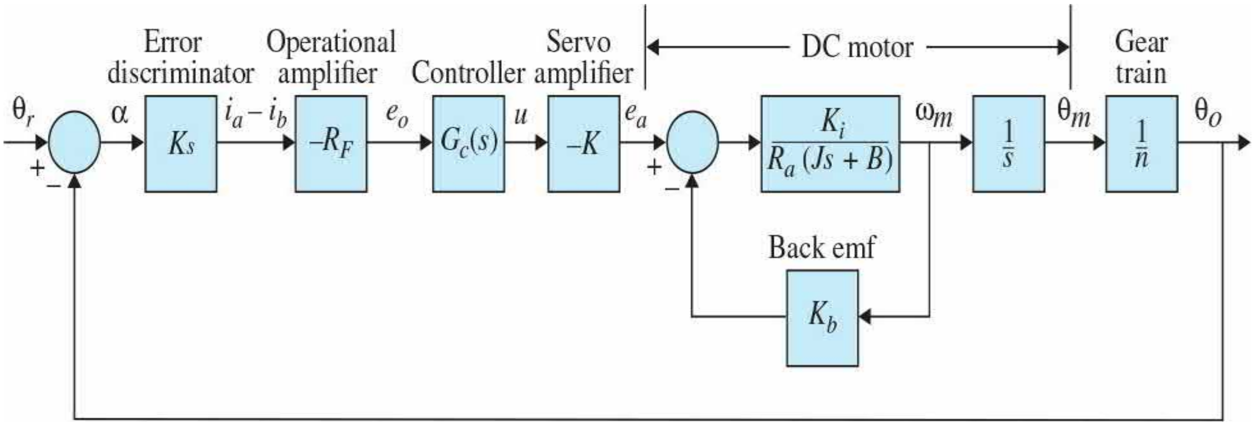


Figure 11-33 Block diagram of sun-seeker control system.

$R_F = 10{,}000\ \Omega$	$K_b = 0.0125$ V/rad/s
$K_i = 0.0125$ N-m/A	$R_a = 6.25\ \Omega$
$J = 10^{-6}$ kg-m$^2 o$	$K_s = 0.1$ A/rad
$K =$ to be determined	$B = 0$
$n = 800$	

The forward-path transfer of the uncompensated system is

$$G_p(s) = \frac{\Theta_o(s)}{A(s)} = \frac{K_s R_F K K_i / n}{R_a J s^2 + K_i K_b s} \tag{11-83}$$

where $\Theta_0(s)$ and A(s) are the Laplace transforms of $\theta_0(t)$ and $\alpha(t)$, respectively. Substituting the numerical values of the system parameters in Eq. (11-83), we get

$$G_p(s) = \frac{\Theta_o(s)}{A(s)} = \frac{2500K}{s(s+25)} \tag{11-84}$$

Time-Domain Design The time-domain specifications of the system are as follows:

1. The steady-state error of $\alpha(t)$ due to a unit-ramp function input for $\theta_r(t)$ should be ≤0.01 rad per rad/s of the final steady-state output velocity. In other words, the steady-state error due to a ramp input should be ≤1%.
2. The maximum overshoot of the step response should be less than 5 percent or as small as possible.
3. Rise time $t_r \le 0.002$ s.
4. Settling time $t_r \le 0.002$ s.

The minimum value of the amplifier gain, K, is determined initially from the steady-state requirement. Applying the final-value theorem to $\alpha(t)$, we have

$$\lim_{t\to\infty}\alpha(t)=\lim_{s\to 0} sA(s)=\lim_{s\to 0}\frac{s\Theta_r(s)}{1+G_p(s)} \tag{11-85}$$

For a unit-ramp input, $\Theta_r(s) = 1/s^2$. By using Eq. (11-84), Eq. (11-85) leads to

$$\lim_{t\to\infty}\alpha(t)=\frac{0.01}{K} \tag{11-86}$$

Thus, for the steady-state value of $\alpha(t)$ to be ≤0.01, K must be ≥1. Let us set $K = 1$, the worst case from the steady-state error standpoint, the characteristic equation of the uncompensated system is

$$s^2+25s+2500=0 \tag{11-87}$$

We can show that the damping ratio of the uncompensated system with $K = 1$ is only 0.25, which corresponds to a maximum overshoot of 44.4 percent. Figure 11-34 shows the unit-step response of the system with K =1 —shown as the uncompensated system.

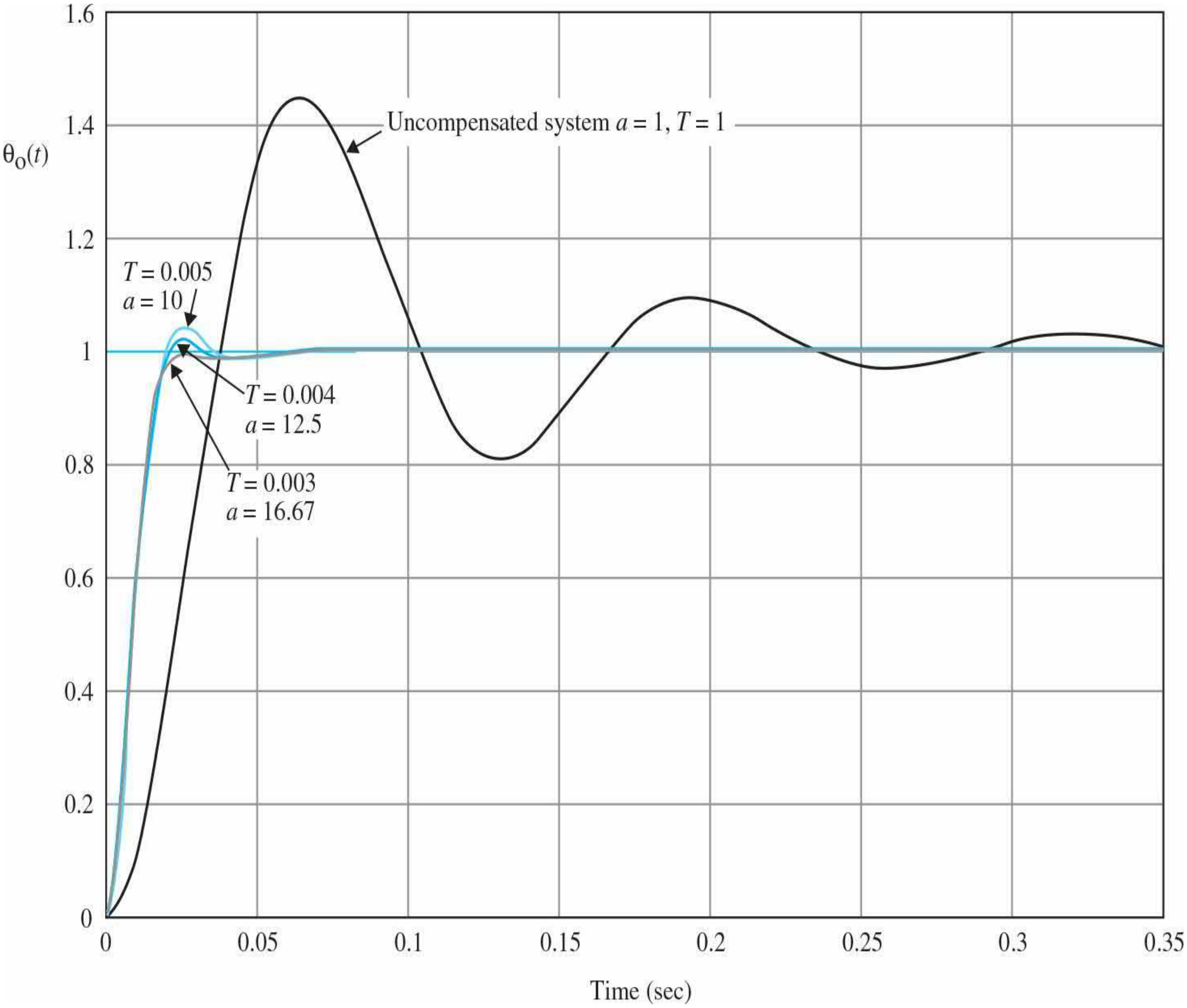


Figure 11-34 Unit-step response of sun-seeker system in Example 11-5-1.

Let us now consider using the phase-lead controller of Eq. (11-75). The forward-path transfer function of the compensated system is written as

$$G(s) = \frac{2500K(1+aTs)}{as(s+25)(1+Ts)} \tag{11-88}$$

For the compensated system to satisfy the steady-state error requirement, K must satisfy

$$K \geq a \tag{11-89}$$

Let us set *K* = a. The characteristic equation of the system is

$$(s^2+25s+2500)+Ts^2(s+25)+2500aTs=0 \tag{11-90}$$

We can use the root-contour method to show the effects of varying *a* and *T* of the phase-lead controller. Let us first set *a* = 0. The characteristic equation of Eq. (11-90) becomes

$$s^2+25s+2500+Ts^2(s+25)=0 \tag{11-91}$$

Dividing both sides of the last equation by the terms that do not contain *T*, we get

$$1+G_{\text{eq1}}(s)=1+\frac{Ts^2(s+25)}{s^2+25s+2500}=0 \tag{11-92}$$

Thus, the root contours of Eq. (11-91) when *T* varies are determined using the pole-zero configuration of $G_{eq1}(s)$ in Eq. (11-92). These root contours are obtained using Toolbox 11-5-2 and are drawn as shown in Fig. 11-35. Notice that the poles of $G_{eq1}(s)$ are the roots of the characteristic equation in Eq. (11-90), when *a* = 0 and *T* = 0. The root contours in Fig. 11-35 clearly show that adding the factor (1+*Ts*) to the denominator of Eq. (11-84) alone would not improve the system performance, since the characteristic equation roots are pushed toward the right-half plane. In fact, the system becomes unstable when *T* is greater than 0.0133. To achieve the full effect of the phase-lead controller, we must ensure that *a*>0 in Eq. (11-90). To prepare for the root contours with *a* as the variable parameter, we divide both sides of Eq. (11-90) by the terms that do not contain *a*, and the following equation results:

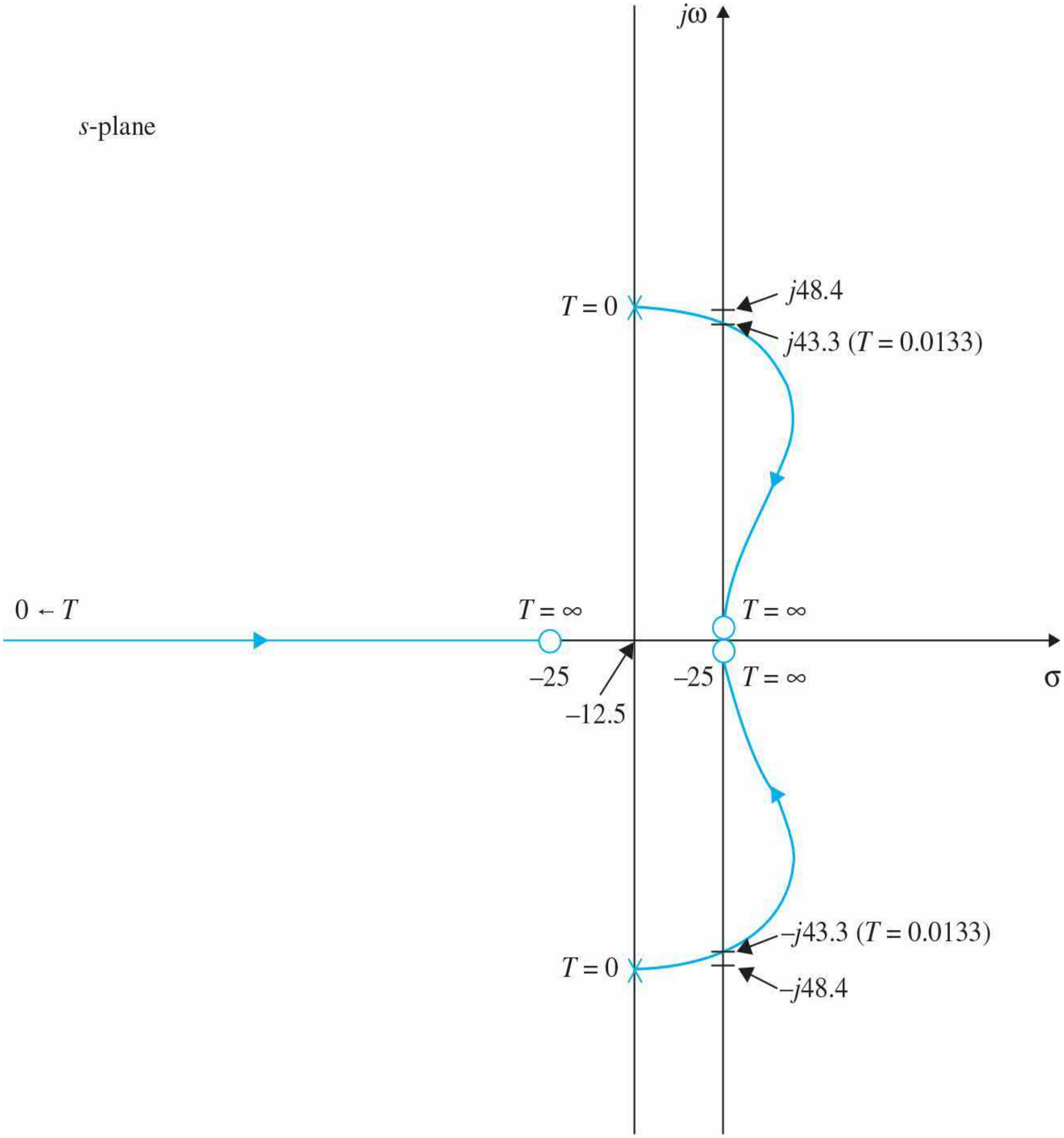


Figure 11-35 Root contours of the sun-seeker system with a = 0, and T varies from 0 to ∞.

$$1+aG_{\mathrm{eq}2}(s)=1+\frac{2500aTs}{s^2+25s+2500+Ts^2(s+25)}=0 \tag{11-93}$$

For a given T, the root contours of Eq. (11-90) when a varies are obtained based on the poles and zeros of $G_{eq2}(s)$. Notice that the poles of $G_{eq2}(s)$ are the same as the roots of Eq. (11-91). Thus, for a given T, the root contours of Eq. (11-90) when a varies must start ($a = 0$) at points on the root contours of Fig. 11-35. These root contours end ($a = \infty$) at $s = 0, \infty, \infty$, which are the zeros of $G_{eq2}(s)$. The complete root contours of Eq. (11-90) are now shown in Fig. 11-36 for several values of T, and a varies from 0 to ∞.

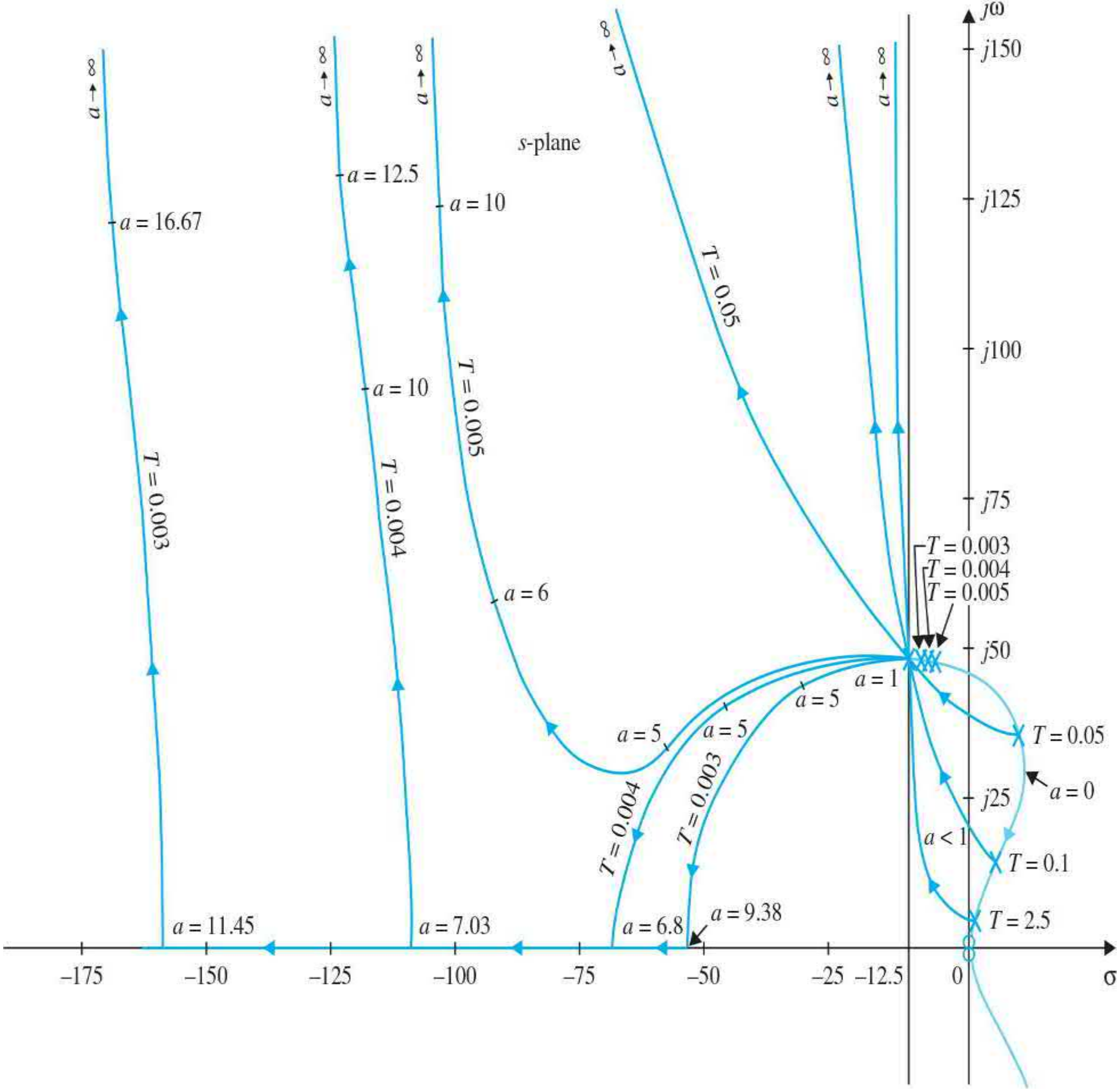
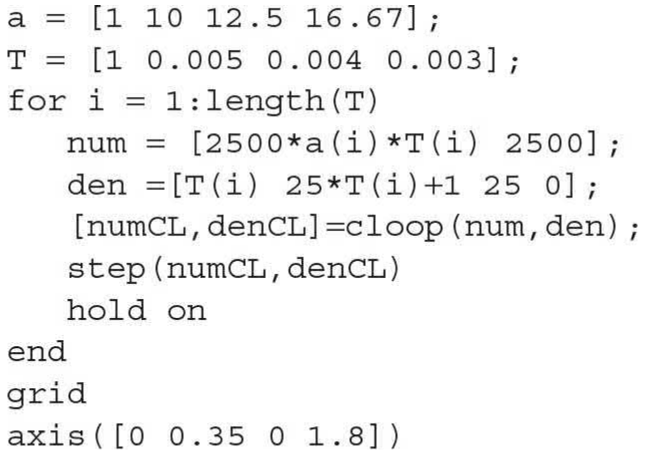


Figure 11-36 Root contours of the sun-seeker system with a phase-lead controller.

Toolbox 11-5-1

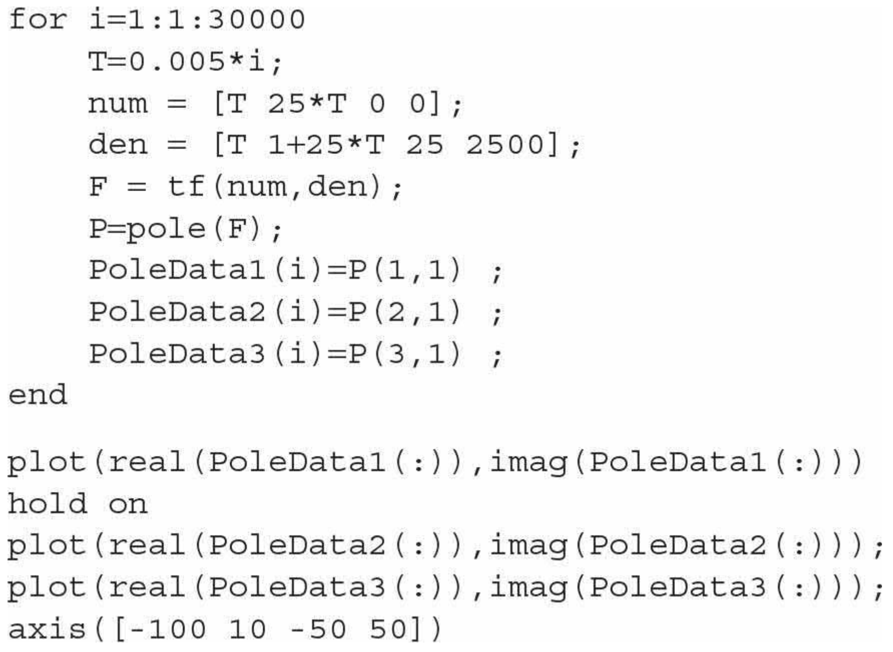
Unit-step responses in Fig. 11-34 are obtained by the following sequence of MATLAB functions:



```
a = [1 10 12.5 16.67];
T = [1 0.005 0.004 0.003];
for i = 1:length(T)
   num = [2500*a(i)*T(i) 2500];
   den =[T(i) 25*T(i)+1 25 0];
    [numCL,denCL]=cloop(num,den);
   step(numCL,denCL)
   hold on
end
grid
axis([0 0.35 0 1.8])
```

Toolbox 11-5-2

Root contours for Fig. 11-35 are obtained by the following sequence of MATLAB functions:

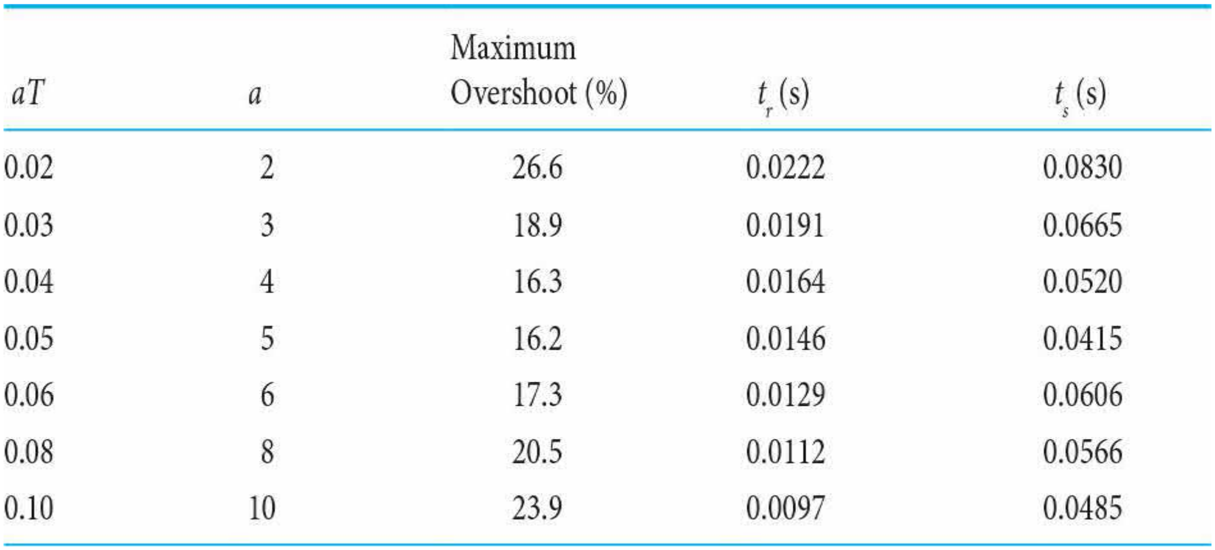


```
for i=1:1:30000
    T=0.005*i;
    num = [T 25*T 0 0];
    den = [T 1+25*T 25 2500];
    F = tf(num,den);
    P=pole(F);
    PoleData1(i)=P(1,1) ;
    PoleData2(i)=P(2,1) ;
    PoleData3(i)=P(3,1) ;
end

plot(real(PoleData1(:)),imag(PoleData1(:)))
hold on
plot(real(PoleData2(:)),imag(PoleData2(:)));
plot(real(PoleData3(:)),imag(PoleData3(:)));
axis([-100 10 -50 50])
```

From the root contours of Fig. 11-36, we see that, for effective phase-lead control, the value of *T* should be small. For large values of *T*, the natural frequency of the system increases rapidly as *a* increases, and very little improvement is made on the damping of the system.

Let us choose *T* = 0.01 arbitrarily. Table 11-12 shows the attributes of the unit-step response when the value of *aT* is varied from 0.02 to 0.1. MATLAB Toolbox 11-5-3 was used for the calculations of the time responses. The results show that the smallest maximum overshoot is obtained when *aT* = 0.05, although the rise and settling times decrease continuously as *aT* increases. However, the smallest value of the maximum overshoot is 16.2 percent, which exceeds the design specification.

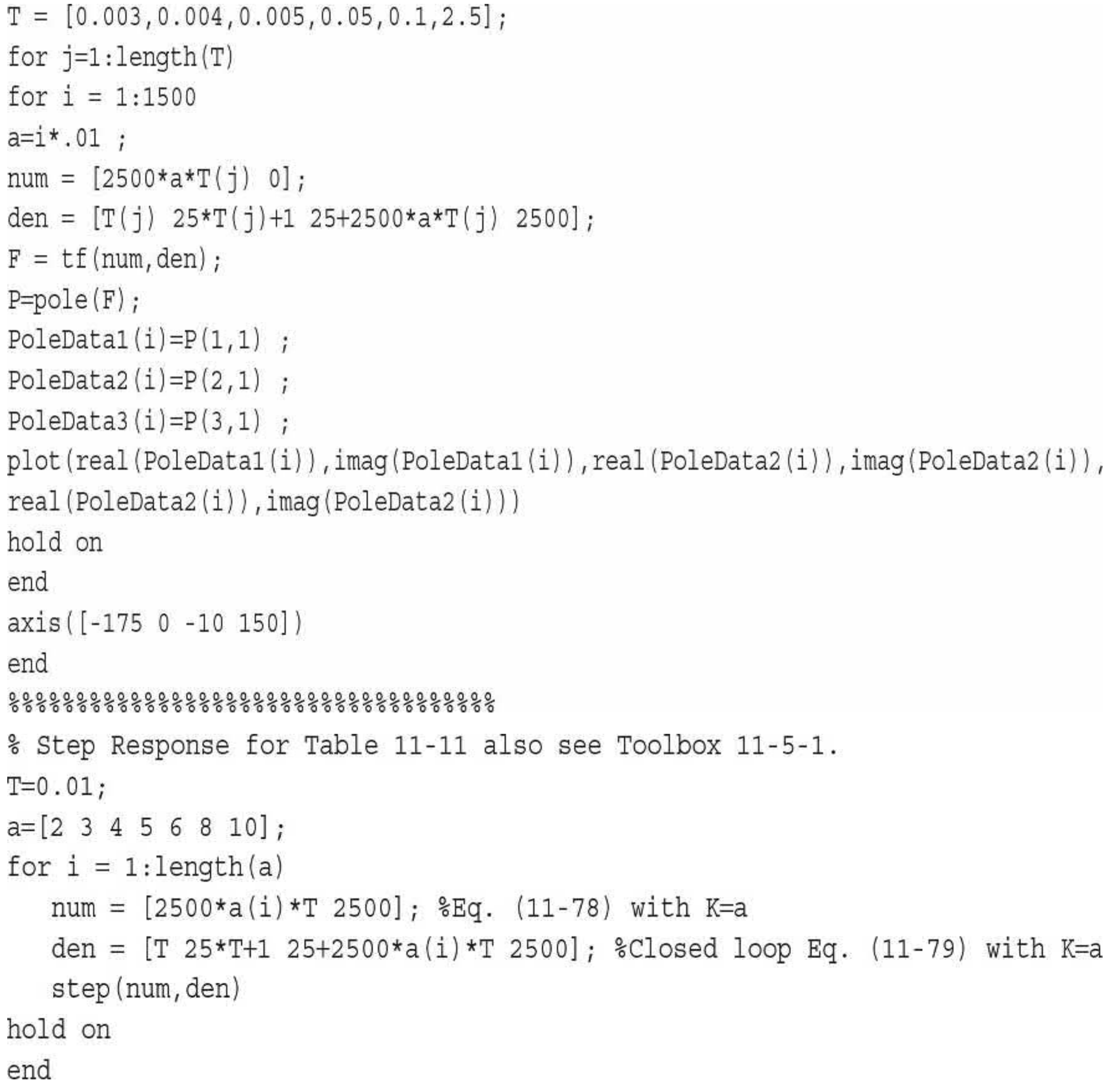
Table 11-12 Attributes of Unit-Step Response of System with Phase-Lead Controller in Example 11-5-1: *T* = 0.01



aT	a	Maximum Overshoot (%)	t_r (s)	t_s (s)
0.02	2	26.6	0.0222	0.0830
0.03	3	18.9	0.0191	0.0665
0.04	4	16.3	0.0164	0.0520
0.05	5	16.2	0.0146	0.0415
0.06	6	17.3	0.0129	0.0606
0.08	8	20.5	0.0112	0.0566
0.10	10	23.9	0.0097	0.0485

Toolbox 11-5-3

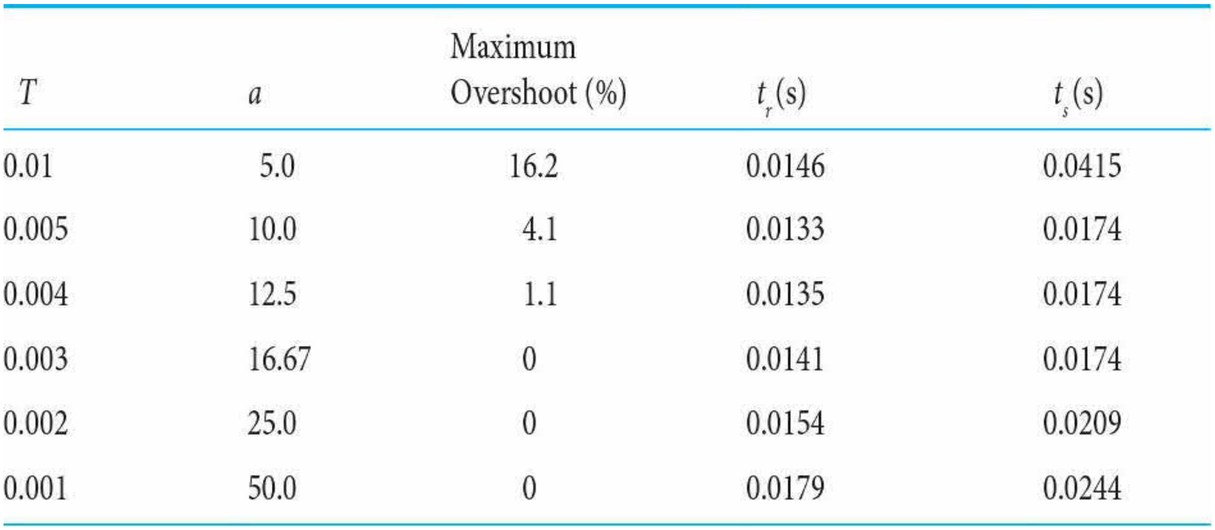
Root contours for Fig. 11-36 are obtained by the following sequence of MATLAB functions:



```
T = [0.003,0.004,0.005,0.05,0.1,2.5];
for j=1:length(T)
for i = 1:1500
a=i*.01 ;
num = [2500*a*T(j) 0];
den = [T(j) 25*T(j)+1 25+2500*a*T(j) 2500];
F = tf(num,den);
P=pole(F);
PoleData1(i)=P(1,1) ;
PoleData2(i)=P(2,1) ;
PoleData3(i)=P(3,1) ;
plot(real(PoleData1(i)),imag(PoleData1(i)),real(PoleData2(i)),imag(PoleData2(i)),
real(PoleData2(i)),imag(PoleData2(i)))
hold on
end
axis([-175 0 -10 150])
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Step Response for Table 11-11 also see Toolbox 11-5-1.
T=0.01;
a=[2 3 4 5 6 8 10];
for i = 1:length(a)
   num = [2500*a(i)*T 2500]; %Eq. (11-78) with K=a
   den = [T 25*T+1 25+2500*a(i)*T 2500]; %Closed loop Eq. (11-79) with K=a
   step(num,den)
hold on
end
```

Next, we set $aT = 0.05$ and vary T from 0.01 to 0.001, as shown in Table 11-13. Table 11-13 shows the attributes of the unit-step responses. As the value of T decreases, the maximum overshoot decreases, but the rise time and settling time increase. The cases that satisfy the design requirements are indicated in Table 11-13 for $aT = 0.05$. Figure 11-34 shows the unit-step responses of the phase-lead-compensated system with three sets of controller parameters. You can use Toolbox 11-5-1 to arrive at these results.

Table 11-13 Attributes of Unit-Step Responses of System with Phase-Lead Controller in Example 11-5-1: *aT* = 0.05



T	a	Maximum Overshoot (%)	t_r(s)	t_s(s)
0.01	5.0	16.2	0.0146	0.0415
0.005	10.0	4.1	0.0133	0.0174
0.004	12.5	1.1	0.0135	0.0174
0.003	16.67	0	0.0141	0.0174
0.002	25.0	0	0.0154	0.0209
0.001	50.0	0	0.0179	0.0244

Choosing T = 0.004, a = 12.5, the transfer function of the phase-lead controller is

$$G_c(s) = a\frac{s+1/aT}{s+1/T} = 12.5\frac{s+20}{s+250} \tag{11-94}$$

The transfer function of the compensated system is

$$G(s) = G_c(s)G_p(s) = \frac{31250(s+20)}{s(s+25)(s+250)} \tag{11-95}$$

To find the op-amp-circuit realization of the phase-lead controller, we arbitrarily set C = 0.1μf, and the resistors of the circuit are found using Eqs. (11-73) and (11-74) as R_1 = 500,000 Ω and R_2 = 40,000 Ω.

Frequency-Domain Design Let us specify that the steady-state error requirement is the same as that given earlier. For frequency-domain design specification, the phase margin is required to be greater than 45$_{\circ}$. The following design steps are taken:

1. The Bode diagram of Eq. (11-84) with K = 1 is plotted as shown in Fig. 11-37.
2. The phase margin of the uncompensated system, read at the gain-crossover frequency, ω'_c rad/s, is 28°. Because the minimum desired phase margin is 45°, at least 17° more phase lead should be added to the loop at the gain-crossover frequency.

3. The phase-lead controller of Eq. (11-75) must provide the additional 17° at the gain-crossover frequency of the compensated system. However, by applying the phase-lead controller, the magnitude curve of the Bode plot is also affected in such a way that the gain-crossover frequency is shifted to a higher frequency. Although it is a simple matter to adjust the corner frequencies, $1/aT$ and $1/T$, of the controller so that the maximum phase of the controller φ_m falls exactly at the new gain-crossover frequency, the original phase curve at this point is no longer 28° (and could be considerably less) because the phase of most control processes decreases with the increase in frequency. In fact, if the phase of the uncompensated process decreases rapidly with increasing frequency near the gain-crossover frequency, the single-stage phase-lead controller will no longer be effective.

In view of the difficulty estimating the necessary amount of phase lead, it is essential to include some safety margin to account for the inevitable phase drop-off. Therefore, in the present case, instead of selecting a φ_m of a mere 17°, let φ_m be 25°. Using Eq. (11-82), we have

$$a=\frac{1+\sin 25^\circ}{1-\sin 25^\circ}=2.46 \qquad (11\text{-}96)$$

4. To determine the proper location of the two corner frequencies ($1/aT$ and $1/T$) of the controller, it is known from Eq. (11-77) that the maximum phase lead φ_m occurs at the geometric mean of the two corner frequencies. To achieve the maximum phase margin with the value of a determined, φ_m should occur at the new gain-crossover frequency ω'_g, which is not known. The following steps are taken to ensure that φ_m occurs at ω'_g.

a. The high-frequency gain of the phase-lead controller of Eq. (11-75) is

$$20\log_{10} a = 20\log_{10} 2.46 = 7.82 \text{ dB} \qquad (11\text{-}97)$$

b. The geometric mean ω_m of the two corner frequencies, $1/aT$ and $1/T$, should be located at the frequency at which the magnitude of the uncompensated process transfer function $G_p(j\omega)$ in dB is equal to the negative value in dB of one-half of this gain. This way, the magnitude curve of the compensated transfer function will pass through the 0-dB axis at $\omega = \omega_m$. Thus, ω_m should be located at the frequency where

$$\left|G_p(j\omega)\right|_{\mathrm{dB}} = -10\log_{10} 2.46 = -3.91\ \mathrm{dB} \tag{11-98}$$

Toolbox 11-5-4

Bode diagram for Fig. 11-37 is obtained by the following sequence of MATLAB functions:

```
a = [0 2.46,12.5,5.828];
T = [0 0.0106,0.004,0.00588];
for i = 1:length(T)
num = [2500*a(i)*T(i) 2500]; % numerator of Eq. (11-78) with K=a
den =[T(i) 1+25*T(i) 25 0]; % denominator of Eq. (11-78) with K=a
bode(num,den);
hold on;
end
grid
```

Toolbox 11-5-5

Nichols plot of G(s) for Fig. 11-38 is obtained by the following sequence of MATLAB functions:

```
a = [0 2.46,12.5,5.828];
T = [0 0.0106,0.004,0.00588];
for i = 1:length(T)
num = [2500*a(i)*T(i) 2500]; % numerator of Eq. (11-78) with K=a
den =[T(i) 1+25*T(i) 25 0]; % denominator of Eq. (11-78) with K=a
t = tf(num,den)
nichols(t); ngrid;
hold on;
end
```

From Fig. 11-37, this frequency is found to be ω_m. Now using Eq. (11-77), we have

$$\frac{1}{T} = \sqrt{a}\omega_m = \sqrt{2.46} \times 60 = 94.1 \text{ rad/s} \tag{11-99}$$

Then, 1/*aT*=94.1/2.46=38.21 rad/s. The transfer function of the phase-lead controller is

$$G_c(s) = a\frac{s+1/aT}{s+1/T} = 2.46\frac{s+38.21}{s+94.1} \tag{11-100}$$

The forward-path transfer function of the compensated system is

$$G(s) = G_c(s)G_p(s) = \frac{6150(s+38.21)}{s(s+25)(s+94.1)} \tag{11-101}$$

Figure 11-37 shows that the phase margin of the compensated system is actually 47.6°.

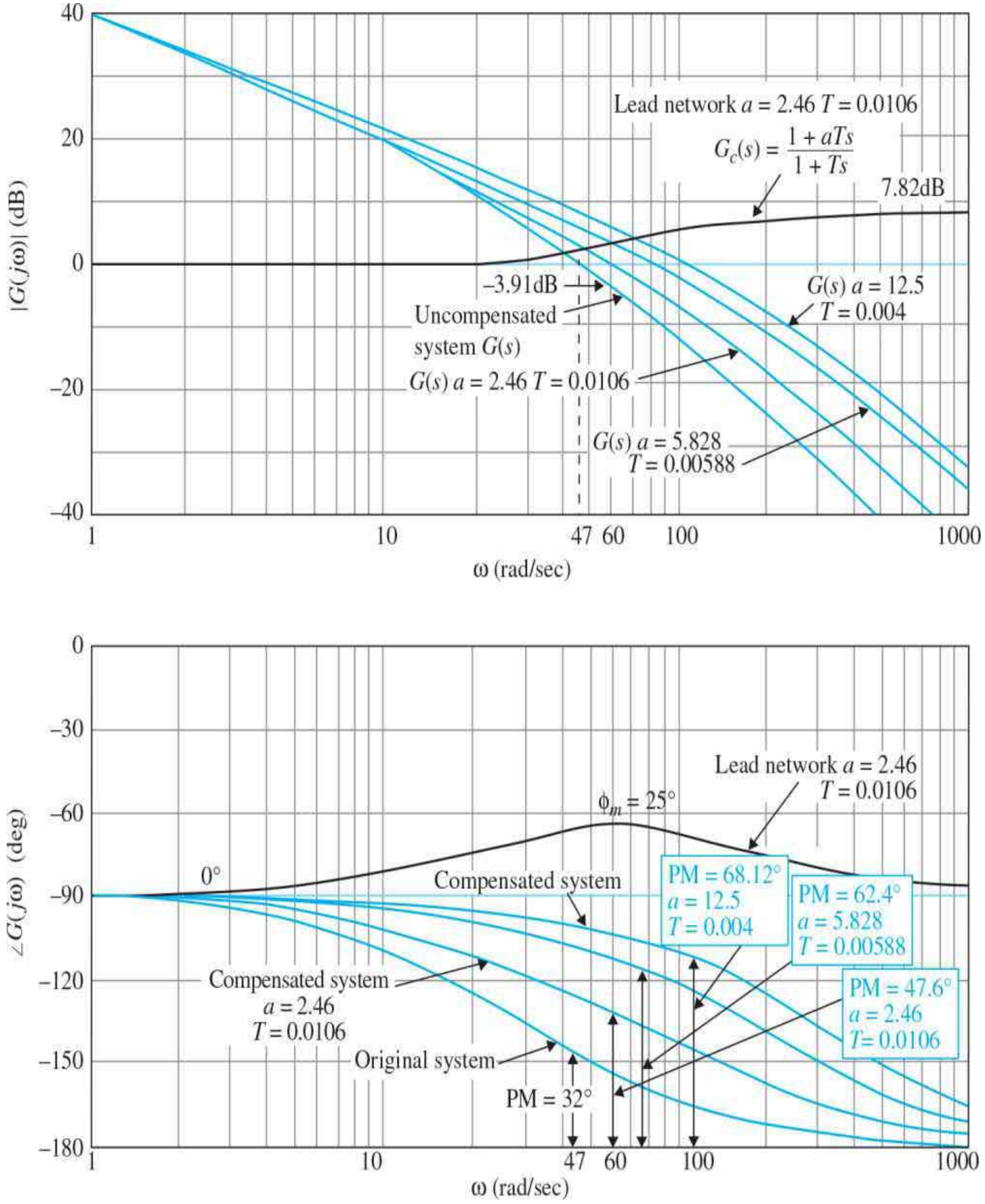


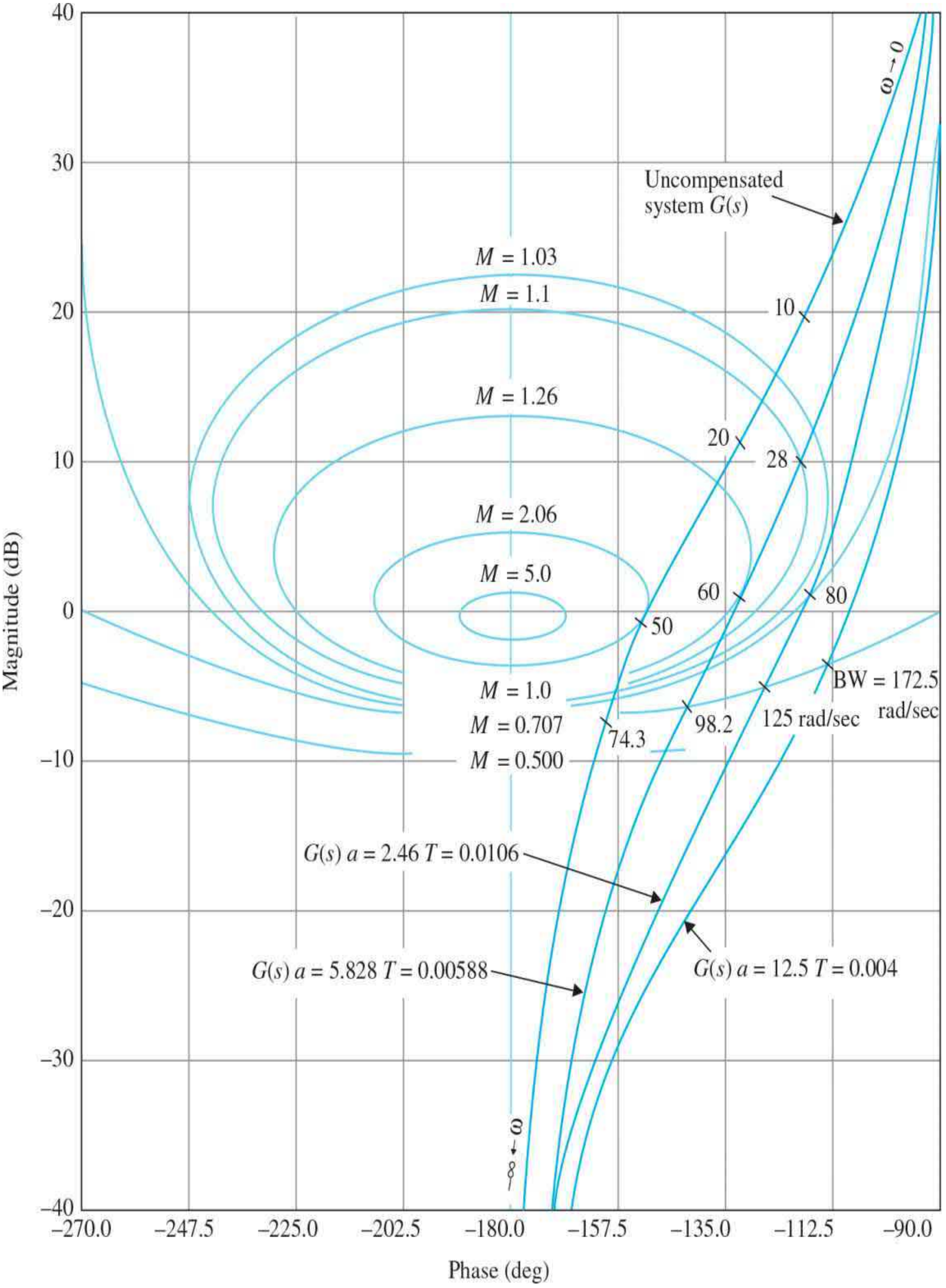
Figure 11-37 Bode diagram of the phase-lead compensation and uncompensated systems in Example 11-5-1. $G(s) = \frac{2500(1 + aTs)}{s(s + 25)(s + Ts)}$.

In Fig. 11-38, the magnitude and phase of the original and the compensated systems are plotted on the Nichols chart for display only. These plots can be made by taking the data directly from the Bode plots of Fig. 11-37. The values of M_r, ω_r, and BW can all be determined from the Nichols chart.

Checking the time-domain performance of the compensated system, we have the following results:

$$\text{Maximum overshoot} = 22.3\% \quad t_r = 0.02045 \text{ s} \quad t_s = 0.07439 \text{ s}$$

which fall short of the time-domain specifications listed earlier. Figure 11-37 also shows the Bode plot of the system compensated with a phase-lead controller with a = 5.828 and T = 0.00588. The phase margin is improved to 62.4°. Using Eq. (11-81), we can show that the result of a = 12.5 obtained in the time-domain design actually corresponds to φ_m = 58.41. Adding this to the original phase of 28°, the corresponding phase margin would be 86.41°. The time-domain and frequency-domain attributes of the system with the three phase-lead controllers are summarized in Table 11-14. The results show that, with a = 12.5 and T = 0.0004, even the projected phase margin is 86.41°; the actual value is 68.12° due to the fall-off of the phase curve at the new gain crossover. ▲



40
30
20
10
0
−10
−20
−30
−40
Magnitude (dB)
−270.0
−247.5
−225.0
−202.5
−180.0
−157.5
−135.0
−112.5
−90.0
Phase (deg)
ω → 0
Uncompensated system G(s)
M = 1.03
M = 1.1
M = 1.26
M = 2.06
M = 5.0
M = 1.0
M = 0.707
M = 0.500
10
20
28
60
80
50
74.3
98.2
125 rad/sec
BW = 172.5 rad/sec
G(s) a = 2.46 T = 0.0106
G(s) a = 5.828 T = 0.00588
G(s) a = 12.5 T = 0.004
ω → ∞

Figure 11-38 Plots of G(s) in the Nichols chart for the system in Example 11-5-1. $G(s)=\dfrac{2500(1+aTs)}{s(s+25)(1+Ts)}$.

Table 11-14 Attributes of System with Phase-Lead Controller in Example 11-5-1

a	T	PM (deg)	M_r	Gain CO (rad/s)	BW (rad/s)	Maximum Overshoot (%)	t_r(s)	t_s(s)
1	1	28.03	2.06	47.0	74.3	44.4	0.0255	0.2133
2.46	0.0106	47.53	1.26	60.2	98.2	22.3	0.0204	0.0744
5.828	0.00588	62.36	1.03	79.1	124.7	7.7	0.0169	0.0474
12.5	0.0040	68.12	1.00	113.1	172.5	1.1	0.0135	0.0174

EXAMPLE 11-5-2[3] In this example, we illustrate the application of a phase-lead controller to a third-order system with relatively high loop gain.

Let us consider that the inductance of the dc motor of the sun-seeker system described in Fig. 11-33 is not zero. The following set of system parameters is given:

$R_F = 10{,}000\ \Omega$

$K_i = 0.0125$ N-m/A

$J = 10^{-6}$ kg-m^2

$K =$ to be determined

$n = 800$

$K_b = 0.0125$ V/rad/s

$R_a = 6.25\ \Omega$

$K_s = 0.3$ A/rad

$B = 0$

$L_a = 10^{-3}$ H

The transfer function of the dc motor is written

$$\frac{\Omega_m(s)}{E_a(s)} = \frac{K_i}{s(L_aJs^2 + JR_as + K_iK_b)} \tag{11-102}$$

The forward-path transfer function of the system is

$$G_p(s) = \frac{\Theta_o(s)}{A(s)} = \frac{K_sR_FKK_i}{s(L_aJs^2 + JR_as + K_iK_b)} \tag{11-103}$$

Substituting the values of the system parameters in Eq. (11-102), we get

$$G_p(s)=\frac{\Theta_o(s)}{A(s)}=\frac{4.6875\times 10^7 K}{s(s^2+625s+156{,}250)} \tag{11-104}$$

Time-Domain Design The time-domain specifications of the system are given as follows:

1. The steady-state error of α(t) due to a unit-ramp function input for $\theta_r(t)$ should be ≤1/300 rad/rad/s of the final steady-state output velocity.
2. The maximum overshoot of the step response should be less than 5 percent or as small as possible.
3. Rise time t_r≤0.004 s.
4. Settling time t_r≤0.02 s.

The minimum value of the amplifier gain K is determined initially from the steady-state requirement. Applying the final-value theorem to α(t), we get

$$\lim_{t\to\infty}\alpha(t)=\lim_{s\to 0} sA(s)=\lim_{s\to 0}\frac{s\Theta_r(s)}{1+G_p(s)} \tag{11-105}$$

Substituting Eq. (11-104) into Eq. (11-105), and Θ_r(s)=1/s^2, we have

$$\lim_{t\to\infty}\alpha(t)=\frac{1}{300\ K} \tag{11-106}$$

Thus, for the steady-state value of α(t) to be ≤1/300, K must be ≥1. Let us set K =1; the forward-path transfer function in Eq. (11-104) becomes

$$G_p(s)=\frac{4.6875\times 10^7}{s(s^2+625s+156{,}250)} \tag{11-107}$$

We can show that the closed-loop sun-seeker system with K =1 has the following attributes for the unit-step response:

$$\text{Maximum overshoot}=43\% \quad \text{Rise time } t_r=0.004797\text{ s} \quad \text{Settling time } t_s=0.04587\text{ s}$$

To improve the system response, let us select the phase-lead controller described by Eq. (11-75). The forward-path transfer function of the compensated system is

$$G(s)=G_c(s)G_p(s)=\frac{4.6875\times 10^7 K(1+aTs)}{as(s^2+625s+156,250)(1+Ts)} \tag{11-108}$$

Now to satisfy the steady-state requirement, K must be readjusted so that $K \geq a$. Let us set K =a. The characteristic equation of the phase-lead compensated system becomes

$$(s^3+625s^2+156,250s+4.6875\times 10^7)+Ts^2(s^2+625s+156,250)+4.6875\times 10^7 aTs=0 \tag{11-109}$$

We can use the root-contour method to examine the effects of varying a and T of the phase-lead controller. Let us first set a to zero. The characteristic equation of Eq. (11-109) becomes

$$(s^3+625s^2+156,250s+4.6875\times 10^7)+Ts^2(s^2+625s+156,250)=0 \tag{11-110}$$

Dividing both sides of the last equation by the terms that do not contain T, we get

$$1+G_{\text{eq1}}(s)=1+\frac{Ts^2(s^2+625s+156,250)}{s^3+625s^2+156,250s+4.6875\times 10^7}=0 \tag{11-111}$$

The root contours of Eq. (11-110) when T varies are determined from the pole-zero configuration of $G_{eq1}(s)$ in Eq. (11-111) and are drawn as shown in Fig. 11-39. When a varies from 0 to ∞, we divide both sides of Eq. (11-109) by the terms that do not contain a, and we have

$$1+G_{\text{eq2}}(s)=1+\frac{4.6875\times 10^7 aTs}{s^3+625s^2+156,250s+4.6875\times 10^7+Ts^2(s^2+625s+156,250)}=0 \tag{11-112}$$

For a given T, the root contours of Eq. (11-109) when a varies are obtained based on the poles and zeros of $G_{eq2}(s)$. The poles of $G_{eq2}(s)$ are the same as the roots of Eq. (11-110). Thus, the root contours when a varies start (a = 0) at the root contours for variable T. Figure 11-39 shows the dominant portions of the root contours when a varies for T = 0.01, 0.0045, 0.001, 0.0005, 0.0001, and 0.00001. Notice that, because the uncompensated system is lightly damped, for the phase-lead controller to be effective, the value of T should be very small. Even for very small values of T, there is only a small range of a

that could bring increased damping, but the natural frequency of the system increases with the increase in *a*. The root contours in Fig. 11-39 show the approximate locations of the dominant characteristic equation roots where maximum damping occurs. Table 11-15 gives the roots of the characteristic equation and the unit-step-response attributes for the cases that correspond to near-smallest maximum overshoot for the *T* selected. Figure 11-40 shows the unit-step response, when *a*=500 and *T*=0.00001. Although the maximum overshoot is only 3.8 percent, the undershoot in this case is greater than the overshoot.

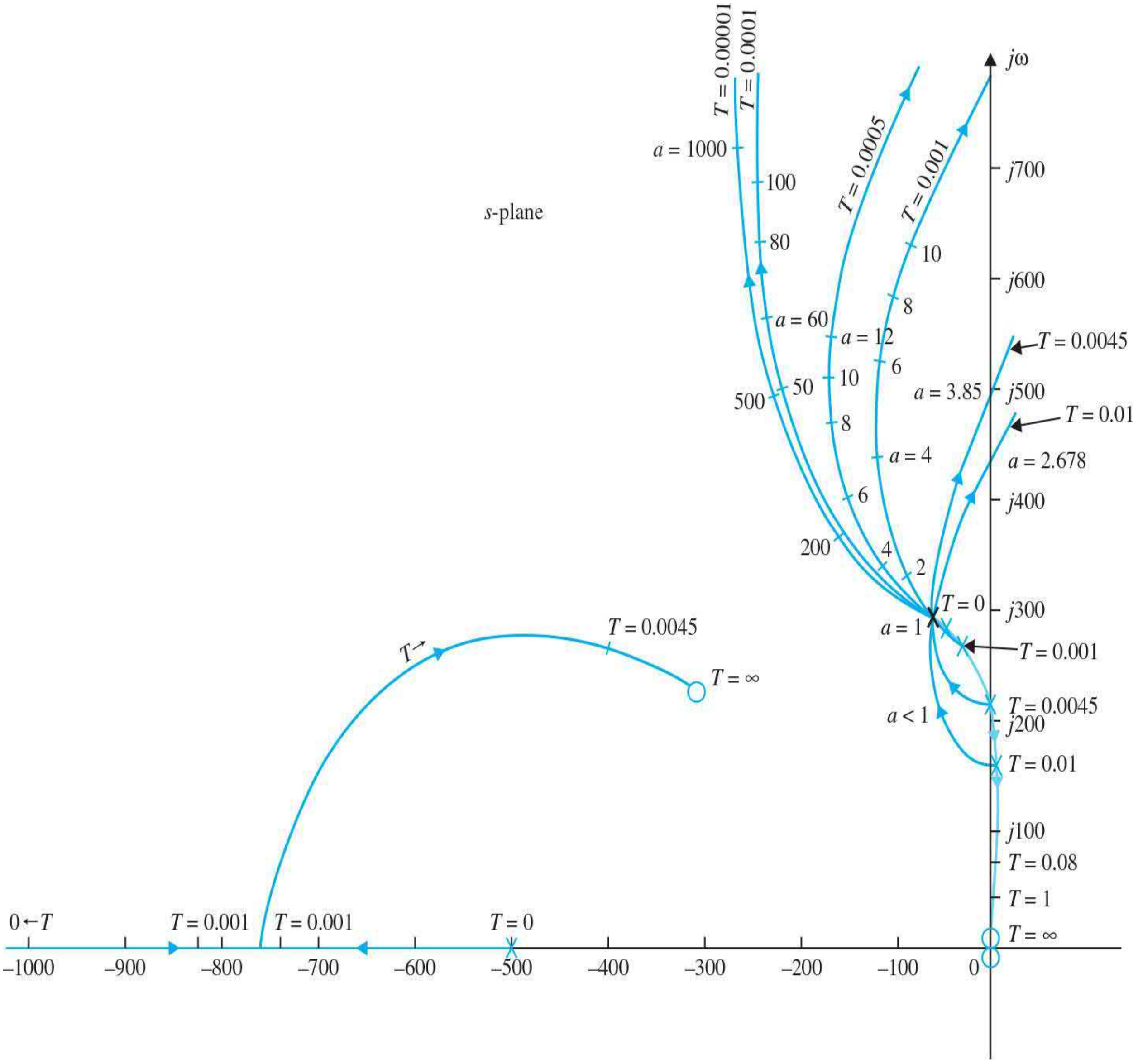


Figure 11-39 Root contours of sun-seeker system in Example 11-5-2

with phase-lead controller. $G_c(s) = \frac{1+aTs}{1+Ts}$.

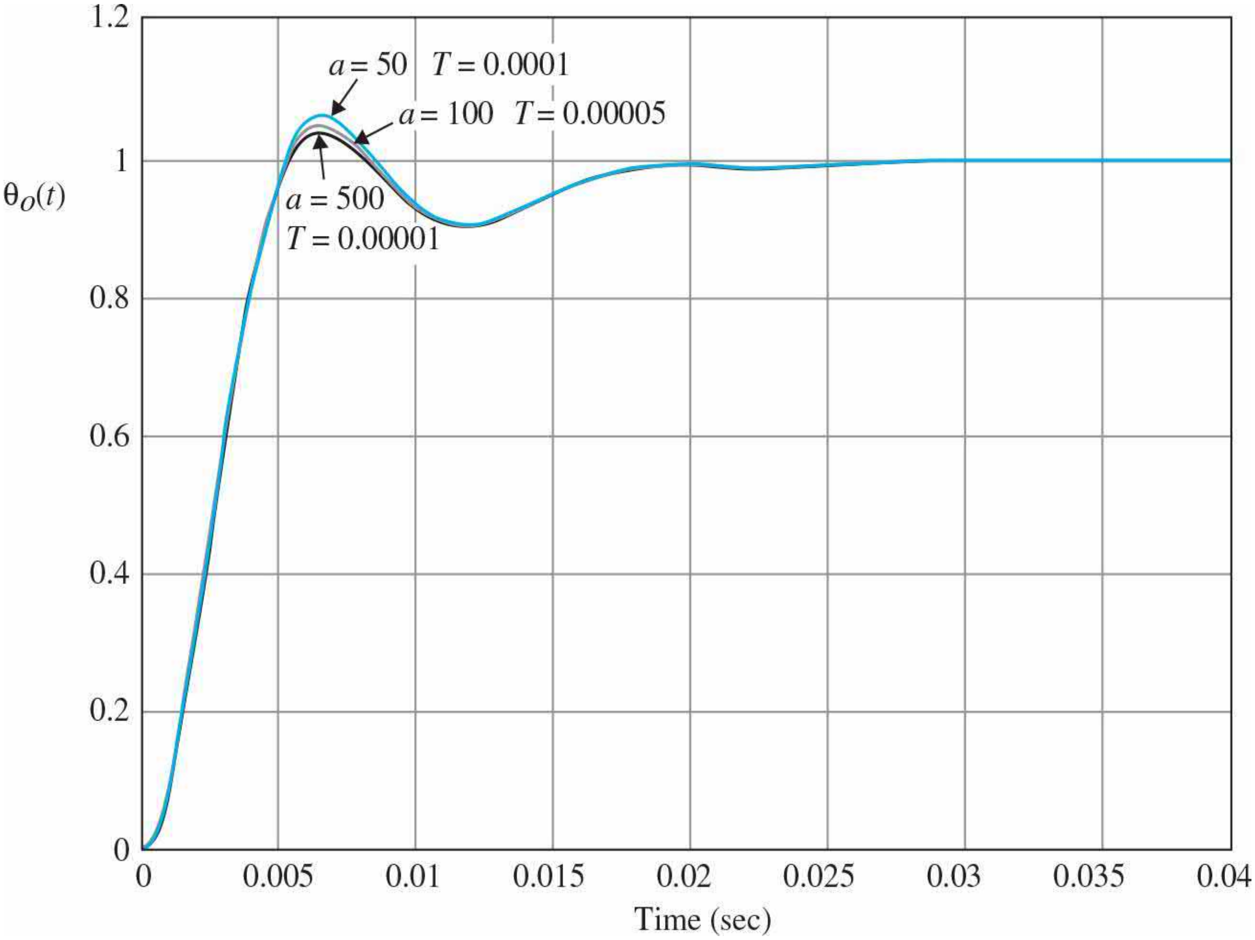
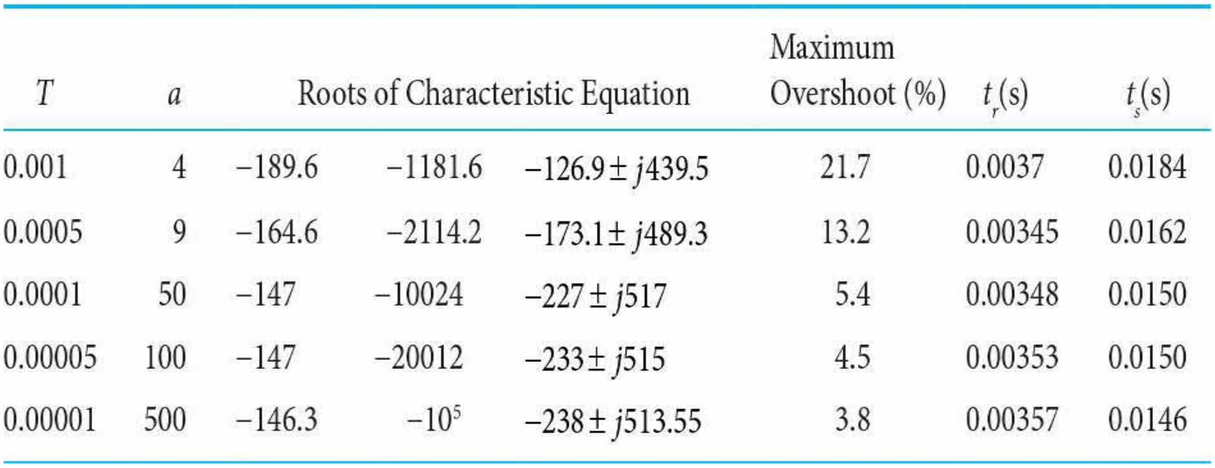


Figure 11-40 Unit-step responses of sun-seeker system in Example 11-5-2 with phase-lead controller. $G_c(s) = \frac{1+aTs}{1+Ts}$.

Table 11-15 Roots of Characteristic Equation and Time Response Attributes of System with Phase-Lead Controller in Example 11-5-2



T	a	Roots of Characteristic Equation			Maximum Overshoot (%)	t_r(s)	t_s(s)
0.001	4	−189.6	−1181.6	$-126.9 \pm j439.5$	21.7	0.0037	0.0184
0.0005	9	−164.6	−2114.2	$-173.1 \pm j489.3$	13.2	0.00345	0.0162
0.0001	50	−147	−10024	$-227 \pm j517$	5.4	0.00348	0.0150
0.00005	100	−147	−20012	$-233 \pm j515$	4.5	0.00353	0.0150
0.00001	500	−146.3	-10^5	$-238 \pm j513.55$	3.8	0.00357	0.0146

Toolbox 11-5-6

Unit-step responses in Fig. 11-40 are obtained by the following sequence of MATLAB functions:

```
a = [50,100,500];
T = [0.0001,0.00005,0.00001];
for i = 1:length(T)
num = 4.6875e7 * [a(i)*T(i) 1];
den = conv([1 625 156250 0],[T(i) 1]);
[numCL,denCL]=cloop(num,den);
format long
roots(denCL)
step(numCL,denCL)
hold on
end
axis([0 .04 0 1.2])
grid
```

Frequency-Domain Design The Bode plot of $G_p(s)$ in Eq. (11-104) is shown in Fig. 11-41. The performance attributes of the uncompensated system are

PM = 29.74°

$M_r = 2.156$

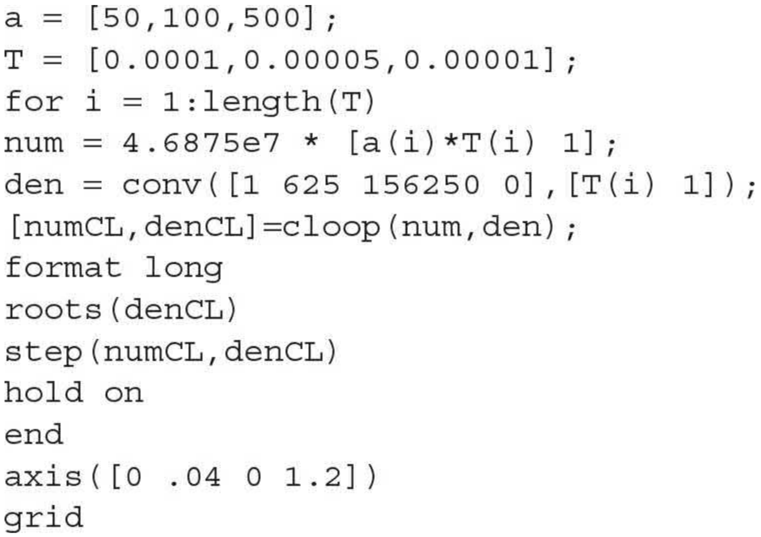
BW = 426.5 rad/s

We would like to show that the frequency-domain design procedure outlined earlier does not work effectively here because the phase curve of $G_p($ $j\omega)$ shown in Fig. 11-41 has a very steep slope near the gain crossover. For example, if we wish to realize a phase margin of 65°, we need at least 65-29.74=35.26° of phase lead. Or, φ_m = 35.26°. Using Eq. (11-82), the value of a is calculated to be

$$a = \frac{1+\sin\phi_m}{1-\sin\phi_m} = \frac{1+\sin 35.26^\circ}{1-\sin 35.26^\circ} = 3.732 \tag{11-113}$$

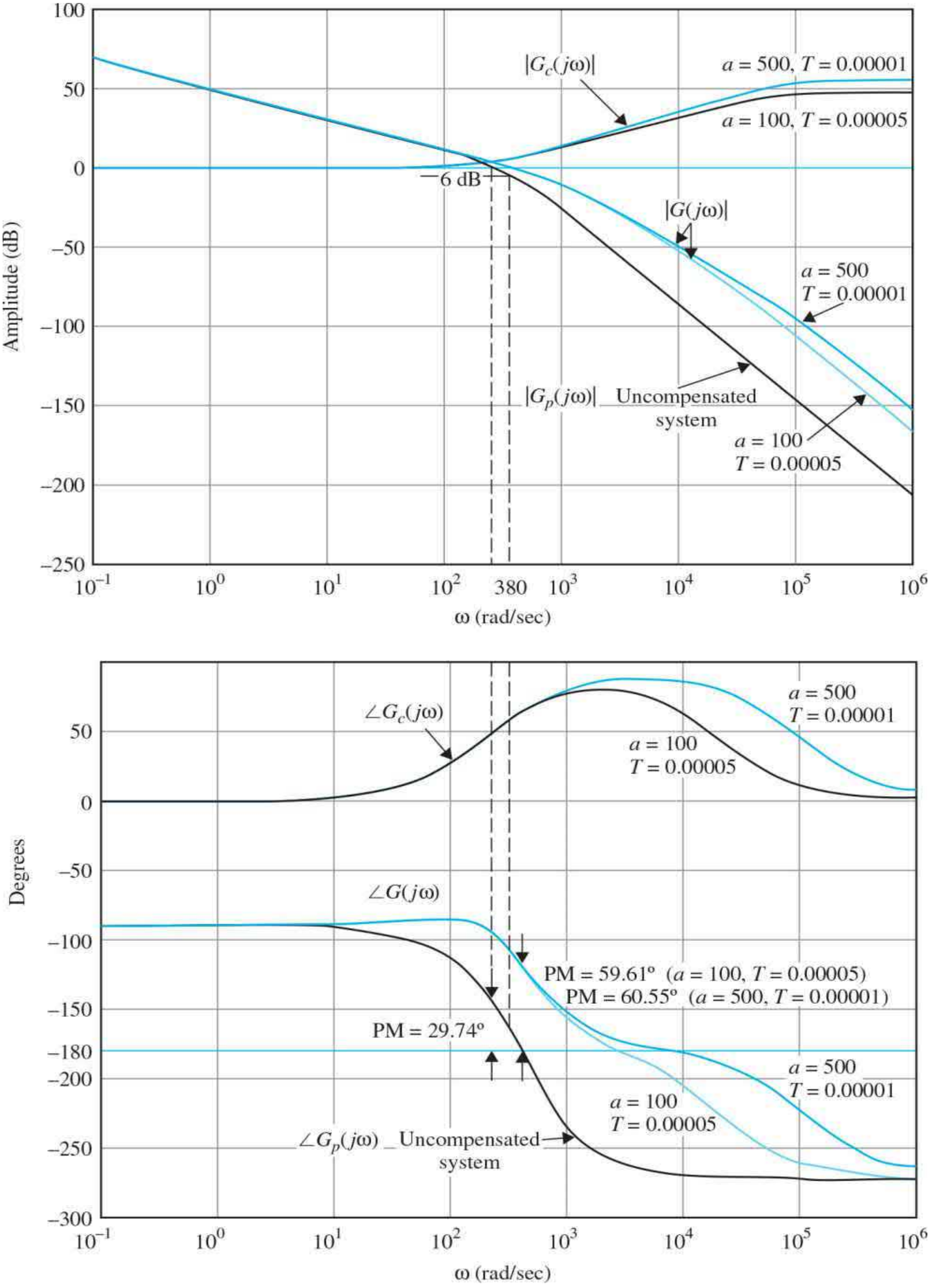
Toolbox 11-5-7

Bode plots shown in Fig. 11-41 are obtained by the following sequence of MATLAB functions:



```
a = [50,100,500];
T = [0.0001,0.00005,0.00001];
for i = 1:length(T)
num = 4.6875e7 * [a(i)*T(i) 1];
den = conv([1 625 156250 0],[T(i) 1]);
[numCL,denCL]=cloop(num,den);
format long
roots(denCL)
step(numCL,denCL)
hold on
end
axis([0 .04 0 1.2])
grid
```

Let us choose a = 4. Theoretically, to maximize the utilization of ω_m, ω_m should be placed at the new gain crossover, which is located at the frequency where the magnitude of $G_p($ $j\omega)$ is -10log$_{10}$ a dB=-10log$_{10}$4=-6. From the Bode plot in Fig. 11-41, this frequency is found to be 380 rad/s. Thus, we let φ_m = 380 rad/s. The value of T is found by using Eq. (11-77):



Amplitude (dB)
100
50
0
−50
−100
−150
−200
−250
10^{-1}
10^{0}
10^{1}
10^{2}
380
10^{3}
10^{4}
10^{5}
10^{6}
ω (rad/sec)
$|G_c(j\omega)|$
$a = 500,\ T = 0.00001$
$a = 100,\ T = 0.00005$
−6 dB
$|G(j\omega)|$
$a = 500$
$T = 0.00001$
$|G_p(j\omega)|$ Uncompensated system
$a = 100$
$T = 0.00005$
Degrees
50
0
−50
−100
−150
−180
−200
−250
−300
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
ω (rad/sec)
$\angle G_c(j\omega)$
$a = 500$
$T = 0.00001$
$a = 100$
$T = 0.00005$
$\angle G(j\omega)$
PM = 59.61° ($a = 100,\ T = 0.00005$)
PM = 60.55° ($a = 500,\ T = 0.00001$)
PM = 29.74°
$a = 500$
$T = 0.00001$
$a = 100$
$T = 0.00005$
$\angle G_p(j\omega)$ Uncompensated system

Figure 11-41 Bode plots of phase-lead controller and forward-path transfer function of sun-seeker system in Example 11-5-2. $G_c(s)=\frac{1+aTs}{1+Ts}$.

$$T=\frac{1}{\omega_m\sqrt{a}}=\frac{1}{380\sqrt{4}}=0.0013 \qquad (11\text{-}114)$$

However, checking the frequency response of the phase-lead compensated system with a = 4 and T = 0.0013, we found that the phase margin is only improved to 38.27°, and M_r = 1.69. The reason is the steep negative slope of the phase curve of $G_p(j\omega)$. The fact is that, at the new gain-crossover frequency of 380 rad/s, the phase of $G_p(j\omega)$ is -170°, as against -150-26° at the original gain crossover—a drop of almost 20°! From the time-domain design, the first line in Table 11-16 shows that, when a = 4 and T = 0.00001, the maximum overshoot is 21.7 percent.

Table 11-16 Attributes of System with Phase-Lead Controller in Example 11-5-2

T	a	PM (deg)	GM (dB)	M_r	BW (rad/s)	Maximum Overshoot (%)	t_r (s)	t_r (s)
1	1	29.74	6.39	2.16	430.4	43.0	0.00478	0.0459
0.00005	100	59.61	31.41	1.009	670.6	4.5	0.00353	0.015
0.00001	500	60.55	45.21	1.000	664.2	3.8	0.00357	0.0146

Checking the frequency response of the phase-lead compensated system with a = 500, and T = 0.00001, the following performance data are obtained:

$$\text{PM}=60.55\text{ degrees} \quad M_r=1 \quad \text{BW}=664.2\text{ rad/s}$$

This shows that the value of a has to be increased substantially just to overcome the steep drop of the phase characteristics when the gain crossover is moved upward.

Figure 11-41 shows the Bode plots of the phase-lead controller and the forward-path transfer functions of the compensated system with a = 100, T =

0.00005, and *a* = 500, *T* = 0.00001. A summary of performance data is given in Table 11-16.

Selecting *a* = 100 and *T* = 0.00005, the phase-lead controller is described by the transfer function

$$G_c(s) = \frac{1}{a}\frac{1+aTs}{1+Ts} = \frac{1}{100}\frac{1+0.005s}{1+0.00005s} \tag{11-115}$$

Using Eqs. (11-73) and (11-74), and letting C = 0.01μF, the circuit parameters of the phase-lead controller are found to be

$$R_2 = \frac{T}{C} = \frac{5\times10^{-5}}{10^{-8}} = 5000\ \Omega \tag{11-116}$$

$$R_1 = aR_2 = 500{,}000\ \Omega \tag{11-117}$$

The forward-path transfer function of the compensated system is

$$\frac{\Theta_o(s)}{A(s)} = \frac{4.6875\times10^7(1+0.005s)}{s(s^2+625s+156{,}250)(1+0.00005s)} \tag{11-118}$$

where the amplifier gain *K* has been set to 100 to satisfy the steady-state requirement. ▲

11-5-3 Effects of Phase-Lead Compensation

From the results of the last two illustrative examples, we can summarize the effects and limitations of the single-stage phase-lead controller as follows:

1. The phase-lead controller adds a zero and a pole, with the zero to the right of the pole, to the forward-path transfer function. The general effect is to add more damping to the closed-loop system. The rise time and settling time are reduced in general.
2. The phase of the forward-path transfer function in the vicinity of the gain-crossover frequency is increased. This improves the phase margin of the closed-loop system.
3. The slope of the magnitude curve of the Bode plot of the forward-path transfer function is reduced at the gain-crossover frequency. This usually corresponds to an improvement in the relative stability of the

system in the form of improved gain and phase margins.

4. The bandwidth of the closed-loop system is increased. This corresponds to faster time response.
5. The steady-state error of the system is not affected.

11-5-4 Limitations of Single-Stage Phase-Lead Control

In general, phase-lead control is not suitable for all systems. Successful application of single-stage phase-lead compensation to improve the stability of a control system is hinged on the following conditions:

1. Bandwidth considerations: If the original system is unstable or with a low stability margin, the additional phase lead required to realize a certain desired phase margin may be excessive. This may require a relatively large value of *a* for the controller, which, as a result, will give rise to a large bandwidth for the compensated system, and the transmission of high-frequency noise entering the system at the input may become objectionable. However, if the noise enters the system near the output, then the increased bandwidth may be beneficial to noise rejection. The larger bandwidth also has the advantage of robustness; that is, the system is insensitive to parameter variations and noise rejection as described before.
2. If the original system is unstable, or with low stability margin, the phase curve of the Bode plot of the forward-path transfer function has a steep negative slope near the gain-crossover frequency. Under this condition, the single-stage phase-lead controller may not be effective because the additional phase lead at the new gain crossover is added to a much smaller phase angle than that at the old gain crossover. The desired phase margin can be realized only by using a very large value of *a* for the controller. The amplifier gain *K* must be set to compensate *a*, so a large value for *a* requires a high-gain amplifier, which could be costly.

   As shown in Example 11-5-2, the compensated system may have a larger undershoot than overshoot. Often, a portion of the phase curve may still dip below the 180°, resulting in a **conditionally stable** system, even though the desired phase margin is satisfied.
3. The maximum phase lead available from a single-stage phase-lead

controller is less than 90°. Thus, if a phase lead of more than 90° is required, a multistage controller should be used.

11-5-5 Multistage Phase-Lead Controller

When the design with a phase-lead controller requires an additional phase of more than 90°, a multistage controller should be used. Figure 11-42 shows an op-amp-circuit realization of a two-stage phase-lead controller. The input-output transfer function of the circuit is

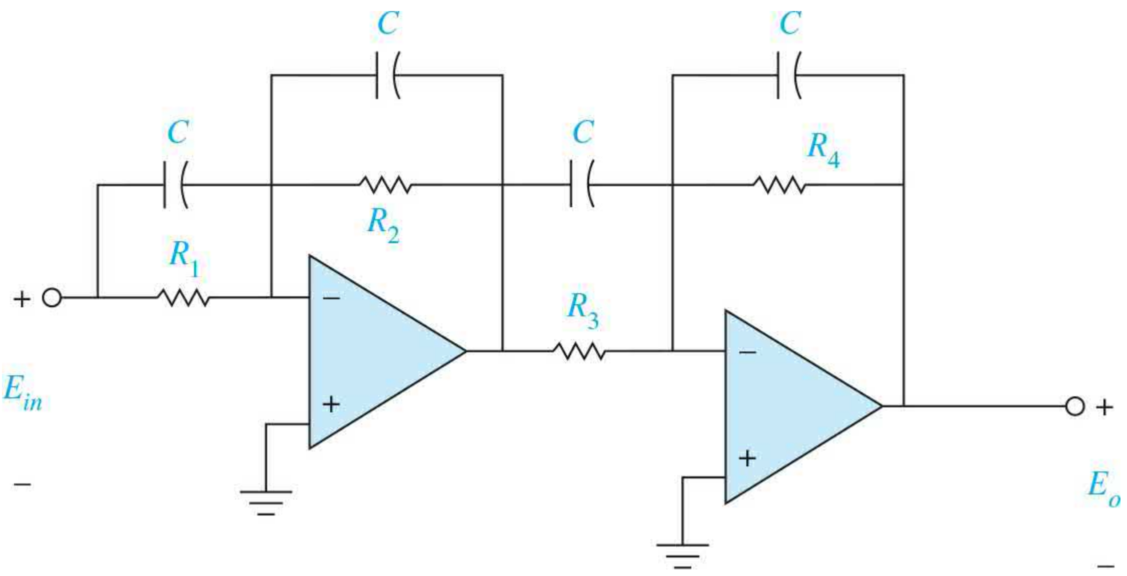


Figure 11-42 Two-stage phase-lead (phase-lag) controller.

$$G_c(s) = \frac{E_o(s)}{E_{\text{in}}(s)} = \left(\frac{s+\dfrac{1}{R_1C}}{s+\dfrac{1}{R_2C}}\right)\left(\frac{s+\dfrac{1}{R_3C}}{s+\dfrac{1}{R_4C}}\right)$$

$$= \frac{R_2R_4}{R_1R_3}\left(\frac{1+R_1Cs}{1+R_2Cs}\right)\left(\frac{1+R_3Cs}{1+R_4Cs}\right) \qquad (11\text{-}119)$$

or

$$G_c(s) = \frac{1}{a_1a_2}\left(\frac{1+a_1T_1s}{1+T_1s}\right)\left(\frac{1+a_2T_2s}{1+T_2s}\right) \qquad (11\text{-}120)$$

where $a_1 = R_1/R_2$, $a_2 = R_3/R_4$, $T_1 = R_2C$, and $T_2 = R_4C$. and $T_2 = R_4C$.

The design of a multistage phase-lead controller in the time domain becomes more cumbersome, since now there are more poles and zeros to be placed. The root-contour method also becomes impractical, since there are more variable parameters. The frequency-domain design in this case does represent a better choice of the design method. For example, for a two-stage controller, we can choose the parameters of the first stage of a two-stage controller so that a portion of the phase margin requirement is satisfied, and then the second stage fulfills the remaining requirement. In general, there is no reason why the two stages cannot be identical. The following example illustrates the design of a system with a two-stage phase-lead controller.

EXAMPLE 11-5-3 For the sun-seeker system designed in Example 11-5-2, let us alter the rise time and settling time requirements to be

Rise time $t_r \leq 0.001$ s

Settling time $t_s \leq 0.005$ s

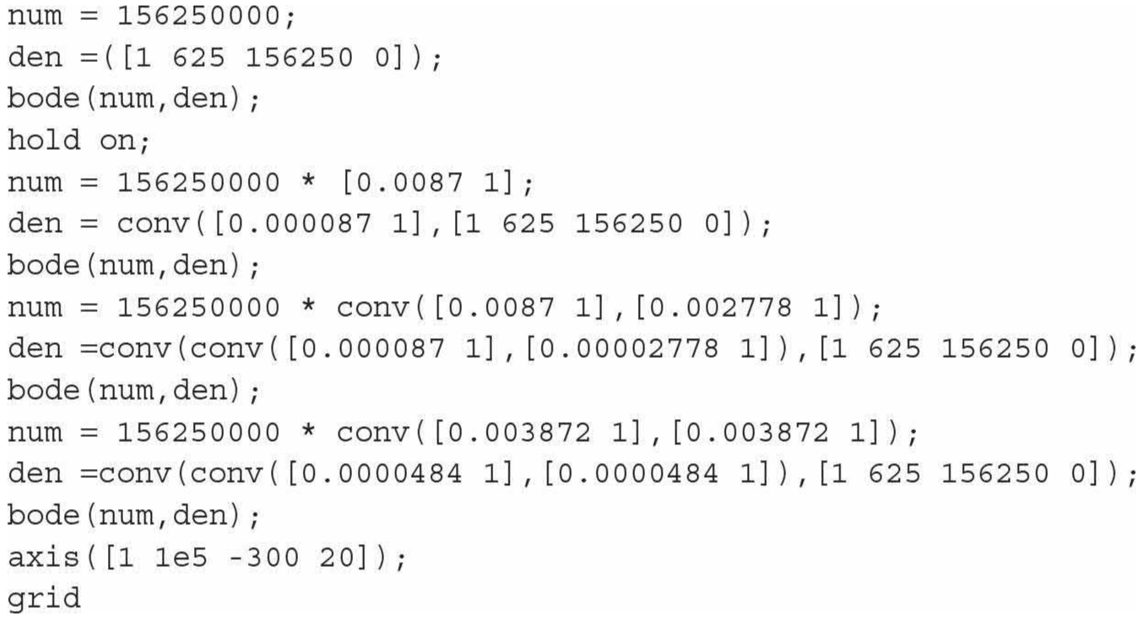
The other requirements are not altered. One way to meet faster rise time and settling time requirements is to increase the forward-path gain of the system. Let us consider that the forward-path transfer function is

$$G_p(s) = \frac{\Theta_o(s)}{A(s)} = \frac{156{,}250{,}000}{s(s^2 + 625s + 156{,}250)} \tag{11-121}$$

Another way of interpreting the change in the forward-path gain is that the ramp-error constant is increased to 1000 (up from 300 in Example 11-5-1). The Bode plot of $G_p(s)$ is shown in Fig. 11-43. The closed-loop system is unstable, with a phase margin of -15.43°.

Toolbox 11-5-8

Bode plots shown in Fig. 11-43 are obtained by the following sequence of MATLAB functions:



```
num = 156250000;
den =([1 625 156250 0]);
bode(num,den);
hold on;
num = 156250000 * [0.0087 1];
den = conv([0.000087 1],[1 625 156250 0]);
bode(num,den);
num = 156250000 * conv([0.0087 1],[0.002778 1]);
den =conv(conv([0.000087 1],[0.00002778 1]),[1 625 156250 0]);
bode(num,den);
num = 156250000 * conv([0.003872 1],[0.003872 1]);
den =conv(conv([0.0000484 1],[0.0000484 1]),[1 625 156250 0]);
bode(num,den);
axis([1 1e5 -300 20]);
grid
```

Because the compensated system in Example 11-5-2 had a phase margin of 60.55°, we would expect that, to satisfy the more stringent time response requirements in this example, the corresponding phase margin would have to be greater. Apparently, this increased phase margin cannot be realized with a single-stage phase-lead controller. It appears that a two-stage controller would be adequate.

The design involves some trial-and-error steps in arriving at a satisfied controller. Because we have two stages of controllers at our disposal, the design has a multitude of flexibility. We can set out by arbitrarily setting a_1 = 100 for the first stage of the phase-lead controller. The phase lead provided by the controller is obtained from Eq. (11-81),

$$\phi_m = \sin^{-1}\left(\frac{a_1 - 1}{a_1 + 1}\right) = \sin^{-1}\left(\frac{99}{101}\right) = 78.58° \qquad (11\text{-}122)$$

To maximize the effect of f_m, the new gain crossover should be at

$$-10\log_{10} a_1 = -10\log_{10} 100 = -20 \text{ dB} \qquad (11\text{-}123)$$

From Fig. 11-43 the frequency that corresponds to this gain on the amplitude curve is approximately 1150 rad/s. Substituting ω_{m1} and a_1 = 100 in Eq. (11-67), we get

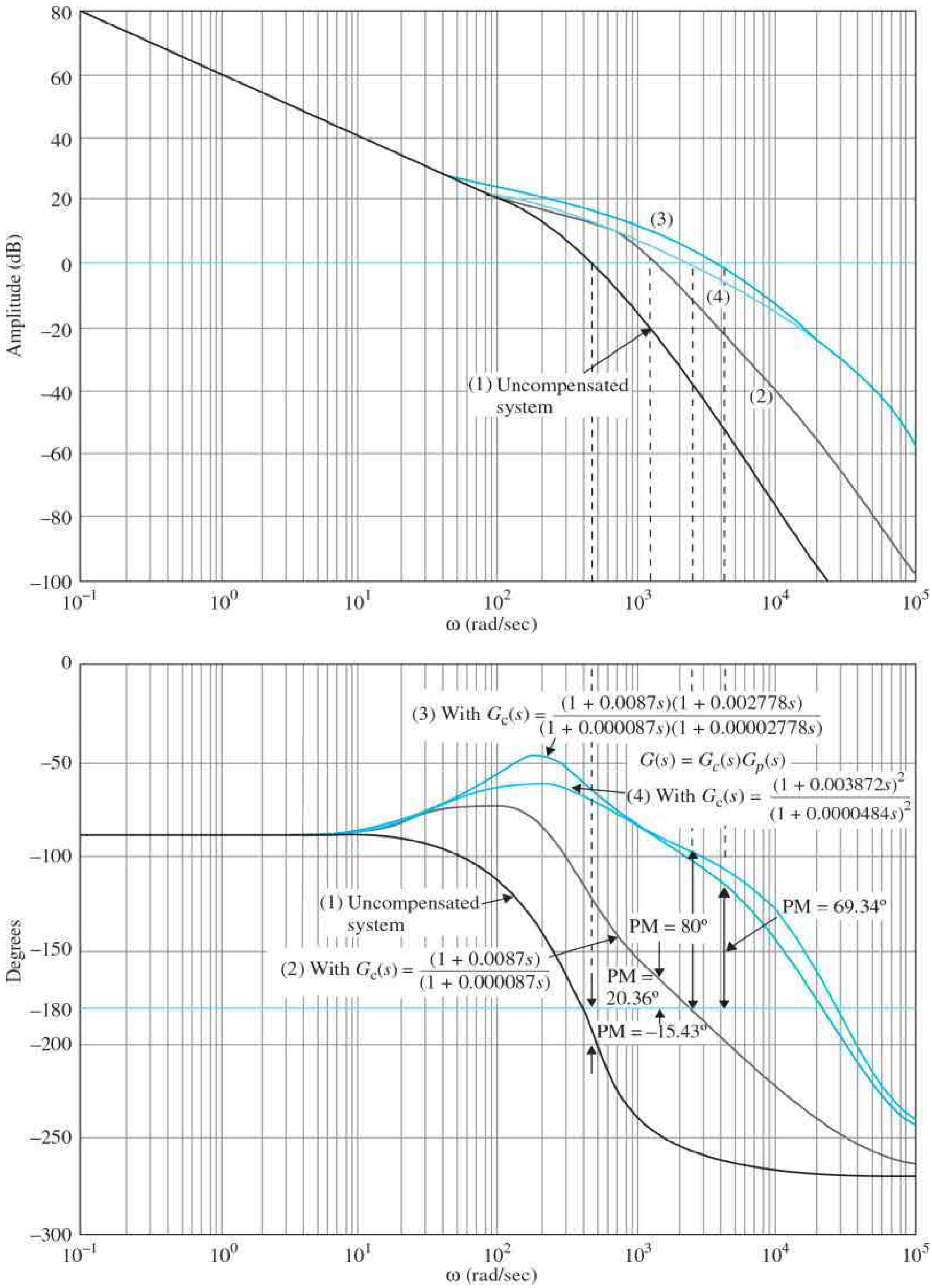

80
60
40
20
0
−20
−40
−60
−80
−100
Amplitude (dB)
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
ω (rad/sec)
(3)
(4)
(2)
(1) Uncompensated system
0
−50
−100
−150
−180
−200
−250
−300
Degrees
(3) With $G_c(s) = \frac{(1 + 0.0087s)(1 + 0.002778s)}{(1 + 0.000087s)(1 + 0.00002778s)}$
$G(s) = G_c(s)G_p(s)$
(4) With $G_c(s) = \frac{(1 + 0.003872s)^2}{(1 + 0.0000484s)^2}$
(1) Uncompensated system
(2) With $G_c(s) = \frac{(1 + 0.0087s)}{(1 + 0.000087s)}$
PM = 80°
PM = 69.34°
PM = 20.36°
PM = −15.43°
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
ω (rad/sec)

Figure 11-43 Bode plots of uncompensated and compensated sun-seeker systems in Example 11-5-2 with two-state phase-lead controller.
$G_p(s)=\dfrac{156{,}250{,}000}{s(s^2+625s+156{,}250)}$.

$$T_1=\frac{1}{\omega_{m1}\sqrt{a_1}}=\frac{1}{1150\sqrt{100}}=0.000087 \tag{11-124}$$

The forward-path transfer function with the one-stage phase-lead controller is

$$G(s)=\frac{156{,}250{,}000(1+0.0087s)}{s(s^2+625s+156{,}250)(1+0.000087s)} \tag{11-125}$$

The Bode plot of the last equation is drawn as curve (2) in Fig. 11-43. We see that the phase margin of the interim design is only 20.36°. Next, we arbitrarily set the value of a_2 of the second stage at 100. From the Bode plot of the transfer function of Eq. (11-125) in Fig. 11-43, we find that the frequency at which the magnitude of $G(j\omega)$ is –20 dB is approximately 3600 rad/s. Thus,

$$T_2=\frac{1}{\omega_{m2}\sqrt{a_2}}=\frac{1}{3600\sqrt{100}}=0.00002778 \tag{11-126}$$

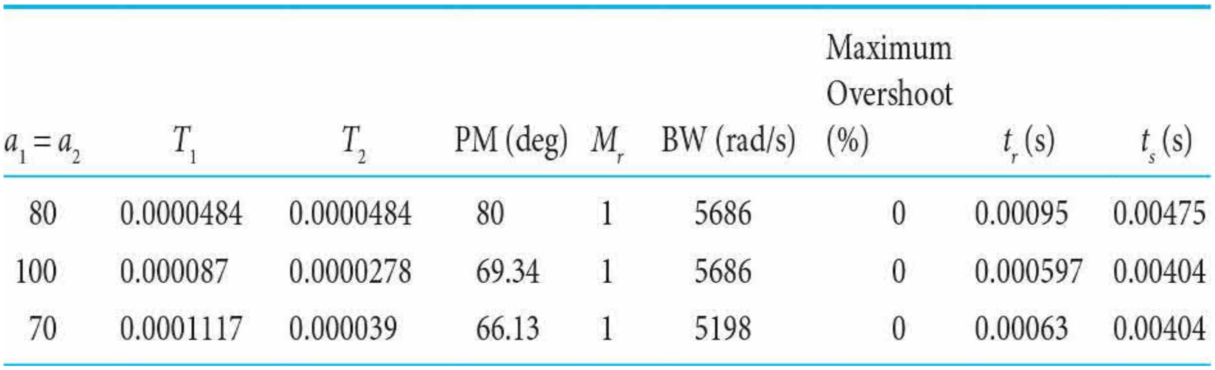
The forward-path transfer function of the sun-seeker system with the two-stage phase-lead controller is ($a_1 = a_2 = 100$)

$$G(s)=\frac{156{,}250{,}000(1+0.0087s)(1+0.002778s)}{s(s^2+625s+156{,}250)(1+0.000087s)(1+0.00002778s)} \tag{11-127}$$

Figure 11-43 shows the Bode plot of the sun-seeker system with the two-stage phase-lead controller designed above [curve (3)]. As seen from Fig. 11-43, the phase margin of the system with $G(s)$ given in Eq. (11-127) is 69.34º. As shown by the system attributes in Table 11-17, the system satisfies all the time-domain specifications. In fact, the selection of $a_1 = a_2 = 100$ appears to be overly stringent. To show that the design is not critical, we can select $a_1 = a_2 = 80$, and then 70 and the time-domain specifications are still satisfied.

Following similar design steps, we arrived at $T_1 = 0.0001117$ and $T_2 = 0.000039$ for $a_1 = a_2 = 70$, and $T_1 = T_2 = 0.0000484$ for $a_1 = a_2 = 80$. Curve (4) of Fig. 11-43 shows the Bode plot of the compensated system with $a_1 = a_2 = 80$. Table 11-17 summarizes all the attributes of the system performance with these three controllers.

Table 11-17 Attributes of Sun-Seeker System in Example 11-5-3 with Two-Stage Phase-Lead Controller



$a_1 = a_2$	T_1	T_2	PM (deg)	M_r	BW (rad/s)	Maximum Overshoot (%)	t_r (s)	t_s (s)
80	0.0000484	0.0000484	80	1	5686	0	0.00095	0.00475
100	0.000087	0.0000278	69.34	1	5686	0	0.000597	0.00404
70	0.0001117	0.000039	66.13	1	5198	0	0.00063	0.00404

The unit-step responses of the system with the two-stage phase-lead controller for $a_1 = a_2 = 80$ and 100 are shown in Fig. 11-44.

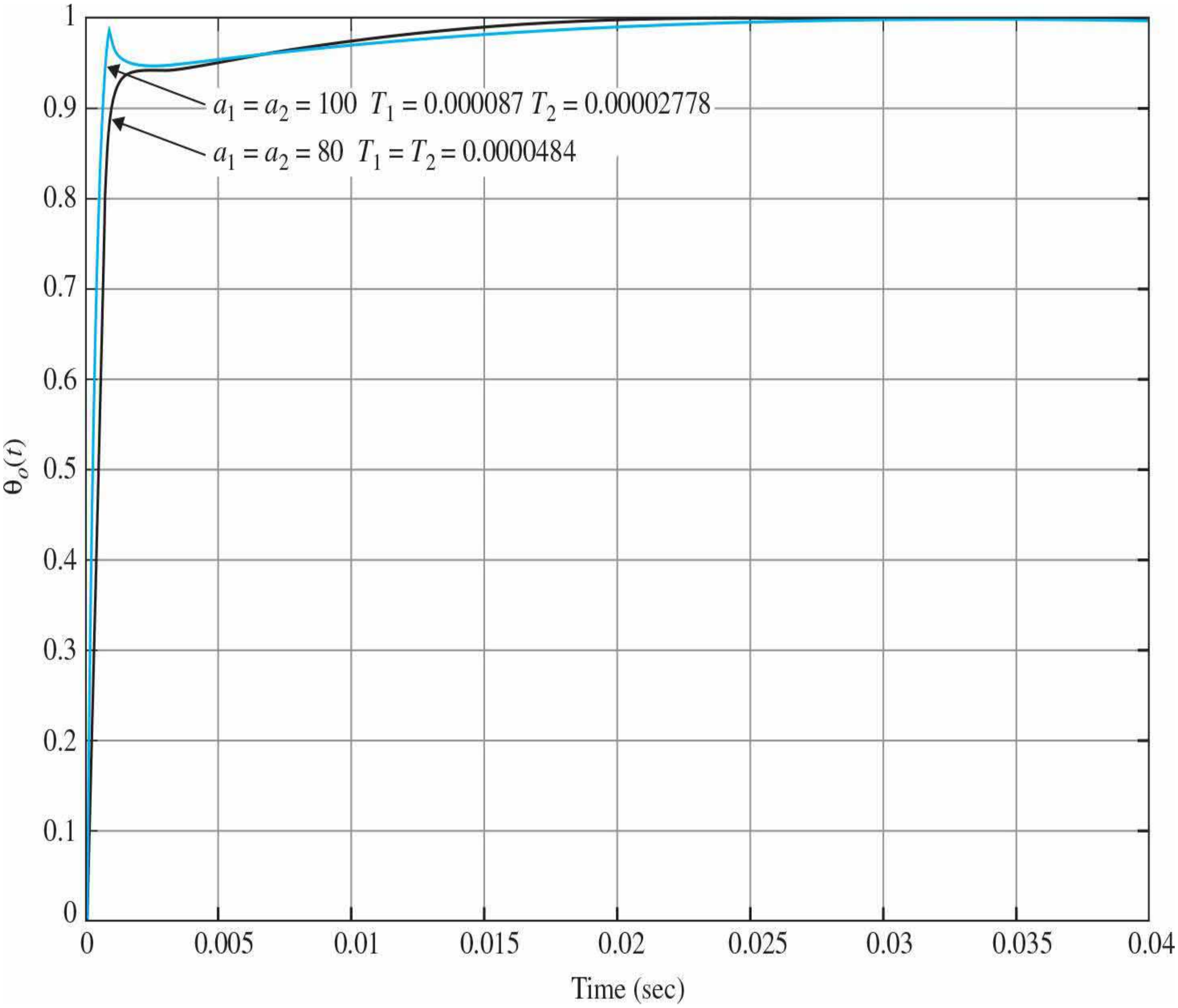
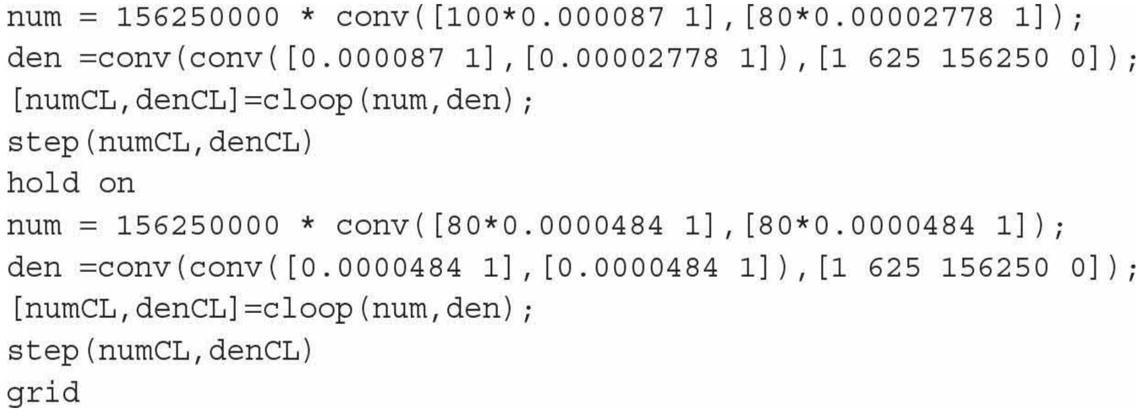


Figure 11-44 Unit-step responses of sun-seeker system in Example 11-5-2 with two-stage phase-lead controller.

$$G_c(s) = \left(\frac{1+a_1T_1s}{1+T_1s}\right)\left(\frac{1+a_2T_2s}{1+T_2s}\right) \quad G_p(s) = \frac{156{,}250{,}000}{s(s^2+625s+156{,}250)}.$$

Toolbox 11-5-9

Figure 11-44 is obtained by the following sequence of MATLAB functions:



```
num = 156250000 * conv([100*0.000087 1],[80*0.00002778 1]);
den =conv(conv([0.000087 1],[0.00002778 1]),[1 625 156250 0]);
[numCL,denCL]=cloop(num,den);
step(numCL,denCL)
hold on
num = 156250000 * conv([80*0.0000484 1],[80*0.0000484 1]);
den =conv(conv([0.0000484 1],[0.0000484 1]),[1 625 156250 0]);
[numCL,denCL]=cloop(num,den);
step(numCL,denCL)
grid
```

11-5-6 Sensitivity Considerations

The sensitivity function defined in Sec. 10-15, Eq. (11-138), can be used as a design specification to indicate the robustness of the system. The sensitivity of the closed-loop transfer function with respect to the variations of the forward-path transfer function is defined as

$$S_G^M(s) = \frac{\partial M(s)/M(s)}{\partial G(s)/G(s)} = \frac{G^{-1}(s)}{1+G^{-1}(s)} = \frac{1}{1+G(s)} \qquad (11\text{-}128)$$

The plot of $\left|S_G^M(j\omega)\right|$ versus frequency gives an indication of the sensitivity of the system as a function of frequency. The ideal robust situation is for $\left|S_G^M(j\omega)\right|$ to assume a small value (<<1) over a wide range of frequencies. As an example, the sensitivity function of the sun-seeker system designed in Example 11-5-2 with the one-stage phase-lead controller with *a* = 100 and *T* = 0.00005 is plotted as shown in Fig. 11-45. Note that the sensitivity function is low at low frequencies and is less than unity for ω <400 rad/s. Although the sun-seeker system in Example 11-5-2 does not need a multistage phase-lead controller, we shall show that, if a two-stage phase-lead controller is used, not only the value of *a* will be substantially reduced, resulting in lower gains for the op-amps, but the system will be more robust. Following the design procedure outlined in Example 11-5-3, a two-stage phase-lead controller is designed for the sun-seeker system with the process transfer function described by Eq. (11-104).

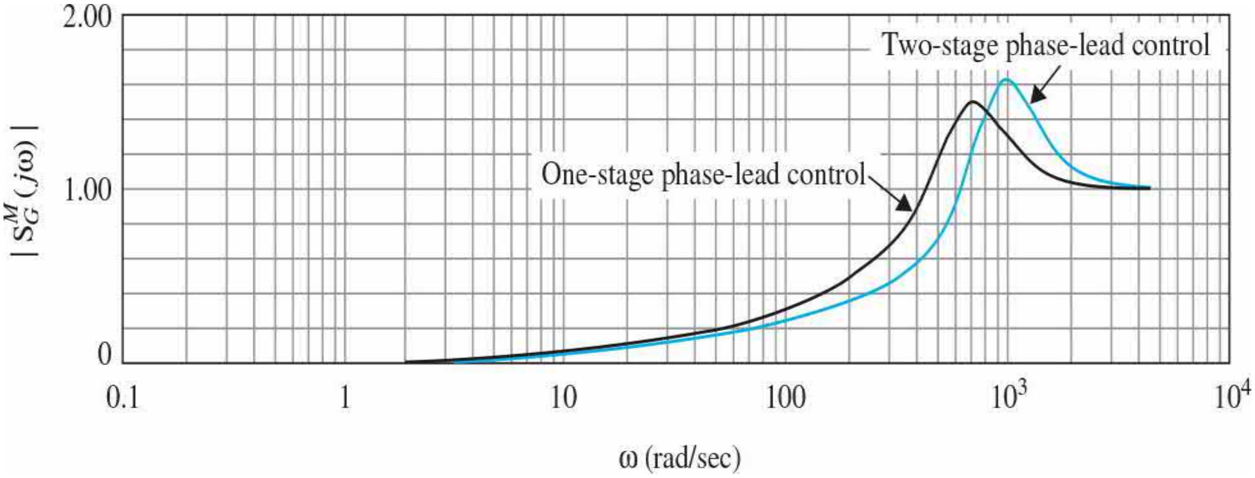


Figure 11-45 Sensitivity functions of sun-seeker system in Example 11-5-2.

The parameters of the controller are $a_1 = a_2 = 5.83$ and $T_1 = T_2 = 0.000673$. The forward-path transfer function of the compensated system is

$$G(s) = \frac{4.6875 \times 10^7 (1 + 0.0039236s)^2}{s(s^2 + 625s + 156{,}250)(1 + 0.000673s)^2} \tag{11-129}$$

Figure 11-45 shows that the sensitivity function of the system with the two-stage phase-lead controller is less than unity for ω<600 rad/s. Thus, the system with the two-stage phase-lead controller is more robust than the system with the single-stage controller. The reason for this is that the more robust system has a higher bandwidth. In general, systems with phase-lead control will be more robust due to the higher bandwidth. However, Fig. 11-45 shows that the system with the two-stage phase-lead controller has a higher sensitivity at high frequencies.

11-5-7 Time-Domain Interpretation and Design of Phase-Lag Control

The transfer function in Eq. (11-72) represents a phase-lag controller or low-pass filter when $a < 1$. The transfer function is repeated as follows:

$$G_c(s) = \frac{1}{a}\left(\frac{1 + aTs}{1 + Ts}\right) \quad a < 1 \tag{11-130}$$

The pole-zero configuration of $G_c(s)$ is shown in Fig. 11-46. Unlike the PI controller, which provides a pole at $s = 0$, the phase-lag controller affects the steady-state error only in the sense that the zero-frequency gain of $G_c(s)$ is greater than unity. Thus, any error constant that is finite and nonzero will be increased by the factor $1/a$ from the phase-lag controller.

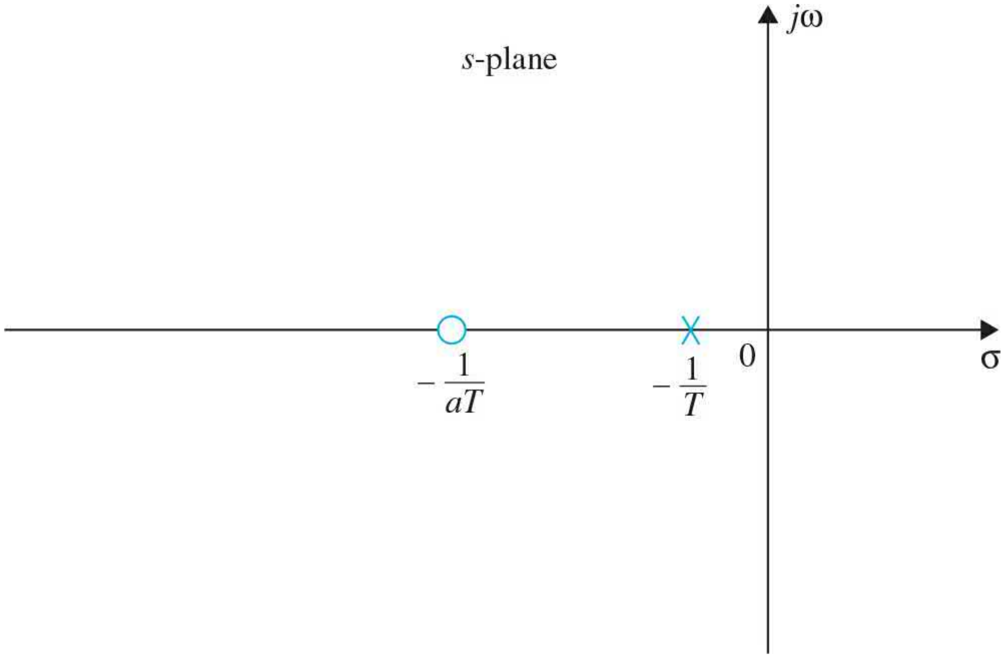


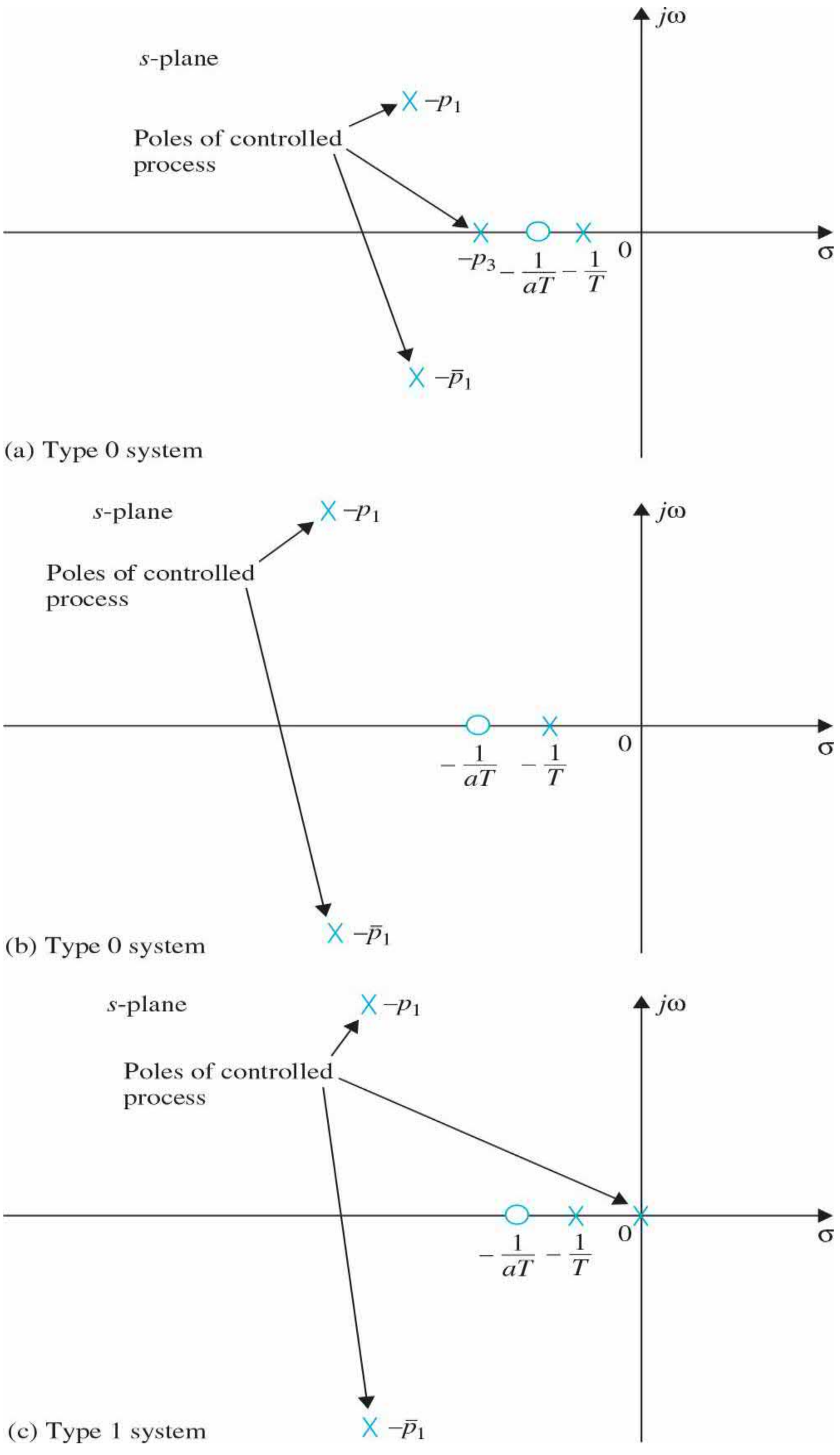
Figure 11-46 Pole-zero configuration of phase-lag controller.

Because the pole at s=-1/T is to the right of the zero at -1/aT , effective use of the phase-lag controller to improve damping would have to follow the same design principle of the PI control presented in Sec. 11-3. Thus, *the proper way of applying the phase-lag control is to place the pole and zero close together. For type 0 and type 1 systems, the combination should be located near the origin in the* s*-plane*. Figure 11-47 illustrates the design strategies in the *s*-plane for type 0 and type 1 systems. Phase-lag control should not be applied to a type 2 system.

The design principle described above can be explained by considering that the controlled process of a type 0 control system is

$$G_p(s) = \frac{K}{(s+p_1)(s+\overline{p}_1)(s+p_3)} \qquad (11\text{-}131)$$

where p_1 and $\overline{p}_1$ are complex-conjugate poles, such as the situation shown in Fig. 11-47.



(a) Type 0 system

(b) Type 0 system

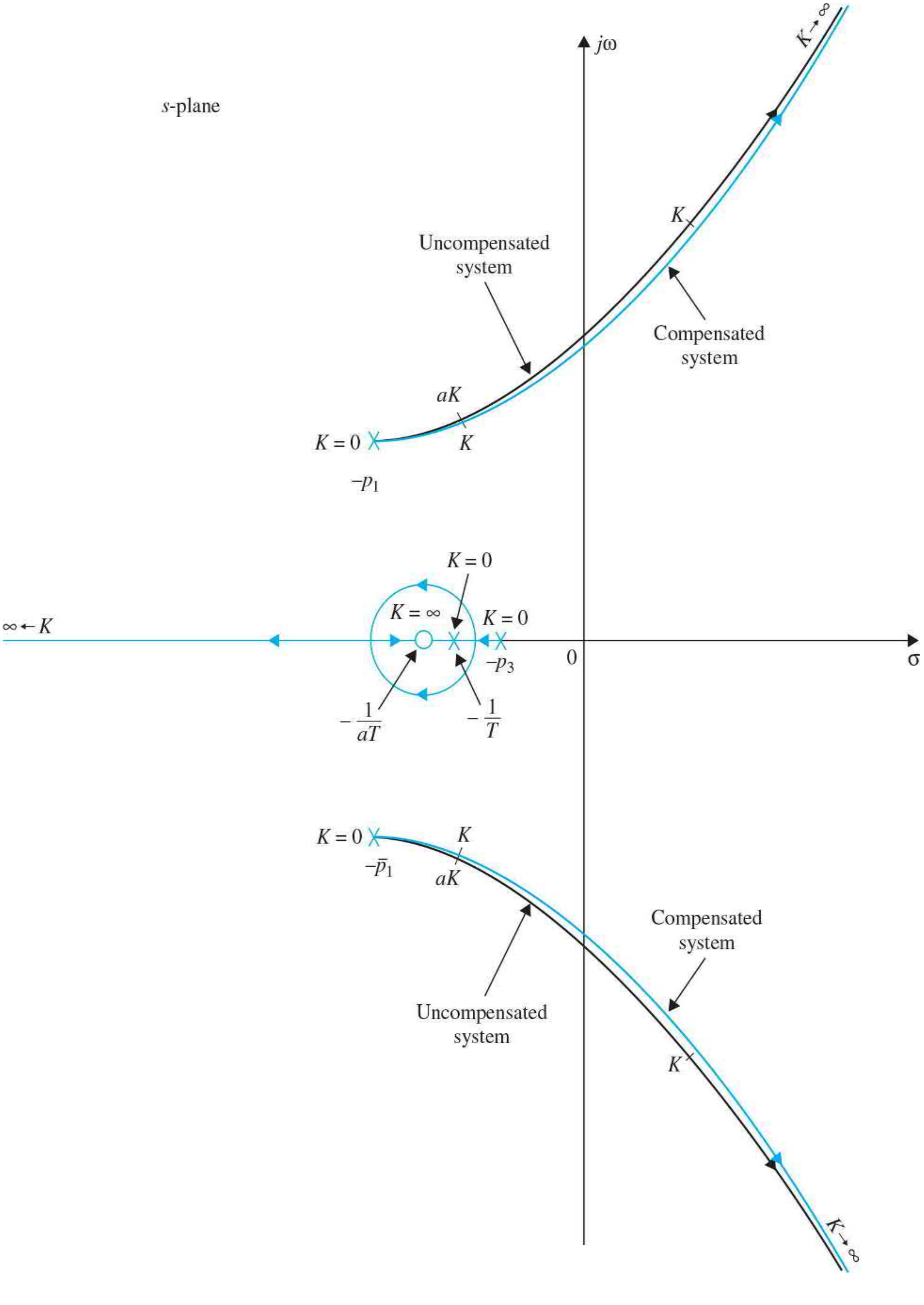
(c) Type 1 system

Figure 11-47 Design strategies for phase-lag control for type 0 and type 1 systems.

Just as in the case of the phase-lead controller, we can drop the gain factor 1/*a* in Eq. (11-131) because whatever the value of *a* is, the value of *K* can be adjusted to compensate for it. Applying the phase-lag controller of Eq. (11-131), without the factor 1/*a*, to the system, the forward-path transfer function becomes

$$G(s)=G_c(s)G_p(s)=\frac{K(1+aTs)}{(s+p_1)(s+p.macr_1)(s+p_3)(1+Ts)} \quad (a<1) \qquad (11\text{-}132)$$

Let us assume that the value of *K* is set to meet the steady-state-error requirement. Also assume that, with the selected value of *K*, the system damping is low or even unstable. Now let $1/T \cong 1/aT$, and place the pole-zero pair near the pole at -1/p_3, as shown in Fig. 11-47. Figure 11-48 shows the root loci of the system with and without the phase-lag controller. Because the pole-zero combination of the controller is very close to the pole at -1/p_3, the shape of the loci of the dominant roots with and without the phase-lag control will be very similar. This is easily explained by writing Eq. (11-132) as



s-plane
$j\omega$
$K \rightarrow \infty$
K
Uncompensated system
Compensated system
aK
$K = 0$
K
$-p_1$
$K = 0$
$K = \infty$
$K = 0$
$\infty \leftarrow K$
$-p_3$
0
σ
$-\frac{1}{aT}$
$-\frac{1}{T}$
$K = 0$
K
$-\bar{p}_1$
aK
Compensated system
Uncompensated system
K
$K \rightarrow \infty$

Figure 11-48 Root loci of uncompensated and phase-lag-compensated systems.

$$G(s) = \frac{Ka(s+1/aT)}{(s+p_1)(s+\bar{p}_1)(s+p_3)(s+1/T)}$$
$$\cong \frac{Ka}{(s+p_1)(s+\bar{p}_1)(s+p_3)} \tag{11-133}$$

Because *a* is less than 1, the application of phase-lag control is equivalent to reducing the forward-path gain from *K* to *Ka*, *while not affecting the steady-state performance of the system*. Figure 11-48 shows that the value of *a* can be chosen so that the damping of the compensated system is satisfactory. Apparently, the amount of damping that can be added is limited if the poles -p_1 and -*p.macr*$_1$ are very close to the imaginary axis. Thus, we can select *a* using the following equation:

$$a = \frac{K \text{ to realize the desired damping}}{K \text{ to realize the steady} - \text{state performance}} \tag{11-134}$$

The value of *T* should be so chosen that the pole and zero of the controller are very close together and close to -1/p_3.

In the time domain, phase-lag control generally has the effect of increasing the rise time and settling time.

11-5-8 Frequency-Domain Interpretation and Design of Phase-Lag Control

The transfer function of the phase-lag controller can again be written as

$$G_c(s) = \frac{1+aTs}{1+Ts} \quad (a<1) \tag{11-135}$$

by assuming that the gain factor -1/*a* is eventually absorbed by the forward gain *K*. The Bode diagram of Eq. (11-135) is shown in Fig. 11-49. The magnitude curve has corner frequencies at ω = 1/*aT* and 1/*T*. Because the transfer functions of the phase-lead and phase-lag controllers are identical in form, except for the value of *a*, the maximum phase lag φ_m of the phase curve of Fig. 11-49 is given by

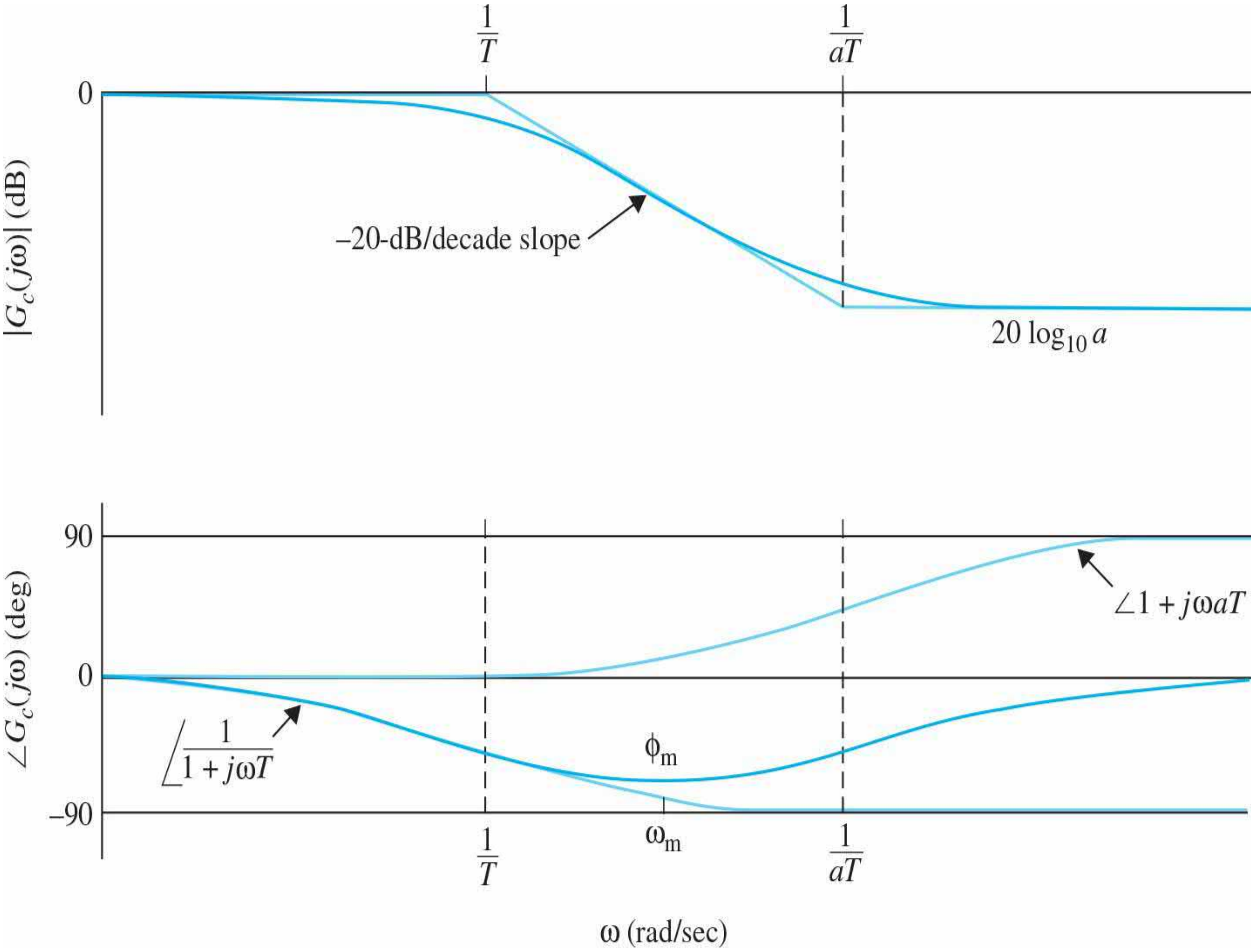


Figure 11-49 Bode diagram of the phase-lag controller.

$$G_c(s) = \frac{(1+aTs)}{(1+Ts)} \quad a < 1 .$$

$$\phi_m = \sin^{-1}\left(\frac{a-1}{a+1}\right) \quad (a<1) \qquad (11\text{-}136)$$

Figure 11-49 shows that the phase-lag controller essentially provides an attenuation of 20 $\log_{10} a$ at high frequencies. Thus, unlike the phase-lead control that utilizes the maximum phase lead of the controller, phase-lag control utilizes the attenuation of the controller at high frequencies. This is parallel to the situation of introducing an attenuation of a to the forward-path gain in the root-locus design. For phase-lead control, the objective of the controller is to increase the phase of the open-loop system in the vicinity of the gain crossover while attempting to locate the maximum phase lead at the new gain crossover. In phase-lag control, the objective is to move the gain crossover to a lower frequency where the desired phase margin is realized, while keeping the phase curve of the Bode plot relatively unchanged at the new gain crossover.

The design procedure for phase-lag control using the Bode plot is outlined as follows:

1. The Bode plot of the forward-path transfer function of the uncompensated system is drawn. The forward-path gain K is set according to the steady-state performance requirement.
2. The phase and gain margins of the uncompensated system are determined from the Bode plot.
3. Assuming that the phase margin is to be increased, the frequency at which the desired phase margin is obtained is located on the Bode plot. This frequency is also the new gain crossover frequency ω'_g, where the compensated magnitude curve crosses the 0-dB-axis.
4. To bring the magnitude curve down to 0 dB at the new gain-crossover frequency ω'_g, the phase-lag controller must provide the amount of attenuation equal to the value of the magnitude curve at ω'_g. In other words,

$$|G_p(j\omega'_g)| = -20\log_{10} a \text{ dB} \quad (a<1) \qquad (11\text{-}137)$$

Solving for *a* from the last equation, we get

$$a = 10^{-|G_p(j\omega'_g)|/20} \quad (a < 1) \tag{11-138}$$

Once the value of *a* is determined, it is necessary only to select the proper value of *T* to complete the design. Using the phase characteristics shown in Fig. 11-45, if the corner frequency 1/*aT* is placed far below the new gain-crossover frequency ω'_g, the phase lag of the controller will not appreciably affect the phase of the compensated system near ω'_g. On the other hand, the value of 1/*aT* should not be too small because the bandwidth of the system will be too low, causing the system to be too sluggish and less robust. Usually, as a general guideline, the frequency 1/*aT* should be approximately one decade below ω'_g; that is,

$$\frac{1}{aT} = \frac{\omega'_g}{10} \text{ rad/s} \tag{11-139}$$

Then,

$$\frac{1}{T} = \frac{a\omega'_g}{10} \text{ rad/s} \tag{11-140}$$

5. The Bode plot of the compensated system is investigated to see if the phase margin requirement is met; if not, the values of *a* and *T* are readjusted, and the procedure is repeated. If design specifications involve gain margin, M_r, or BW, then these values should be checked and satisfied.

Because the phase-lag control brings in more attenuation to a system, then if the design is proper, the stability margins will be improved but at the expense of lower bandwidth. The only benefit of lower bandwidth is reduced sensitivity to high-frequency noise and disturbances.

The following example illustrates the design of the phase-lag controller and all its ramifications.

EXAMPLE 11-5-4 For the sun-seeker system designed in Example 11-5-1, the forward-path transfer function of the uncompensated system is

$$G_p(s) = \frac{\Theta_o(s)}{A(s)} = \frac{2500K}{s(s+25)} \qquad (11\text{-}141)$$

Time-Domain Design The time-domain specifications of the system are as follows:

1. The steady-state error of α(*t*) due to a unit-ramp function input for $\theta_r(t)$ should be ≤1/300 rad/rad/s of the final steady-state output velocity.
2. The maximum overshoot of the step response should be less than 5% or as small as possible.
3. Rise time t_r ≤0.5 s.
4. Settling time t_s ≤0.5 s.
5. Due to noise problems, the bandwidth of the system must be <50 rad/s.

Notice that the rise-time and settling-time requirements have been relaxed considerably from the phase-lead design in Example 11-5-1. The root loci of the uncompensated system are shown in Fig. 11-50*a*.

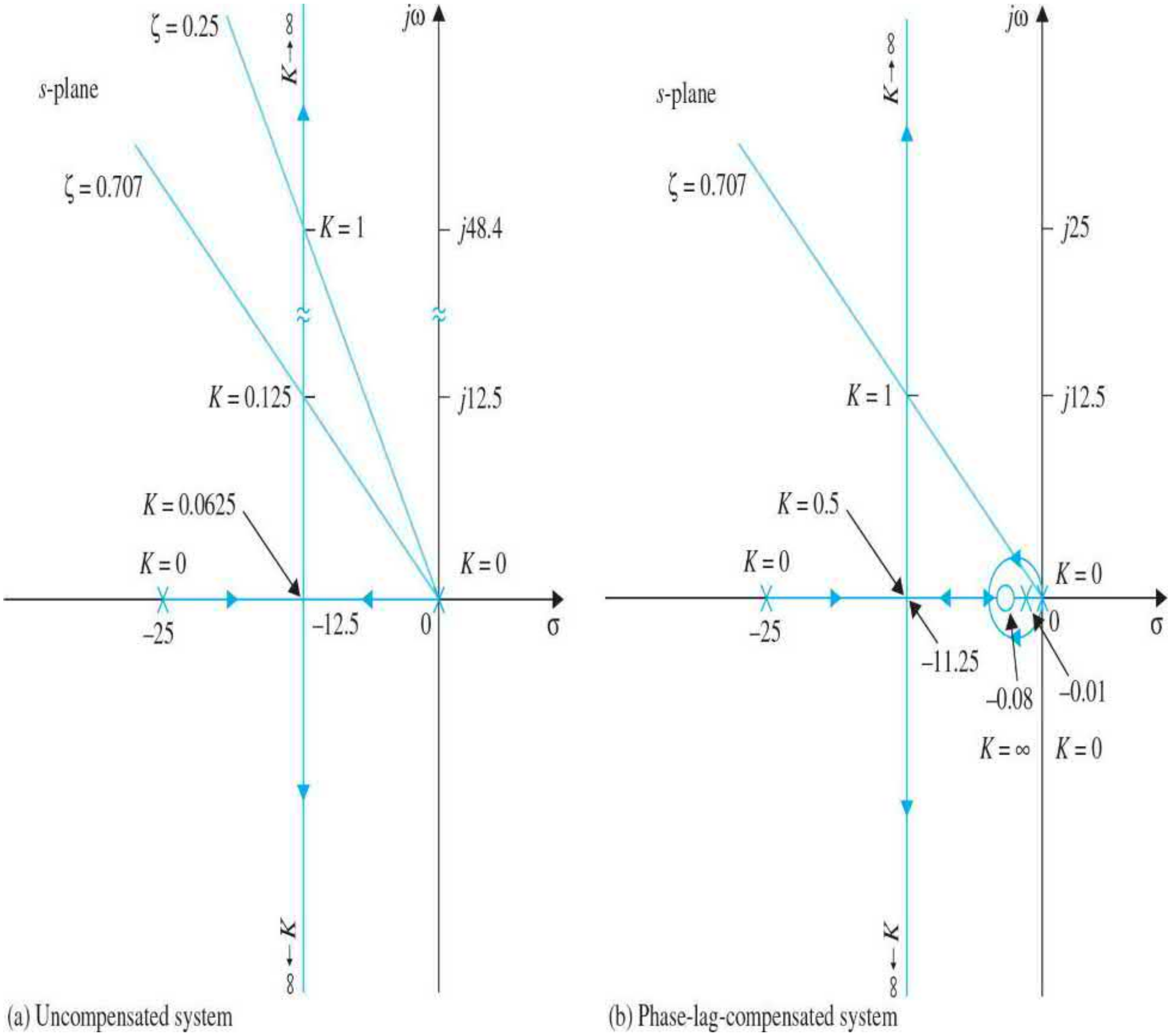


Figure 11-50 Root loci of sun-seeker system in Example 11-5-4.

$$G_p(s) = \frac{2500K}{s(s+25)} \quad G_c(s) = \frac{1+aTs}{1+Ts} \quad a = 0.125 \quad T = 100.$$

As in Example 11-5-1, we set $K = 1$ initially. The damping ratio of the uncompensated system is 0.25, and the maximum overshoot is 44.4 percent. Figure 11-51 shows the unit-step response of the system with $K = 1$.

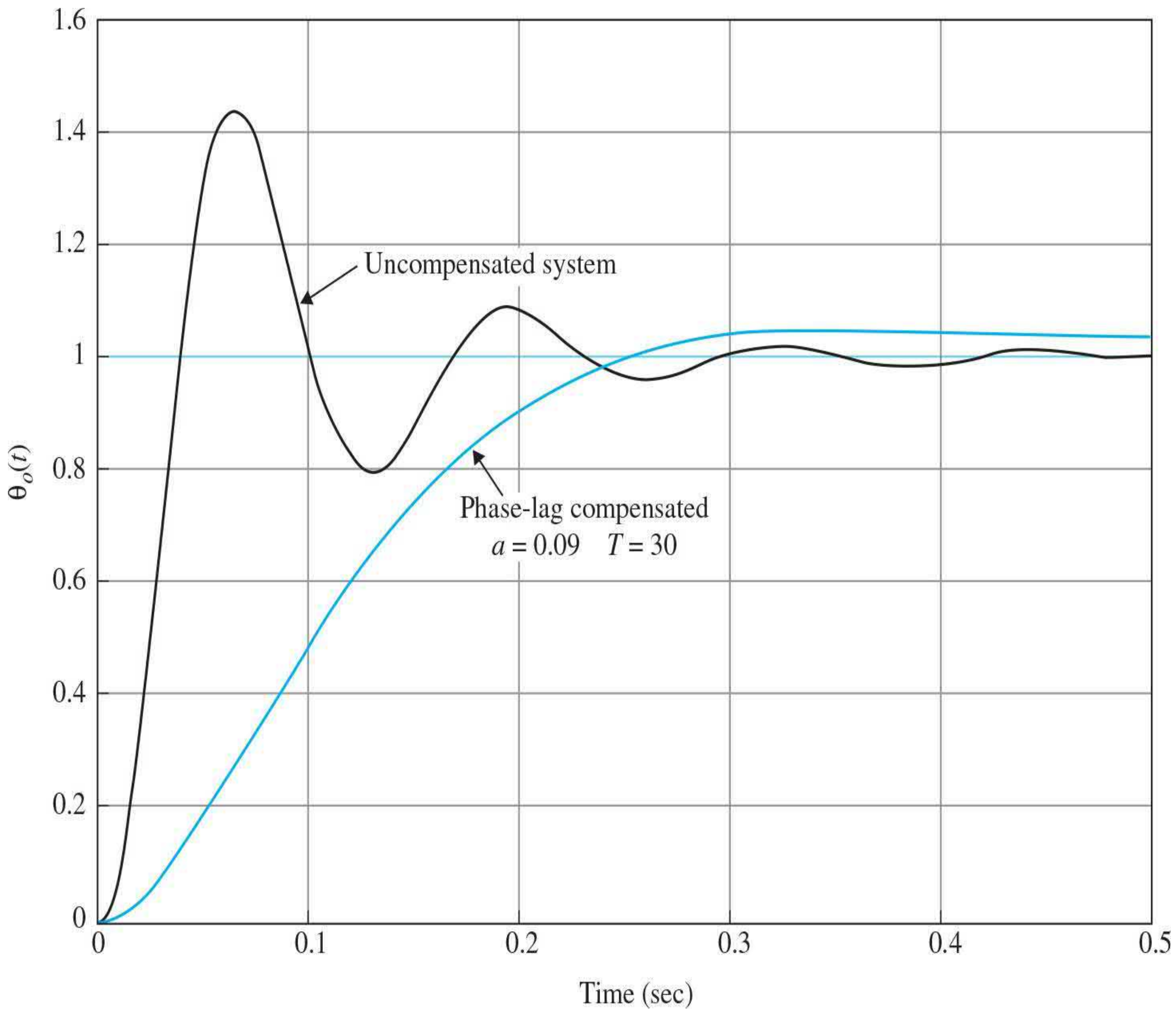


Figure 11-51 Unit-step responses of uncompensated and compensated sun-seeker systems with phase-lag controller in Example 11-5-4.

$$G_p(s) = \frac{2500\ K}{s(s+25)} \quad G_c(s) = \frac{1+aTs}{1+Ts} \quad a = 0.09 \quad T = 30.$$

Let us select the phase-lag controller with the transfer function given in Eq. (11-130). The forward-path transfer function of the compensated system is

$$G(s) = G_c(s)G_p(s) = \frac{2500K(s+1/aT)}{s(s+25)(s+1/T)} \tag{11-142}$$

If the value of K is maintained at 1, the steady-state error will be a percent, which is better than that of the uncompensated system, since $a < 1$. *For* *effective phase-lag control, the pole and zero of the controller transfer function should be placed close together, and then for the type 1 system, the combination should be located relatively close to the origin of the* s*-plane*. From the root loci of the uncompensated system in Fig. 11-50*a*, we see that, if K could be set to 0.125, the damping ratio would be 0.707, and the maximum overshoot of the system would be 4.32 percent. By setting the pole and zero of the controller close to $s = 0$, the shape of the loci of the dominant roots of the compensated system will be very similar to those of the uncompensated system. We can find the value of a using Eq. (11-124), that is,

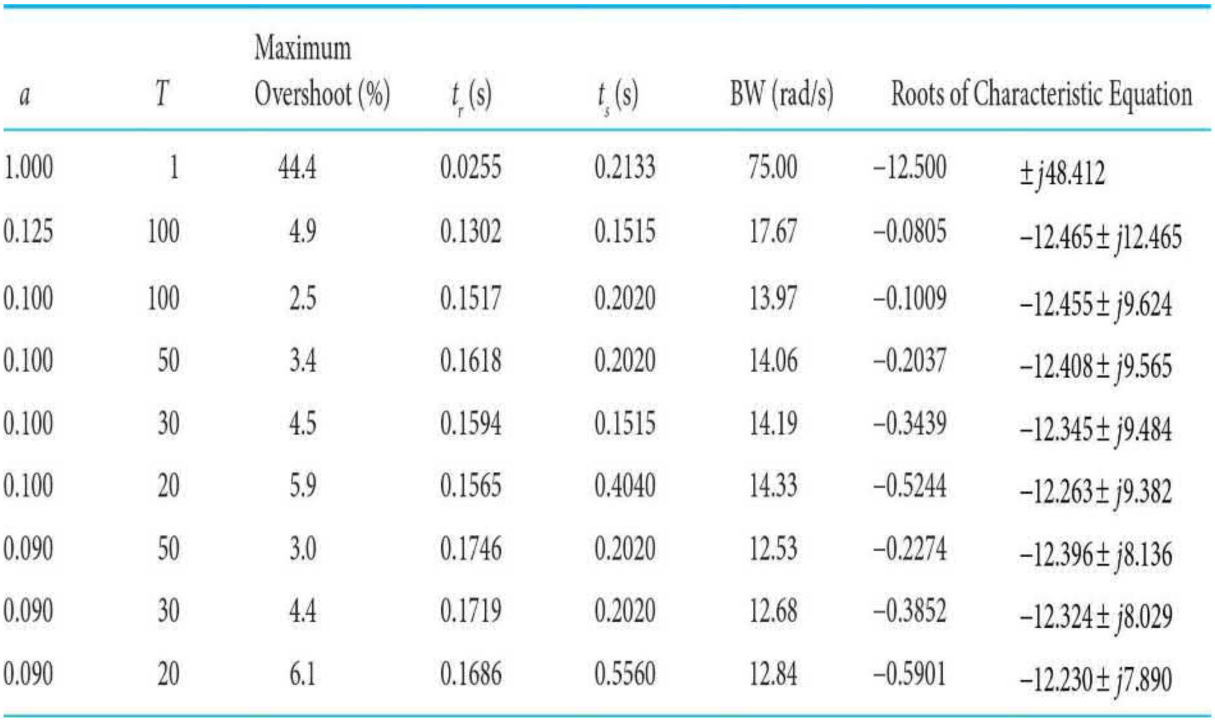
$$a = \frac{K \text{ to realize the desired damping}}{K \text{ to realize the steady} - \text{state performance}} = \frac{0.125}{1} = 0.125 \qquad (11\text{-}143)$$

Thus, if the value of T is sufficiently large, when $K = 1$, the dominant roots of the characteristic equation will correspond to a damping ratio of approximately 0.707. Let us arbitrarily select $T = 100$. The root loci of the compensated system are shown in Fig. 11-50*b*. The roots of the characteristic equation when $K = 1$, $a = 0.125$, and $T = 100$ are

$$s = -0.0805, \quad -12.465 + j12.465, \quad \text{and} \quad -12.465 - j12.465$$

which corresponds to a damping ratio of exactly 0.707. If we had chosen a smaller value for T, then the damping ratio would be slightly off 0.707. From a practical standpoint, the value of T cannot be too large, since from Eq. (11-74), $T = R_2C$, a large T would correspond to either a large capacitor or an unrealistically large resistor. To reduce the value of T and simultaneously satisfy the maximum overshoot requirement, a should also be reduced. However, a cannot be reduced indefinitely, or the zero of the controller at $-1/aT$ would be too far to the left on the real axis. Table 11-18 gives the attributes of the time-domain performance of the phase-lag compensated sun-seeker system with various values for a and T. The ramifications of the various design parameters are clearly displayed.

Table 11-18 Attributes of Performance of Sun-Seeker System in Example 11-6-1 with Phase-Lag Controller



a	T	Maximum Overshoot (%)	t_r (s)	t_s (s)	BW (rad/s)	Roots of Characteristic Equation	
1.000	1	44.4	0.0255	0.2133	75.00	−12.500	$\pm j48.412$
0.125	100	4.9	0.1302	0.1515	17.67	−0.0805	$-12.465 \pm j12.465$
0.100	100	2.5	0.1517	0.2020	13.97	−0.1009	$-12.455 \pm j9.624$
0.100	50	3.4	0.1618	0.2020	14.06	−0.2037	$-12.408 \pm j9.565$
0.100	30	4.5	0.1594	0.1515	14.19	−0.3439	$-12.345 \pm j9.484$
0.100	20	5.9	0.1565	0.4040	14.33	−0.5244	$-12.263 \pm j9.382$
0.090	50	3.0	0.1746	0.2020	12.53	−0.2274	$-12.396 \pm j8.136$
0.090	30	4.4	0.1719	0.2020	12.68	−0.3852	$-12.324 \pm j8.029$
0.090	20	6.1	0.1686	0.5560	12.84	−0.5901	$-12.230 \pm j7.890$

Thus, a suitable set of controller parameters would be $a = 0.09$ and $T = 30$. With $T = 30$, selecting C = 1 μF would require R_2 to be 30 M Ω. A smaller value for T can be realized by using a two-stage phase-lag controller. The unit-step response of the compensated system with $a = 0.09$ and $T = 30$ is shown in Fig. 11-52. Notice that the maximum overshoot is reduced at the expense of rise time and settling time. Although the settling time of the compensated system is shorter than that of the uncompensated system, it actually takes much longer for the phase-lag-compensated system to reach steady state.

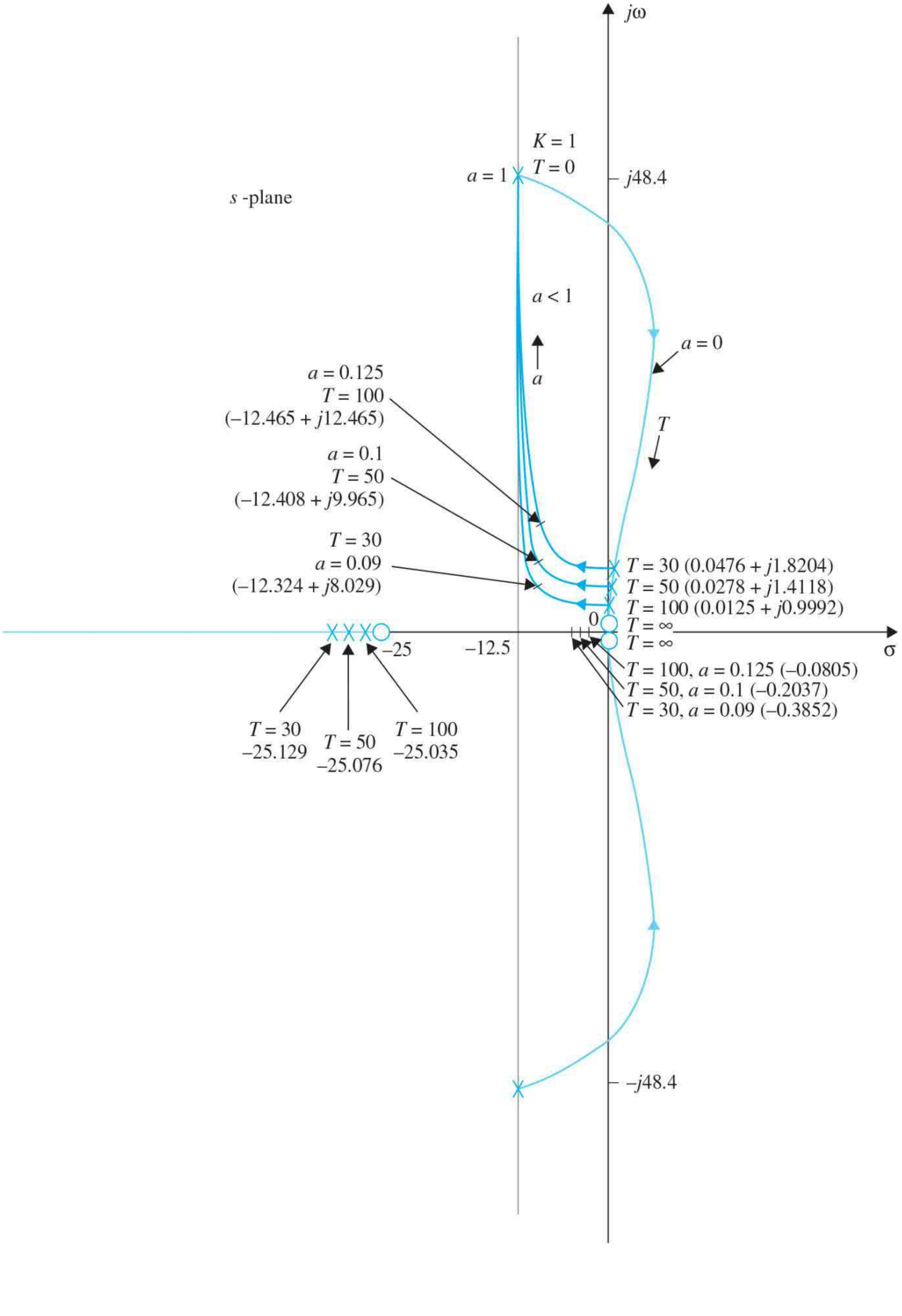
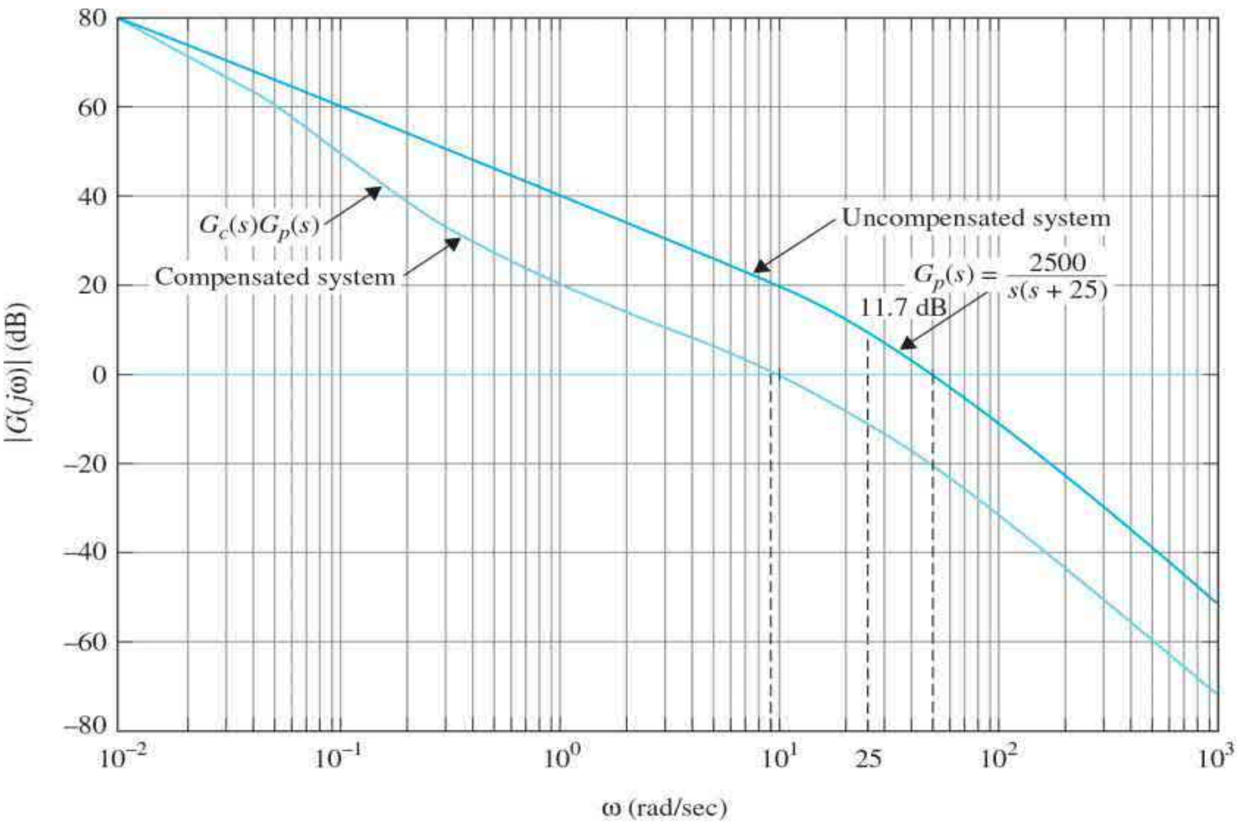

jω
s -plane
K = 1
T = 0
a = 1
j48.4
a < 1
a
a = 0
T
a = 0.125
T = 100
(–12.465 + j12.465)
a = 0.1
T = 50
(–12.408 + j9.965)
T = 30
a = 0.09
(–12.324 + j8.029)
T = 30 (0.0476 + j1.8204)
T = 50 (0.0278 + j1.4118)
T = 100 (0.0125 + j0.9992)
0
T = ∞
T = ∞
–25
–12.5
σ
T = 100, a = 0.125 (–0.0805)
T = 50, a = 0.1 (–0.2037)
T = 30, a = 0.09 (–0.3852)
T = 30
–25.129
T = 50
–25.076
T = 100
–25.035
–j48.4

Figure 11-52 Root contours of sun-seeker system in Example 11-5-4 with phase-lag controller.

The root-contour design conducted earlier in Example 11-5-1 using Eqs. (11-90) through (11-93) for phase-lead control and Figs. 11-35 and 11-36 is still valid for phase-lag control, except that in the present case, $a < 1$. Thus, in Fig. 11-36 only the portions of the root contours that correspond to $a < 1$ are applicable for phase-lag control. These root contours clearly show that, for effective phase-lag control, the value of T should be relatively large. In Fig. 11-52, we illustrate further that the complex poles of the closed-loop transfer function are rather insensitive to the value of T when the latter is relatively large.

Frequency-Domain Design The Bode plot of $G_p(j\omega)$ of Eq. (11-141) is shown in Fig. 11-53 for $K = 1$. The Bode plot shows that the phase margin of the uncompensated system is only 28°. Not knowing what phase margin will correspond to a maximum overshoot of less than 5 percent, we conduct the following series of designs using the Bode plot in Fig. 11-53. Starting with a phase margin of 45°, we observe that this phase margin can be realized if the gain-crossover frequency ω'_g is at 25 rad/s. This means that the phase-lag controller must reduce the magnitude curve of $G_p(j\omega)$ to 0 dB at $\omega = 25$ rad/s while it does not appreciably affect the phase curve near this frequency. Because the phase-lag controller still contributes a small negative phase when the corner frequency $1/aT$ is placed at 1/10 of the value of ω'_g, it is a safe measure to choose ω'_g at somewhat less than 25 rad/s, say, 20 rad/s.


80
60
40
20
0
−20
−40
−60
−80
$|G(j\omega)|$ (dB)
$G_c(s)G_p(s)$
Compensated system
Uncompensated system
$G_p(s) = \frac{2500}{s(s+25)}$
11.7 dB
10^{-2}
10^{-1}
10^{0}
10^{1}
25
10^{2}
10^{3}
ω (rad/sec)

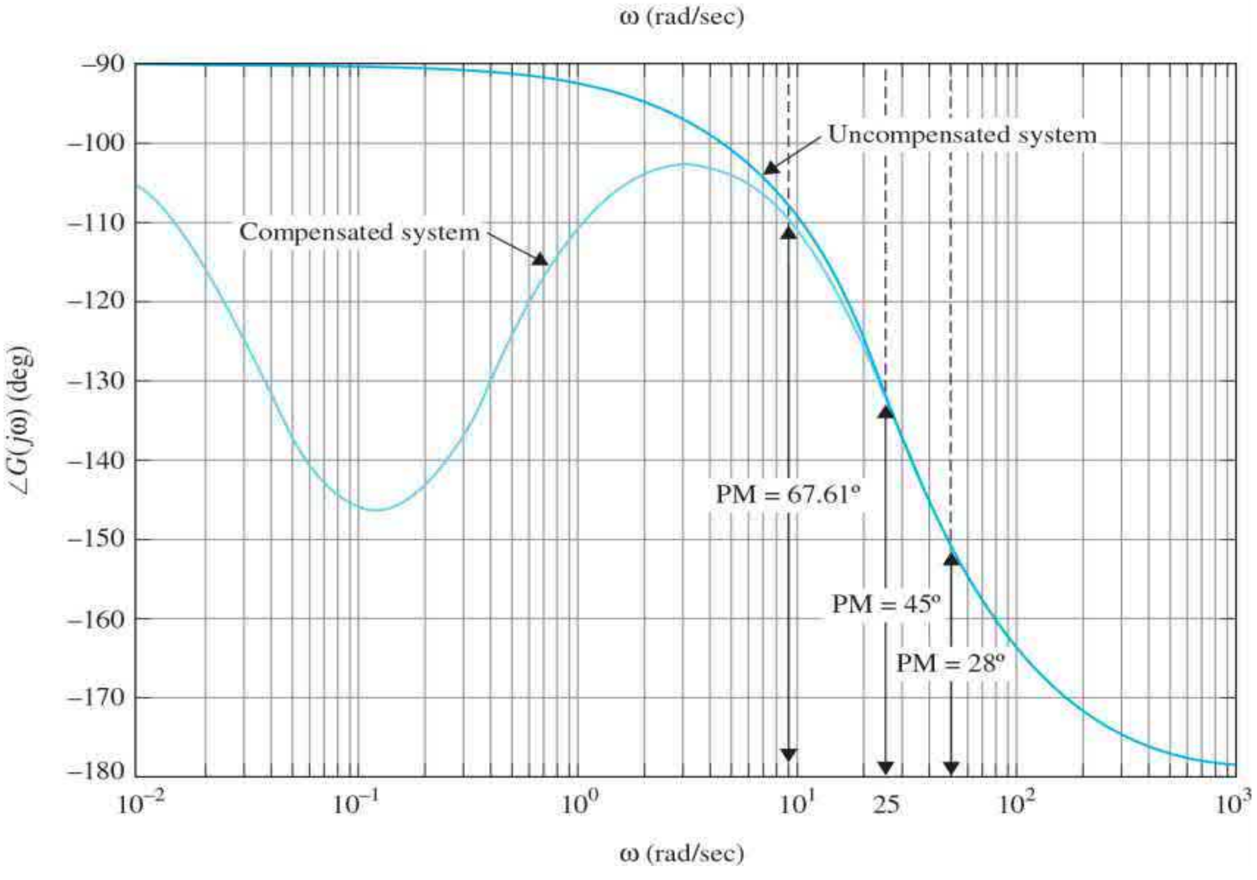

ω (rad/sec)
−90
−100
−110
−120
−130
−140
−150
−160
−170
−180
$\angle G(j\omega)$ (deg)
Uncompensated system
Compensated system
PM = 67.61°
PM = 45°
PM = 28°
10^{-2}
10^{-1}
10^{0}
10^{1}
25
10^{2}
10^{3}
ω (rad/sec)

Figure 11-53 Bode plot of uncompensated and compensated systems with phase-lag controller in Example 11-5-4. $G_c(s)=\dfrac{1+3s}{1+30s} \quad G_p(s)=\dfrac{2500}{s(s+25)}$.

From the Bode plot, the value of $|G_p(j\omega'_g)|_{\text{dB}}$ at ω'_g = 20 is 11.7 dB. Thus, using Eq. (11-138), we have

$$a = 10^{-|G_p(j\omega'_g)|/20} = 10^{-11.7/20} = 0.26 \qquad (11\text{-}144)$$

The value of 1/*aT* is chosen to be at 1/10 the value of ω'_g = 20 rad/s. Thus,

$$\frac{1}{aT} = \frac{\omega'_g}{10} = \frac{20}{10} = 2 \text{ rad/s} \qquad (11\text{-}145)$$

and

$$T = \frac{1}{2a} = \frac{1}{0.52} = 1.923 \qquad (11\text{-}146)$$

Checking out the unit-step response of the system with the designed phase-lag control, we found that the maximum overshoot is 24.5 percent. The next step is to try aiming at a higher phase margin. Table 11-19 gives the various design results by using various desired phase margins up to 80°.

Table 11-19 Performance Attributes of Sun-Seeker System in Example 11-5-4 with Phase-Lag Controller

Desired PM (deg)	a	T	Actual PM (deg)	M_r	BW (rad/s)	Maximum Overshoot (%)	t_r (s)	t_s (s)
45	0.26	1.923	46.78	1.27	33.37	24.5	0.0605	0.2222
60	0.178	3.75	54.0	1.19	25.07	17.5	0.0823	0.303
70	0.1	10	63.87	1.08	14.72	10.0	0.1369	0.7778
80	0.044	52.5	74.68	1.07	5.7	7.1	0.3635	1.933

Examining the results in Table 11-18, we see that none of the cases satisfies the maximum overshoot requirement of ≤5%. The *a* = 0.044 and *T* =

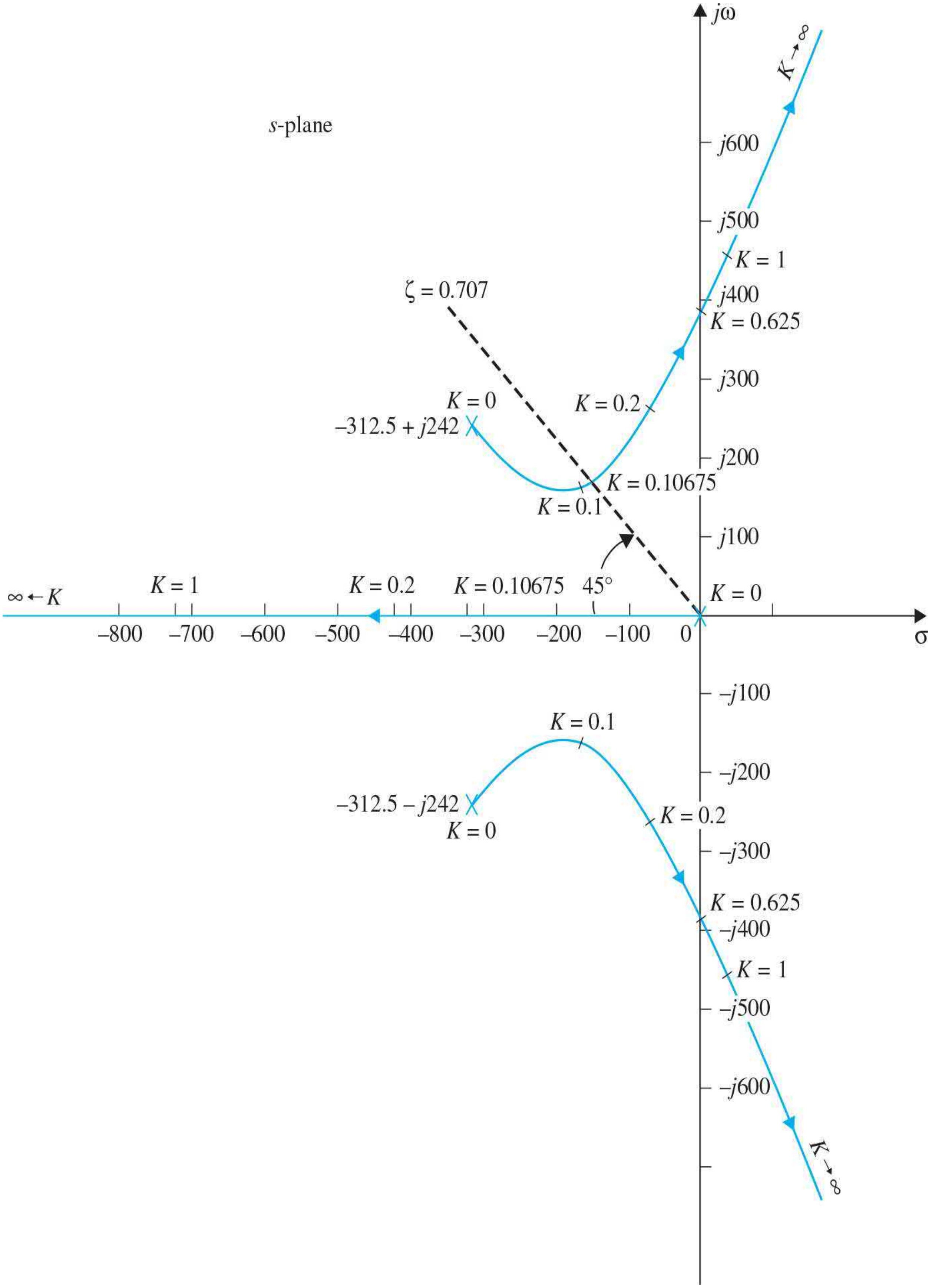
52.5 case yields the best maximum overshoot, but the value of *T* is too large to be practical. Thus, we single out the case with *a* = 0.1 and *T* = 10 and refine the design by increasing the value of *T*. As shown in Table 11-19, when *a* = 0.1 and *T* = 30, the maximum overshoot is reduced to 4.5 percent. The Bode plot of the compensated system is shown in Fig. 11-49. The phase margin is 67.61°.

The unit-step response of the phase-lag-compensated system shown in Fig. 11-52 points out a major disadvantage of the phase-lag control. Because the phase-lag controller is essentially a low-pass filter, the rise time and settling time of the compensated system are usually increased. However, we shall show by the following example that phase-lag control can be more versatile and has a wider range of effectiveness in improving stability than the single-stage phase-lead controller, especially if the system has low or negative damping. ▲

EXAMPLE 11-5-5 Consider the sun-seeker system designed in Example 11-5-3, with the forward-path transfer function given in Eq. (11-121). Let us restore the gain *K*, so that a root-locus plot can be made for the system. Then, Eq. (11-121) is written

$$G_p(s) = \frac{156{,}250{,}000\, K}{s(s^2 + 625s + 156{,}250)} \qquad (11\text{-}147)$$

The root loci of the closed-loop system are shown in Fig. 11-54. When *K* = 1, the system is unstable, and the characteristic equation roots are at -713.14,44.07+*j*466.01.



jω
s-plane
K → ∞
j600
j500
K = 1
ζ = 0.707
j400
K = 0.625
j300
K = 0.2
K = 0
−312.5 + j242
j200
K = 0.10675
K = 0.1
j100
45°
∞ ← K
K = 1
K = 0.2
K = 0.10675
K = 0
−800
−700
−600
−500
−400
−300
−200
−100
0
σ
−j100
K = 0.1
−j200
−312.5 − j242
K = 0.2
K = 0
−j300
K = 0.625
−j400
K = 1
−j500
−j600
K → ∞

Figure 11-54 Root loci of uncompensated system in Example 11-5-5.

$$G_p(s) = \frac{156{,}250{,}000\ K}{s(s^2 + 625s + 156{,}250)}.$$

Example 11-5-3 shows that the performance specification on stability cannot be achieved with a single-stage phase-lead controller. Let the performance criteria be as follows:

Maximum overshoot ≤5%

Rise time $t_r \leq 0.02$ s

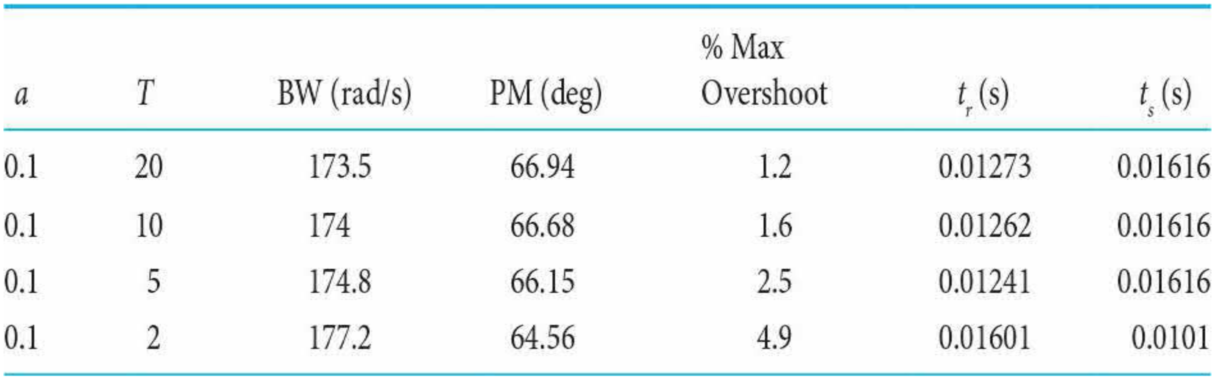
settling time $t_r \leq 0.02$ s

Let us assume that the desired relative damping ratio is 0.707. Figure 11-54 shows that, when $K = 0.10675$, the dominant characteristic equation roots of the uncompensated system are at $-172.77 \pm j172.73$, which correspond to a damping ratio of 0.707. Thus, the value of a is determined from Eq. (11-134),

$$a = \frac{K \text{ to realize the desired damping}}{K \text{ to realize the steady} - \text{state performance}} = \frac{0.10675}{1} = 0.10675 \qquad (11\text{-}148)$$

Let $a = 0.1$. Because the loci of the dominant roots are far away from the origin in the s-plane, the value of T has a wide range of flexibility. Table 11-20 shows the performance results when $a = 0.1$ and for various values of T.

Table 11-20 Performance Attributes of Sun-Seeker System in Example 11-5-6 with Phase-Lag Controller



a	T	BW (rad/s)	PM (deg)	% Max Overshoot	t_r (s)	t_s (s)
0.1	20	173.5	66.94	1.2	0.01273	0.01616
0.1	10	174	66.68	1.6	0.01262	0.01616
0.1	5	174.8	66.15	2.5	0.01241	0.01616
0.1	2	177.2	64.56	4.9	0.01601	0.0101

Therefore, the conclusion is that only one stage of the phase-lag controller is needed to satisfy the stability requirement, whereas two stages of the phase-lead controller are needed, as shown in Example 11-5-3.

Sensitivity Function The sensitivity function $|S_G^M(j\omega)|$ of the phase-lag compensated system with $a = 0.1$ and $T = 20$ is shown in Fig. 11-55. Notice that the sensitivity function is less than unity for frequencies up to only 102 rad/s. This is due to the low bandwidth of the system as a result of phase-lag control. ▲

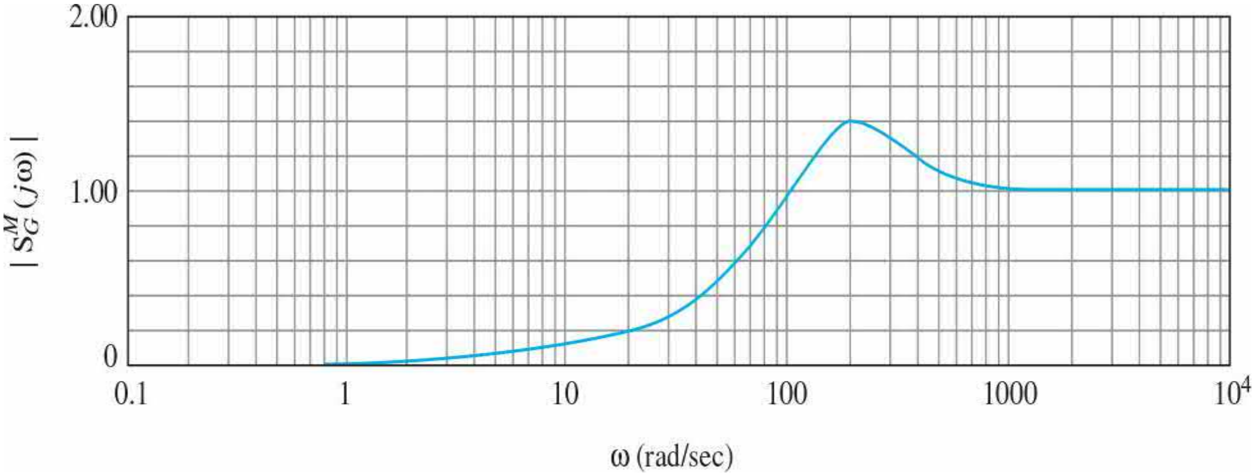


Figure 11-55 Sensitivity function of phase-lag-compensated system in Example 11-5-6.

11-5-9 Effects and Limitations of Phase-Lag Control

From the results of the preceding illustrative examples, the effects and limitations of phase-lag control on the performance of linear control systems can be summarized as follows:

1. For a given forward-path gain *K*, the magnitude of the forward-path transfer function is attenuated near the gain-crossover frequency, thus improving the relative stability of the system.
2. The gain-crossover frequency is decreased, and thus the bandwidth of the system is reduced.
3. The rise time and settling time of the system are usually longer because the bandwidth is usually decreased.
4. The system is more sensitive to parameter variations because the sensitivity function is greater than unity for all frequencies

approximately greater than the bandwidth of the system.

11-5-10 Design with Lead-Lag Controller

We have learned from preceding sections that phase-lead control generally improves rise time and damping but increases the natural frequency of the closed-loop system. However, phase-lag control when applied properly improves damping but usually results in a longer rise time and settling time. Therefore, each of these control schemes has its advantages, disadvantages, and limitations, and there are many systems that cannot be satisfactorily compensated by either scheme acting alone. It is natural, therefore, whenever necessary, to consider using a combination of the lead and lag controllers, so that the advantages of both schemes are utilized.

The transfer function of a simple lag-lead (or lead-lag) controller can be written

$$G_c(s) = G_{c1}(s)G_{c2}(s) = \underbrace{\left(\frac{1+a_1T_1s}{1+T_1s}\right)}_{|\leftarrow \text{lead} \rightarrow|}\underbrace{\left(\frac{1+a_2T_2s}{1+T_2s}\right)}_{|\leftarrow \text{lag} \rightarrow|} \quad (a_1 > 1,\ a_2 < 1) \qquad (11\text{-}149)$$

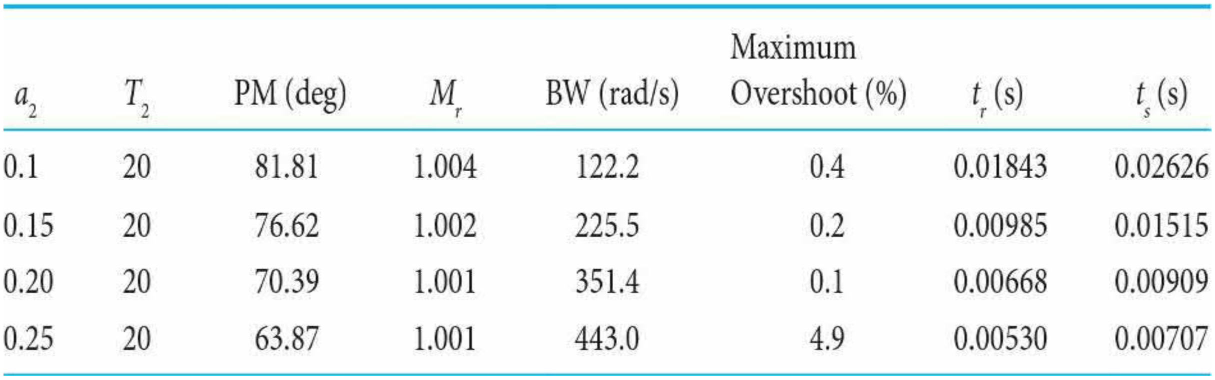
The gain factors of the lead and lag controllers are not included because, as shown previously, these gain and attenuation are compensated eventually by the adjustment of the forward gain K.

Because the lead-lag controller transfer function in Eq. (11-149) now has four unknown parameters, its design is not as straightforward as the single-stage phase-lead or phase-lag controller. *In general, the phase-lead portion of the controller is used mainly to achieve a shorter rise time and higher bandwidth, and the phase-lag portion is brought in to provide major damping of the system*. Either the phase-lead or the phase-lag control can be designed first. We shall use the next example to illustrate the design steps.

EXAMPLE 11-5-6 Consider the sun-seeker system of Example 11-5-3. The uncompensated system with $K = 1$ was shown to be unstable. A two-stage phase-lead controller was designed in Example 11-5-4, and a single-stage phase-lag controller was designed in Example 11-5-5.

Based on the design in Example 11-5-3, we can first select a phase-lead control with *a* = 70 and T_1 = 0.00004. The remaining phase-lag control can be designed using either the root-locus method or the Bode plot method. Table 11-21 gives the results for and various values of *a*. The results in Table 11-21 show that the optimal value of a_2, from the standpoint of minimizing the maximum overshoot, for a_1 = 70 and T_2 = 0.00004, is approximately 0.2.

Table 11-21 Performance Attributes of Sun-Seeker System in Example 11-7-1 with Lead-Lag Controller: a_1 = 70, T_1 = 0.00004



a_2	T_2	PM (deg)	M_r	BW (rad/s)	Maximum Overshoot (%)	t_r (s)	t_s (s)
0.1	20	81.81	1.004	122.2	0.4	0.01843	0.02626
0.15	20	76.62	1.002	225.5	0.2	0.00985	0.01515
0.20	20	70.39	1.001	351.4	0.1	0.00668	0.00909
0.25	20	63.87	1.001	443.0	4.9	0.00530	0.00707

Compared with the phase-lag controller designed in Example 11-5-4, the BW is increased to 351.4 rad/s from 66.94 rad/s, and the rise time is reduced to 0.00668 s from 0.01273 s. The system with the lead-lag controller is more robust because the magnitude of the sensitivity function does not increase to unity until near the BW of 351.4 rad/s. As a comparison, the unit-step responses of the system with the two-stage phase-lead control, the single-stage phase-lag control, and the lead-lag control are shown in Fig. 11-56.

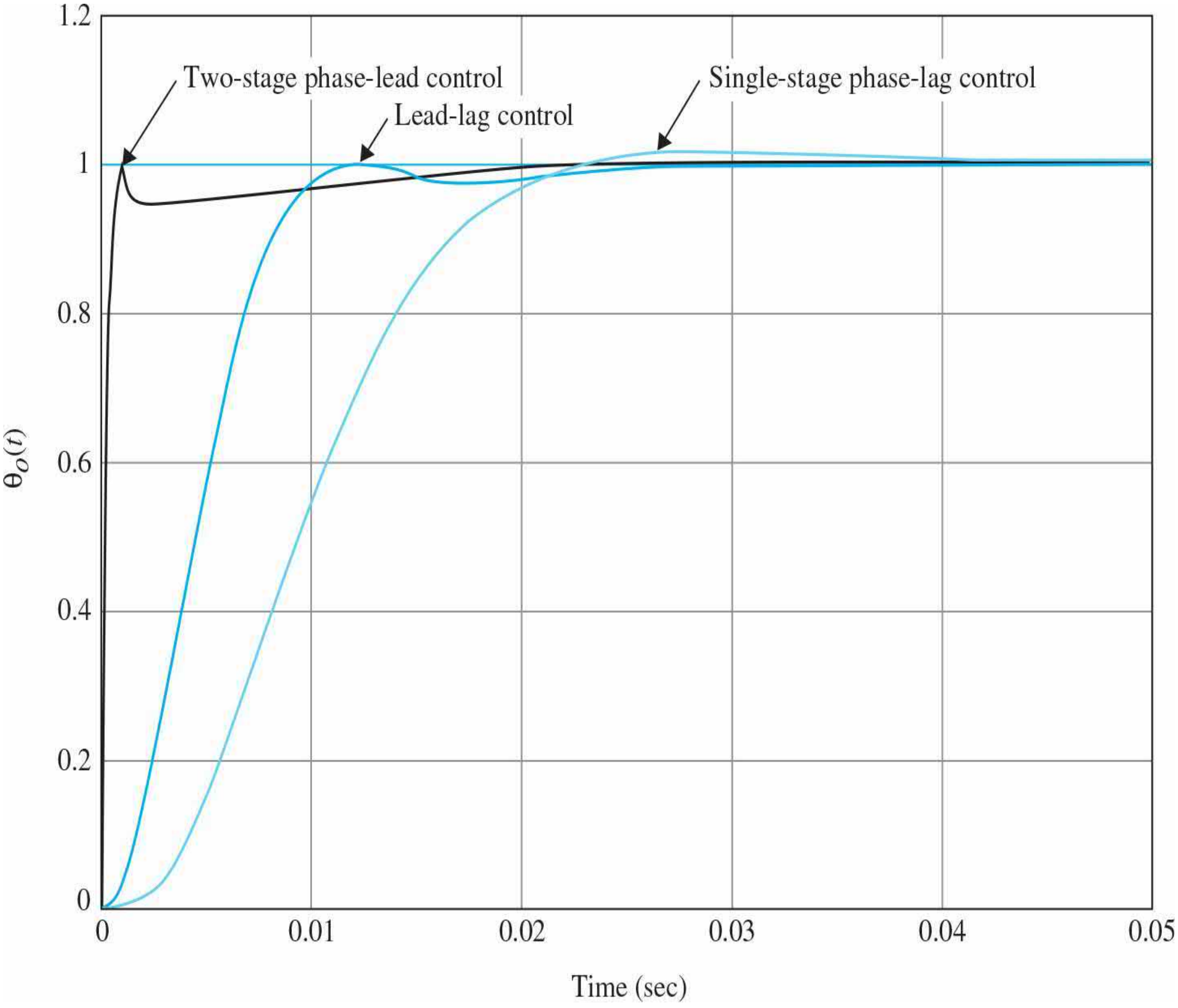


Figure 11-56 Sun-seeker system in Example 11-5-6 with single-stage phase-lag controller, lead-lag controller, and two-stage phase-lead controller.

It should be noted that the bandwidth and rise time of the sun-seeker system can be further increased and reduced, respectively, by using a larger value of a_1 for the phase-lead portion of the controller. However, the resulting step response will have a large undershoot, although the maximum overshoot can be kept small. ▲

11-6 POLE-ZERO-CANCELLATION DESIGN: NOTCH FILTER

The transfer functions of many controlled processes contain one or more pairs of complex-conjugate poles that are very close to the imaginary axis of the *s*-plane. These complex poles usually cause the closed-loop system to be lightly damped or unstable. One immediate solution is to use a controller that has a transfer function with zeros selected to cancel the undesirable poles of the system, and controller poles at more desirable locations in the *s*-plane to achieve the desired dynamic performance. For example, if the transfer function of a process is

$$G_p(s) = \frac{K}{s(s^2 + s + 10)} \tag{11-150}$$

in which the complex-conjugate poles may cause stability problems in the closed-loop system when the value of *K* is large, the suggested series controller may be of the form

$$G_c(s) = \frac{s^2 + s + 10}{s^2 + as + b} \tag{11-151}$$

The constants *a* and *b* may be selected according to the performance specifications of the closed-loop system.

While this type of controller may be an ideal one, it requires an exact model of the process, $G_p(s)$, which is rarely possible in practice. No matter how accurate, the transfer function model of the process, $G_p(s)$, usually deviates from the actual system because of unmodeled plant dynamics or nonlinear behavior. Thus, the *true* poles and zeros of the transfer function of the process may not be accurately modeled. In fact, the true order of the system may even be higher than that represented by the transfer function used for modeling purposes. Another difficulty is that the dynamic properties of the process may vary, even very slowly, due to aging of the system components or changes in the operating environment, so the poles and zeros of the transfer function may move during the operation of the system. The parameters of the controller are constrained by the actual physical components available and cannot be assigned arbitrarily. For these and other reasons, even if we could precisely design the poles and zeros of the transfer function of the controller, exact pole-zero cancellation is almost never possible in practice. We will now show that, in most cases, exact cancellation is *not* really necessary to effectively negate the influence of the undesirable poles using pole-zero-cancellation compensation schemes.

> In practice, exact pole–zero cancellation is almost never possible because of modeling inaccuracies.

Let us assume that a controlled process is represented by

$$G_p(s) = \frac{K}{s(s+p_1)(s+p.macr_1)} \qquad (11\text{-}152)$$

where p_1 and $p.macr_1$ are the two complex-conjugate poles that are to be canceled. Let the transfer function of the series controller be

$$G_c(s) = \frac{(s+p_1+\varepsilon_1)(s+p.macr_1+\varepsilon.macr_1)}{s^2+as+b} \qquad (11\text{-}153)$$

where ϵ_1 is a complex number whose magnitude is very small and $\epsilon.macr_1$ is its complex conjugate. The open-loop transfer function of the compensated system is

$$G(s) = G_c(s)G_p(s) = \frac{K(s+p_1+\varepsilon_1)(s+p.macr_1+\varepsilon.macr_1)}{s(s+p_1)(s+p.macr_1)(s^2+as+b)} \qquad (11\text{-}154)$$

Because of inexact cancellation, we cannot discard the terms $(s+p_1)(s+p.macr_1)$ in the denominator of Eq. (11-154). The closed-loop transfer function is

$$\frac{Y(s)}{R(s)} = \frac{K(s+p_1+\varepsilon_1)(s+p.macr_1+\varepsilon.macr_1)}{s(s+p_1)(s+p.macr_1)(s^2+as+b)+K(s+p_1+\varepsilon_1)(s+p.macr_1+\varepsilon_1)} \qquad (11\text{-}155)$$

The root-locus diagram in Fig. 11-57 explains the effect of inexact pole-zero cancellation. Notice that the two closed-loop poles as a result of inexact cancellation lie between the pairs of poles and zeros at $s=-p_1,-p.macr_1$ and $-p_1-\varepsilon_1,-p.macr_1-\varepsilon.macr_1$ respectively. Thus, these closed-loop poles are very close to the open-loop poles and zeros that are meant to be canceled. Eq. (11-

155) can be approximated as

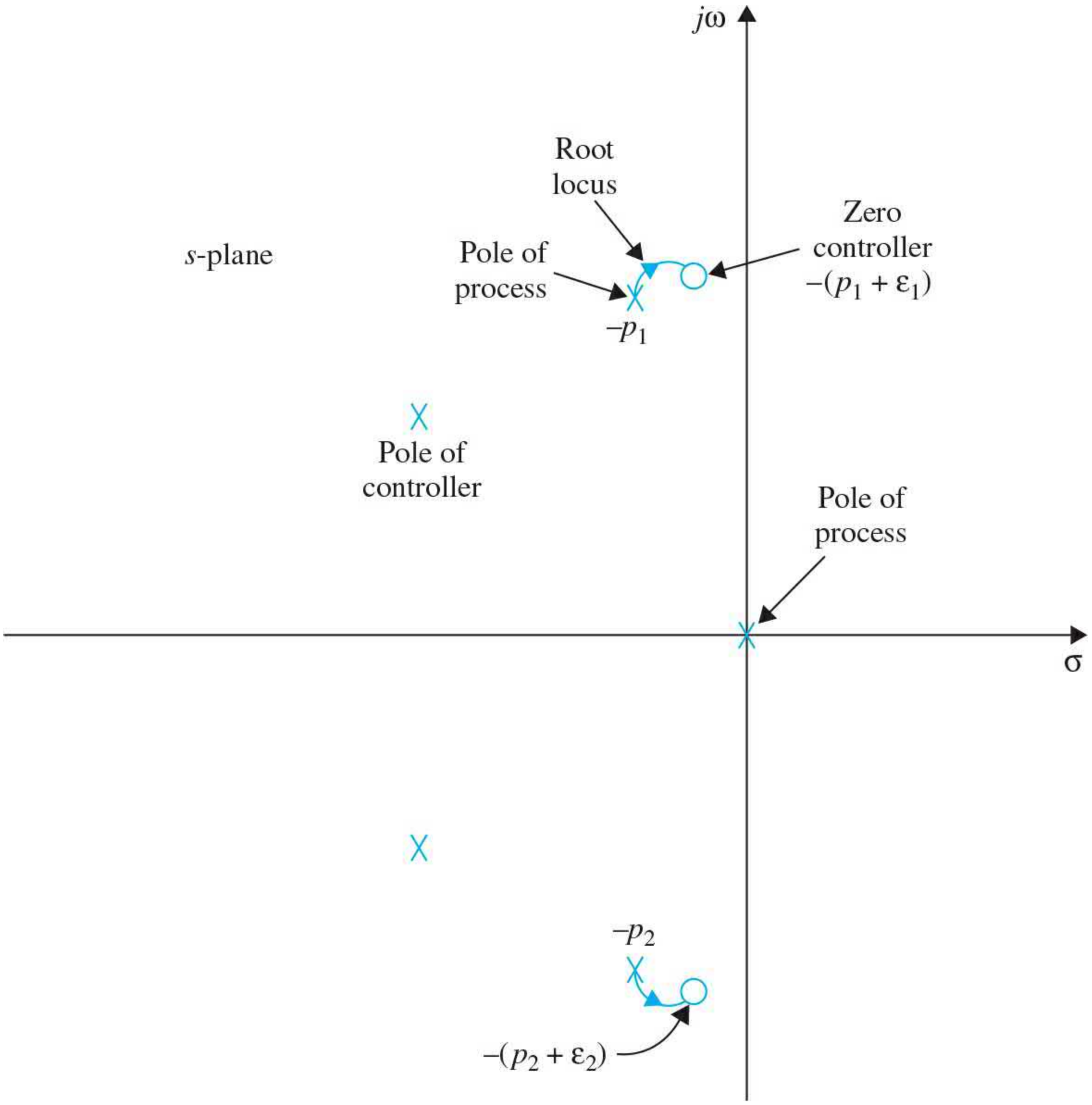


Figure 11-57 Pole-zero configuration and root loci of inexact cancellation.

$$\frac{Y(s)}{R(s)} \cong \frac{K(s+p_1+\varepsilon_1)(s+p.macr_1+\varepsilon.macr_2)}{(s+p_1+\delta_1)(s+\bar{p}_1+\delta_1)(s^3+as+b+K)} \qquad (11\text{-}156)$$

where δ_1 and δ_1 are a pair of very small complex-conjugate numbers that depend on ϵ_1, $\epsilon.macr_1$, and all the other parameters. The partial-fraction expansion of Eq. (11-156) is

$$\frac{Y(s)}{R(s)} \cong \frac{K_1}{s+p_1+\delta_1} + \frac{K_2}{s+\bar{p}_1+\bar{\delta}_1} + \text{terms due to the remaining poles} \qquad (11\text{-}157)$$

We can show that K_1 is proportional to $\varepsilon_1 - \delta_1$, which is a very small number. Similarly, K_2 is also very small. *This exercise simply shows that, although the poles at* $-p_1$ *and* $-p_2$ *cannot be canceled precisely, the resulting transient-response terms due to inexact cancellation will have insignificant amplitudes, so unless the controller zeros earmarked for cancellation are too far off target, the effect can be neglected for all practical purposes*. Another way of viewing this problem is that the zeros of $G(s)$ are retained as the zeros of closed-loop transfer function $Y(s)/R(s)$, so from Eq. (11-156), we see that the two pairs of poles and zeros are close enough to be canceled from the transient-response standpoint.

Keep in mind that we should never attempt to cancel poles that are in the right-half s-plane because any inexact cancellation will result in an unstable system. Inexact cancellation of poles could cause difficulties if the unwanted poles of the process transfer function are very close to or right on the imaginary axis of the s-plane. In this case, inexact cancellation may also result in an unstable system. Figure 11-58*a* illustrates a situation in which the relative positions of the poles and zeros intended for cancellation result in a stable system, whereas in Fig. 11-58*b*, the inexact cancellation is unacceptable. The relative distance between the poles and zeros intended for cancellation is small, which results in residual terms in the time response solution. Although these terms have very small amplitudes, they tend to grow without bound as time increases. Hence the system response becomes unstable.

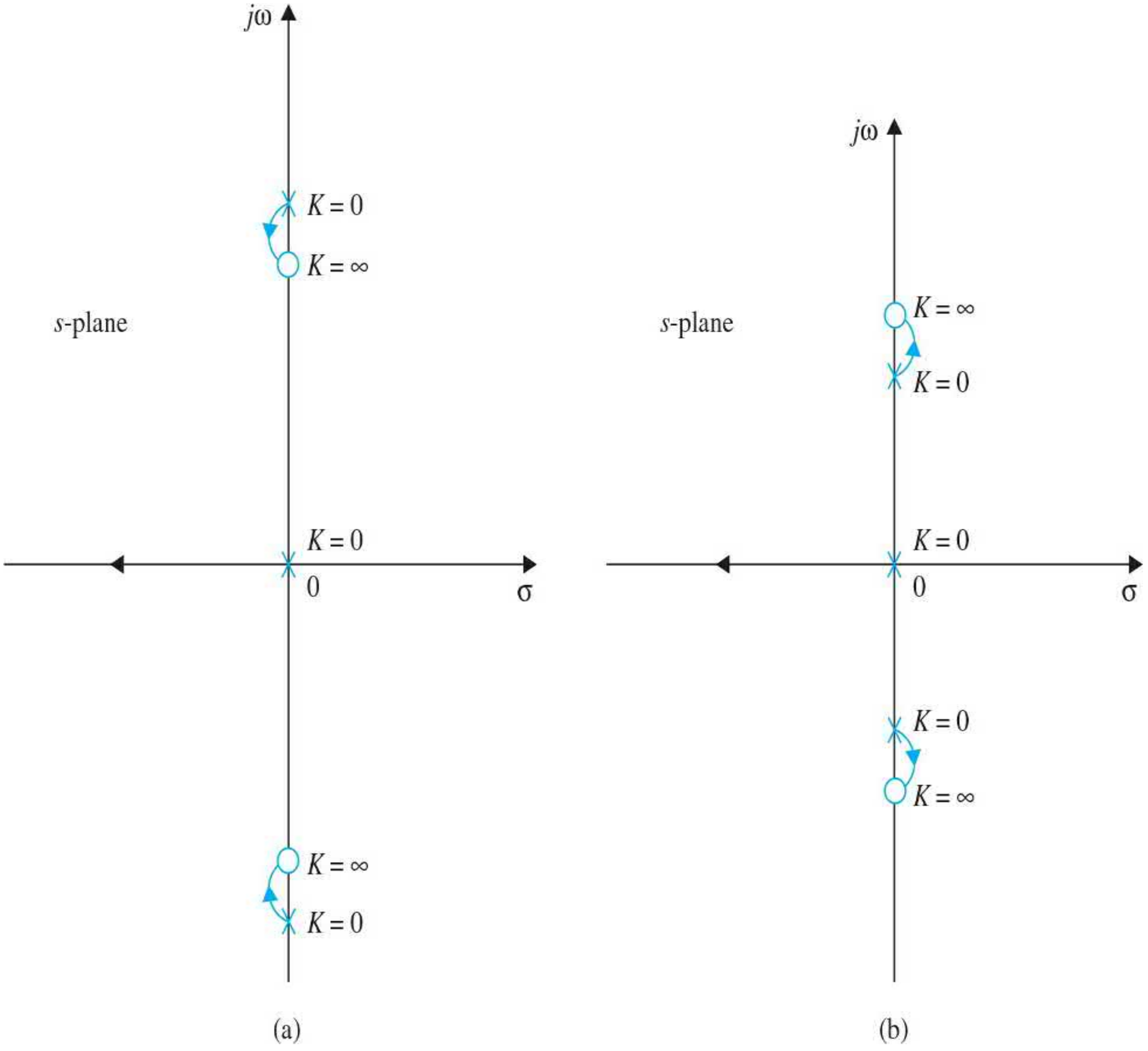


Figure 11-58 Root loci showing the effects of inexact pole-zero cancellations.

11-6-1 Second-Order Active Filter

Transfer functions with complex poles and/or zeros can be realized by electric circuits with op-amps. Consider the transfer function

$$G_c(s) = \frac{E_2(s)}{E_1(s)} = K\frac{s^2 + b_1 s + b_2}{s^2 + a_1 s + a_2} \tag{11-158}$$

where a_1, a_2, b_1, and b_2 are real constants. The active-filter realization of Eq.

(11-158) can be accomplished by using the direct decomposition scheme of state variables discussed in Sec. 11-10. A typical op-amp circuit is shown in Fig. 11-59. The parameters of the transfer function in Eq. (11-158) are related to the circuit parameters as follows:

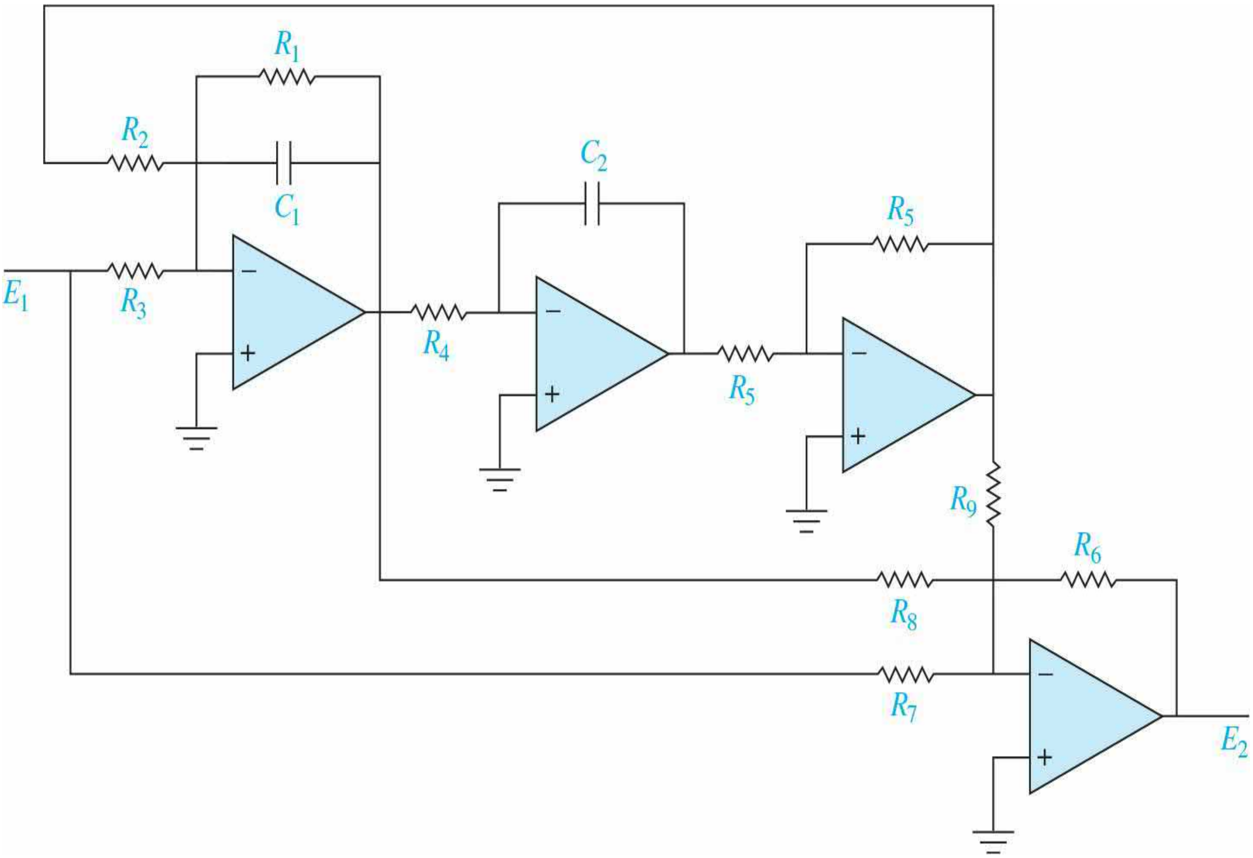


Figure 11-59 Op-amp circuit realization of the second-order transfer function $\frac{E_2(s)}{E_1(s)} = K\frac{s^2+b_1s+b_2}{s^2+a_1s+a_2}$.

$$K = -\frac{R_6}{R_7} \tag{11-159}$$

$$a_1 = \frac{1}{R_1C_1} \tag{11-160}$$

$$a_2 = \frac{1}{R_2R_4C_1C_2} \tag{11-161}$$

$$b_1 = \left(1 - \frac{R_1 R_7}{R_3 R_8}\right) a_1 \quad (b_1 < a_1) \tag{11-162}$$

$$b_2 = \left(1 - \frac{R_2 R_7}{R_3 R_9}\right) a_2 \quad (b_2 < a_2) \tag{11-163}$$

Because $b_1 < a_1$, the zeros of $G_c(s)$ in Eq. (11-158) are less damped and are closer to the origin in the s-plane than the poles. By setting various combinations of R_7 and R_8, and R_9 to infinity, a variety of second-order transfer functions can be realized. Note that all the parameters can be adjusted independently of one another. For example, R_1 can be adjusted to set a_1; R_4 can be adjusted to set a_2; and b_1 and b_2 are set by adjusting R_8 and R_9, respectively. The gain factor K is controlled independently by R_6.

11-6-2 Frequency-Domain Interpretation and Design

While it is simple to grasp the idea of pole-zero-cancellation design in the s-domain, the frequency-domain provides added perspective to the design principles. Figure 11-60 illustrates the Bode plot of the transfer function of a typical second-order controller with complex zeros. The magnitude plot of the controller typically has a “notch” at the resonant frequency ω_n. The phase plot is negative below and positive above the resonant frequency, while passing through zero degrees at the resonant frequency. The attenuation of the magnitude curve and the positive-phase characteristics can be used effectively to improve the stability of a linear system. Because of the “notch” characteristic in the magnitude curve, the controller is also referred to in the industry as a **notch filter** or **notch controller**.

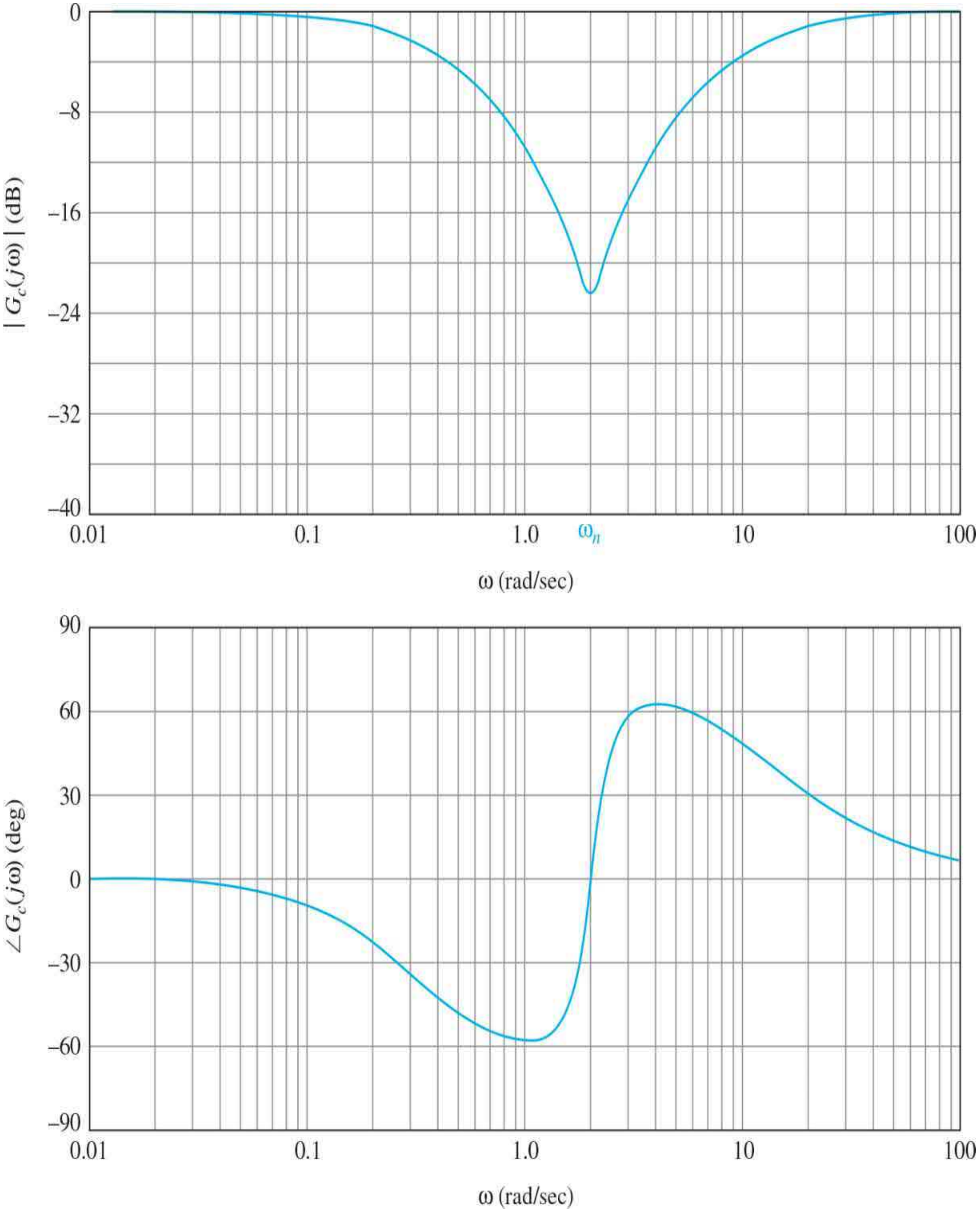


Figure 11-60 Bode plot of a notch controller with the transfer function.

$$G(s) = \frac{(s^2 + 0.8s + 4)}{(s + 0.384)(s + 10.42)}.$$

From the frequency-domain standpoint, the notch controller has advantages over the phase-lead and phase-lag controllers in certain design conditions because the magnitude and phase characteristics do not affect the high- and low-frequency properties of the system. Without using the pole-zero-cancellation principle, the design of the notch controller for compensation in the frequency domain involves the determination of the amount of attenuation required and the resonant frequency of the controller.

Let us express the transfer function of the notch controller in Eq. (11-158) as

$$G_c(s) = \frac{s^2 + 2\zeta_z \omega_n s + \omega_n^2}{s^2 + 2\zeta_p \omega_n s + \omega_n^2} \tag{11-164}$$

where we have made the simplification by assuming that $a_2 = b_2$.

The attenuation provided by the magnitude of $G_c(j\omega)$ at the resonant frequency ω_n is

$$|G_c(j\omega_n)| = \frac{\zeta_z}{\zeta_p} \tag{11-165}$$

Thus, knowing the maximum attenuation required at ω_n, the ratio of ω_z/ω_p is known.

The following example illustrates the design of the notch controller based on pole-zero cancellation and required attenuation at the resonant frequency.

EXAMPLE 11-6-1 Complex-conjugate poles in system transfer functions are often due to compliances in the coupling between mechanical elements. For instance, if the shaft between the motor and load is nonrigid, the shaft is modeled as a torsional spring, which could lead to complex-conjugate poles in the process transfer function. Figure 11-61 shows a speed-control system in which the coupling between the motor and the load is modeled as a torsional spring. The system equations are

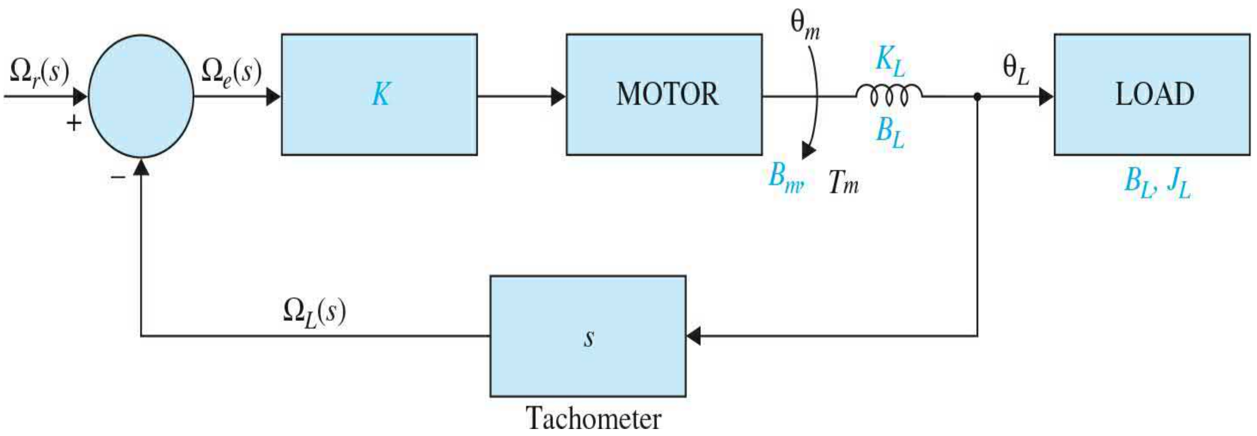


Figure 11-61 Block diagram of speed-control system in Example 11-6-1.

$$T_m(t) = J_m \frac{d\omega_m(t)}{dt} + B_m\omega_m(t) + J_L \frac{d\omega_L(t)}{dt} \tag{11-166}$$

$$K_L[\theta_m(t) - \theta_L(t)] + B_L[\omega_m(t) - \omega_L(t)] = J_L \frac{d\omega_L(t)}{dt} \tag{11-167}$$

$$T_m(t) = K\omega_e(t) \tag{11-168}$$

$$\omega_e(t) = \omega_r(t) - \omega_L(t) \tag{11-169}$$

where $T_m(t)$ = motor torque

$\omega_m(t)$ = motor angular velocity

$\omega_L(t)$ = load angular velocity

$\omega_L(t)$ = load angular displacement

$\omega_m(t)$ = motor angular displacement

J_m = motor inertia 0.0001 oz-in.-s^2

J_L = load inertia 0.0005 oz-in.-s^2

B_m = viscous-friction coefficient of motor 0.01 oz-in.-s

B_L = viscous-friction coefficient of shaft 0.001 oz-in.-s

K_L = spring constant of shaft 100 ozin./rad

K = amplifier gain = 1

The loop transfer function of the system is

$$G_p(s)=\frac{\Omega_L(s)}{\Omega_e(s)}=\frac{B_L s+K_L}{J_m J_L s^3+(B_m J_L+B_L J_m+B_L J_L)s^2+(K_L J_L+B_m B_L+K_L J_m)s+B_m K_L} \tag{11-170}$$

By substituting the system parameters in the last equation, $G_p(s)$ becomes

$$\begin{aligned}G_p(s)&=\frac{20{,}000(s+100{,}000)}{s^3+112s^2+1{,}200{,}200s+20{,}000{,}000}\\&=\frac{20{,}000(s+100{,}000)}{(s+16.69)(s+47.66+j1094)(s+47.66-j1094)}\end{aligned} \tag{11-171}$$

Thus, the shaft compliance between the motor and the load creates two complex-conjugate poles in $G_p(s)$ that are lightly damped. The resonant frequency is approximately 1095 rad/s, and the closed-loop system is unstable. The complex poles of $G_p(s)$ would cause the speed response to oscillate even if the system were stable.

Pole-Zero-Cancellation Design with Notch Controller The following are the performance specifications of the system:

The steady-state speed of the load due to a unit-step input should have an error of not more than 1%.

Maximum overshoot of output speed ≤ 5%.

Rise time t_r <0.5s.

Settling time t_s <0.5s.

To compensate the system, we need to get rid, or, perhaps more realistically, minimize the effect, of the complex poles of $G_p(s)$ at s = -47.66+j1094 and -47.66+j1094. Let us select a *notch controller* with the transfer function given in Eq. (11-164) to improve the performance of the system. The complex-conjugate zeros of the controller should be so placed that they will cancel the undesirable poles of the process. Therefore, the transfer function of the notch controller should be

$$G_c(s)=\frac{s^2+95.3s+1{,}198{,}606.6}{s^2+2\zeta_p\omega_n s+\omega_n^2} \tag{11-172}$$

The forward-path transfer function of the compensated system is

$$G(s)=G_c(s)G_p(s)=\frac{20{,}000(s+100{,}000)}{(s+16.69)(s^2+2\zeta_p\omega_n s+\omega_n^2)} \tag{11-173}$$

Because the system is type 0, the step-error constant is

$$K_P=\lim_{s\to 0}G(s)=\frac{2\times 10^9}{16.69\times\omega_n^2}=\frac{1.198\times 10^8}{\omega_n^2} \tag{11-174}$$

For a unit-step input, the steady-state error of the system is written

$$e_{ss}=\lim_{t\to\infty}\omega_e(t)=\lim_{s\to 0}s\Omega_e(s)=\frac{1}{1+K_P} \tag{11-175}$$

Thus, for the steady-state error to be less than or equal to 1 percent, K_p≥99. The corresponding requirement on ω_n is found from Eq. (11-174),

$$\omega_n\le 1210 \tag{11-176}$$

We can show that, from the stability standpoint, it is better to select a large value for ω_n. Thus, let ω_n = 1200 rad/s, which is at the high end of the allowable range from the steady-state error standpoint. However, the design specifications given above can only be achieved by using a very large value for ζ_p. For example, when ζ_p =15,000, the time response has the following performance attributes:

Maximum overshoot = 3.7%

Rise time t_r = 0.1897 s

settling time t_s = 0.256 s

Although the performance requirements are satisfied, the solution is unrealistic because the extremely large value for ζ_p cannot be realized by physically available controller components.

Let us choose ζ_p = 10 and ω_n = 1000 rad/s. The forward-path transfer function of the system with the notch controller is

$$G(s)=G_c(s)G_p(s)=\frac{20{,}000(s+100{,}000)}{(s+16.69)(s+50)(s+19{,}950)} \qquad (11\text{-}177)$$

We can show that the system is stable, but the maximum overshoot is 71.6 percent. Now we can regard the transfer function in Eq. (11-177) as a new design problem. There are a number of possible solutions to the problem of meeting the design specifications given. We can introduce a phase-lag controller or a PI controller, among other possibilities.

Second-Stage Phase-Lag Controller Design Let us design a phase-lag controller as the second-stage controller for the system. The roots of the characteristic equation of the system with the notch controller are at s=-19954, -31.328 + *j*316.36, and -31.328 + *j*316.36. The transfer function of the phase-lag controller is

$$G_{c1}(s)=\frac{1+aTs}{1+Ts} \quad (a<1) \qquad (11\text{-}178)$$

where for design purposes we have omitted the gain factor 1/*a* in Eq. (11-178).

Let us select *T* = 10 for the phase-lag controller. Table 11-22 gives time-domain performance attributes for various values of *a*. The best value of *a* from the overall performance standpoint appears to be 0.005. Thus, the transfer function of the phase-lag controller is

Table 11-22 Time-Domain Performance Attributes of System in Example 11-6-1 with Notch-Phase-Lag Controller

a	T	aT	Maximum Overshoot (%)	t_r (s)	t_s (s)
0.001	10	0.01	14.8	0.1244	0.3836
0.002	10	0.02	10.0	0.1290	0.3655
0.004	10	0.04	3.2	0.1348	0.1785
0.005	10	0.05	1.0	0.1375	0.1818
0.0055	10	0.055	0.3	0.1386	0.1889
0.006	10	0.06	0	0.1400	0.1948

$$G_{c1}(s) = \frac{1 + aTs}{1 + Ts} = \frac{1 + 0.05s}{1 + 10s} \tag{11-179}$$

The forward-path transfer function of the compensated system with the notch-phase-lag controller is

$$G(s) = G_c(s)G_{c1}(s)G_p(s) = \frac{20{,}000(s + 100{,}000)(1 + 0.05s)}{(s + 16.69)(s + 50)(s + 19{,}950)(1 + 10s)} \tag{11-180}$$

The unit-step response of the system is shown in Fig. 11-62. Because the step-error constant is 120.13, the steady-state speed error due to a step input is 1/120.13, or 0.83 percent.

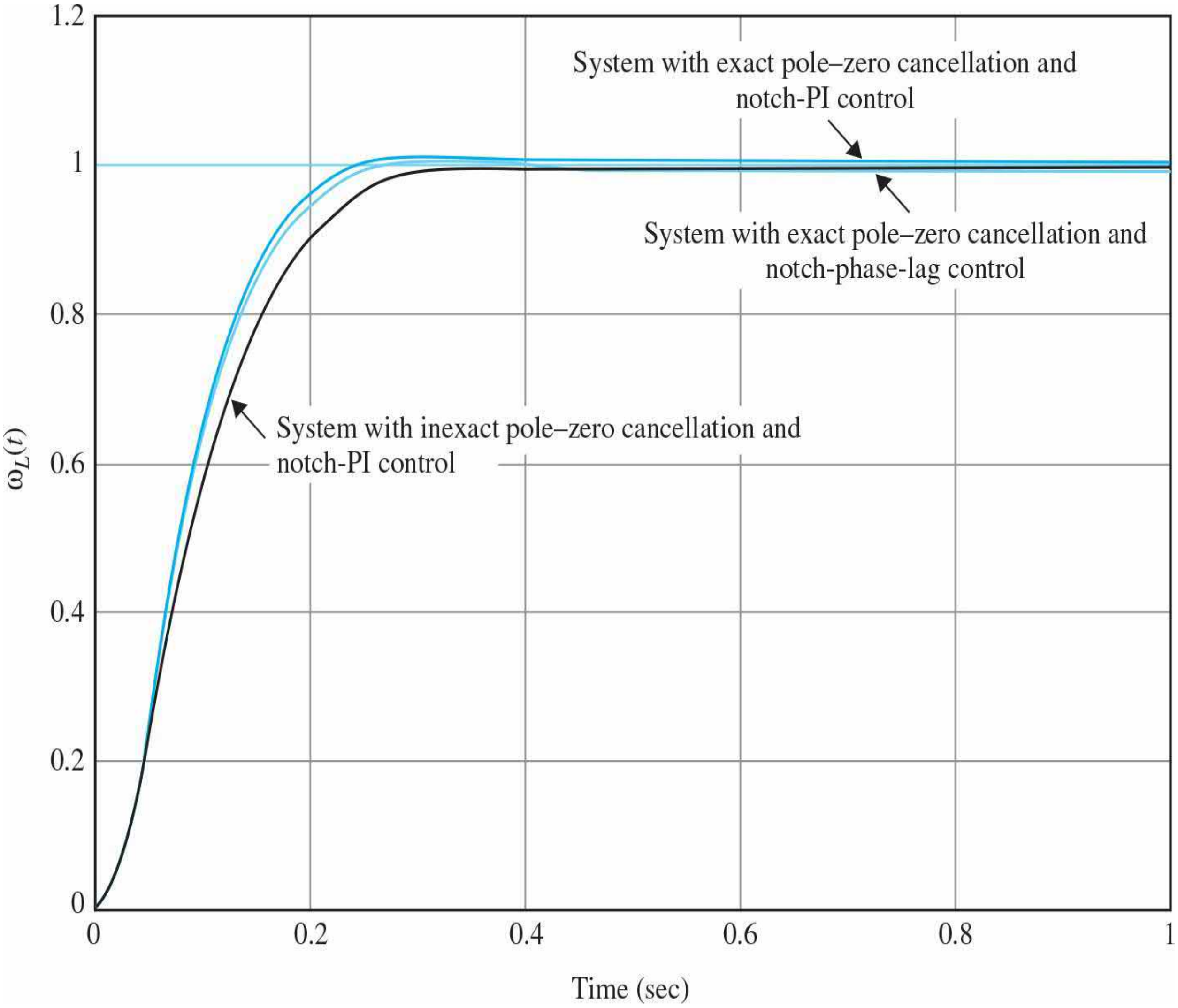


Figure 11-62 Unit-step responses of speed-control system in Example

11-6-1.

Second-Stage PI Controller Design A PI controller can be applied to the system to improve the steady-state error and the stability simultaneously. The transfer function of the PI control is written

$$G_{c2}(s) = K_P + \frac{K_I}{s} = K_P\left(\frac{s + K_I/K_P}{s}\right) \tag{11-181}$$

We can design the PI controller based on the phase-lag controller by writing Eq. (11-179) as

$$G_{c1}(s) = 0.005\left(\frac{s+20}{s+0.1}\right) \tag{11-182}$$

Thus, we can set $K_p = 0.005$ and $K_I/k_p = 20$. Then, $K_I = 0.1$. Figure 11-62 shows the unit-step response of the system with the notch-PI controller. The attributes of the step response are as follows:

% Maximumovershoot = 1%

Rise time $t_r = 0.1380$ s

settling time $t_s = 0.1818$ s

which are extremely close to those with the notch-phase-lag controller, except that in the notch-PI case the steady-state velocity error is zero when the input is a step function.

Sensitivity Due to Imperfect Pole-Zero Cancellation As mentioned earlier, exact cancellation of poles and zeros is almost never possible in real life. Let us consider that the numerator polynomial of the notch controller in Eq. (11-152) cannot be exactly realized by physical resistor and capacitor components. Rather, the transfer function of the notch controller is more realistically chosen as

$$G_c(s) = \frac{s^2 + 100s + 1{,}000{,}000}{s^2 + 20{,}000s + 1{,}000{,}000} \tag{11-183}$$

Figure 11-62 shows the unit-step response of the system with the notch controller in Eq. (11-183). The attributes of the unit-step response are as follows:

% Maximumovershoot = 0.4%

Rise time t_r = 0.17 s

settling time t_s = 0.2323 s

Frequency-Domain Design To carry out the design of the notch controller, we refer to the Bode plot of Eq. (11-171) shown in Fig. 11-63. Due to the complex-conjugate poles of $G_p(s)$, the magnitude plot has a peak of 24.86 dB at 1095 rad/s. From the Bode plot in Fig. 11-63 we see that we may want to bring the magnitude plot down to –20 dB at the resonant frequency of 1095 rad/s so that the resonance is smoothed out. This requires an attenuation of –44.86 dB. Thus, from Eq. (11-165),

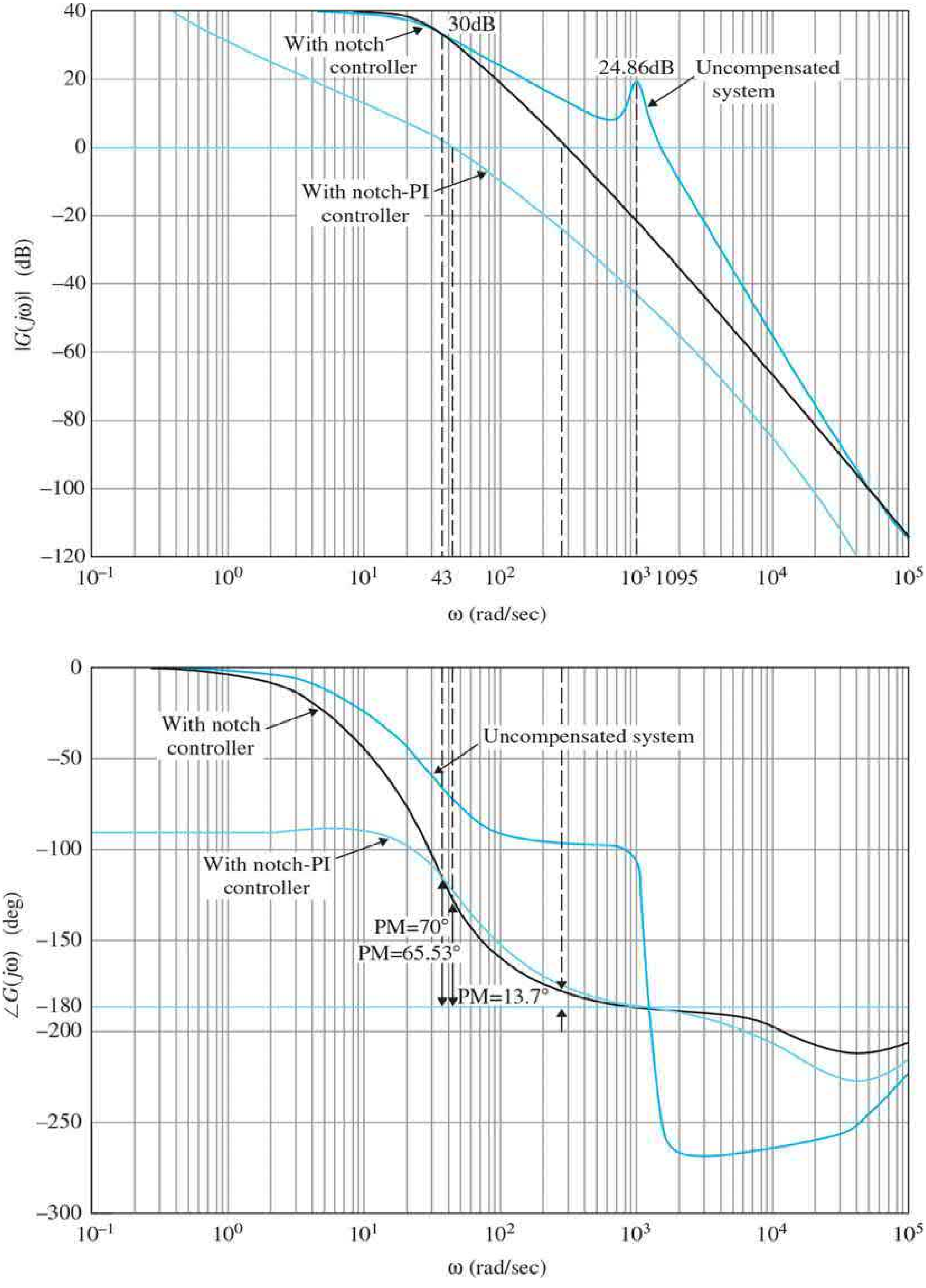

40
20
0
−20
−40
−60
−80
−100
−120
|G(jω)| (dB)
30dB
With notch controller
24.86dB
Uncompensated system
With notch-PI controller
10−1
10^0
10^1
43
10^2
10^3
1095
10^4
10^5
ω (rad/sec)
0
−50
−100
−150
−180
−200
−250
−300
∠G(jω) (deg)
With notch controller
Uncompensated system
With notch-PI controller
PM=70°
PM=65.53°
PM=13.7°
10−1
10^0
10^1
10^2
10^3
10^4
10^5
ω (rad/sec)

Figure 11-63 Bode plots of the uncompensated speed-control system in Example 11-6-1, with notch controller and with notch-PI controller.

$$\left|G_c(j\omega_c)\right| = -44.86\ \text{dB} = \frac{\zeta_z}{\zeta_p} = \frac{0.0435}{\zeta_p} \tag{11-184}$$

where $\zeta_z = 95.3/2\sqrt{1198606.6} = 0.0435$ is found from the numerator of Eq. (11-172). Solving for ζ_p from the last equation, we get ζ_p = 7.612. The attenuation should be placed at the resonant frequency of 1095 rad/s; thus, ω_n = 1095 rad/s. The notch controller of Eq. (11-162) becomes

$$G_c(s) = \frac{s^2 + 95.3s + 1{,}198{,}606.6}{s^2 + 16{,}670.28s + 1{,}199{,}025} \tag{11-185}$$

The Bode plot of the system with the notch controller in Eq. (11-185) is shown in Fig. 11-63. We can see that the system with the notch controller has a phase margin of only 13.7°, and M_r is 3.92.

To complete the design, we can use a PI controller as a second-stage controller. Following the guideline given in Sec. 11-3 on the design of a PI controller, we assume that the desired phase margin is 80°. From the Bode plot in Fig. 11-63, we see that, to realize a phase margin of 80°, the new gain-crossover frequency should be ω_g = 43 rad/s, and the magnitude of $G(j\omega_g)$ is 30 dB. Thus, from Eq. (11-42),

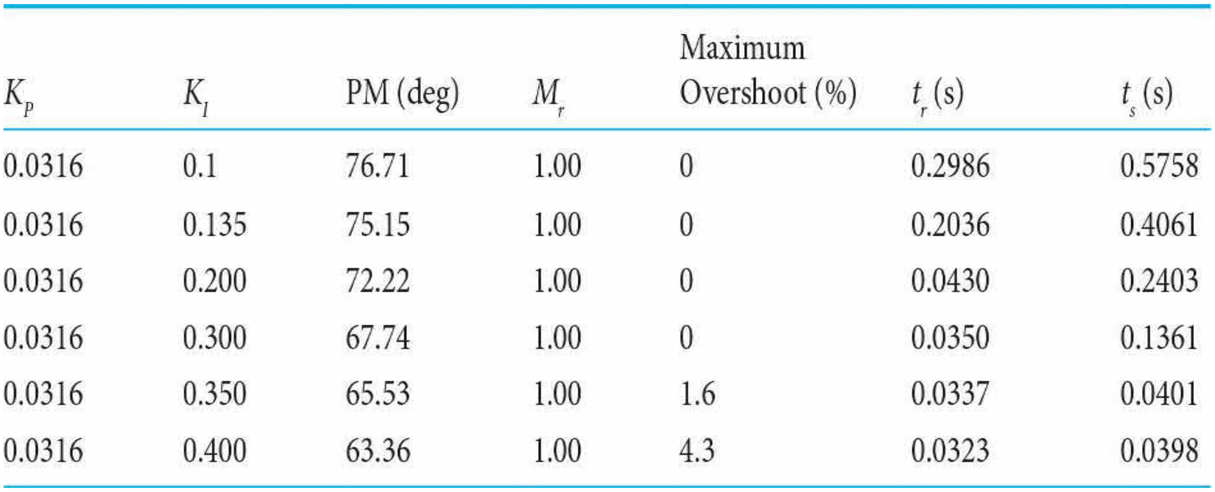
$$K_P = 10^{-\left|G(j\omega'_g)\right|_{\text{dB}}/20} = 10^{-30/20} = 0.0316 \tag{11-186}$$

The value of K_I is determined using the guideline given by Eq. (11-43),

$$K_I = \frac{\omega_g K_P}{10} = \frac{43 \times 0.0316}{10} = 0.135 \tag{11-187}$$

Because the original system is type 0, the final design needs to be refined by adjusting the value of K_I. Table 11-23 gives the performance attributes when K_p = 0.0316 and K_I is varied from 0.135. From the best maximum overshoot, rise time, and settling time measures, the best value of K_I appears to be 0.35. The forward-path transfer function of the compensated system with the notch-PI controller is

Table 11-23 Performance Attributes of System in Example 11-6-1 with Notch-PI Controller Designed in Frequency Domain



K_P	K_I	PM (deg)	M_r	Maximum Overshoot (%)	t_r (s)	t_s (s)
0.0316	0.1	76.71	1.00	0	0.2986	0.5758
0.0316	0.135	75.15	1.00	0	0.2036	0.4061
0.0316	0.200	72.22	1.00	0	0.0430	0.2403
0.0316	0.300	67.74	1.00	0	0.0350	0.1361
0.0316	0.350	65.53	1.00	1.6	0.0337	0.0401
0.0316	0.400	63.36	1.00	4.3	0.0323	0.0398

$$\frac{Y(s)}{R(s)} = \frac{G_{cf}(s)G_c(s)G_p(s)}{1+G_c(s)G_p(s)} \tag{11-189}$$

Figure 11-63 shows the Bode plot of the system with the notch-PI controller, with $K_P = 0.0316$ and $K_I = 0.35$. The unit-step responses of the compensated system with $K_P = 0.0316$ and $K_I = 0.135, 0.35$ and 0.40 are shown in Fig. 11-64.

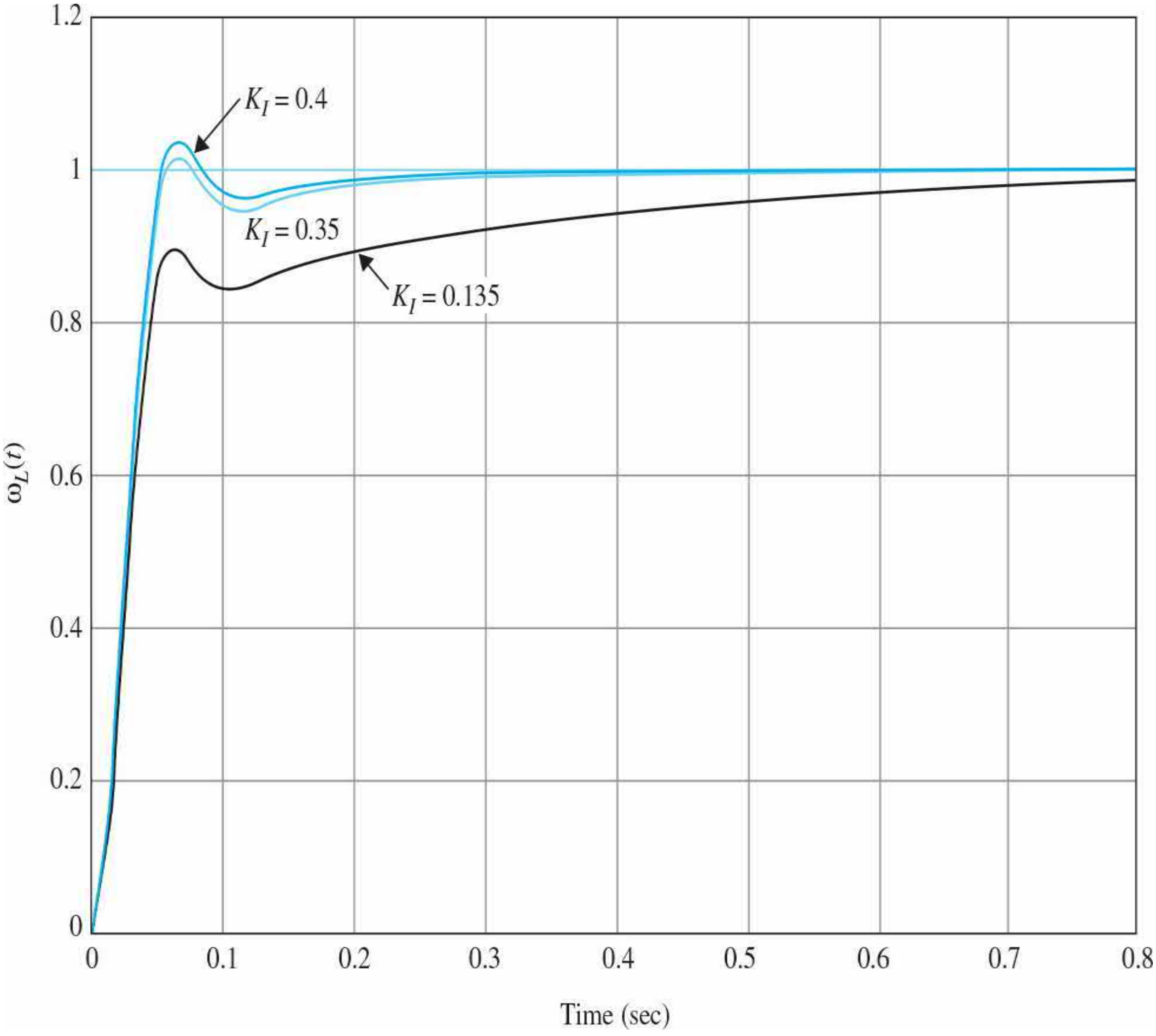


Figure 11-64 Unit-step responses of speed-control system in Example 11-6-1 with notch-PI controller,

$$G_c(s) = \frac{s^2 + 95.3s + 1{,}198{,}606.6}{s^2 + 16{,}670.28s + 1{,}199{,}025} \quad G_{c2}(s) = 0.0316 + \frac{0.35}{s}.$$

11-7 FORWARD AND FEEDFORWARD CONTROLLERS

The compensation schemes discussed in the preceding sections all have one degree of freedom in that there is essentially one controller in the system, although the controller can contain several stages connected in series or in parallel. The limitations of a one-degree-of-freedom controller were discussed in Sec. 11-1. The two-degree-of-freedom compensation scheme shown in Fig. 11-2*d* through *f* offer design flexibility when a multiple number of design criteria have to be satisfied simultaneously.

From Fig. 11-2*e*, the closed-loop transfer function of the system is

$$\frac{Y(s)}{R(s)}=\frac{G_{cf}(s)G_c(s)G_p(s)}{1+G_c(s)G_p(s)} \qquad (11\text{-}189)$$

and the error transfer function is

$$\frac{E(s)}{R(s)}=\frac{1}{1+G_c(s)G_p(s)} \qquad (11\text{-}190)$$

Thus, the controller G_c(s) can be designed so that the error transfer function will have certain desirable characteristics, and the controller $G_{cf}(s)$ can be selected to satisfy performance requirements with reference to the input-output relationship. Another way of describing the flexibility of a two-degree-of-freedom design is that the controller $G_c(s)$ is usually designed to provide a certain degree of system stability and performance, but because the zeros of $G_c(s)$ always become the zeros of the closed-loop transfer function, unless some of the zeros are canceled by the poles of the process transfer function, $G_p(s)$, these zeros may cause a large overshoot in the system output even when the relative damping as determined by the characteristic equation is satisfactory. In this case and for other reasons, the transfer function $G_{cf}(s)$ may be used for the control or cancellation of the undesirable zeros of the closed-loop transfer function, while keeping the characteristic equation intact. Of course, we can also introduce zeros in $G_{cf}(s)$ to cancel some of the undesirable poles of the closed-loop transfer function that could not be otherwise affected by the controller $G_c(s)$. The feedforward compensation scheme shown in Fig. 11-2*f* serves the same purpose as the forward compensation, and the difference between the two configurations depends on system and hardware implementation considerations.

It should be kept in mind that, while the forward and feedforward compensations may seem powerful because they can be used directly for the addition or deletion of poles and zeros of the closed-loop transfer function, there is a fundamental question involving the basic characteristics of feedback. If the forward or feedforward controller is so powerful, then why do we need feedback at all? Because $G_{cf}(s)$ in the systems of Fig. 11-2*e* and *f* are outside the feedback loop, the system is susceptible to parameter variations in $G_{cf}(s)$. Therefore, in reality, these types of compensation cannot be satisfactorily applied to all situations.

EXAMPLE 11-7-1 As an illustration of the design of the forward and feedforward compensators, consider the second-order sun-seeker system with phase-lag control designed in Example 11-5-4. One of the disadvantages of phase-lag compensation is that the rise time is usually quite long. Let us consider that the phase-lag-compensated sun-seeker system has the forward-path transfer function:

$$G(s) = G_c(s)G_p(s) = \frac{2500(1+10s)}{s(s+25)(1+100s)} \qquad (11\text{-}191)$$

The time-response attributes are as follows:

Maximumovershoot = 2.5%

t_r = 0.1637 s

t_r = 0.2020 s

We can improve the rise time and the settling time while not appreciably increasing the overshoot by adding a PD controller $G_{cf}(s)$ to the system, as shown in Fig. 11-65*a*. This effectively adds a zero to the closed-loop transfer function while not affecting the characteristic equation. Selecting the PD controller as

$$G_{cf}(s) = 1 + 0.05s \qquad (11\text{-}192)$$

the time-domain performance attributes are as follows:

Maximumovershoot = 4.3%

t_r = 0.1069 s

t_r = 0.1313 s

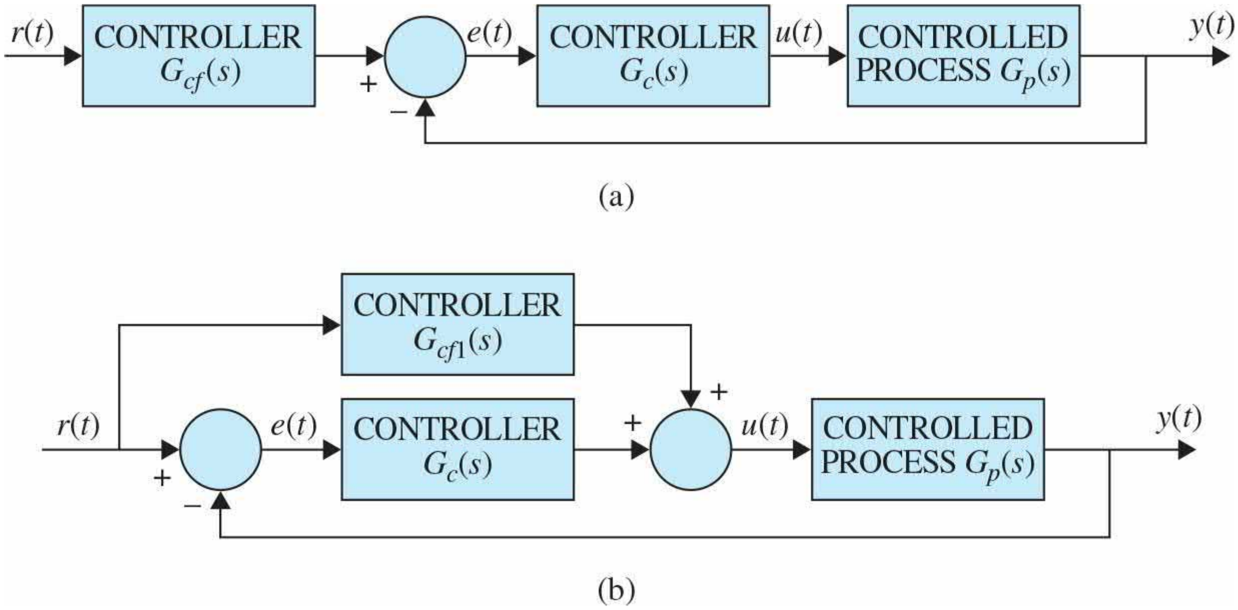


Figure 11-65 (*a*) Forward compensation with series compensation. (*b*) Feedforward compensation with series compensations.

If instead the feedforward configuration of Fig. 11-65*b* is chosen, the transfer function of $G_{cf1}(s)$ is directly related to $G_{cf}(s)$, that is, equating the closed-loop transfer functions of the two systems in Fig. 11-65*a* and *b*, we have

$$\frac{[G_{cf1}(s)+G_c(s)]G_p(s)}{1+G_c(s)G_p(s)}=\frac{G_{cf}G_c(s)G_p(s)}{1+G_c(s)G_p(s)} \tag{11-193}$$

Solving for $G_{cf}1(s)$ from Eq. (11-193) yields

$$G_{cf1}(s)=[G_{cf}(s)-1]G_c(s) \tag{11-194}$$

Thus, with $G_{cf}(s)$ as given in Eq. (11-189), we have the transfer function of the feedforward controller:

$$G_{cf1}(s)=0.05s\left(\frac{1+10s}{1+100s}\right) \tag{11-195}$$

11-8 DESIGN OF ROBUST CONTROL SYSTEMS

In many control-system applications, not only must the system satisfy the damping and accuracy specifications, but the control must also yield performance that is **robust** (insensitive) to external disturbance and parameter variations. We have shown that feedback in conventional control systems has the inherent ability to reduce the effects of external disturbance and parameter variations. Unfortunately, robustness with the conventional feedback configuration is achieved only with a high loop gain, which is normally detrimental to stability. Let us consider the control system shown in Fig. 11-66. The external disturbance is denoted by the signal *d*(*t*), and we assume that the amplifier gain *K* is subject to variation during operation. The input-output transfer function of the system when *d*(*t*)is

$$M(s)=\frac{Y(s)}{R(s)}=\frac{KG_{cf}(s)G_c(s)G_p(s)}{1+KG_c(s)G_p(s)} \qquad (11\text{-}196)$$

and the disturbance-output transfer function when *r*(*t*) is

$$T(s)=\frac{Y(s)}{D(s)}=\frac{1}{1+KG_c(s)G_p(s)} \qquad (11\text{-}197)$$

In general, the design strategy is to select the controller $G_c(s)$ so that the output *y*(*t*) is insensitive to the disturbance over the frequency range in which the latter is dominant and to design the feedforward controller $G_{cf}(s)$ to achieve the desired transfer function between the input *r*(*t*) and the output *y*(*t*).

Let us define the sensitivity of *M*(*s*) due to the variation of *K* as

$$S_K^M=\frac{\text{percent change in } M(s)}{\text{percent change in } K}=\frac{dM(s)/M(s)}{dK/K} \qquad (11\text{-}198)$$

Then, for the system in Fig. 11-66,

$$S_K^M=\frac{1}{1+KG_c(s)G_p(s)} \qquad (11\text{-}199)$$

which is identical to Eq. (11-197). Thus, the sensitivity function and the disturbance-output transfer function are identical, which means that disturbance suppression and robustness with respect to variations of *K* can be designed with the same control schemes.

The following example shows how the two-degree-of-freedom control system of Fig. 11-66 can be used to achieve a high-gain system that will satisfy the performance and robustness requirements, as well as noise rejection.

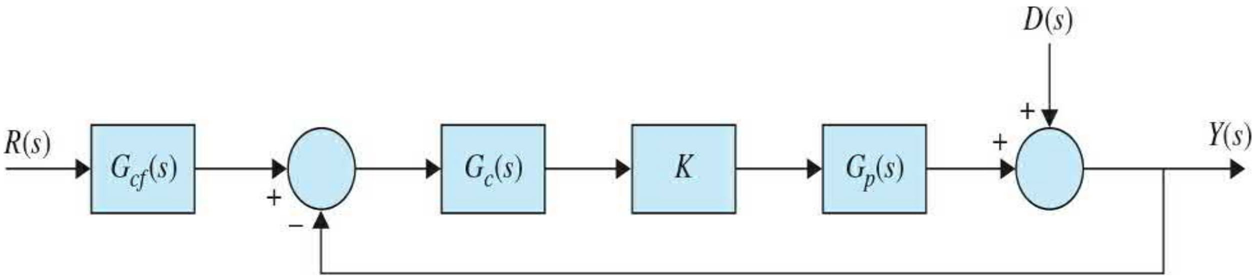


Figure 11-66 Control system with disturbance.

EXAMPLE 11-8-1 Let us consider the second-order sun-seeker system in Example 11-5-4, which is compensated with phase-lag control. The forward-path transfer function is

$$G_p(s)=\frac{2500K}{s(s+25)} \qquad (11\text{-}200)$$

where *K* = 1. The forward-path transfer function of the phase-lag-compensated system with *a* = 0.1 and *T* = 100 is

$$G(s)=G_c(s)G_p(s)=\frac{2500K(1+10s)}{s(s+25)(1+100s)} \quad (K=1) \qquad (11\text{-}201)$$

Because the phase-lag controller is a low-pass filter, the sensitivity of the closed-loop transfer function *M*(*s*) with respect to *K* is poor. The bandwidth of the system is only 13.97 rad/s, but it is expected that $|S_K^M(j\omega)|$ will be greater than unity at frequencies beyond 13.97 rad/s. Figure 11-67 shows the unit-step responses of the system when *K* = 1, the nominal value, and *K* = 0.5 and 2.0. Notice that, if for some reason, the forward gain *K* is changed from its nominal value, the system response of the phase-lag-compensated system would vary substantially. The attributes of the step responses and the characteristic equation roots are shown in Table 11-24 for the three values of

K. Figure 11-68 shows the root loci of the system with the phase-lag controller. The two complex roots of the characteristic equation vary substantially as *K* varies from 0.5 to 2.0.

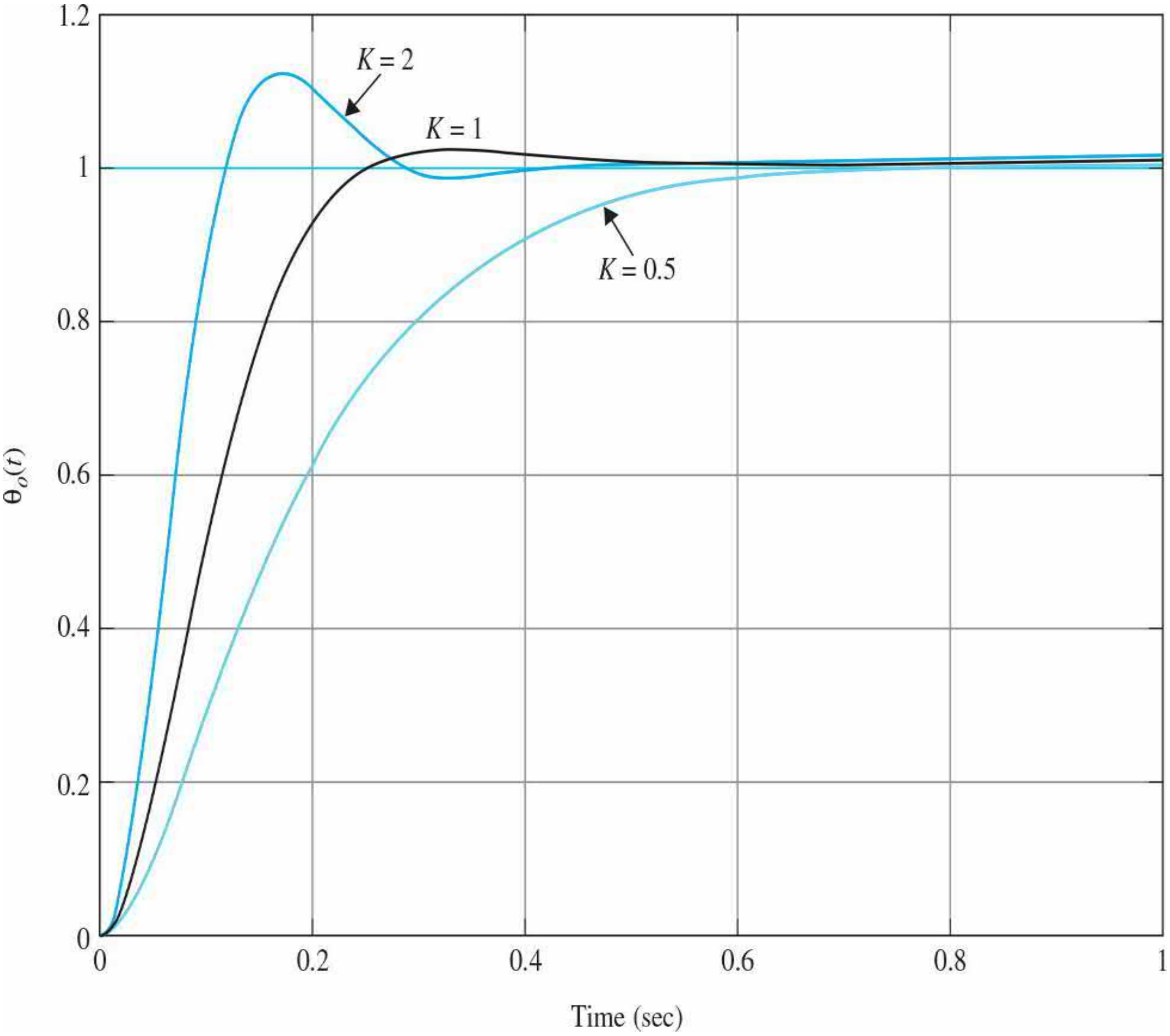


Figure 11-67 Unit-step responses of the second-order sun-seeker system with phase-lag controller, $G(s) = \dfrac{2500(1+10s)}{s(s+25)(1+100s)}$.

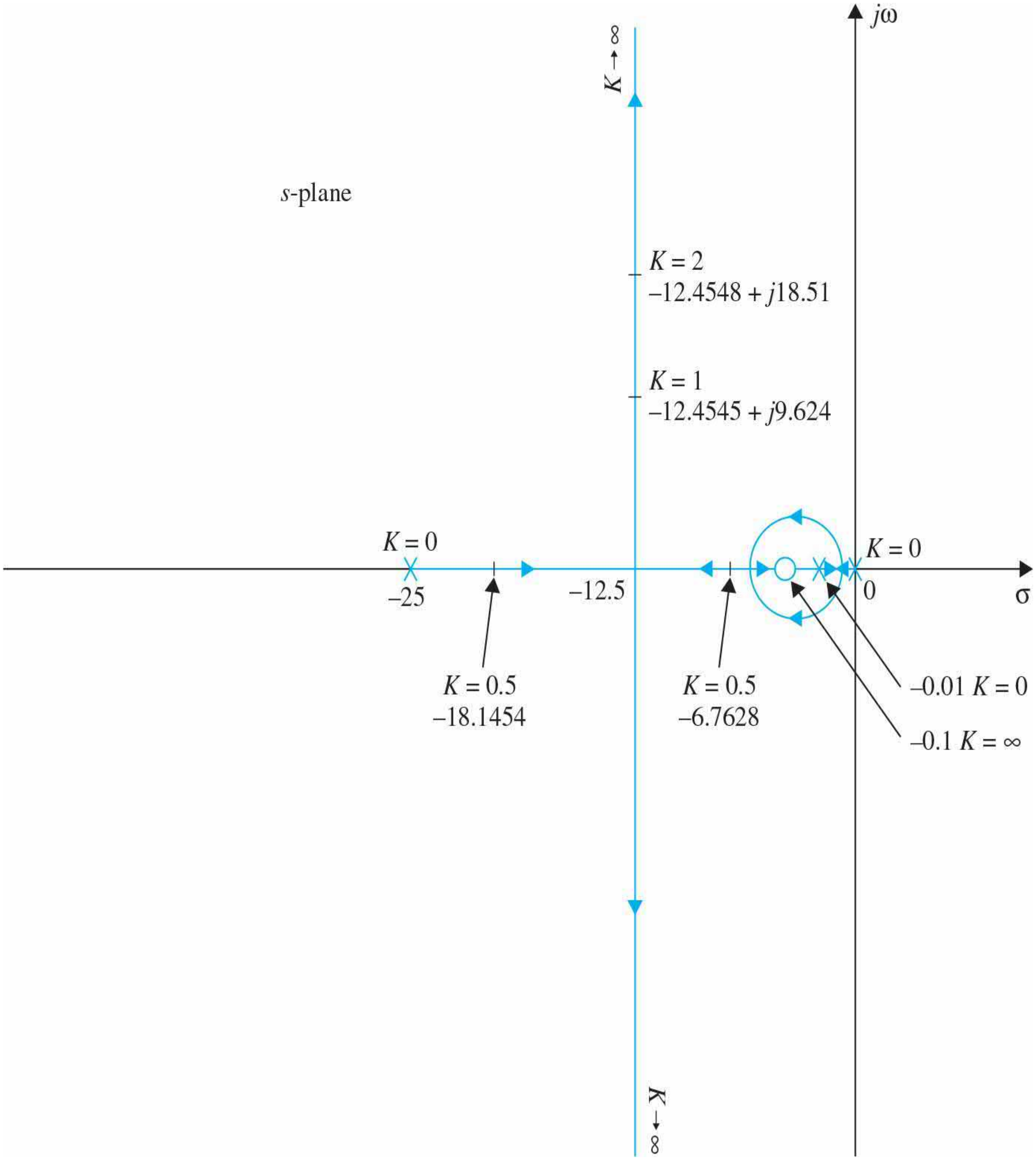
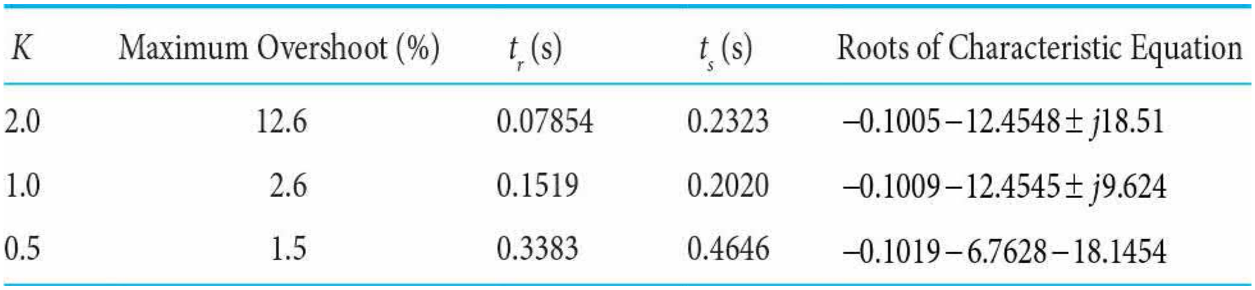


Figure 11-68 Root loci of the second-order sun-seeker system with phase-lag controller, $G(s)=\frac{2500(1+10s)}{s(s+25)(1+100s)}$.

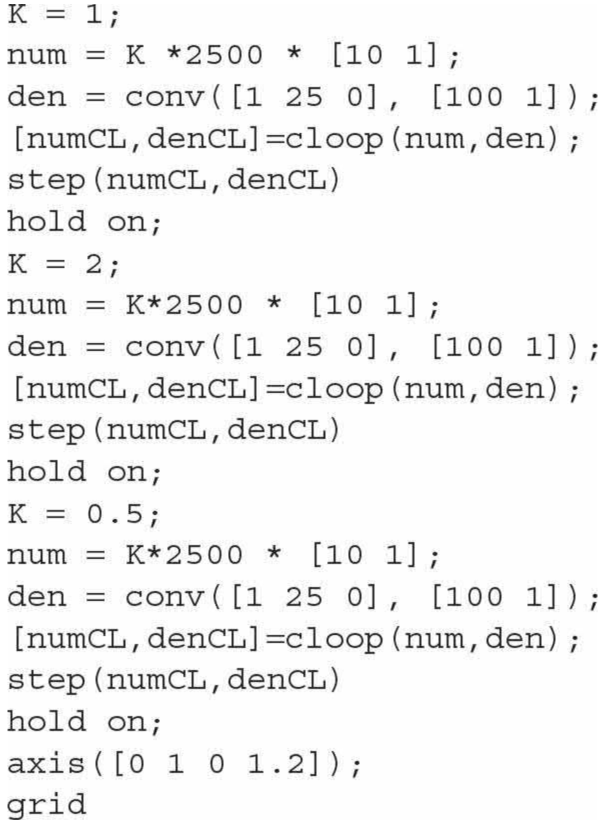
Table 11-24 Attributes of Unit-Step Response of Second-Order Sun-Seeker System with Phase-Lag Controller in Example 11-8-1



K	Maximum Overshoot (%)	t_r (s)	t_s (s)	Roots of Characteristic Equation
2.0	12.6	0.07854	0.2323	$-0.1005 - 12.4548 \pm j18.51$
1.0	2.6	0.1519	0.2020	$-0.1009 - 12.4545 \pm j9.624$
0.5	1.5	0.3383	0.4646	$-0.1019 - 6.7628 - 18.1454$

Toolbox 11-8-1

Figure 11-67 is obtained by the following sequence of MATLAB functions:



```
K = 1;
num = K *2500 * [10 1];
den = conv([1 25 0], [100 1]);
[numCL,denCL]=cloop(num,den);
step(numCL,denCL)
hold on;
K = 2;
num = K*2500 * [10 1];
den = conv([1 25 0], [100 1]);
[numCL,denCL]=cloop(num,den);
step(numCL,denCL)
hold on;
K = 0.5;
num = K*2500 * [10 1];
den = conv([1 25 0], [100 1]);
[numCL,denCL]=cloop(num,den);
step(numCL,denCL)
hold on;
axis([0 1 0 1.2]);
grid
```

The design strategy of the robust controller is to place two zeros of the controller near the desired closed-loop poles, which according to the phase-lag-compensated system are at $s = -12.455 \pm j9.624$. Thus, we let the controller transfer function be

$$G_c(s) = \frac{(s+13+j10)(s+13-j10)}{269} = \frac{(s^2+26s+269)}{269} \tag{11-202}$$

The forward-path transfer function of the system with the robust controller is

$$G(s) = \frac{9.2937K(s^2+26s+269)}{s(s+25)} \tag{11-203}$$

Figure 11-69 shows the root loci of the system with the robust controller. By placing the two zeros of $G_c(s)$ near the desired characteristic equation roots, the sensitivity of the system is greatly improved. In fact, the root sensitivity near the two complex zeros at which the root loci terminate is very low. Figure 11-69 shows that, when K approaches infinity, the two characteristic equation roots approach $-13 \pm j10$.

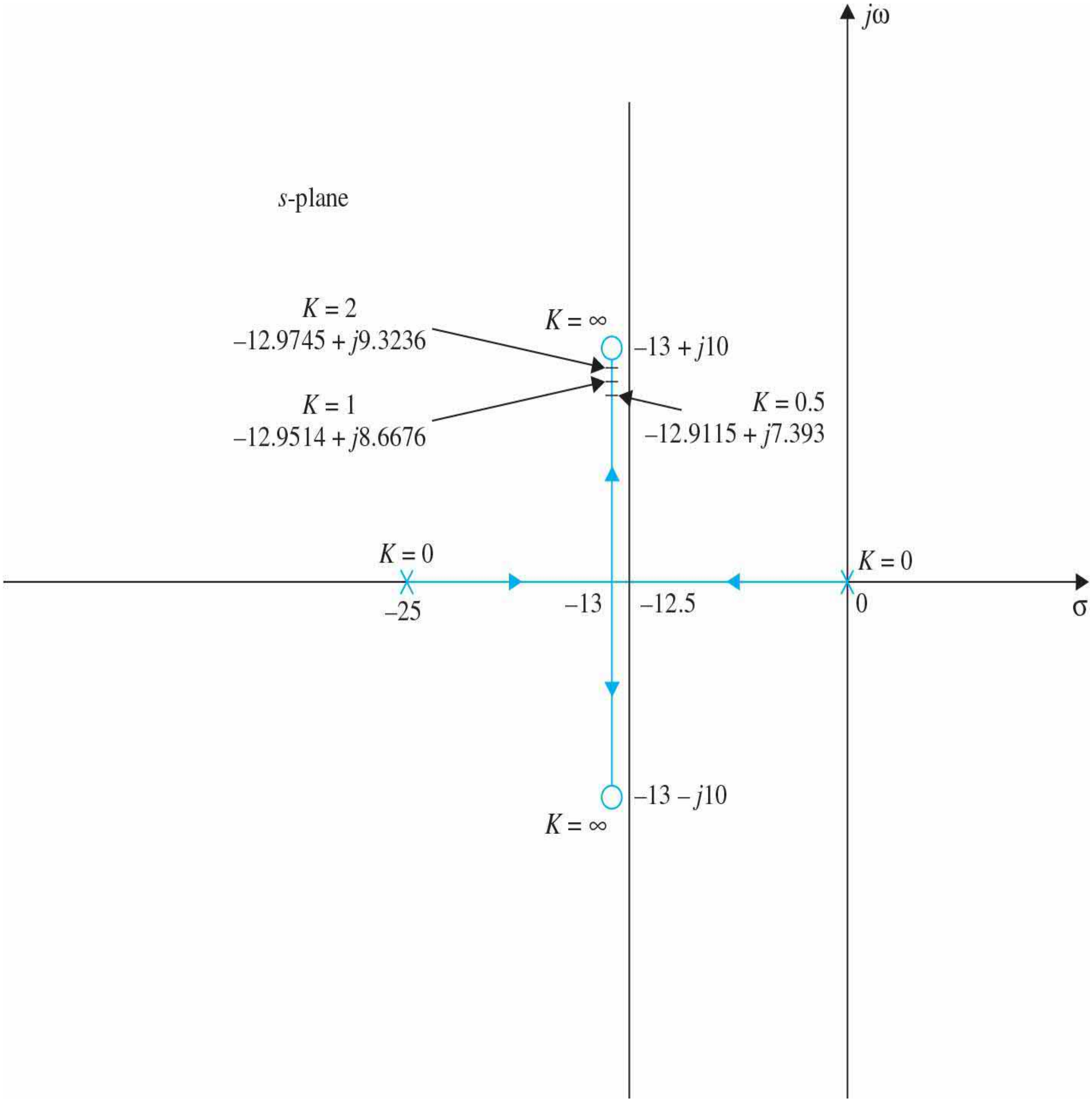


Figure 11-69 Root loci of the second-order sun-seeker system with robust controller, $G(s)=\frac{2500(1+10s)}{s(s+25)(1+100s)}$.

Toolbox 11-8-2

Figure 11-68 is obtained by the following sequence of MATLAB functions:

```
num = 2500 * [10 1];
den = conv([1 25 0], [100 1]);
rlocus(num,den);
axis([-30 10 -20 20])
% Use the cursor to obtain values of K and the poles
```

Toolbox 11-8-3

Figure 11-69 is obtained by the following sequence of MATLAB functions:

```
num = 9.2937*[1 26 269];
den = [1 25 0];
rlocus(num,den);
% Use the cursor to obtain values of K and the poles
```

Because the zeros of the forward-path transfer function are identical to the zeros of the closed-loop transfer function, the design is not complete by using only the series controller $G_c(s)$ because the closed-loop zeros will essentially cancel the closed-loop poles. This means we must add the forward controller, as shown in Fig. 11-70, where $G_{cf}(s)$ should contain poles to cancel the zeros of $s^2 + 26s + 269$ of the closed-loop transfer function. Thus, the transfer function of the forward controller is

$$G_{cf}(s) = \frac{269}{s^2 + 26s + 269} \tag{11-204}$$

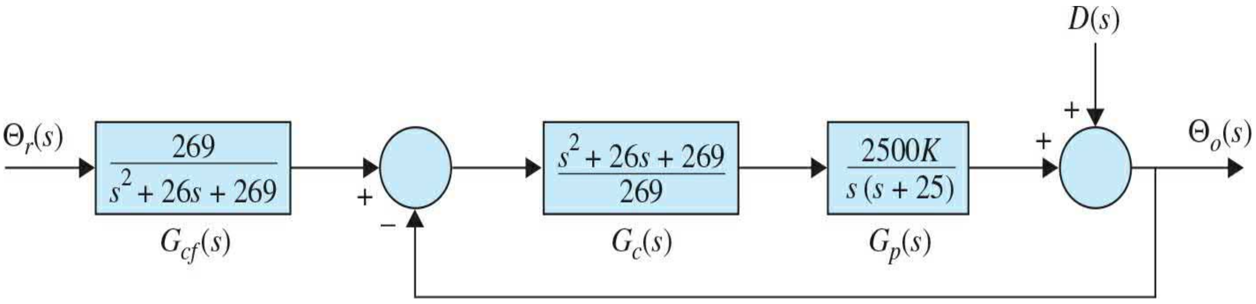


Figure 11-70 Second-order sun-seeker system with robust controller and forward controller.

The block diagram of the overall system is shown in Fig. 11-70. The closed-loop transfer function of the compensated system with *K* = 1 is

$$\frac{\Theta_o(s)}{\Theta_r(s)} = \frac{242.88}{s^2 + 25.903s + 242.88} \qquad (11\text{-}205)$$

The unit-step responses of the system for *K* = 0.5, 1.0, and 2.0 are shown in Fig. 11-71, and their attributes are given in Table 11-25. As shown, the system is now very insensitive to the variation of *K*.

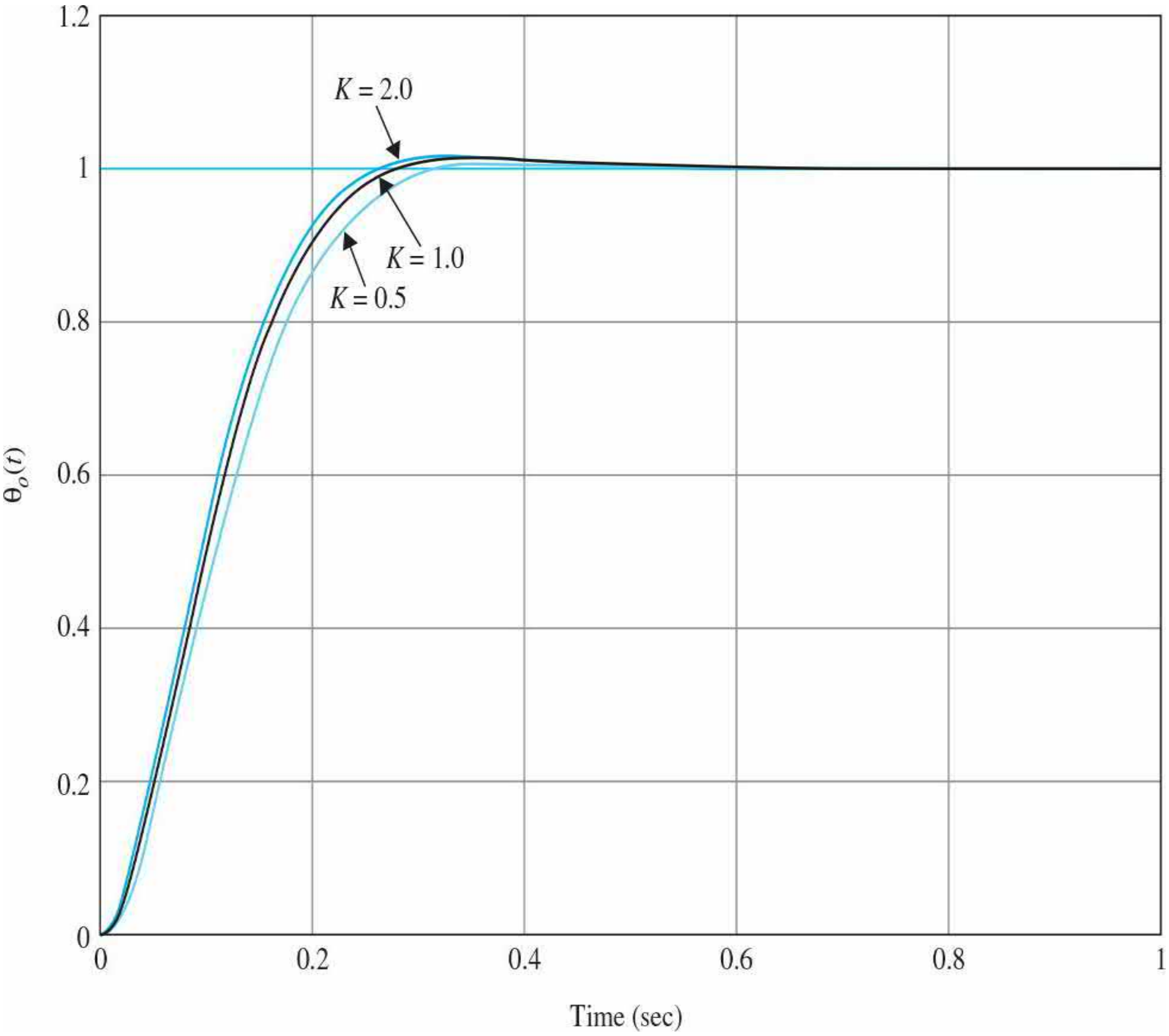
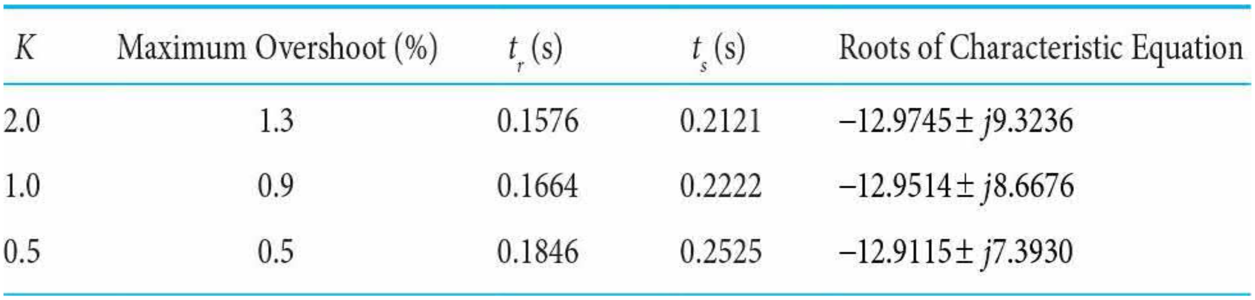


Figure 11-71 Unit-step responses of the second-order sun-seeker system with robust controller and forward controller.

Table 11-25 Attributes of Unit-Step Response of Second-Order Sun-Seeker System with Robust Controller in Example 11-8-1



K	Maximum Overshoot (%)	t_r (s)	t_s (s)	Roots of Characteristic Equation
2.0	1.3	0.1576	0.2121	$-12.9745 \pm j9.3236$
1.0	0.9	0.1664	0.2222	$-12.9514 \pm j8.6676$
0.5	0.5	0.1846	0.2525	$-12.9115 \pm j7.3930$

Because the system in Fig. 11-70 is now more robust, it is expected that the disturbance effect will be reduced. However, we cannot evaluate the effect of the controllers in the system of Fig. 11-70 by applying a unit-step function as $d(t)$. The true improvement on the noise rejection properties is more appropriately analyzed by investigating the frequency response of $\Theta_o(s)/D(s)$. The noise-to-output transfer function, written from Fig. 11-70, is

$$\frac{\Theta_o(s)}{D(s)} = \frac{1}{1+G_c(s)G_p(s)} = \frac{s(s+25)}{10.2937s^2+266.636s+2500} \qquad (11\text{-}206)$$

The amplitude Bode plot of Eq. (11-206) is shown in Fig. 11-72, along with those of the uncompensated system and the system with phase-lag control. Notice that the magnitude of the frequency response between $D(s)$ and $\Theta_o(s)$ is much smaller than those of the system without compensation and with phase-lag control. The phase-lag control also accentuates the noise for frequencies up to approximately 40 rad/s, adding more stability to the system.

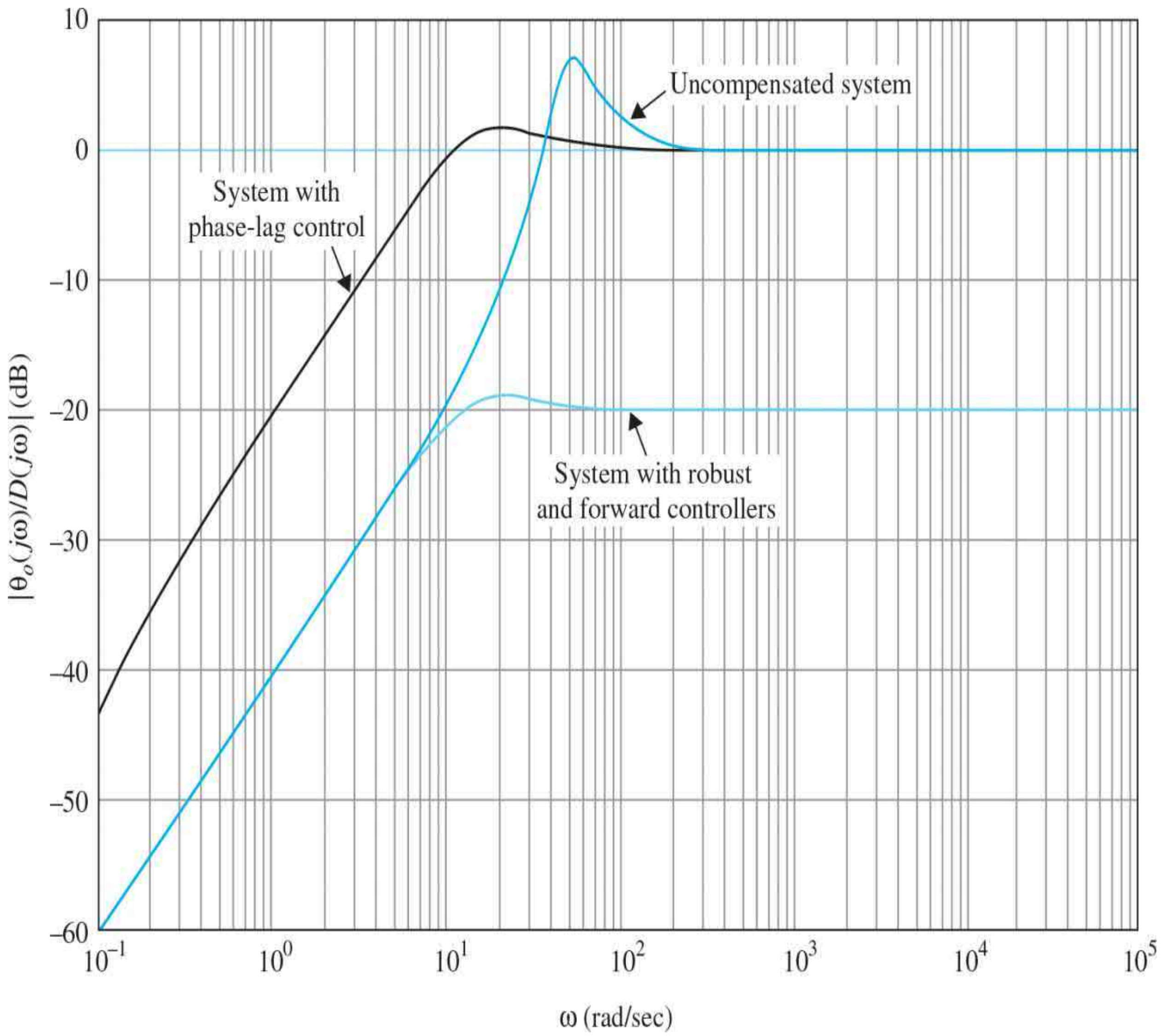


Figure 11-72 Amplitude Bode plot of response due to noise of second-order sun-seeker system.

EXAMPLE 11-8-2 In this example, a robust controller with forward compensation is designed for the third-order sun-seeker system in Example 11-5-5 with phase-lag control. The forward-path transfer function of the uncompensated system is

$$G_p(s)=\frac{156,250,000K}{s(s^2+625s+156,250)} \tag{11-207}$$

where K = 1. The root loci of the closed-loop system are shown in Fig. 11-

54, which lead to the phase-lag control with results shown in Table 11-20. Let us select the parameters of the phase-lag controller as *a* = 0.1 and *T* = 20. The dominant roots of the characteristic equation are $s = -187.73 \pm j164.93$.

Let us place the two zeros of the second-order robust controller at $-180 \pm j166.13$, so that the controller transfer function is

$$G_c(s) = \frac{s^2 + 360s + 60{,}000}{60{,}000} \qquad (11\text{-}208)$$

To ease the high-frequency realization problem of the controller, we may add two nondominant poles to $G_c(s)$. The following analysis is carried out with $G_c(s)$ given in Eq. (11-208), however. The root loci of the compensated system are shown in Fig. 11-73. Thus, by placing the zeros of the controller very close to the desired dominant roots, the system is very insensitive to changing values of *K* near and beyond the nominal value of *K*. The forward controller has the transfer function

$$G_{cf}(s) = \frac{60{,}000}{s^2 + 360s + 60{,}000} \qquad (11\text{-}209)$$

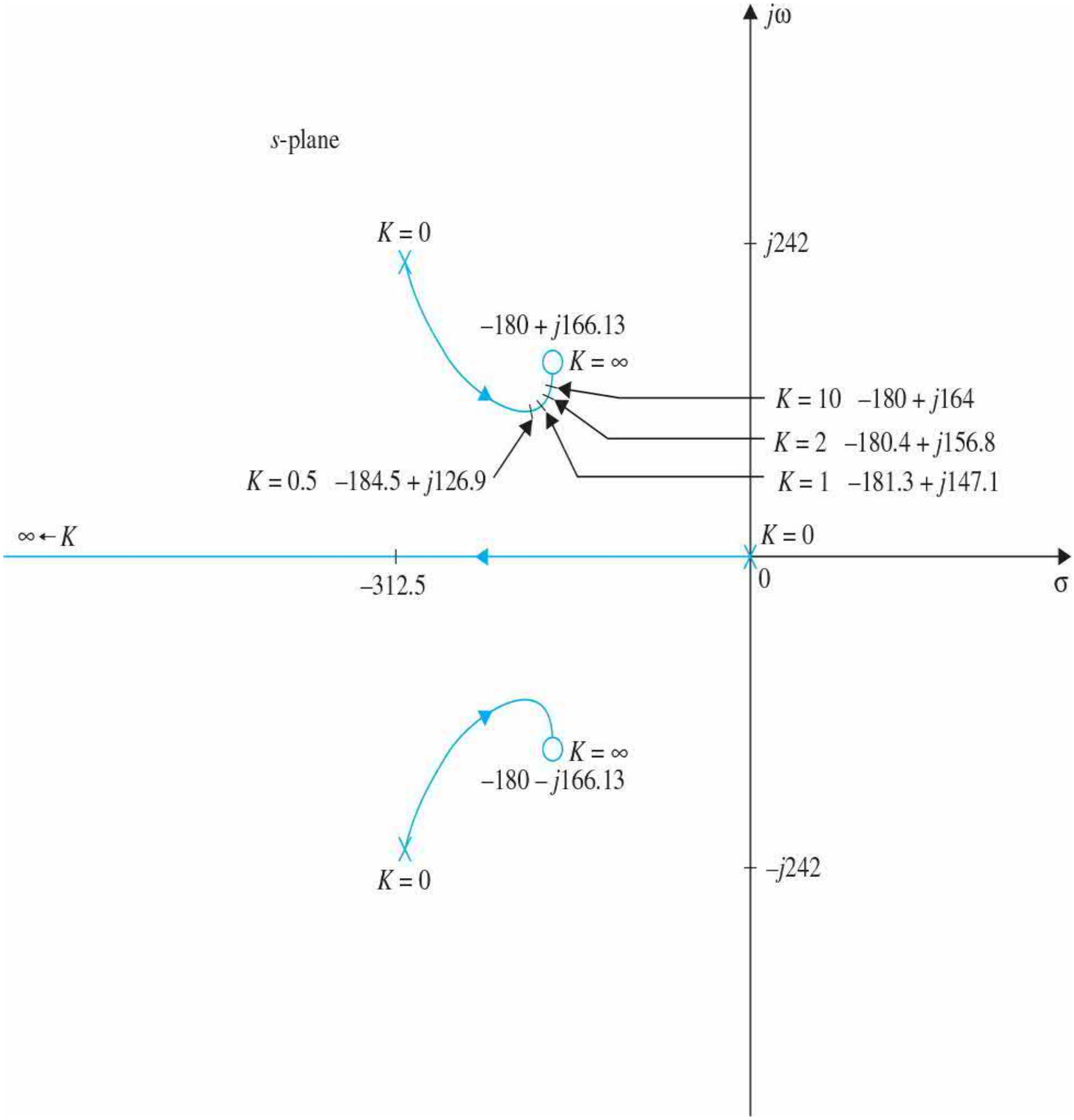


Figure 11-73 Root loci of the third-order sun-seeker system with robust and forward controllers.

The attributes of the unit-step response for $K = 0.5$, 1.0, 2.0, and 10.0 and the corresponding characteristic equation roots are given in Table 11-26.

Table 11-26 Attributes of Unit-Step Response and Characteristic Equation Roots of Third-Order Sun-Seeker System with Robust and

Forward Controllers in Example 11-8-2

K	Maximum Overshoot (%)	t_r (s)	t_s (s)	Characteristic Equation Roots
0.5	1.0	0.01115	0.01616	$-1558.1 - 184.5 \pm j126.9$
1.0	2.1	0.01023	0.01414	$-2866.6 - 181.3 \pm j147.1$
2.0	2.7	0.00966	0.01313	$-5472.6 - 180.4 \pm j156.8$
10.0	3.2	0.00924	0.01263	$-26307 - 180.0 \pm j164.0$

EXAMPLE 11-8-3 In this example, we consider the design of a position-control system that has a variable load inertia. This type of situation is quite common in control systems. For example, the load inertia seen by the motor in an electronic printer will change when different printwheels are used. The system should have satisfactory performance for all the printwheels intended to be used with the system.

To illustrate the design of a robust system that is insensitive to the variation of load inertia, consider that the forward-path transfer function of a unity-feedback control system is

$$G_p(s) = \frac{KK_i}{s[(Js+B)(Ls+B)+K_iK_b]} \tag{11-210}$$

The system parameters are as follows:

K_i = motor torque constant = 1 N·m/A

K_b = motor back-emf constant = 1 V/rad/s

R = motor resistance = 1 Ω

L = motor inductance = 0.01 H

B = motor and load viscous-friction coefficient $\cong 0$

J = motor and load inertia, varies between 0.01 and 0.02 N·m/rad/s^2

K = amplifier again

Substituting these system parameters into Eq. (11-210), we get

$$\text{For} \quad J = 0.01 \quad G_p(s) = \frac{10{,}000K}{s(s^2 + 100s + 10{,}000)} \tag{11-211}$$

$$\text{For} \quad J = 0.02 \quad G_p(s) = \frac{5000K}{s(s^2 + 100s + 5000)} \tag{11-212}$$

The performance specifications are as follows:

Ramp error constant $K_v \geq 200$

Maximum overshoot ≥ 5% or as small as possible

Rise time $t_r \geq 0.05$ s

Settling time $t_s \geq 0.05$ s

These specifications are to be maintained for $0.01 \leq J \leq 0.02$.

Figure 11-74 shows the root loci of the uncompensated system for $J = 0.01$ and $J = 0.02$. We see that, regardless of the value of J, the uncompensated system is unstable for $K > 100$.

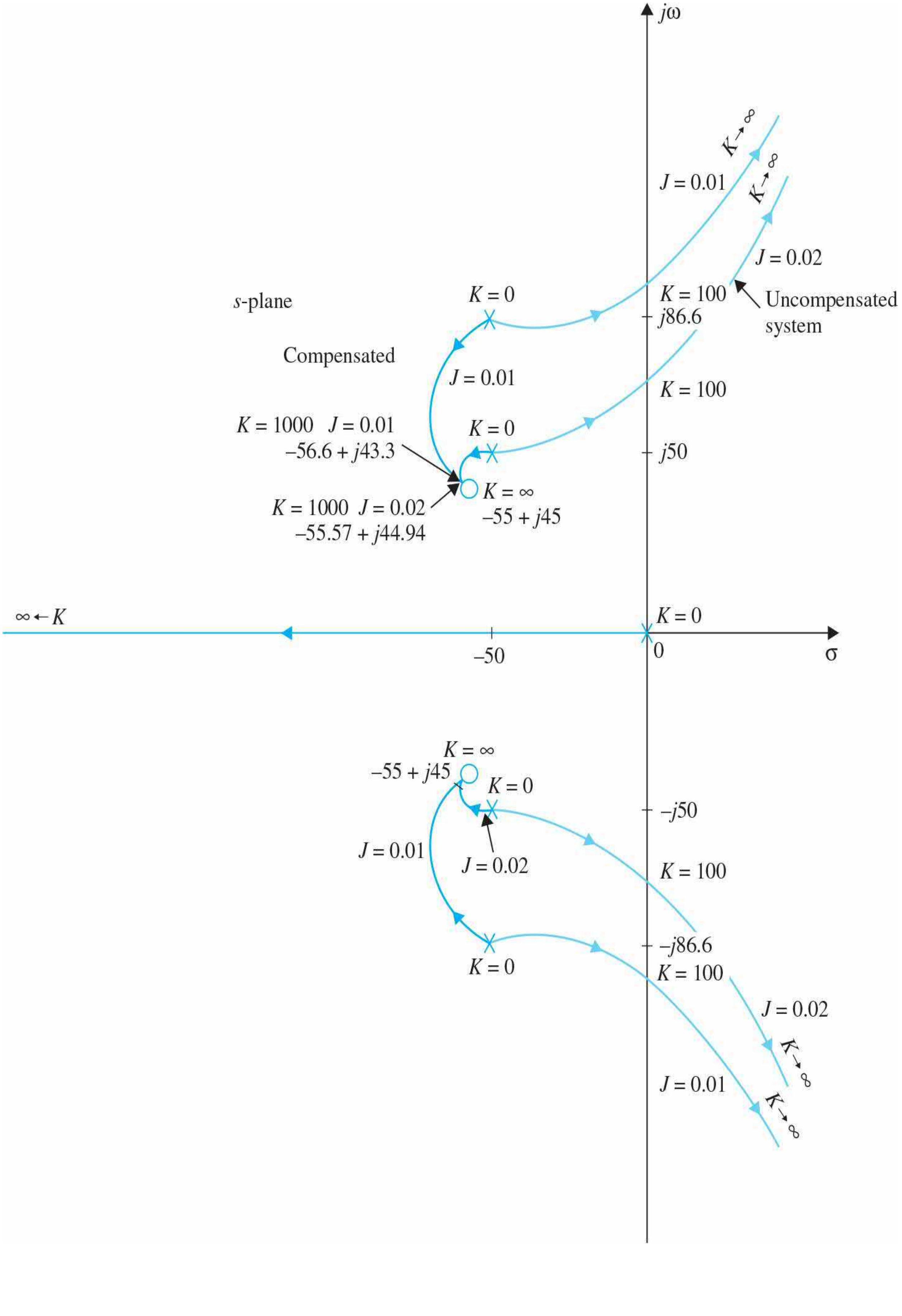

jω
K→∞
K→∞
J = 0.01
J = 0.02
s-plane
K = 0
K = 100
Uncompensated
system
j86.6
Compensated
J = 0.01
K = 100
K = 1000   J = 0.01
–56.6 + j43.3
K = 0
j50
K = ∞
–55 + j45
K = 1000   J = 0.02
–55.57 + j44.94
∞ ← K
K = 0
–50
0
σ
K = ∞
–55 + j45
K = 0
–j50
J = 0.01
J = 0.02
K = 100
–j86.6
K = 0
K = 100
J = 0.02
K→∞
J = 0.01
K→∞

Figure 11-74 Root loci of the position-control system in Example 11-8-3 with robust and forward controllers.

To achieve robust control, let us choose the system configuration of Fig. 11-65*a*. We introduce a second-order series controller with the zeros placed near the desired dominant characteristic equation of the compensated system. The zeros should be so placed that the dominant characteristic equation roots would be insensitive to the variation in *J*. This is done by placing the two zeros at –55 ± *j*45, although the exact location is unimportant. By choosing the two controller zeros as designated, the root loci of the compensated system show that the two complex roots of the characteristic equation will be very close to these zeros for various values of *J*, especially when the value of *K* is large. The transfer function of the robust controller is

$$G_c(s) = \frac{(s^2 + 110s + 5050)}{5050} \tag{11-213}$$

As in the last example, we may add two nondominant poles to $G_c(s)$ to ease the high-frequency realization problem of the controller. The analysis is carried out with $G_c(s)$ given in Eq. (11-213).

Let $K = 1000$, although 200 would have been adequate to satisfy the K_v requirement. Then, for $J = 0.01$, the forward-path transfer function of the compensated system is

$$G(s) = G_c(s)G_p(s) = \frac{1980.198(s^2 + 110s + 5050)}{s(s^2 + 100s + 10{,}000)} \tag{11-214}$$

and for $J = 0.02$,

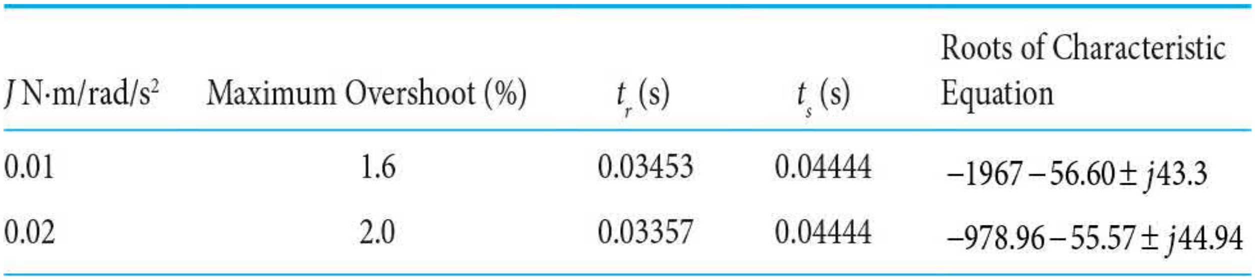
$$G(s) = \frac{990.99(s^2 + 110s + 5050)}{s(s^2 + 100s + 5000)} \tag{11-215}$$

To cancel the two zeros of the closed-loop transfer function, the transfer function of the forward controller is

$$G_{cf}(s) = \frac{5050}{s^2 + 110s + 5050} \tag{11-216}$$

The attributes of the unit-step response and the characteristic equation roots of the compensated system with $K = 1000$, $J = 0.01$ and $J = 0.02$, are given in Table 11-27.

Table 11-27 Attributes of Unit-Step Response and Characteristic Equation Roots of System with Robust and Forward Controllers in Example 11-8-3



J N·m/rad/s^2	Maximum Overshoot (%)	t_r (s)	t_s (s)	Roots of Characteristic Equation
0.01	1.6	0.03453	0.04444	$-1967 - 56.60 \pm j43.3$
0.02	2.0	0.03357	0.04444	$-978.96 - 55.57 \pm j44.94$

11-9 MINOR-LOOP FEEDBACK CONTROL

The control schemes discussed in the preceding sections have all utilized series controllers in the forward path of the main loop or feedforward path of the control system. Although series controllers are the most common because of their simplicity in implementation, depending on the nature of the system, sometimes there are advantages in placing the controller in a minor feedback loop, as shown in Fig. 11-2*b*. For example, a tachometer may be coupled directly to a dc motor not only for the purpose of speed indication but more often for improving the stability of the closed-loop system by feeding back the output signal of the tachometer. The motor speed can also be generated by processing the back emf of the motor electronically. In principle, the PID controller or phase-lead and phase-lag controllers can all, with varying degrees of effectiveness, be applied as minor-loop feedback controllers. Under certain conditions, minor-loop control can yield systems that are more robust, that is, less sensitive to external disturbance or internal parameter variations.

11-9-1 Rate-Feedback or Tachometer-Feedback Control

The principle of using the derivative of the actuating signal to improve the damping of a closed-loop system can be applied to the output signal to achieve a similar effect. In other words, the derivative of the input signal is fed back and added algebraically to the actuating signal of the system. In practice, if the output variable is mechanical displacement, a tachometer may be used to convert mechanical displacement into an electrical signal that is proportional to the derivative of the displacement. Figure 11-75 shows the block diagram of a control system with a secondary path that feeds back the derivative of the output. The transfer function of the tachometer is denoted by $K_t s$, where K_t is the tachometer constant, usually expressed in volts per radian per second for analytical purposes. Commercially, K_t is given in the data sheet of the tachometer, typically in volts per 1000 rpm. The effects of rate or tachometer feedback can be illustrated by applying it to a second-order prototype system. Consider that the controlled process of the system shown in Fig. 11-75 has the transfer function

$$G_p(s) = \frac{\omega_n^2}{s(s+2\zeta\omega_n)} \qquad (11\text{-}217)$$

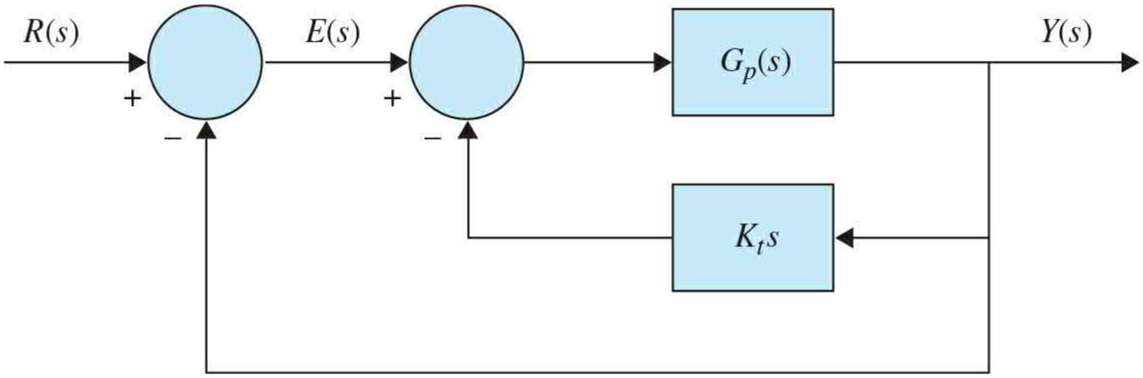


Figure 11-75 Control system with tachometer feedback.

The closed-loop transfer function of the system is

$$\frac{Y(s)}{R(s)} = \frac{\omega_n^2}{s^2 + \left(2\zeta\omega_n + K_t\omega_n^2\right)s + \omega_n^2} \qquad (11\text{-}218)$$

and the characteristic equation is

$$s^2 + \left(2\zeta\omega_n + K_t\omega_n^2\right)s + \omega_n^2 = 0 \qquad (11\text{-}219)$$

From the characteristic equation, *it is apparent that the effect of the tachometer feedback is the increase of the damping of the system, since K_t appears in the same term as the damping ratio ζ.*

In this respect, tachometer-feedback control has exactly the same effect as the PD control. However, the closed-loop transfer function of the system with PD control in Fig. 11-3 is

$$\frac{Y(s)}{R(s)} = \frac{\omega_n^2(K_P + K_D s)}{s^2 + \left(2\zeta\omega + K_D\omega_n^2\right)s + \omega_n^2 K_P} \qquad (11\text{-}220)$$

Comparing the two transfer functions in Eqs. (11-218) and (11-220), we see that the two characteristic equations are identical if $K_P = 1$ and $K_D = K_t$. However, Eq. (11-220) has a zero at $s = -K_P/K_D$, whereas Eq. (11-208) does not. Thus, the response of the system with tachometer feedback is uniquely defined by the characteristic equation, whereas the response of the system with the PD control also depends on the zero at $s = -K_P/K_D$, which could have a significant effect on the overshoot of the step response.

With reference to the steady-state analysis, the forward-path transfer function of the system with tachometer feedback is

$$\frac{Y(s)}{E(s)} = \frac{\omega_n^2}{s\left(s + 2\zeta\omega_n + K_t\omega_n^2\right)} \qquad (11\text{-}221)$$

Because the system is still type 1, the basic characteristics of the steady-state error are not altered by the tachometer feedback; that is, when the input is a step function, the steady-state error is zero. For a unit-ramp function input, the steady-state error of the system is $(2\zeta + K_t\omega_n)/\omega_n$, whereas that of the system with PD control in Fig. 11-3 is $2\zeta/\omega_n$. Thus, *for a type 1 system, tachometer feedback decreases the ramp-error constant K_v* but does not affect the step-error constant K_P.

11-9-2 Minor-Loop Feedback Control with Active Filter

Instead of using a tachometer, an active filter with *RC* elements and op-amps can be used to reduce cost and save space in the minor feedback loop for compensation. We illustrate this approach with the following example.

EXAMPLE 11-9-1 Consider that, for the second-order sun-seeker system in Example 11-5-4, instead of using a series controller in the forward path, we adopt the minor-loop feedback control, as shown in Fig. 11-76*a*, with

$$G_p(s)=\frac{2500}{s(s+25)} \tag{11-222}$$

and

$$H(s)=\frac{K_t s}{1+Ts} \tag{11-223}$$

To maintain the system as type 1, it is necessary that *H*(*s*) contain a zero at *s* = 0. Equation (11-223) can be realized by the op-amp circuit shown in Fig. 11-76*b*. This circuit cannot be applied as a series controller in the forward path because it acts as an open circuit in the steady state when the frequency is zero. As a minor-loop controller, the zero-transmission property to dc signals does not pose any problems.

The forward-path transfer function of the system in Fig. 11-76*a* is

$$\begin{aligned}\frac{\Theta_o(s)}{\Theta_e(s)}=G(s)&=\frac{G_p(s)}{1+G_p(s)H(s)}\\&=\frac{2500(1+Ts)}{s[(s+25)(1+Ts)+2500K_t]}\end{aligned} \tag{11-224}$$

The characteristic equation of the system is

$$Ts^3+(25T+1)s^2+(25+2500T+2500K_t)s+2500=0 \tag{11-225}$$

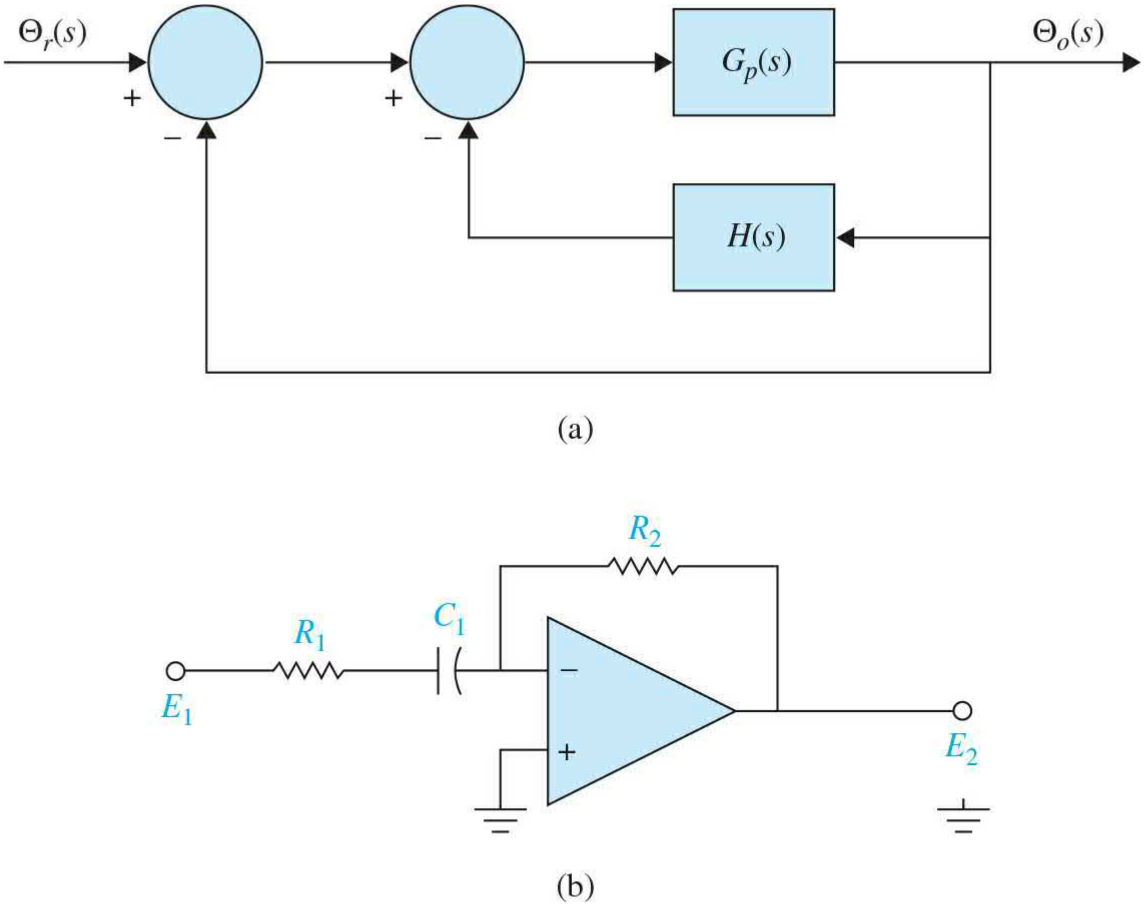


Figure 11-76 (*a*) Sun-seeker control system with minor-loop control. (*b*) Op-amp circuit realization of $\frac{K_t s}{1+Ts}$.

To show the effects of the parameters K_t and T, we construct the root contours of Eq. (11-225) by first considering that K_t is fixed and T is variable. Dividing both sides of Eq. (11-225) by the terms that do not contain T, we get

$$1+\frac{Ts(s^2+25s+2500)}{s^2+(25+2500K_t)s+5000}=0 \qquad (11\text{-}226)$$

When the value of K_t is relatively large, the two poles of the last equation are real with one very close to the origin. It is more effective to choose K_t so that the poles of Eq. (11-226) are complex.

Figure 11-77 shows the root contours of Eq. (11-225) with K_t = 0.02, and T varies from 0 to ∞.

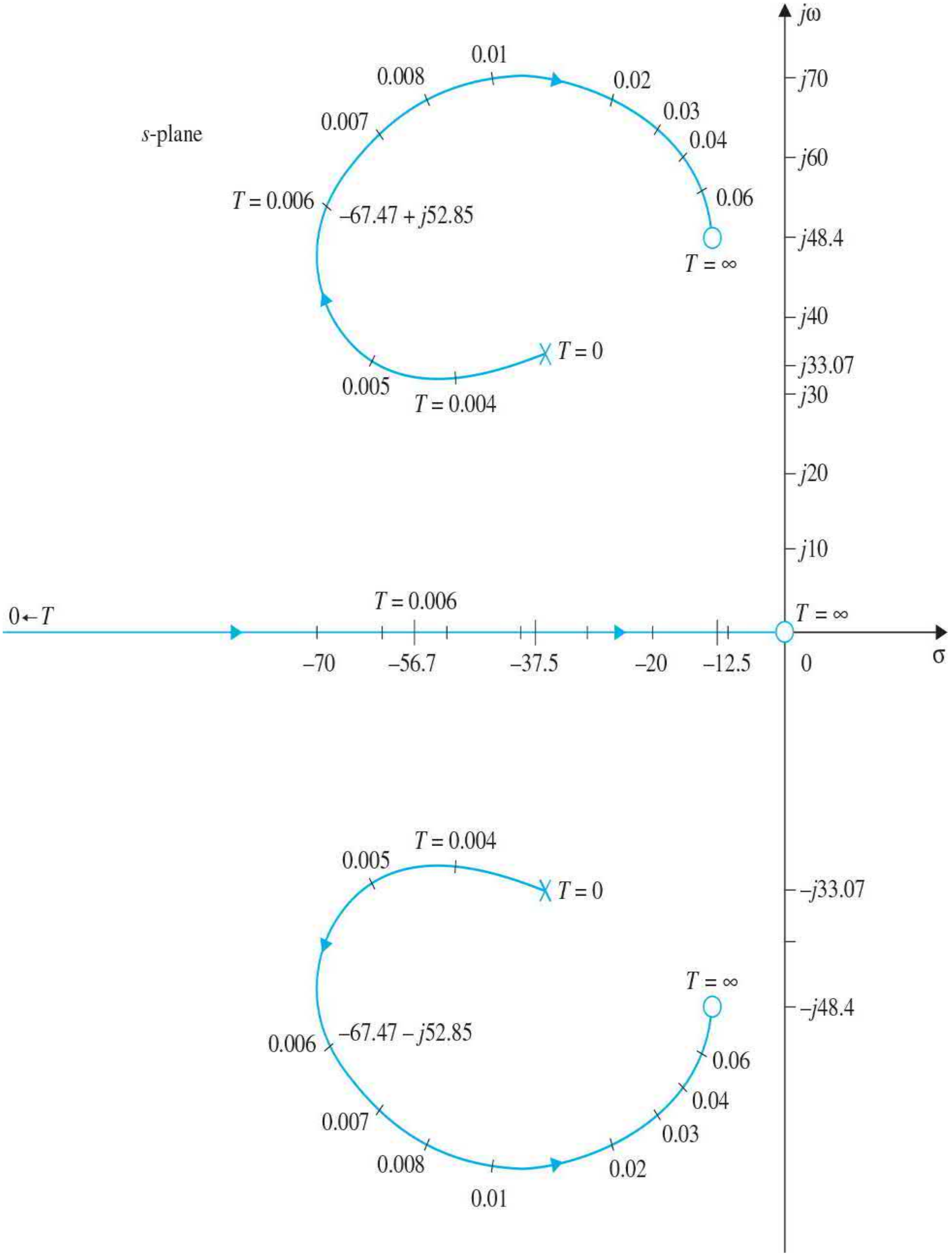

jω
s-plane
0.01
0.008
0.02
0.03
0.007
0.04
0.06
T = 0.006
−67.47 + j52.85
T = ∞
T = 0
0.005
T = 0.004
j70
j60
j48.4
j40
j33.07
j30
j20
j10
0 ← T
T = 0.006
T = ∞
−70
−56.7
−37.5
−20
−12.5
0
σ
T = 0.004
0.005
T = 0
−j33.07
T = ∞
−j48.4
0.006
−67.47 − j52.85
0.06
0.04
0.007
0.03
0.008
0.02
0.01

Figure 11-77 Root contours of $Ts^3 + (25T + 1)s^2 + (25 + 2500K_t + 2500T)s + 2500 = 0$, $K_t = 0.02$.

When $T = 0.006$, the characteristic equation roots are at –56.72, $-67.47 + j52.85$, and $-67.47 - j52.85$. The attributes of the unit-step response are as follows:

Maximum overshoot = 0

$t_r = 0.04485$ s

$t_s = 0.06061$ s

$t_{max} = 0.4$ s

The ramp-error constant of the system is

$$K_v = \lim_{s \to 0} sG(s) = \frac{100}{1 + 100K_t} \tag{11-227}$$

Thus, just as with tachometer feedback, the minor-loop feedback controller of Eq. (11-223) reduces the ramp-error constant K_v, although the system is still type 1.

11-10 MATLAB TOOLS AND CASE STUDIES

In this section, we will go through the steps involved in finding some of the results and displaying many of the graphics from the examples in this chapter using the MATLAB **SISO** design tool. Through the graphical-user interface (GUI) approach, SISO creates a user-friendly environment to reduce the complexity of control systems design. The SISO design tool allows you to design a single-input/single-output (SISO) compensator using root locus, Bode diagram, and Nichols and Nyquist techniques.[4] By default, the SISO design tool assumes the feedback block diagram representation shown in Fig. 11-78 with feedback gain H, a compensator C placed in series with the plant G, and the prefilter F.

We illustrate the usage of the SISO tool through the following examples.

EXAMPLE 11-10-1 Recall from Example 11-2-1 the forward-path transfer function for the altitude-control system was

$$G(s)=\frac{4500K}{s(s+361.2)} \tag{11-228}$$

The design constraints for this problem were as follows:

Steady-state error due to unit-ramp input = 0.000443

Maximum overshoot ≤ 5%

Rise time $t_r \leq$ s

Settling time $t_s \leq$ s

Time-Domain Design

Proportional controller: Let us first design a proportional controller using the SISO design tool. To satisfy the maximum value of the specified steady-state error requirement, *K* should be set at 181.17. To examine the performance of the proportional controller, we need to find the system root locus. To implement the root-locus design approach we utilize MATLAB Toolbox 11-10-1 to enter the plant *G* and controller *C* transfer functions in Fig. 11-78. Note that by default the value of feedback gain *H* and prefilter *F* are automatically set to unity.

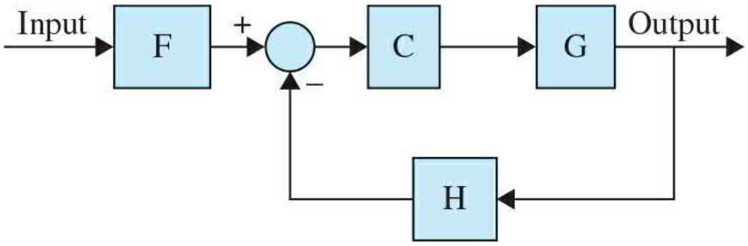
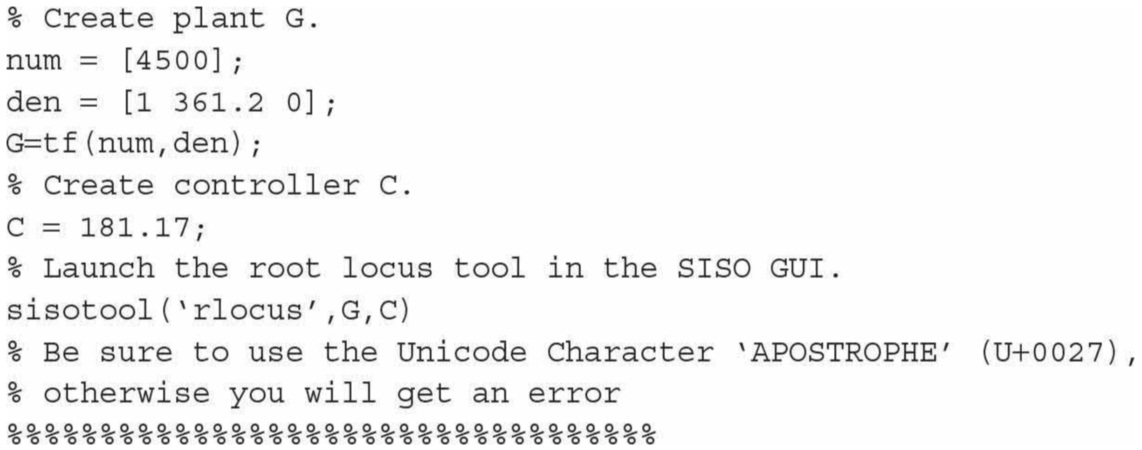


Figure 11-78 Default control configuration used in MATLAB design tool.

Toolbox 11-10-1

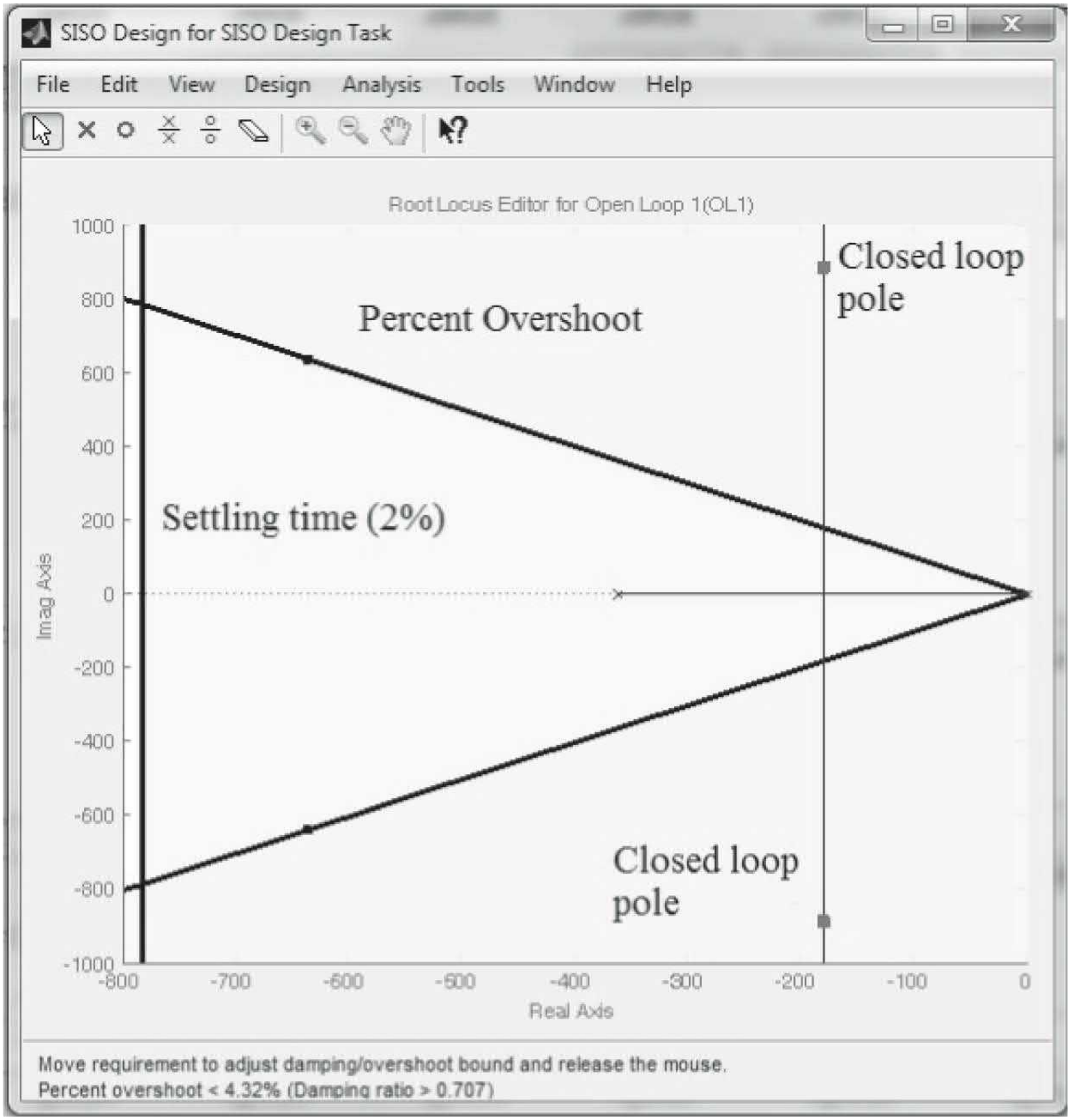
The root-locus design tool in SISO for Example 11-10-1 is activated by the following sequence of MATLAB functions:



```
% Create plant G.
num = [4500];
den = [1 361.2 0];
G=tf(num,den);
% Create controller C.
C = 181.17;
% Launch the root locus tool in the SISO GUI.
sisotool('rlocus',G,C)
% Be sure to use the Unicode Character 'APOSTROPHE' (U+0027),
% otherwise you will get an error
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 11-79 shows the root locus of the system where the closed-loop system poles for K = 181.17 are located at $s_{1,2}$ = –181 ± J885 (MATLAB has rounded this value off). To view the closed loop systems poles and zeros, go to the View menu and select Closed-Loop Poles, as shown in Fig. 11-80. Recall the poles of the closed-loop system are

$$s_{1,2} = -180.6 \pm \sqrt{32616 - 4500K} \tag{11-229}$$



Move requirement to adjust damping/overshoot bound and release the mouse.
Percent overshoot < 4.32% (Damping ratio > 0.707)

Figure 11-79 Root locus of $G(s)=\frac{4500K}{s(s+361.2)}$ for Example 11-10-1, using the MATLAB SISO design tool, incorporating the percent overshoot and the settling time as design constraints.

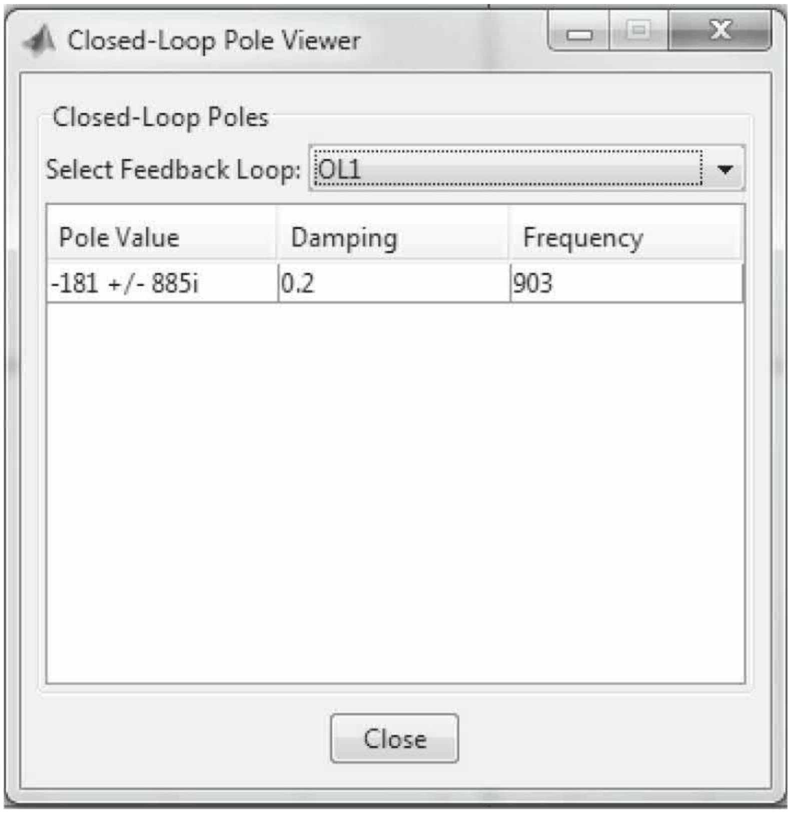
Incorporation of the design criteria: As a first step to design a controller, we use the built-in design criteria option within the SISO design tool to establish the desired pole regions on the root locus. Next, to select the Design

Requirements option, right click on the Root Locus plot each time you want to add a new design requirement. To do this, select “New” to enter one of the following:

- Settling time
- Percent overshoot
- Damping ratio
- Natural frequency

In this case, we have included settling time and percent overshoot as design constrains. Enter the two constraints as they are listed at the beginning of this example, one at a time. In order also to enter the rise time as a constraint, the user must first establish a relation between the damping ratio and the natural frequency using an equation for rise time. Recall that the approximate equations for rise time for a second-order prototype system were provided in Chap. 7. Because the settling time and the percent overshoot are more important criteria in this example, we will use them as primary constraints. After designing a controller based on these constraints, we will determine whether the system complies with the rise-time constraint.

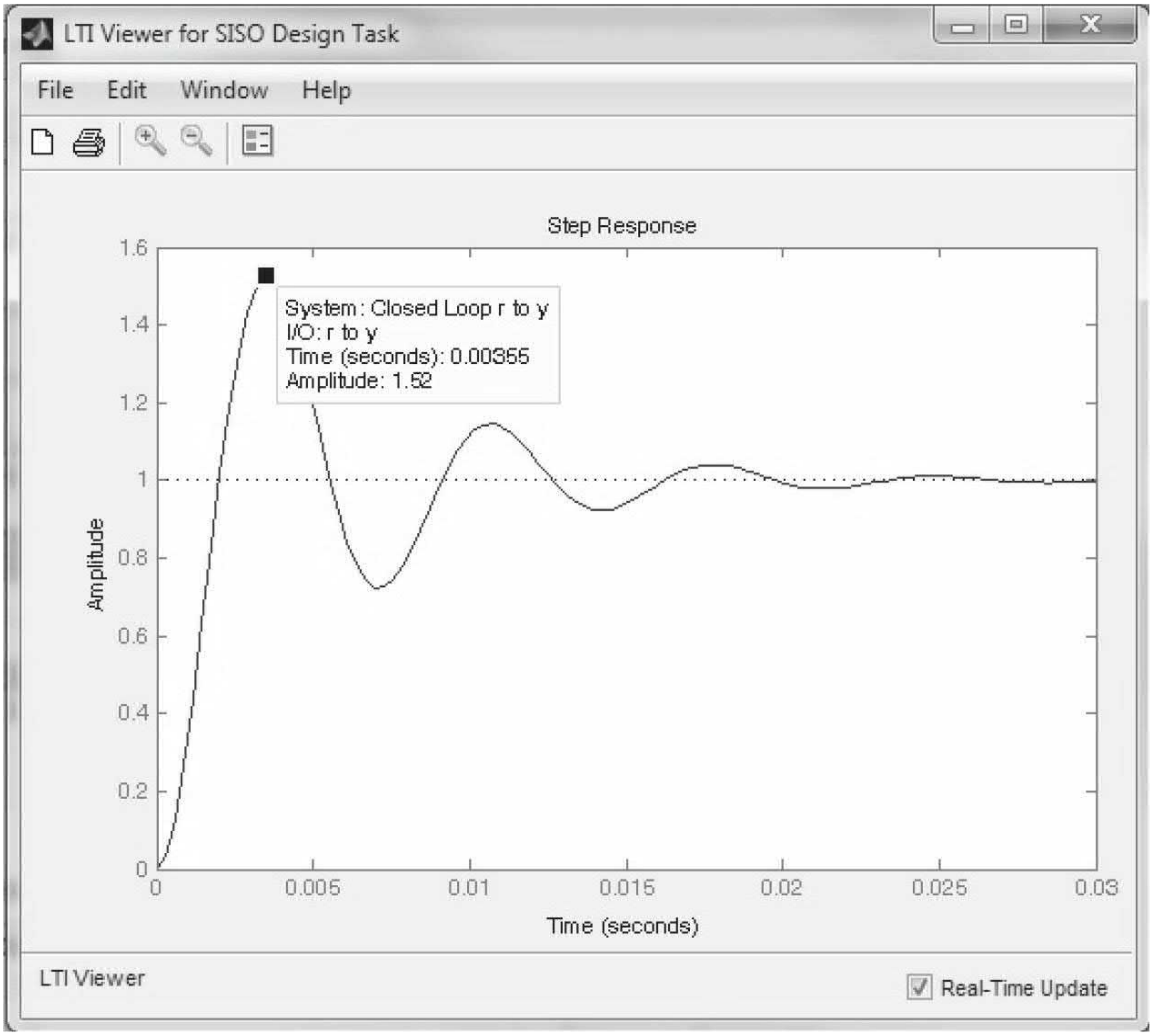
Figure 11-79 shows the closed-loop system pole for $K = 181.17$ are not in the desired area. As shown, with this value of K, the damping ratio of the system is 0.2, and the maximum overshoot is 52.7 percent, as shown by the unit-step response in Fig. 11-81. In order to achieve the desired response, the desired poles of the system must lie within the percent overshoot wedge and to the left of the boundary imposed by the settling time close to the –800 markers in Fig. 11-79. Obviously, it is impossible to use the proportional controller (for any value of K) to move the poles of the closed-loop system farther to the left-half plane.



Select Feedback Loop: OL1

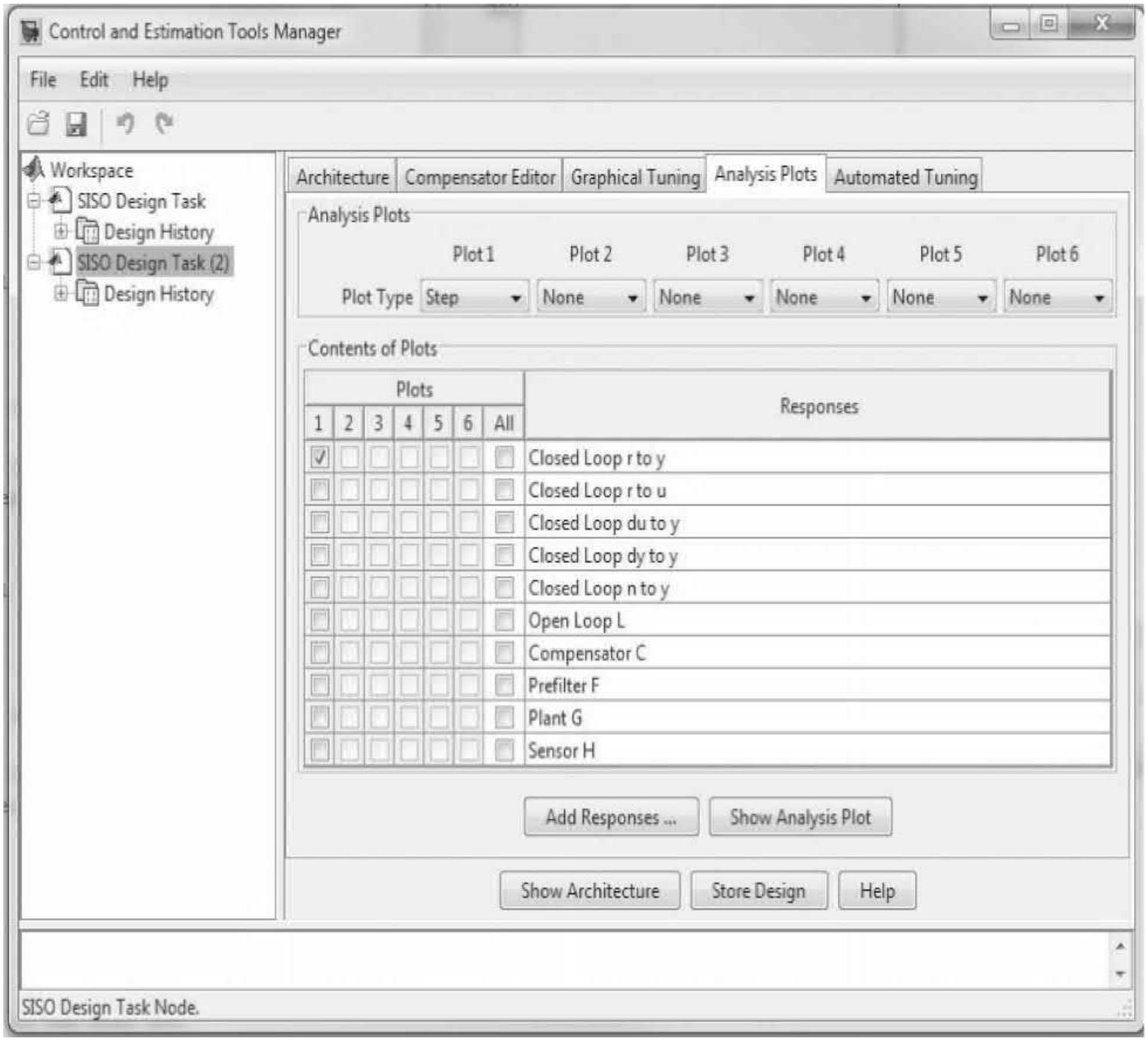
Pole Value	Damping	Frequency
-181 +/- 885i	0.2	903

Figure 11-80 Closed loop poles of $G(s)=\frac{4500K}{s(s+361.2)}$ with unity feedback, for Example 11-10-1.



System: Closed Loop r to y
I/O: r to y
Time (seconds): 0.00355
Amplitude: 1.52

Figure 11-81 Unit-step response of the system from Example 11-10-1.



	Plot 1	Plot 2	Plot 3	Plot 4	Plot 5	Plot 6
Plot Type	Step	None	None	None	None	None

Plots 1	2	3	4	5	6	All	Responses
☑	□	□	□	□	□	□	Closed Loop r to y
□	□	□	□	□	□	□	Closed Loop r to u
□	□	□	□	□	□	□	Closed Loop du to y
□	□	□	□	□	□	□	Closed Loop dy to y
□	□	□	□	□	□	□	Closed Loop n to y
□	□	□	□	□	□	□	Open Loop L
□	□	□	□	□	□	□	Compensator C
□	□	□	□	□	□	□	Prefilter F
□	□	□	□	□	□	□	Plant G
□	□	□	□	□	□	□	Sensor H

Figure 11-82 The Control and Estimation Tools Manager in the MATLAB SISO design tool for selecting the closed-loop system time response.

To see the closed-loop system time response to a unit-step input, select the Response to Step Command option from the Analysis menu, which is located at the top of the screen shown in Fig. 11-79. To get this plot, you must select the "Closed Loop r to y" response in the Analysis Plot menu in the Control and Estimation Tools Manager window as shown in Fig. 11-82.

PD controller: With the PD controller of Eq. (11-228) and $K = 181.17$, the forward-path transfer function of the system becomes

$$G(s)=\frac{815{,}265(K_P+K_D s)}{s(s+361.2)} \tag{11-230}$$

The closed-loop transfer function is

$$\frac{\Theta_y(s)}{\Theta_r(s)}=\frac{815{,}265K_D\left(s+\dfrac{K_P}{K_D}\right)}{s^2+(361.2+815{,}265K_D)s+815{,}265K_P} \tag{11-231}$$

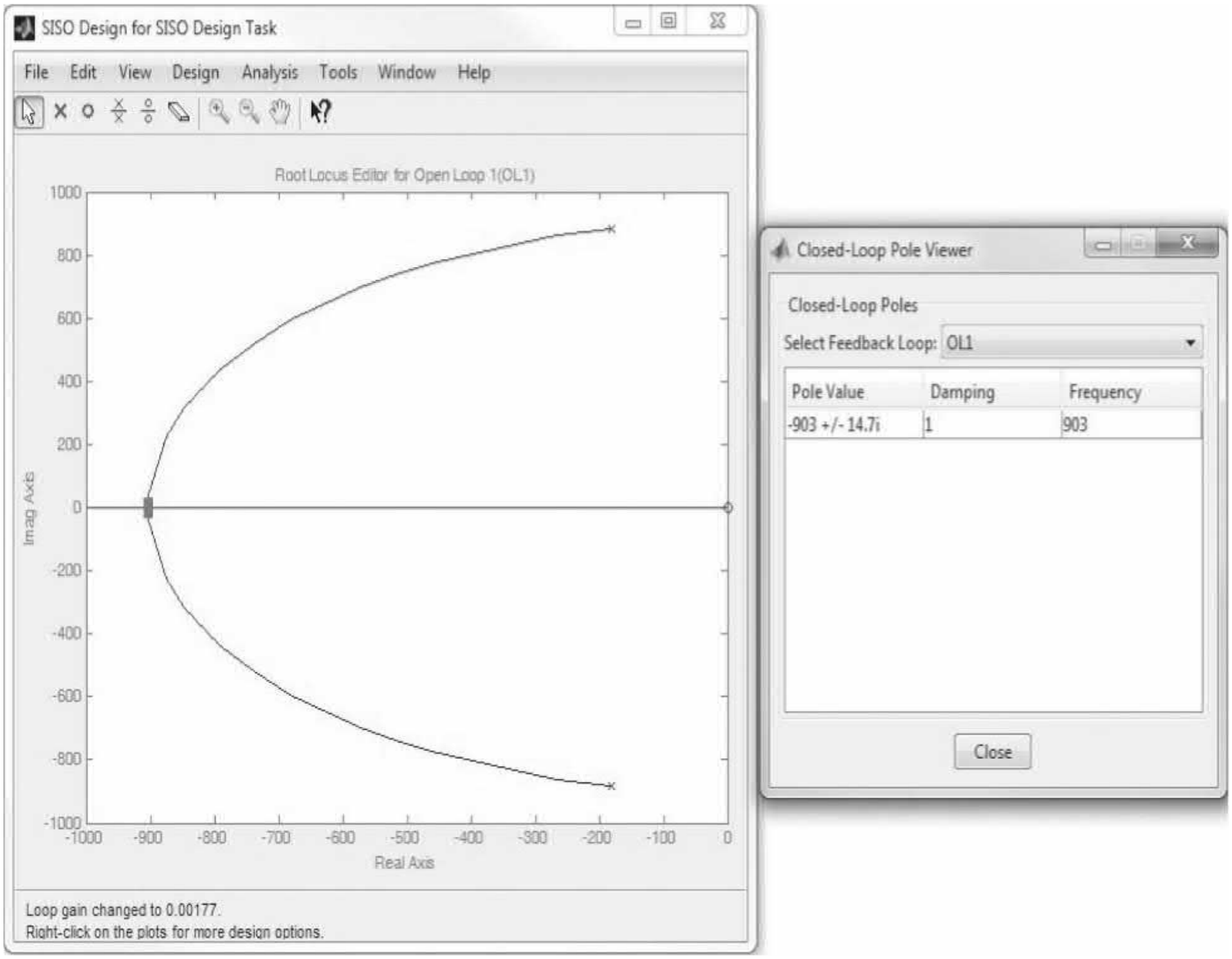
The characteristic equation is written as

$$s^2+(361.2+815{,}265K_D)s+815{,}265K_P=0 \tag{11-232}$$

As discussed in Example 11-2-1, we can apply the **root-contour method** to the characteristic equation in Eq. (11-232) to examine the effect of varying K_P and K_D by finding the root locus of G_{eq} (s) in the following equation:

$$1+G_{\text{eq}}(s)=1+\frac{815{,}265K_D s}{s^2+361.2s+815{,}265K_P}=0 \tag{11-233}$$

where the root contours are plotted by fixing the K_P value and varying $K_{D.}$ Hence, using Toolbox 11-10-2 we get the root contours of Eq. (11-232), as shown in Fig. 11-83 for K_P = 1. As discussed in Example 11-2-1, when K_D is set to 0.00177, the roots are real and equal at –903, and the damping is critical.



Select Feedback Loop: OL1

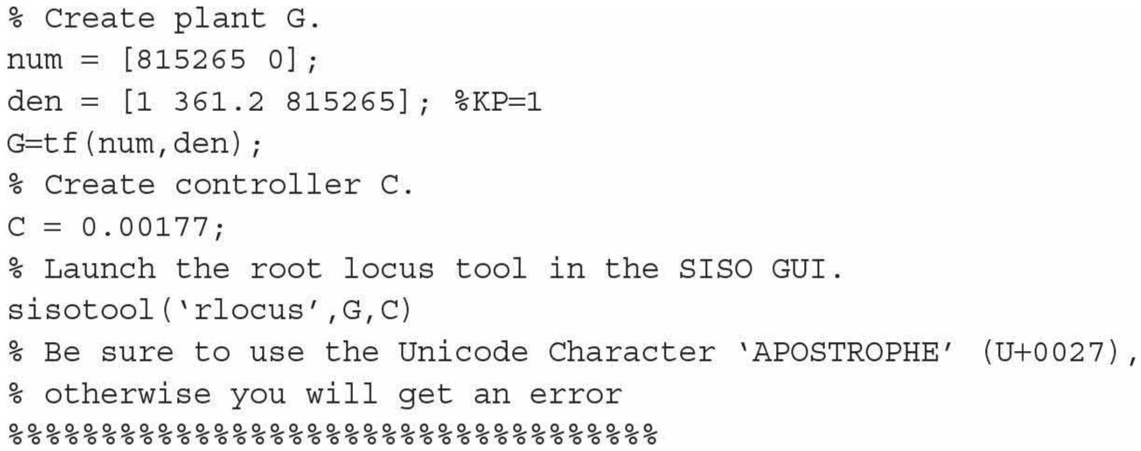
Pole Value	Damping	Frequency
-903 +/- 14.7i	1	903

Loop gain changed to 0.00177.
Right-click on the plots for more design options.

Figure 11-83 Root contours of PD-controlled system from Example 11-10-1.

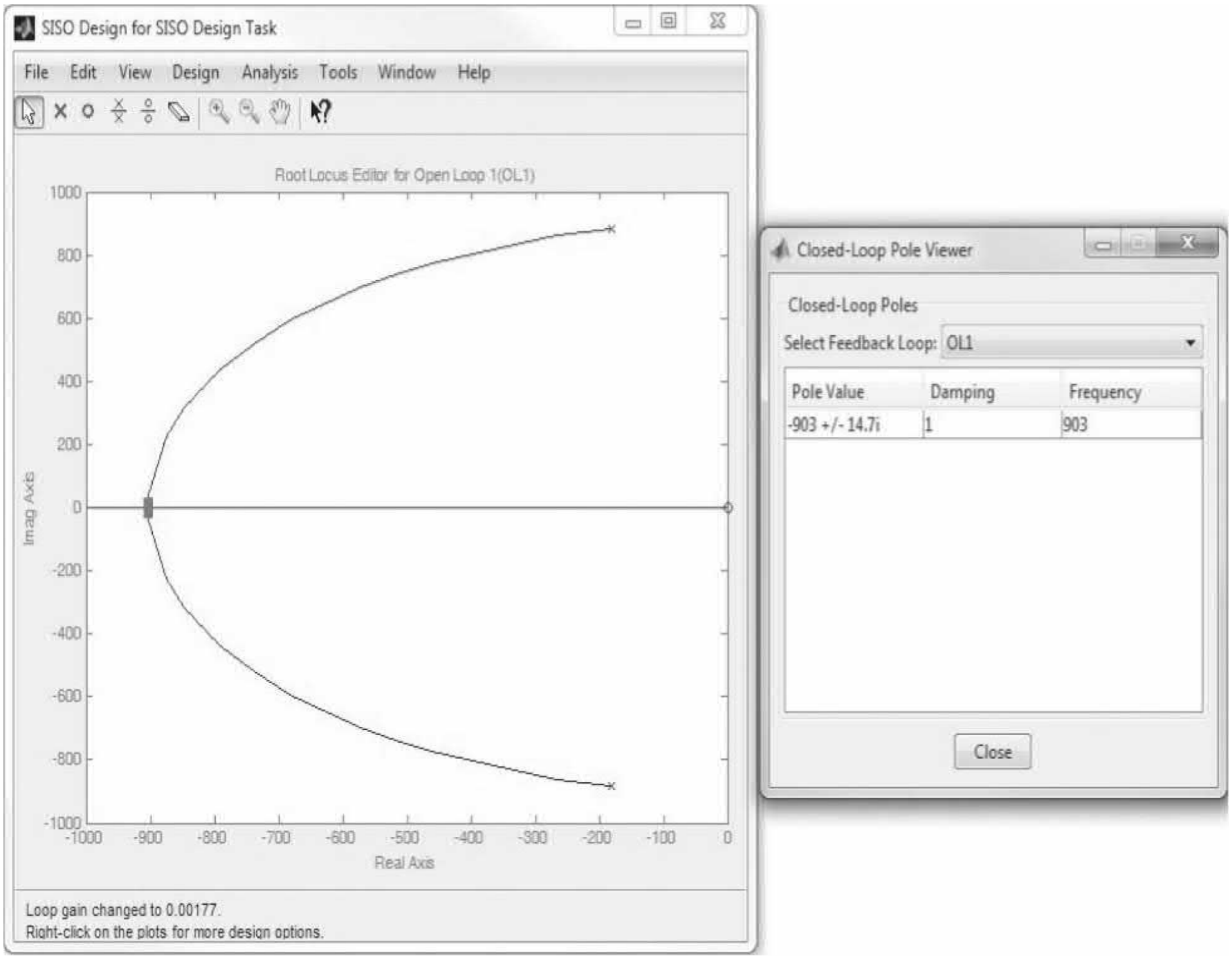
Toolbox 11-10-2

Root contours of Eq. (11-232) shown in Fig. 11-83 are obtained by the following sequence of MATLAB functions:



```
% Create plant G.
num = [815265 0];
den = [1 361.2 815265]; %KP=1
G=tf(num,den);
% Create controller C.
C = 0.00177;
% Launch the root locus tool in the SISO GUI.
sisotool(‘rlocus’,G,C)
% Be sure to use the Unicode Character ‘APOSTROPHE’ (U+0027),
% otherwise you will get an error
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Next we obtain the root locus of the PD controller by adding a zero to controller *C* in the Compensator Editor menu within the Control and Estimation Tools Manager window, as shown in Fig. 11-84*a*. To enter a zero to the controller, right-click the mouse inside the “Dynamics” area; select add Pole/Zero, followed by Real Zero option. Click the new “Real Zero” row inside the “Dynamics” box to enter the new zero value of the PD controller. Recall from Example 11-2-1 that for the root-contour approach the zero value for $K_P = 1$ and $K_D = 0.00177$ corresponds to a zero at $s = –K_P/K_D = –565$. Figure 11-84*b* shows the Compensator Editor window after the PD controller has been added.

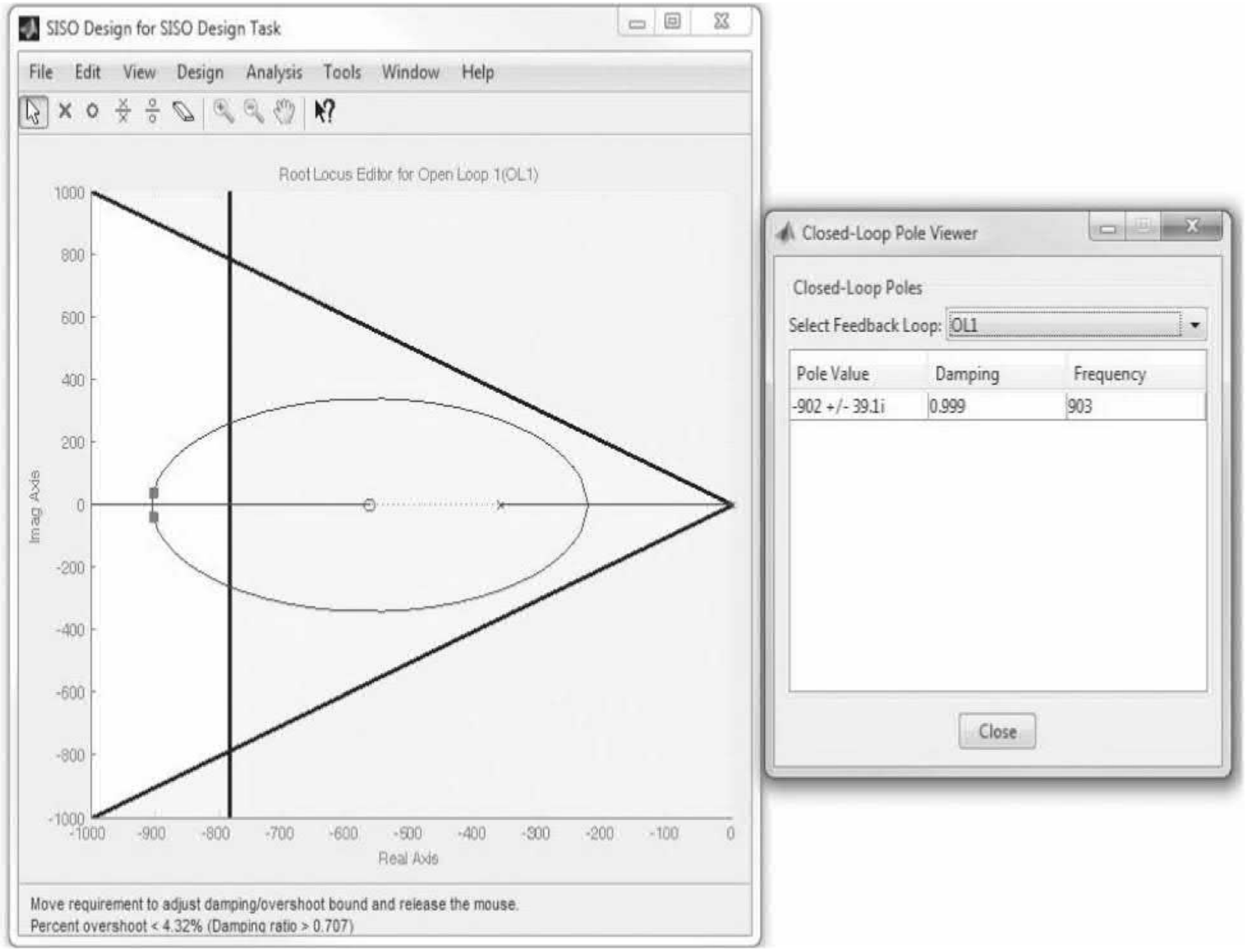


Loop gain changed to 0.00177.
Right-click on the plots for more design options.

Pole Value	Damping	Frequency
-903 +/- 14.7i	1	903

Figure 11-84 Addition of a zero to the controller *C* to create a PD controller. (*a*) Addition process. (*b*) After addition of a zero at $s = -K_P/K_D = -565$.

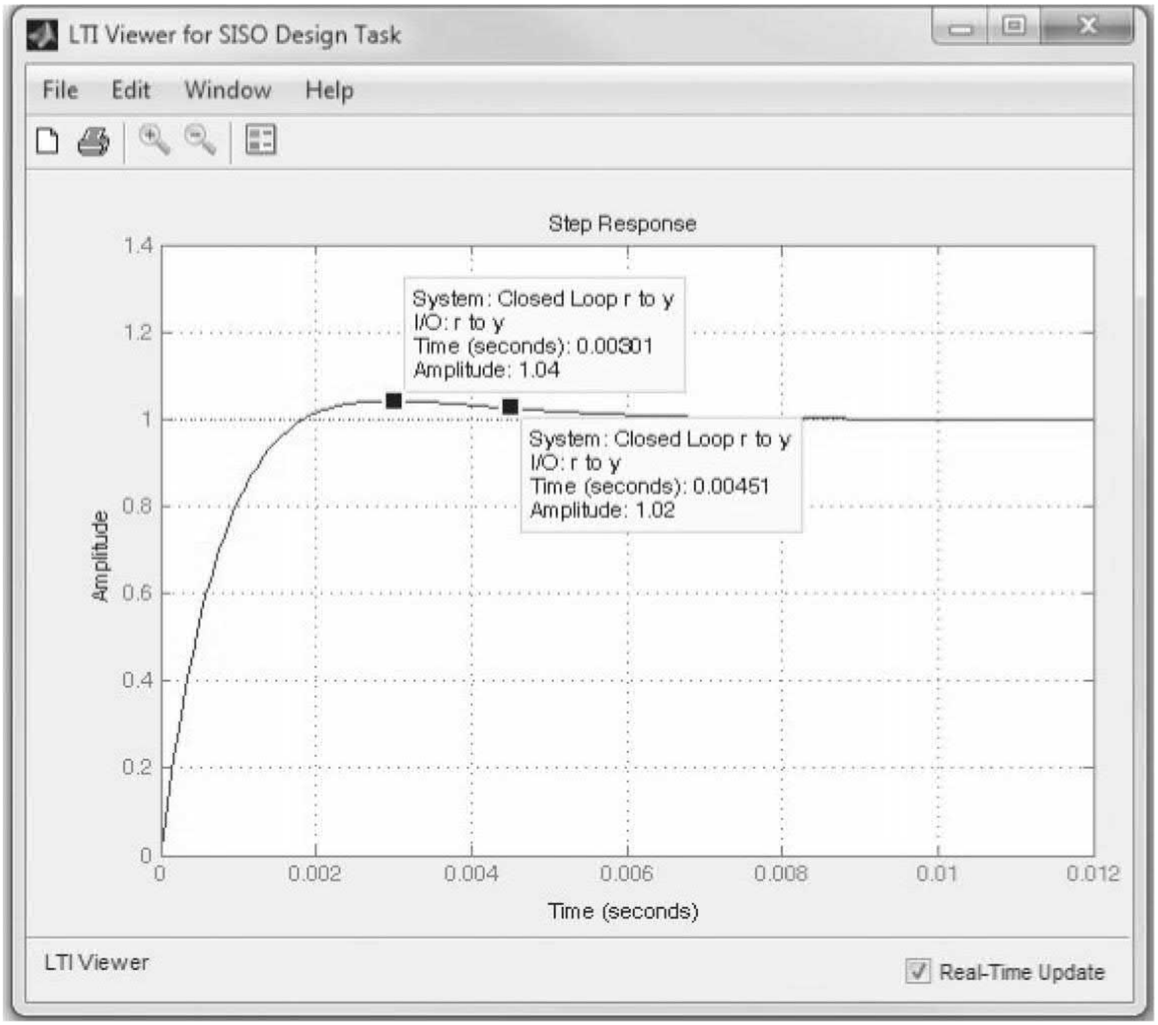
The new root-locus diagram along with the closed loop pole values for the controller gain of 181.17 appear in Fig. 11-85—you can view the poles' current locations from the View menu. Note there is a slight discrepancy between the pole values in Figs. 11-83 and 11-85 because of MATLAB rounding off process, which is not a concern. The step response of the controlled system in Fig. 11-86 shows the system has now complied with all design criteria. The 2% settling time is now 0.0451 s, while the overshoot is 4%. It is interesting that, although the closed-loop poles are both real, the system has a non-oscillatory response with an overshoot because of the dominant effect of the PD controller zero on the response. These results perfectly match with Example 11-2-1.



Move requirement to adjust damping/overshoot bound and release the mouse.
Percent overshoot < 4.32% (Damping ratio > 0.707)

Pole Value	Damping	Frequency
-902 +/- 39.1i	0.999	903

Figure 11-85 The root-locus diagram for Example 11-10-1 after incorporating a zero in the PD controller at *s* = 565 = –1/0.00177 and using a gain of 181.17.



System: Closed Loop r to y
I/O: r to y
Time (seconds): 0.00301
Amplitude: 1.04

System: Closed Loop r to y
I/O: r to y
Time (seconds): 0.00451
Amplitude: 1.02

Figure 11-86 Step response of the system in Example 11-10-1 with controller $C(s) = 181.17(s + 1/0.00177)$.

In addition to these findings, recall in Example 11-2-1 that the motor torque limit is the most important constraint in any design. In practice, always verify that the actuator used has enough torque or load to create such response. In fact, as a safety factor, it is best to operate the motor at 50 percent of its maximum torque value. To check this, you need to obtain the torque transfer function of the system. From Fig. 7-52, by applying the SFG gain formula, the closed loop torque transfer function of the system is written as

$$T(s)=\frac{T_m(s)}{\Theta_r(s)}$$

$$=\frac{K_s K_1 K_i (K_P + K_D s)s}{(R_a J_t + K_1 K_2 J_t)s^2 + (R_a B_t + K_1 K_2 B_t + K_i K_b + K_s K_1 K_i K_D N)s + K_s K_1 K_i K_P N} \qquad (11\text{-}236)$$

Entering the parameter values from Sec. 7-9 and using Toolbox 11-10-3, we obtain the motor torque response. The question here is if the motor is capable of generating almost 14,000 oz·in of torque almost instantly. Or, do we need to change our controller parameters?

If you require fine-tuning your existing design with minimal changes, you can easily move the closed loop poles in Fig. 11-85 to locations just to the left of the settling time constraint, as shown in Fig. 11-88*a*. In that case K_P = 0.85 and K_D = 0.0015, and the poles of the system are at $s_{1,2}$ = –794 ± *j*250. The step response of the system, shown in Fig. 11-88*b*, obviously meets the requirements.

Toolbox 11-10-3

Unit-step response of Eq. (11-236) shown in Fig. 11-87 is obtained by the following sequence of MATLAB functions:

```
num = 10*9*[181.17*0.00177 1 0];
den1=5*0.0002+10*0.5*0.0002;
den2=5*0.015+10*0.5*0.015+9*0.0636+10*9*181.17*0.00177/10;
den3=10*9*181.17/10;
den = [den1 den2 den3];
step(num,den)
```

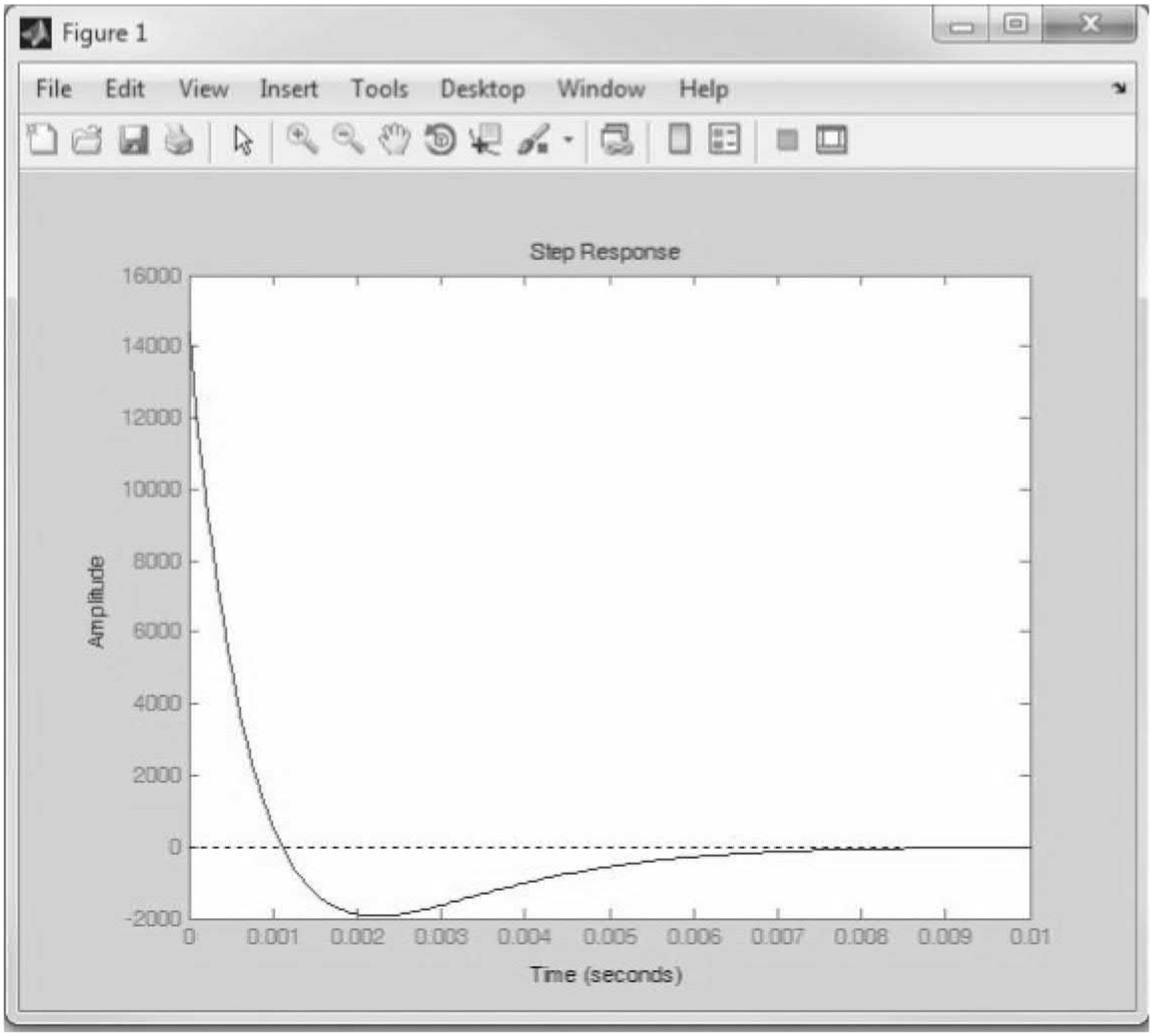
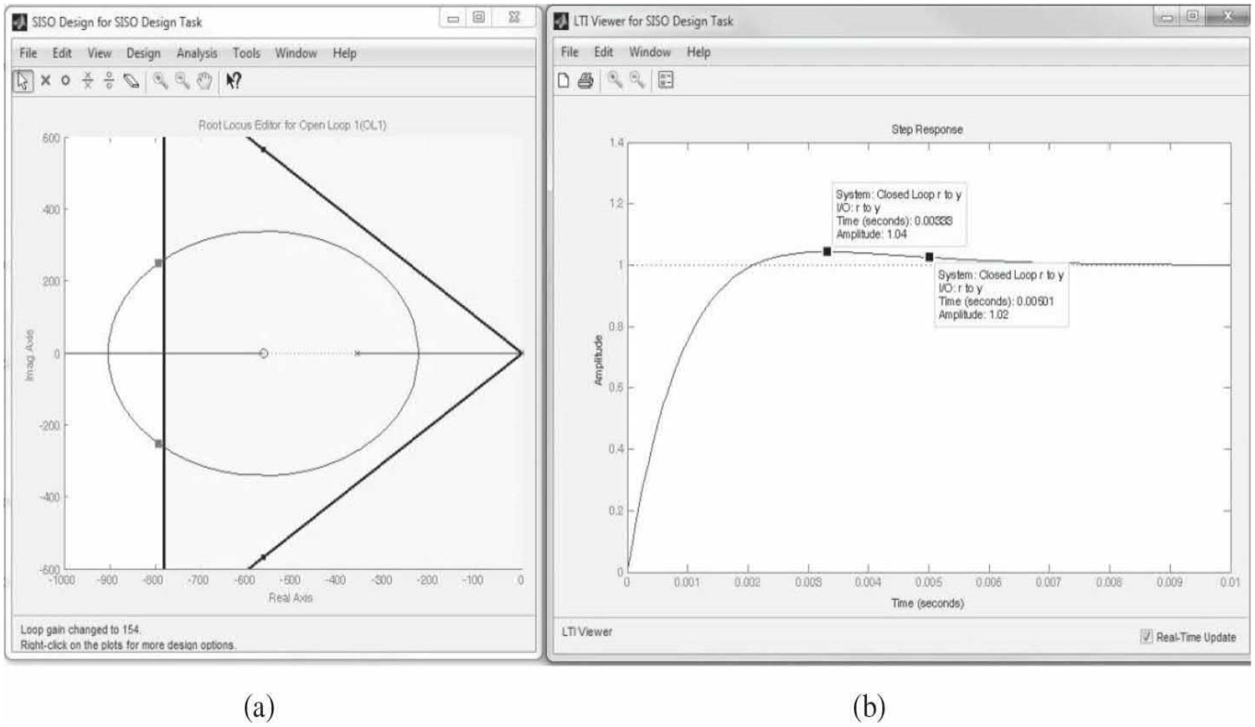


Figure 11-87 Motor torque step response of the system in Example 11-10-1 with a controller $C(s) = 181.17(s + 1/0.00177)$.



Loop gain changed to 154.
Right-click on the plots for more design options.

System: Closed Loop r to y
I/O: r to y
Time (seconds): 0.00333
Amplitude: 1.04

System: Closed Loop r to y
I/O: r to y
Time (seconds): 0.00501
Amplitude: 1.02

Figure 11-88 (*a*) The root-locus diagram. (*b*) Step response of the system in Example 11-10-1 with a PD controller, $C(s) = 154.03(s + 1/0.00177)$.

Frequency-Domain Design Now let us carry out the design of the PD controller in the frequency domain using the following performance criteria:

Steady-state error due to a unit–ramp input ≤ 0.00443

Phase margin ≥ 80°

Resonant peak $M_r \leq 1.05$

BW ≤ 2000 rad/s

Use Toolbox 11-10-4 to activate the MATLAB Bode design tool SISO design. Using this design tool we can arrive at the loop magnitude and phase diagrams for Example 11-10-1 with controller $C(s) = 181.17(s + 1/0.00177)$, as shown in Fig. 11-89.

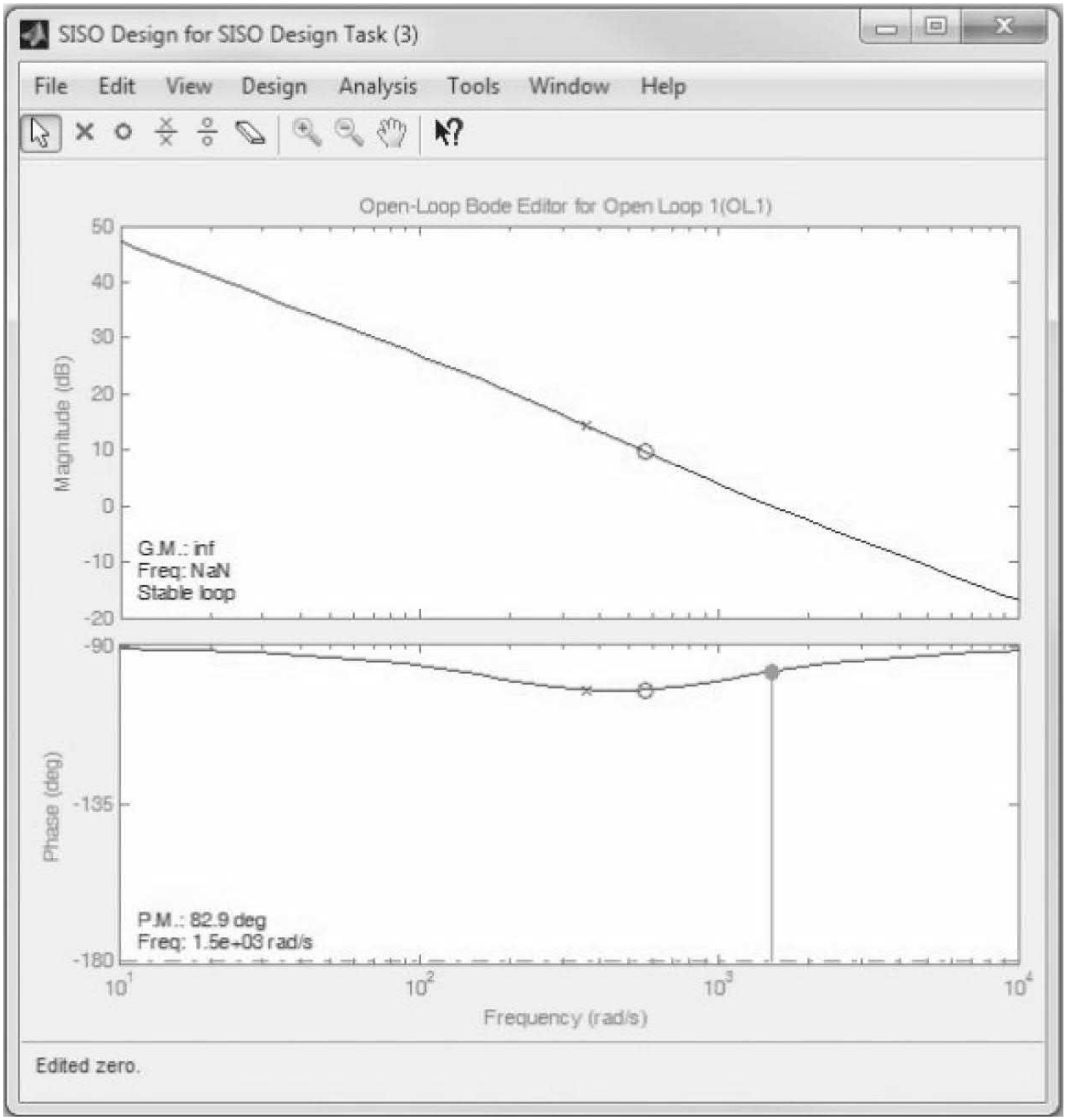
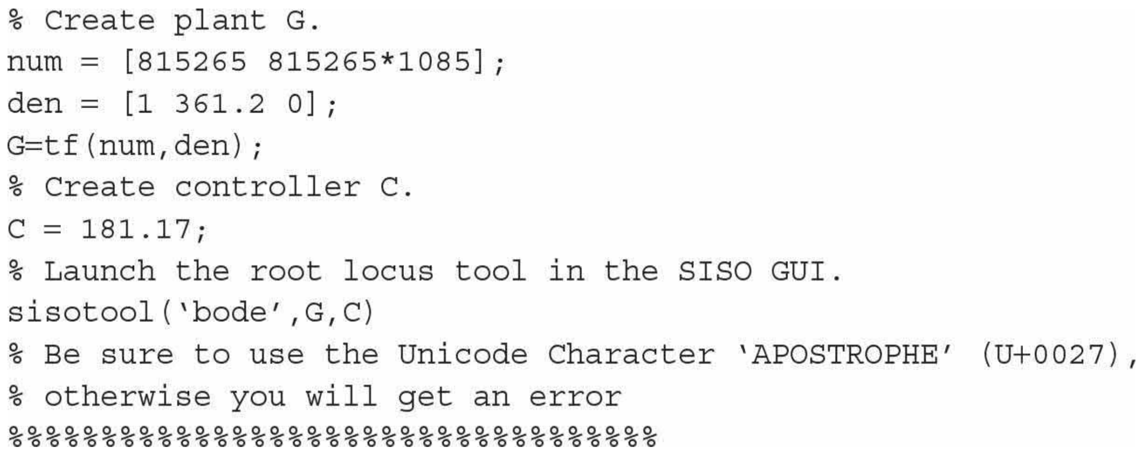


Figure 11-89 The loop magnitude and phase diagrams for Example 11-10-1 with controller $C(s) = 181.17(s + 1/0.00177)$.

Toolbox 11-10-4

The Bode design tool in SISO for Example 11-10-1 is activated by the following sequence of MATLAB functions:



```
% Create plant G.
num = [815265 815265*1085];
den = [1 361.2 0];
G=tf(num,den);
% Create controller C.
C = 181.17;
% Launch the root locus tool in the SISO GUI.
sisotool('bode',G,C)
% Be sure to use the Unicode Character 'APOSTROPHE' (U+0027),
% otherwise you will get an error
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

As in the root-locus design approach, a PD controller with a zero at $s = -K_P/K_D = -565$ was used for this purpose. The gain margin in this case is infinity, while the phase margin is 83°. As a result, the system has also fully complied with all the design criteria in the frequency domain.

EXAMPLE 11-10-2 **Example 11-2-2 Revisited** Use the following toolbox to get the uncompensated root locus and bode plots corresponding to Eq. (11-30). Using the root locus and Bode plots, as shown in Fig. 11-90, you can follow the process discussed in Example 11-10-1 to design your PD controller.

Toolbox 11-10-5

5The SISO design tool for Example 11-10-1 is activated by the following sequence of MATLAB functions:

```
% Create plant G.
num = [1.5e-7];
den = [1 3408.2 1204000 0];
G=tf(num,den);
% Create controller C.
C = 181.17;
% Launch the root locus tool in the SISO GUI.
sisotool(G,C)
% Be sure to use the Unicode Character ‘APOSTROPHE’ (U+0027),
% otherwise you will get an error
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

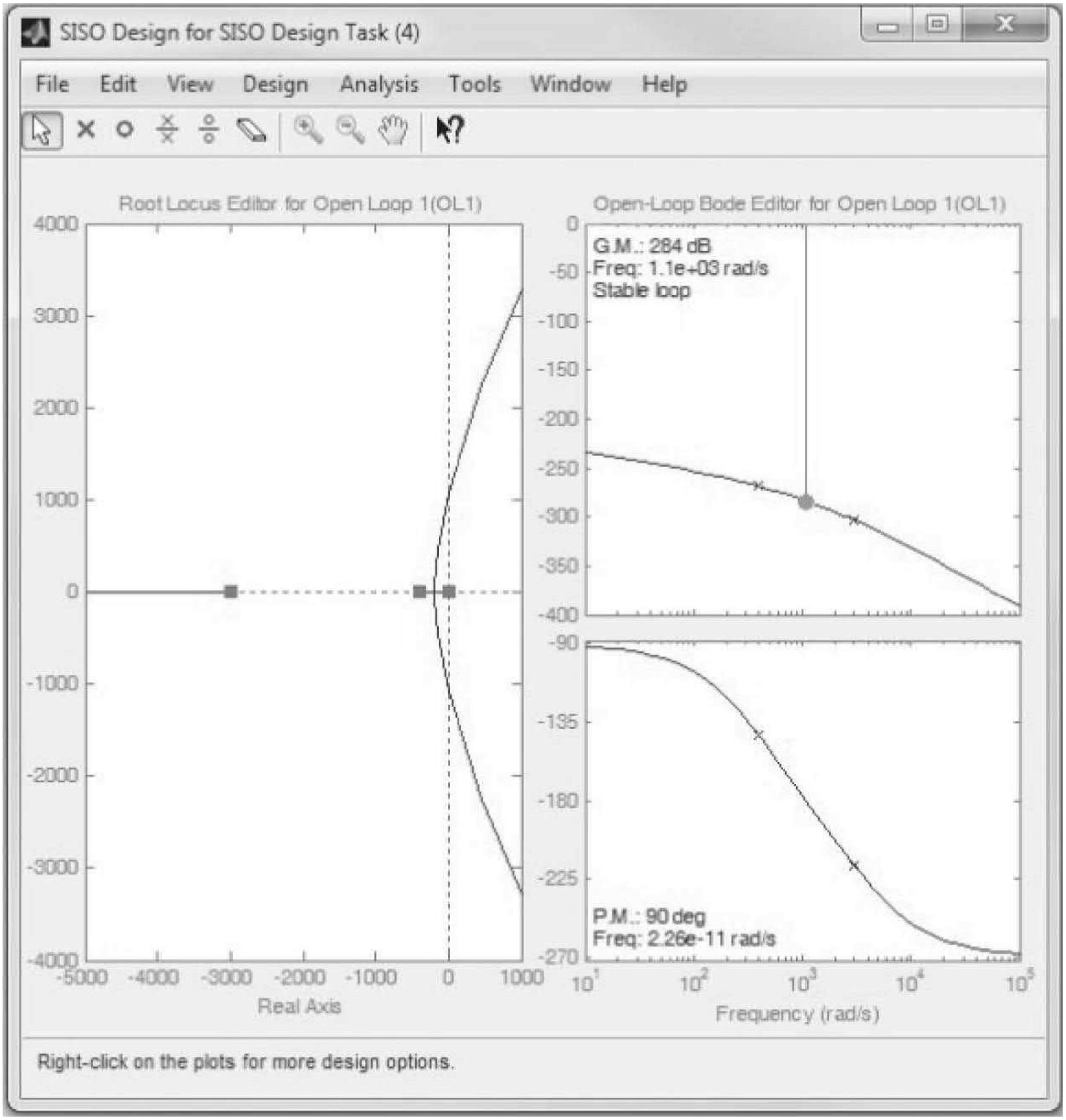
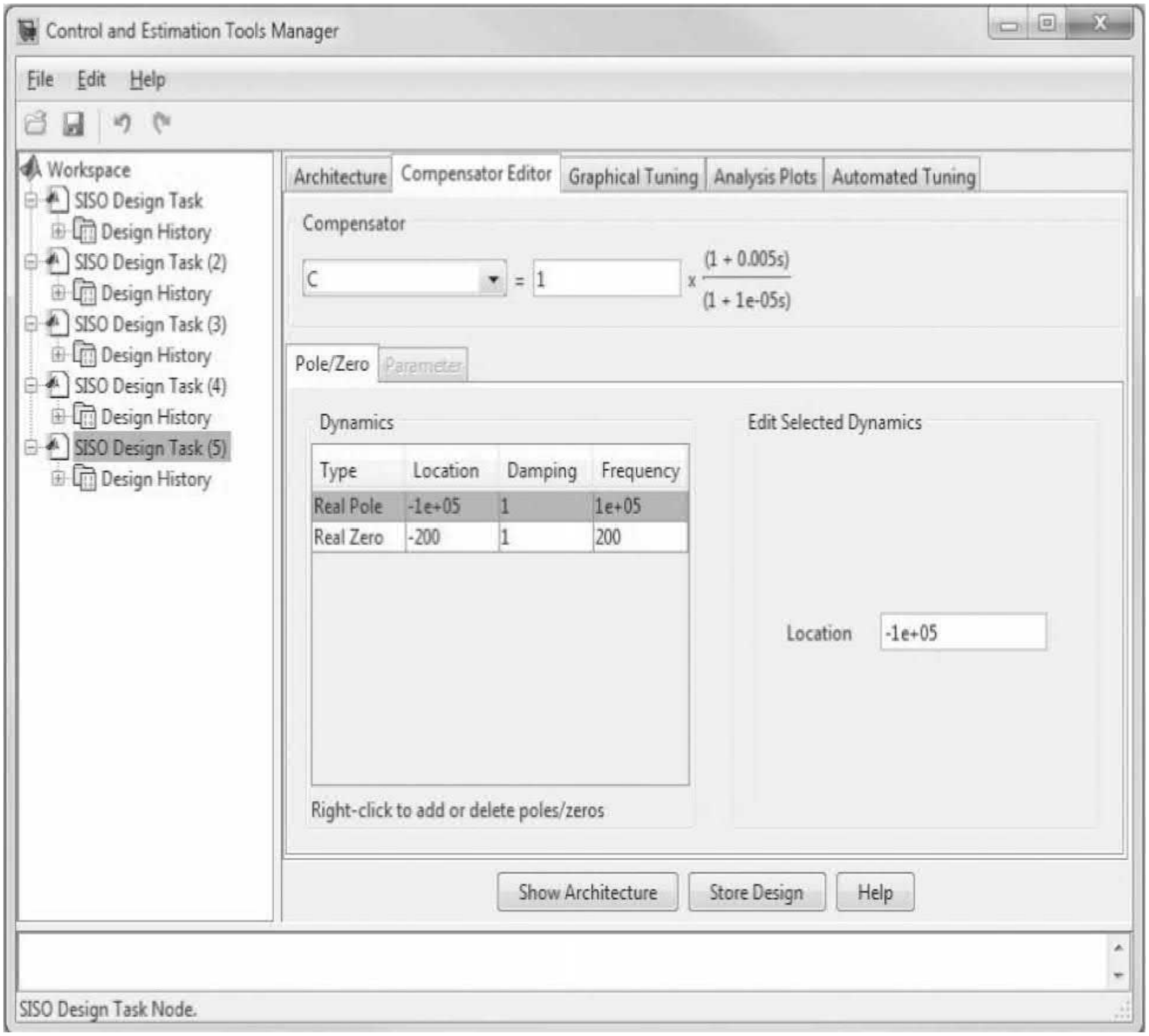


Figure 11-90 Root loci and Bode plots of the PD-controlled system from Example 11-2-2.

EXAMPLE 11-10-3 Example 11-5-2 Revisited

Use the following toolbox to get the uncompensated root locus and bode plots corresponding to Eq. (11-104). Using the root locus and Bode plots, as shown in Fig. 11-91, you can follow the process discussed in Example 11-10-1 to design your phase-lead controller.



Compensator

C = 1 x $\frac{(1 + 0.005s)}{(1 + 1e\text{-}05s)}$

Dynamics

Type	Location	Damping	Frequency
Real Pole	-1e+05	1	1e+05
Real Zero	-200	1	200

Right-click to add or delete poles/zeros

Edit Selected Dynamics

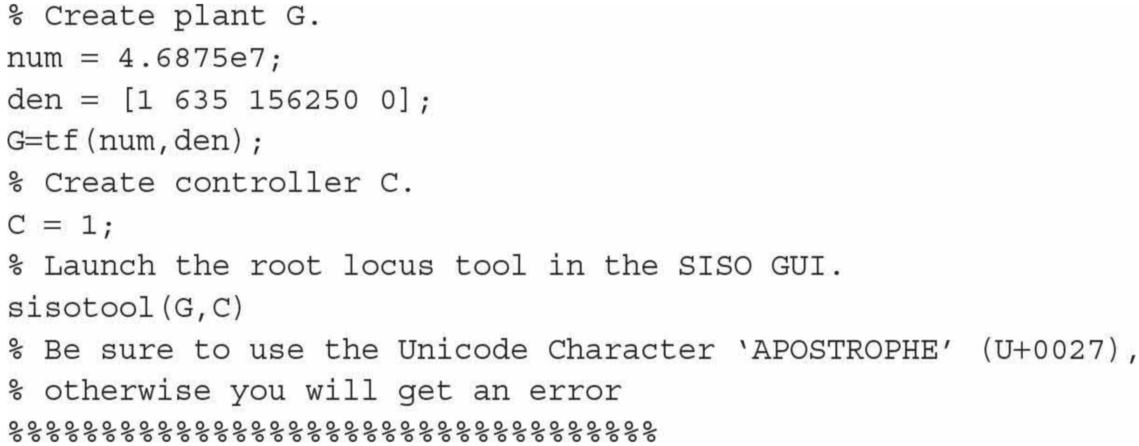
Location -1e+05

SISO Design Task Node.

Figure 11-91 Implementation of a lead controller $G_c(s)=\dfrac{1+aTs}{1+Ts}$ when a = 500 and T = 0.00001, in the SISO design tool for Example 11-10-3.

Toolbox 11-10-6

The SISO design tool for Example 11-10-1 is activated by the following sequence of MATLAB functions:



```
% Create plant G.
num = 4.6875e7;
den = [1 635 156250 0];
G=tf(num,den);
% Create controller C.
C = 1;
% Launch the root locus tool in the SISO GUI.
sisotool(G,C)
% Be sure to use the Unicode Character ‘APOSTROPHE’ (U+0027),
% otherwise you will get an error
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Add the controller $C(s)=\frac{1+aTs}{1+Ts}$ when *a* = 500 and *T* = 0.00001, using the Control and Estimation Tools Manager, as shown in Fig. 11-91. The final system root locus, Bode plots and step response are shown in Fig. 11-92. Also review Example 11-5-2 to confirm the results.

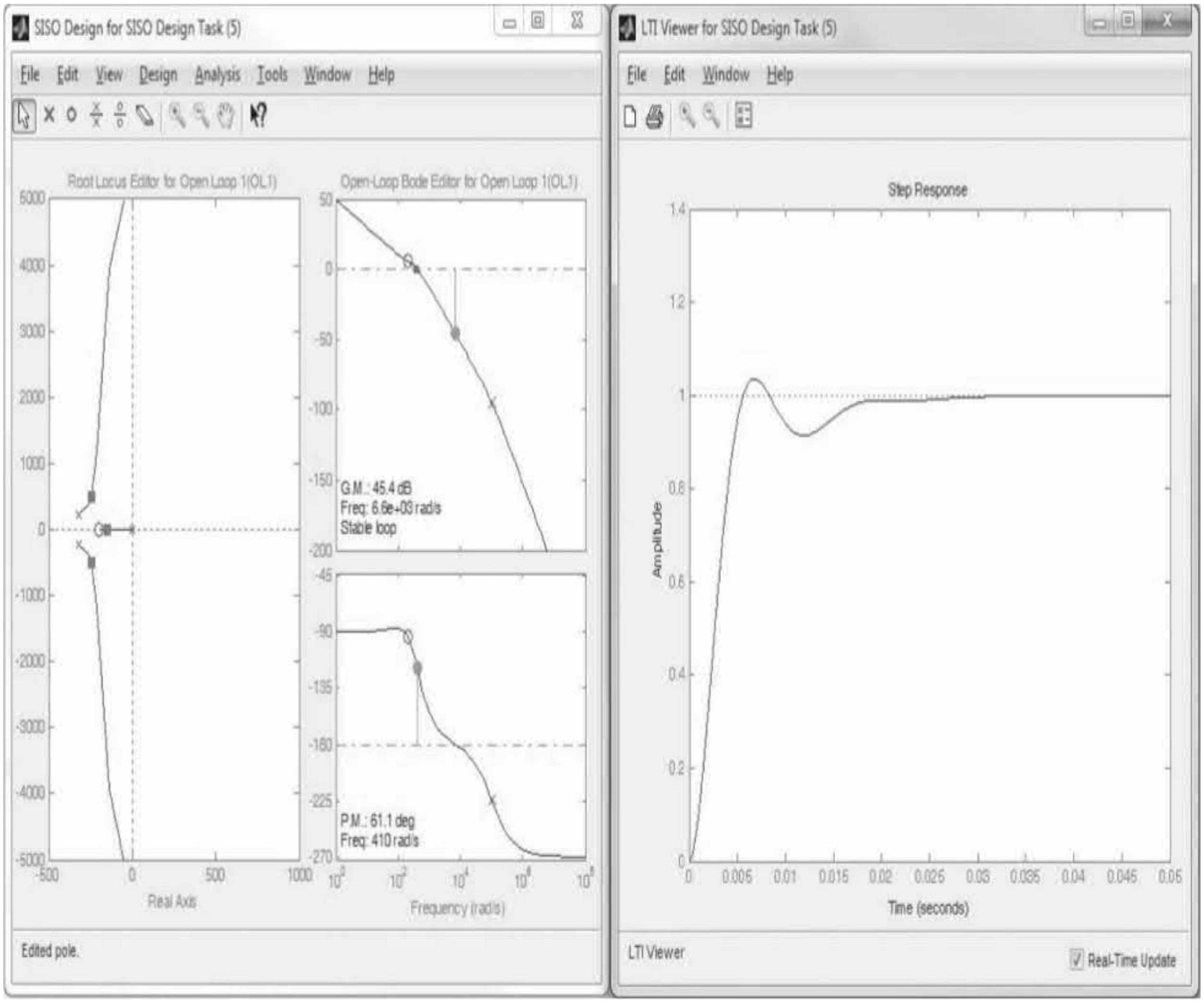


Figure 11-92 Root locus, Bode plots and unit-step response of sun-seeker system in Example 11-5-2 with phase-lead controller. $G_c(s)=\dfrac{1+aTs}{1+Ts}$ when a = 500 and T = 0.00001.

In this example, we illustrate the application of a phase-lead controller to a third-order system

$$G_p(s)=\frac{\Theta_o(s)}{A(s)}=\frac{4.6875\times10^7 K}{s(s^2+625s+156,250)} \tag{11-237}$$

11-11 THE CONTROL LAB

In App. D, we provide examples of LEGO MINDSTORMS labs including a pick and place robot and an elevator positioning system. As a course project, you can use the ideas discussed in this chapter to design different controllers for these systems. You may use the MATLAB **SISO** design tool, Simulink (App. E) or the toolboxes used in this chapter to develop your control systems. Finally, refer to the end of the Problems section for LEGO MINDSTORMS term projects.

REFERENCES

1. Bailey, F. N., and S. Meshkat, "Root Locus Design of a Robust Speed Control," *Proc. Incremental Motion Control Symposium*, pp. 49–54, Jun. 1983.
2. Graebel, W. P., *Engineering Fluid Mechanics*, Taylor & Francis, New York, 2001.
3. Kleman, A., *Interfacing Microprocessors in Hydraulic Systems*, Marcel Dekker, New York, 1989.
4. Kuo, B. C., and F. Golnaraghi, *Automatic Control Systems*, John Wiley & Sons, New York, 2003.
5. Manring, N. D., *Hydraulic Control Systems*, John Wiley & Sons, New York, 2005.
6. McCloy, D., and H. R. Martin, *The Control of Fluid Power*, Longman Group Limited, London, 1973.
7. Ogata, K., *Modern Control Engineering*, Prentice-Hall, New Jersey, 1997.
8. Smith, H. W., and E. J. Davison, "Design of Industrial Regulators," *Proc. IEE (London)*, Vol. 119, pp. 1210–1216, Aug. 1972.
9. Willems, J. C., and S. K. Mitter, "Controllability, Observability, Pole Allocation, and State Reconstruction," *IEEE Trans. Automatic Control*, Vol. AC-16, pp. 582– 595, Dec. 1971.
10. Woods, R. L., and K. L. Lawrence, *Modeling and Simulation of Dynamic Systems*, Prentice-Hall, New Jersey, 1997.

PROBLEMS

Most of the following problems can be solved using a computer program. This is highly recommended if the reader has access to such a program.

11-1. The block diagram of a control system with a series controller is shown in Fig. 11P-1. Find the transfer function of the controller $G_c(s)$ so that the following specifications are satisfied:

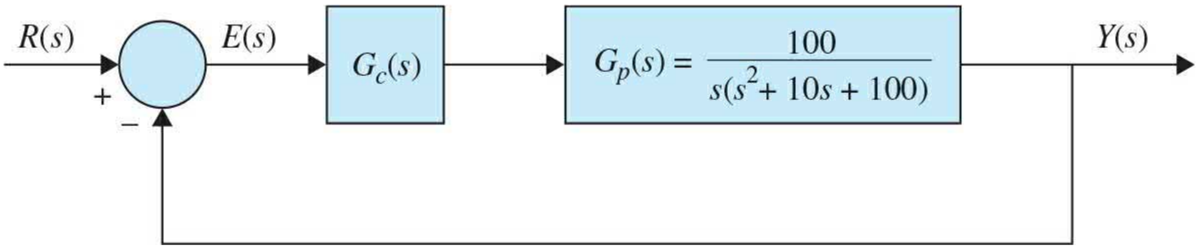


Figure 11P-1

The ramp-error constant K_v is 5.

The closed-loop transfer function is of the form

$$M(s) = \frac{Y(s)}{R(s)} = \frac{K}{(s^2 + 20s + 200)(s + a)}$$

where K and a are real constants. Find the values of K and a.

The design strategy is to place the closed-loop poles at $-10 + j10$ and $-10 - j10$, and then adjust the values of K and a to satisfy the steady-state requirement. The value of a is large so that it will not affect the transient response appreciably. Find the maximum overshoot of the designed system.

11-2. Repeat Prob. 11-1 if the ramp-error constant is to be 9. What is the maximum value of K_v that can be realized? Comment on the difficulties that may arise in attempting to realize a very large K_v.

11-3. The forward-path transfer function of a unity-feedback control system is

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

Find the value of K and t so that the overshoot of the system is 25.4 percent at the $\zeta = 0.4$.

11-4. The forward-path transfer function of a system is

$$G(s)H(s)=\frac{24}{s(s+1)(s+6)}$$

Design a PD controller that satisfies the following factors:

(a) The steady-state error is less than $p / 10$ when the input is a ramp with a slope of $2p$ rad/s.

(b) The phase margin is between 40° and 50°.

(c) Gain-crossover frequency is greater than 1 rad/s.

11-5. A control system with a PD controller is shown in Fig. 11P-5.

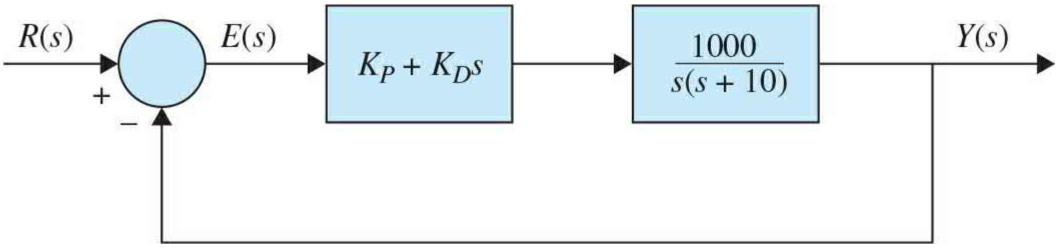


Figure 11P-5

(a) Find the values of K_P and K_D so that the ramp-error constant K_v is 1000 and the damping ratio is 0.5.

(b) Find the values of K_P and K_D so that the ramp-error constant K_v is 1000 and the damping ratio is 0.707.

(c) Find the values of K_P and K_D so that the ramp-error constant K_v is 1000 and the damping ratio is 1.0.

11-6. For the control system shown in Fig. 11P-5, set the value of K_P so that the ramp-error constant is 1000.

(a) Vary the value of K_D from 0.2 to 1.0 in increments of 0.2 and determine the values of phase margin, gain margin, M_r, and BW of the system. Find the value of K_D so that the phase margin is maximum.

(b) Vary the value of K_D from 0.2 to 1.0 in increments of 0.2 and find the value of K_D so that the maximum overshoot is minimum.

11-7. The forward-path transfer function of a system is

$$G(s)H(s) = \frac{1}{(2s+1)(s+1)(0.5s+1)}$$

Design a PD controller such that the K_p = 9 and the phase margin is greater than 25°.

11-8. The forward-path transfer function of a system is

$$G(s)H(s) = \frac{60}{s(0.4s+1)(s+1)(s+6)}$$

(a) Design a PD controller to satisfy the following specifications:

(i) K_v = 10

(ii) The phase margin is 45°.

(b) Use MATLAB to plot the Bode diagram of the compensated system.

11-9. Consider the second-order model of the aircraft attitude control system shown in Fig. 7-51. The transfer function of the process is

$$G_p(s) = \frac{4500\ K}{s(s+361.2)}$$

(a) Design a series PD controller with the transfer function $G_c(s) = K_D + K_P s$ so that the following performance specifications are satisfied:

Steady – state error due to a unit – ramp input ≤ 0.001

Maximum overshoot ≤ 5%

Rise time $t_r \leq 0.005$ s

Setting time $t_s \leq 0.005$ s

(b) Repeat part (a) for all the specifications listed, and, in addition, the bandwidth of the system must be less than 850 rad/s.

11-10. Figure 11P-10 shows the block diagram of the liquid-level control system described in Prob. 2-36. The number of inlets is denoted by N. Set N = 20. Design the PD controller so that with a unit-step input the tank is filled to within 5 percent of the reference level in less than 3 s without overshoot.

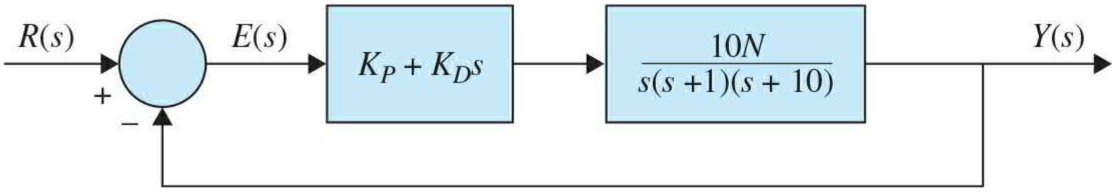


Figure 11P-10

11-11. For the liquid-level control system described in Prob. 11-10.

(a) Set K_P so that the ramp-error constant is 1. Vary K_D from 0 to 0.5 and find the value of K_D that gives the maximum phase margin. Record the gain margin, M_r, and BW.

(b) Plot the sensitivity functions $\left|S_G^M(j\omega)\right|$ of the uncompensated system and the compensated system with the values of K_D and K_P determined in part (a). How does the PD controller affect the sensitivity?

11-12. The block diagram of a servo system is shown in Fig. 11P-12.

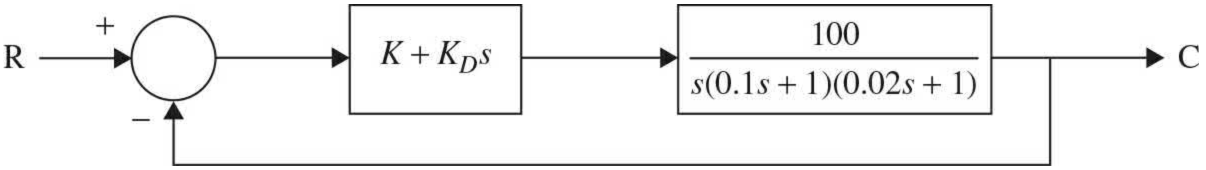


Figure 11P-12

Design the PD controller so that the phase margin is greater than 50° and the BW is greater than 20 rad/s. Use MATLAB to verify your answer.

11-13. The forward-path transfer function of a unity-feedback system is

$$G(s)H(s) = \frac{1000\ K}{s(0.2s+1)(0.005s+1)}$$

Design a compensator such that the steady-state error to the unit-step input is less than 0.01 and the closed-loop damping ratio $\zeta > 0.4$.

Use MATLAB to plot the Bode diagram of the compensated system.

11-14. The open-loop transfer function of a dc motor is

$$G(s) = \frac{250}{s(0.2s+1)}$$

Design a PD controller so that the steady-state error to the input ramp is less than 0.005, the maximum overshoot is 20 percent for the unit-step input, and the BW must be maintained at a value approximately the same as that of the uncompensated system.

11-15. The open-loop plant model of a plastic extrusion is given by

$$G(s)=\frac{40}{(s+1)(0.25s+1)}$$

Design a series of lead compensator, which is described by

$$G_c(s)=\frac{r(\tau s+1)}{(r\tau s+1)}$$

so that the phase margin is 45° and the BW must be maintained at a value approximately the same as that of the uncompensated system.

11-16. Repeat Prob. 11-15 assuming that the $r < 0.1$.

11-17. The forward-path transfer function of a unity-feedback control system is

$$G(s)H(s)=\frac{1000K}{s(0.2s+1)(0.05s+1)}$$

(a) Design a compensator such that

(i) The steady-state error is less than 0.01 for a unit-ramp input.

(ii) The phase margin is greater than 45°.

(iii) The steady-state error is less than 0.004 for a sinusoidal input with $\omega > 0.2$.

(iv) The noise for the frequencies greater than 200 rad/s reduced to 100 at the output.

(b) Use MATLAB to plot the Bode diagram of the compensated system and verify or refine your design in part (a).

11-18. A control system with a type 0 process $G_p(s)$ and a PI controller is shown in Fig. 11P-18.

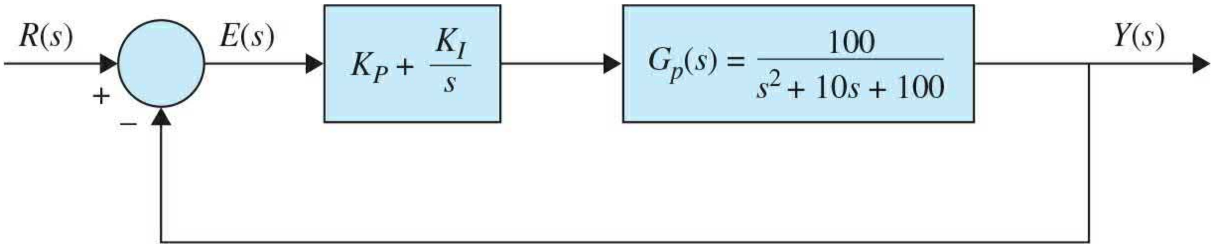


Figure 11P-18

(a) Find the value of K_I so that the ramp-error constant K_v is 10.

(b) Find the value of K_P so that the magnitude of the imaginary parts of the complex roots of the characteristic equation of the system is 15 rad/s. Find the roots of the characteristic equation.

(c) Sketch the root contours of the characteristic equation with the value of K_I as determined in part (a) and for $0 \leq K_P > \infty$.

11-19. For the control system described in Prob. 11-18,

(a) Set K_I so that the ramp-error constant is 10. Find the value of K_P so that the phase margin is minimum. Record the values of the phase margin, gain margin, M_r, and BW.

(b) Plot the sensitivity functions $\left|S_G^M(j\omega)\right|$ of the uncompensated and the compensated systems with the values of K_I and K_P selected in part (a). Comment on the effect of the PI control on sensitivity.

11-20. For the control system shown in Fig. 11P-18, perform the following:

(a) Find the value of K_I so that the ramp-error constant K_v is 100.

(b) With the value of K_I found in part (a), find the critical value of K_P so that the system is stable. Sketch the root contours of the characteristic equation for $0 \leq K_P > \infty$.

(c) Show that the maximum overshoot is high for both large and small values of K_P. Use the value of K_I found in part (a). Find the value of K_P when the maximum overshoot is a minimum. What is the value of this maximum overshoot?

11-21. Repeat Prob. 11-20 for $K_v = 10$.

11-22. The forward-path transfer function of a system is

$$G(s)=\frac{24}{s(s+1)(s+6)}$$

(a) Design a PI controller that satisfies the following factors:

(i) The ramp error constant $K_v < 20$.

(ii) The phase margin is between 40° and 50°.

(iii) Gain-crossover frequency is greater than 1 rad/s.

(b) Use MATLAB to plot the Bode diagram of the closed-loop system.

11-23. The forward-path transfer function of a robot arm–positioning system is represented by

$$G(s)=\frac{40}{s(s+2)(s+20)}$$

(a) Design a PI controller such that

(i) The steady-state error is less than 5 percent of the slope for a ramp input.

(ii) The phase margin is between 32.5° and 37.5°.

(iii) Gain-crossover frequency is 1 rad/s.

(b) Use MATLAB to plot the Bode diagram of the closed-loop system and verify your design in part (a).

11-24. The forward-path transfer function of a system is

$$G(s)=\frac{210}{s(5s+7)(s+3)}$$

Design a PI controller with a unity dc gain so that the phase margin of the system is greater than 40°, and then find the BW of the system.

11-25. The transfer function of the steering of a ship is given by

$$G(s)=\frac{2353K(71-500s)}{71s(40s+13)(5000s+181)}$$

Design a PI controller such that

(a) The ramp error constant $K_v = 2$.

(b) The phase margin is greater than 50°.

(c) For all frequencies greater than crossover frequency, PM > 0. This means the system is always stable without any condition.

(d) Show the closed-loop poles in the root locus with respect to values of *K*.

11-26. The transfer function of a unity-feedback system is

$$G(s)=\frac{2\times10^{5}}{s(s+20)(s^{2}+50s+10000)}$$

(a) A PD controller with the transfer function of $H(s)=\dfrac{(\tau s+1)}{(r\tau s+1)}$ is designed with $r = 0.2$ and $\tau = 0.05$. It is desired to find the gain so that the crossover frequency is 31.6 rad/s.

(b) Find the ramp error constant *Kv* by applying the controller designed in part (a).

(c) Consider the PD controller designed in part (a) is applied to the system. Find the value of *K* for a PI controller so that the ramp error constant K_v = 100.

(d) If the PI controller pole is at 3.16 rad/s and the crossover frequency maintains at 31.6 rad/s, what is the zero of the PI controller? [Consider the transfer function of the PI controller is $H(s)=\dfrac{r(\tau s+1)}{(r\tau s+1)}$.]

(e) Use MATLAB to plot the Bode diagram of the compensated system and find the phase margin.

11-27. A control system with a type 0 process and a PID controller is shown in Fig. 11P-27. Design the controller parameters so that the following specifications are satisfied:

Ramp-error contant K_v = 100.

Rise time $t_v < 0.01$ s

Maximum overshoot < 2%

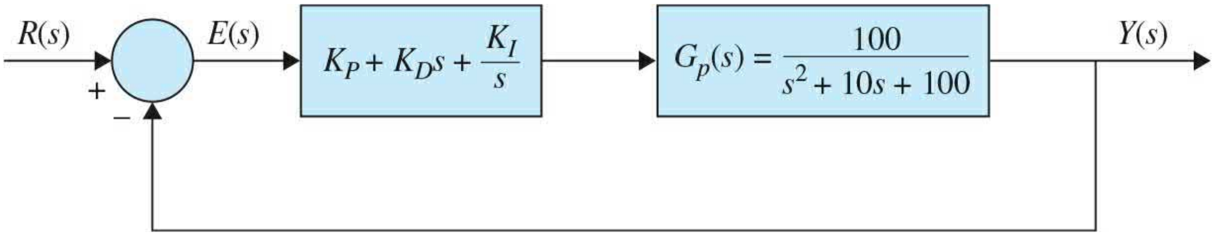

$R(s)$
$E(s)$
$K_P + K_D s + \frac{K_I}{s}$
$G_p(s) = \frac{100}{s^2 + 10s + 100}$
$Y(s)$
+
−

Figure 11P-27

Plot the unit-step response of the designed system.

11-28. A considerable amount of effort is being spent by automobile manufacturers to meet the exhaust-emission-performance standards set by the government. Modern automotive-power-plant systems consist of an internal combustion engine that has an internal cleanup device called the catalytic converter. Such a system requires control of the engine air-fuel ratio (A/F), the ignition-spark timing, exhaust-gas recirculation, and injection air. The control system problem considered in this exercise deals with the control of the air-fuel ratio. In general, depending on fuel composition and other factors, a typical stoichiometric A/F is 14.7:1, that is, 14.7 g of air to each gram of fuel. An A/F greater or less than stoichiometry will cause high hydrocarbons, carbon monoxide, and nitrous oxide in the tailpipe emission. The control system whose block diagram is shown in Fig. 11P-28 is devised to control the air-fuel ratio so that a desired output variable is maintained for a given command signal. Figure 11P-28 shows that the sensor senses the composition of the exhaust-gas mixture entering the catalytic converter. The electronic controller detects the difference or the error between the command and the sensor signals and computes the control signal necessary to achieve the desired exhaust-gas composition. The output variable $y(t)$ denotes the effective air-fuel ratio. The transfer function of the engine is given by

$$\frac{Y(s)}{U(s)} = G_p(s) = \frac{e^{-T_d s}}{1+\tau s}$$

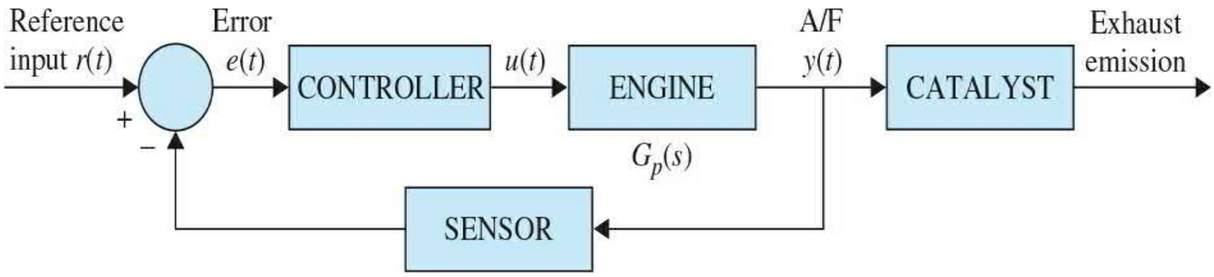


Figure 11P-28

where T_d is the time delay and is 0.2 s. The time constant τ is 0.25 s. Approximate the time delay by a power series:

$$e^{-T_d s} \cong \frac{1}{1+T_d s+T_d^2 s^2/2}$$

(a) Let the controller be a PI controller so that

$$G_c(s)=\frac{U(s)}{E(s)}=K_P+\frac{K_I}{s}$$

Find the value of K_I so that the ramp-error constant K_v is 2. Determine the value of K_P so that the maximum overshoot of the unit-step response is a minimum and the settling time is a minimum. Give the values of the maximum overshoot and the settling time. Plot the unit-step response of $y(t)$. Find the marginal value of K_P for system stability.

(b) Can the system performance be further improved by using a PID controller?

$$G_c(s)=\frac{U(s)}{E(s)}=K_P+K_D s+\frac{K_I}{s}$$

11-29. One of the advantages of the frequency-domain analysis and design methods is that systems with pure time delays can be treated without approximation. Consider the automobile-engine control system treated in Prob. 11-28.

The process has the transfer function

$$G_p(s) = \frac{e^{-0.2s}}{1+0.25s}$$

Let the controller be of the PI type so that $G_c(s) = K_P + K_I/s$. Set the value of K_I so that the ramp-error constant K_v is 2. Find the value of K_P so that the phase margin is a maximum. How does this "optimal" K_P compare with the value of K_P found in Prob. 11-28*a*? Find the critical value of K_P for system stability. How does this value of K_P compare with the critical value of K_P found in Prob. 11-28?

11-30. Figure 11P-30 shows a simplified design of an airplane attitude controller.

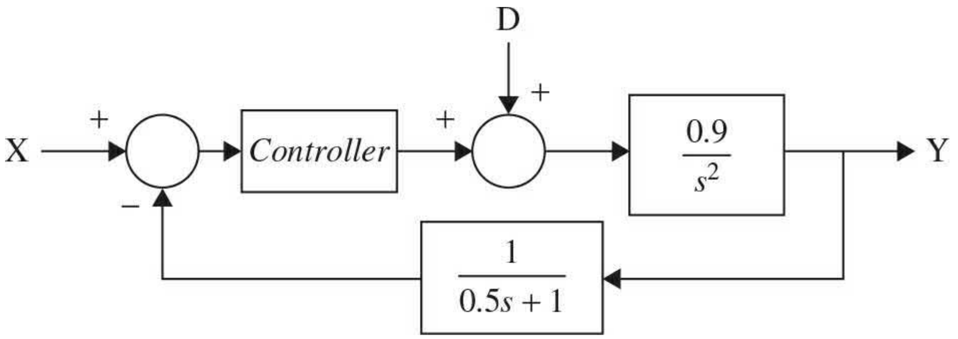


Figure 11P-30

where *D* is the disturbance torque. Design a PID controller with the following satisfactions:

(a) Zero steady-state error

(b) PM = 65°

(c) High bandwidth (as high as possible)

11-31. Consider the open-loop plant model of a plastic extrusion given in Prob. 11-15.

Design a series of lead-lag compensator that is described by

$$H(s) = \frac{(\tau_1 s+1)(\tau_2 s+1)}{\tau_1\tau_1 s^2 + \left(\tau_1 + \frac{\tau_2}{r}\right)s + 1}$$

and satisfies the following:

(a) The phase margin is 45°.

(b) The steady-state error of a closed-loop system to the unit-step input is less than 1 percent.

(c) The gain-crossover frequency is 5 rad/s.

11-32. A telescope for tracking stars and asteroids on a space shuttle may be modeled as a pure mass *M*. It is suspended by magnetic bearings so that there is no friction, and its attitude is controlled by magnetic actuators located at the base of the payload. The dynamic model for the control of the *z*-axis motion is shown in Fig. 11P-32*a*. Because there are electrical components on the telescope, electric power must be brought to the telescope through a cable. The spring shown is used to model the wire-cable attachment, which exerts a spring force on the mass. The force produced by the magnetic actuators is denoted by *f* (*t*). The force equation of motion in the *z* direction is

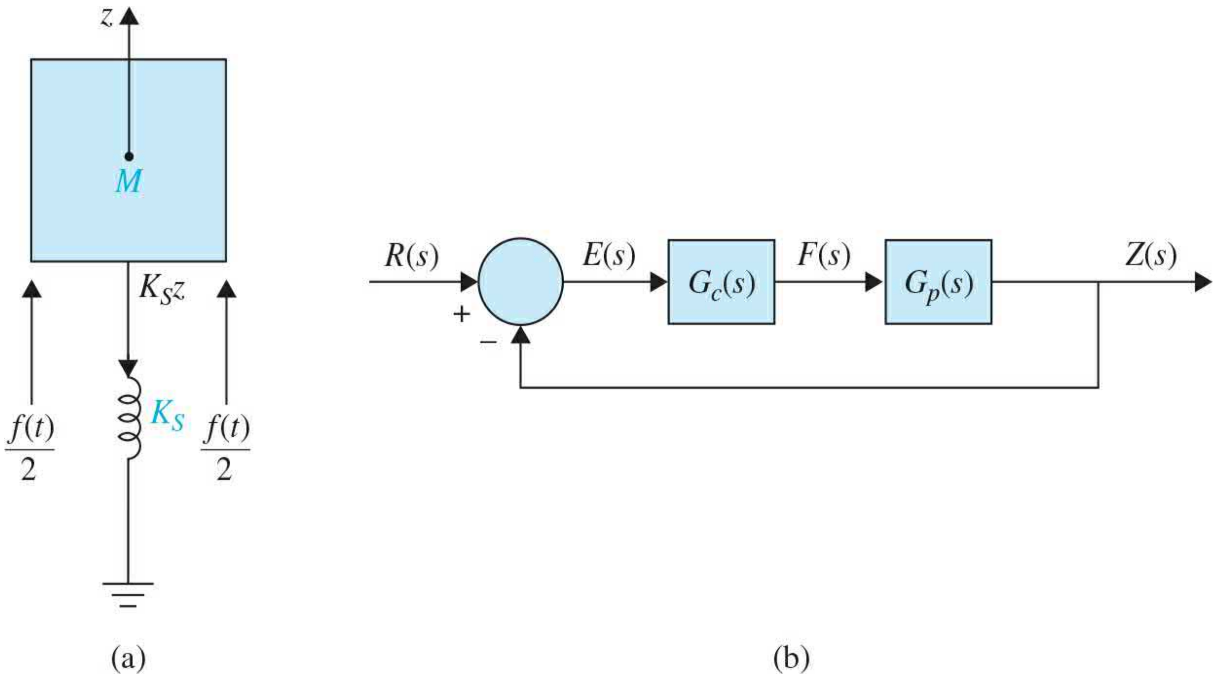


Figure 11P-32

where K_s = 1 lb/ft, and *M* = 150 lb (mass); *f*(*t*) is in pounds, and *z*(*t*) is measured in feet.

(a) Show that the natural response of the system output *z*(*t*) is oscillatory without damping. Find the natural undamped frequency of the open-loop space-shuttle system.

(b) Design the PID controller

$$G_c(s) = K_P + K_D s + \frac{K_I}{s}$$

shown in Fig. 9P-32*b* so that the following performance specifications are satisfied:

$$\text{Ramp-error constant } K_v = 100.$$

The complex characteristic equation roots correspond to a relative damping ratio of 0.707 and a natural undamped frequency of 1 rad/s.

Compute and plot the unit-step response of the designed system. Find the maximum overshoot. Comment on the design results.

(c) Design the PID controller so that the following specifications are satisfied:

Ramp-error constant K_v = 100.

Maximum overshoot < 5%

Compute and plot the unit-step response of the designed system. Find the roots of the characteristic equation of the designed system.

11-33. Repeat Prob. 11-32*b* with the following specifications:

$$\text{Ramp-error constant } K_v = 100.$$

The complex characteristic equation roots correspond to a relative damping ratio of 1.0 and a natural undamped frequency of 1 rad/s.

11-34. Consider a cruise control system shown in Fig. 11P-34.

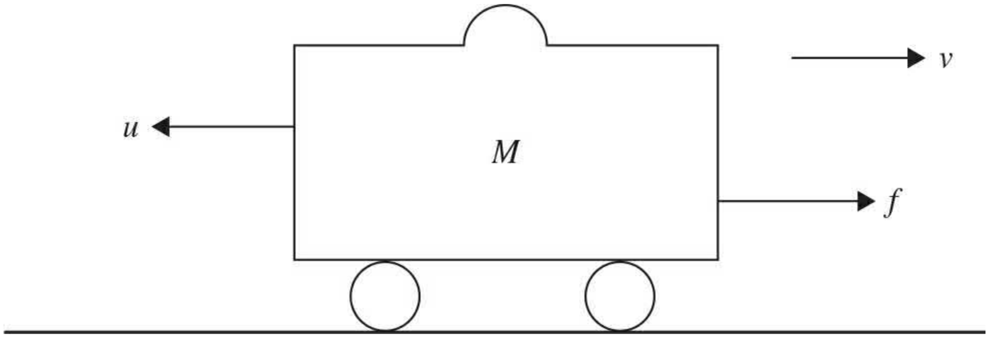


Figure 11P-34

where f is the engine force, v is the velocity, u is the friction force, and $u = \mu v$.

Assuming $M = 1000$ kg, $\mu = 50$ Nsec/m, and $f = 500$ N:

(a) Find the transfer function of the system.

(b) Design a PID controller that satisfies the following:

(i) Rise time is less than 5 s.

(ii) Maximum overshoot is less than 10 percent.

(iii) Steady-state error is less than 2 percent.

11-35. An inventory control system is modeled by the following state equations:

$$\frac{dx_1(t)}{dt} = -2x_2(t)$$

$$\frac{dx_2(t)}{dt} = -2u(t)$$

where $x_1(t)$ = level of inventry, $x_2(t)$ = rate of sales of product, and $u(t)$ = prodcution rate. The output equation is $y(t) = x_1(t)$. One unit of time is one day. Fig. 11P-35 shows the block diagram of the closed-loop inventory control system with a series controller. Let the controller be a PD controller, $G_c(s) = K_P + K_D s$.

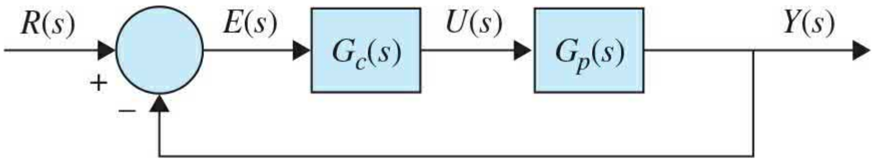


Figure 11P-35

(a) Find the parameters of the PD controller, K_P and K_D, so that the roots of the characteristic equation correspond to a relative damping ratio of 0.707 and $\omega_n = 1$ rad/s. Plot the unit-step response of $y(t)$ and find the maximum overshoot.

(b) Find the values of K_P and K_D so that the overshoot is zero and the rise time is less than 0.06 s.

(c) Design the PD controller so that $M_r = 1$ and BW ≤ 40 rad/s.

11-36. The block diagram of a type 2 control system with a series controller $G_c(s)$ is shown in Fig. 11P-36.

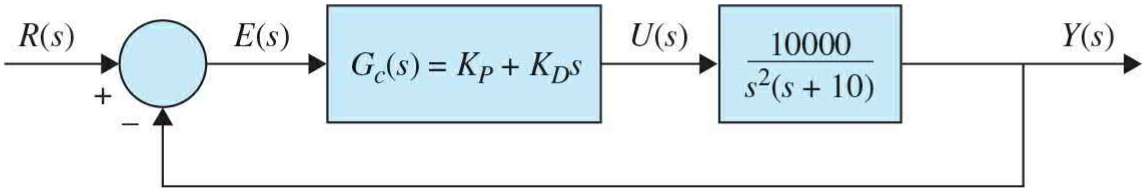


Figure 11P-36

The objective is to design a PD controller so that the following specifications are satisfied:

Maximum overshoot < 10%

Rise Time < 0.5 s

(a) Obtain the characteristic equation of the closed-loop system, and determine the ranges of the values of K_P and K_D for stability. Show the region of stability in the K_D-versus-K_P plane.

(b) Construct the root loci of the characteristic equation with $K_D = 0$ and $0 \le K_P < \infty$. Then construct the root contours for $0 \le K_P < \infty$ and several fixed values of K_P ranging from 0.001 to 0.01.

(c) Design the PD controller to satisfy the performance specifications given. Use the information on the root contours to help your design. Plot the unit-step response of $y(t)$.

(d) Check the design results obtained in part (c) in the frequency domain. Determine the phase margin, gain margin, M_r, and BW of the designed system.

11-37. Consider a dc motor shown in Fig. 11P-37 and described in Sec. 4-7-3.

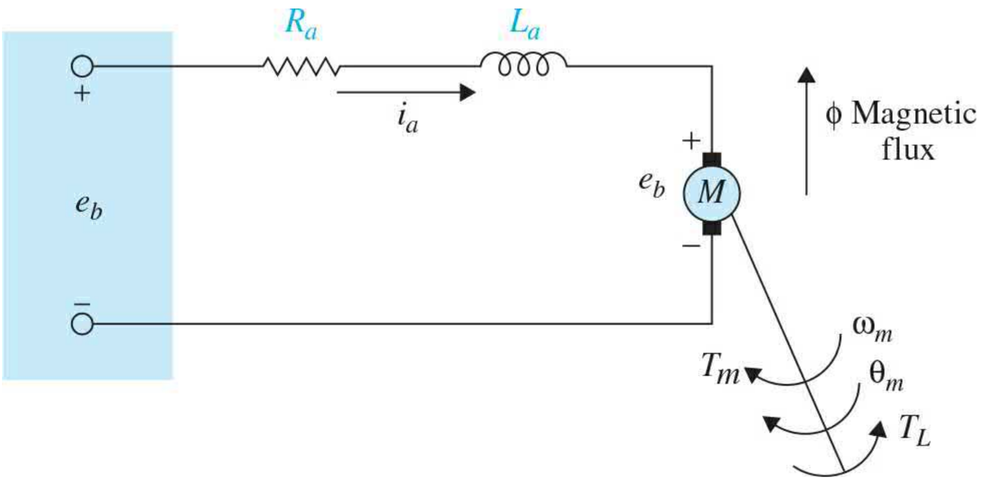


Figure 11P-37

Assuming the following:

The rotor inertia (J) = 0.01 kg · m²/s²

Damping ratio of the mechanical system (ζ) = 0.1 Nms

Back-emf constant (K_b) = 0.01 Nm/amp

Torque constant (K_t) = 0.01 Nm/amp

Armature resistance (R_a) = 1 Ω

Armature inductance (L_a) = 0.5 H

Design a PID controller that satisfies the following:

(a) Settling time is less than 2 s.

(b) Maximum overshoot is less than 5 percent.

(c) Steady-state error is less than 1 percent.

11-38. For the dc motor described in Prob. 11-37, assuming the following:

The rotor inertia (J) = 3.2284E-6 kg · m²/s²

Damping ratio of the mechanical system (ζ) = 3.5077E-6 Nms

Back-emf constant (K_b) = 0.0274 Nm/amp

Torque constant (K_t) = 0.0274 Nm/amp

Armature resistance (R_a) = 4 Ω

Armature inductance (L_a) = 2.75E-6 H

Design a PID controller that satisfies the following:

(a) Settling time is less than 40 ms.

(b) Maximum overshoot is less than 16 percent.

(c) Zero steady-state error is less than 1 percent.

(d) Zero steady-state error due to a disturbance.

11-39. Consider the broom-balancing control system described in Probs. 3-43 and 8-51. The $\mathbf{A}^*$ and $\mathbf{B}^*$ matrices are given in Prob. 8-51 for the small-signal linearized model.

$$\Delta\dot{\mathbf{x}} = \mathbf{A}^*\,\Delta\mathbf{x}(t) + \mathbf{B}^*\,\Delta r(t)$$
$$\Delta y(t) = \mathbf{C}\Delta\mathbf{x}(t)$$
$$\mathbf{D}^* = [0 \quad 0 \quad 1 \quad 0]$$

Figure 11P-39 shows the block diagram of the system with a series PD controller. Determine if the PD controller can stabilize the system; if so, find the values of K_P and K_D. If the PD controller cannot stabilize the system, explain why not.

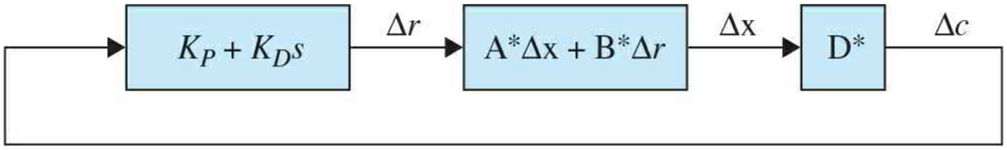


Figure 11P-39

11-40. The process of a unity-feedback control system has the transfer function

$$G_p(s) = \frac{100}{s^2 + 10s + 100}$$

Design a series controller (PD, PI, or PID) so that the following performance specifications are satisfied:

Steady-state error due to a step input = 0

Maximum overshoot < 2%

Rise time < 0.02 s

Carry out the design in the frequency domain and check the design in the time domain.

11-41. The forward path of a unity-feedback control system that includes a disturbance signal $D(s)$ is given by

$$G(s)=\frac{1}{(s^2+3.6s+9)}$$

(a) Design a PID controller with the transfer function of

$$H(s)=\frac{K(\tau_1 s+1)(\tau_2 s+1)}{s}$$

so that the response to any step disturbance is damped in less than 3 s at the 2 percent settling time.

(b) Use MATLAB to plot the response of the closed-loop system to various step disturbance inputs and verify your design in part (a).

11-42. For the inventory control system shown in Fig. 11P-35, let the controller be of the phase-lead type:

$$G_c(s)=\frac{1+aTs}{1+Ts} \quad a>1$$

Determine the values of a and T so that the following performance specifications are satisfied:

Steady-state error due to a step input = 0

Maximum overshoot < 5%

(a) Design the controller using the root contours with T and a as variable parameters. Plot the unit-step response of the designed system. Plot the Bode diagram of $G_s(s) = G_c(s)G_p(s)$, and find PM, GM, M_r, and BW of the designed system.

(b) Design the phase-lead controller so that the following performance specifications are satisfied:

Steady-state error due to a step input = 0

Phasemargin > 75°

$M_r < 1.1$

Construct the Bode diagram of $G(s)$ and carry out the design in the frequency domain. Find the attributes of the time response of the designed system.

11-43. Consider that the process of a unity-feedback control system is

$$G_p(s)=\frac{1000}{s(s+10)}$$

Let the series controller be a single-stage phase-lead controller:

$$G_c(s)=\frac{1+aTs}{1+Ts}a>1$$

(a) Determine the values of *a* and *T* so that the zero of $G_c(s)$ cancels the pole of $G_p(s)$ at $s = -10$.

The damping ratio of the designed system should be unity. Find the attributes of the unit-step response of the designed system.

(b) Carry out the design in the frequency domain using the Bode plot. The design specifications are as follows:

Phasemargin > 75°

$M_r < 1.1$

Find the attributes of the unit-step response of the designed system.

11-44. Figure 11P-44 shows the quarter-car model realization with 2° of freedom.

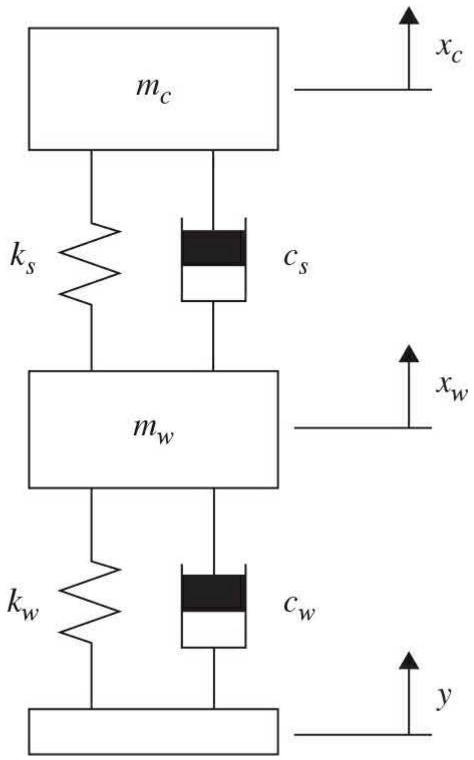


Figure 11P-44

Assuming:

Body mass (m_c) = 2500 kg

Suspension mass (m_w) = 320 kg

Spring constant of suspension system (k_c) = 80,000 N/m

Spring constant of wheel and tire (k_w) = 500,000 N/m

Damping constant of suspension system (C_s) = 350 Ns/m

Damping constant of wheel and tire (X_w) = 15,020 Ns/m

When the vehicle is experiencing any road disturbance, the vehicle body should not have large oscillations, and the oscillations should be damped quickly. If the deformation tire is negligible, and the road disturbance (**D**) is considered a step input,

(a) Design a PID controller that satisfies the following requirements:

(i) Overshoot less than 5 percent

(ii) Settling time shorter than 5 s

(b) Use MATLAB to plot the response of the closed-loop system to various step disturbance inputs and verify your design in part (a).

11-45. Consider that the controller in the liquid-level control system shown in Fig. 11P-10 is a phase-lead controller:

$$G_c(s) = \frac{1 + aTs}{1 + Ts} a > 1$$

(a) For $N = 20$, select the values of a and T so that the maximum overshoot is barely 0 percent. The value of a must not exceed 1000. Find the attributes of the unit-step response of the designed system. Plot the unit-step response.

(b) For $N = 20$, design the phase-lead controller in the frequency domain. Find the values of a and T so that the phase margin is maximized subject to the condition that BW > 100. The value of a must not exceed 1000.

11-46. The transfer function of the process of a unity-feedback control system is

$$G_p(s) = \frac{6}{s(1 + 0.2s)(1 + 0.5s)}$$

(a) Construct the Bode diagram of $G_p(j\omega)$ and determine the PM, GM, M_r, and BW of the system.

(b) Design a series single-stage series phase-lead controller with the transfer function

$$G_c(s) = \left(\frac{1 + aTs}{1 + Ts}\right) \quad a > 1$$

so that the phase margin is maximum. The value of a must not be greater than 1000. Determine PM and M_r of the designed system. Determine the attributes of the unit-step response.

(c) Using the system designed in part (b) as a basis, design a two-stage phase-lead controller so that the phase margin is at least 85°. The transfer function of the two-stage phase-lead controller is

$$G_c(s) = \left(\frac{1 + aT_1 s}{1 + T_1 s}\right)\left(\frac{1 + bT_2 s}{1 + T_2 s}\right) \quad a > 1 \quad b > 1$$

where a and T_1 are determined in part (b). The value of T_2 should not exceed 1000. Find the values of PM and M_r of the designed system. Find the attributes of the unit-step response.

(d) Plot the unit-step responses of the output in parts (a), (b), and (c).

Figure 11P-47 shows an inverted pendulum on a cart.

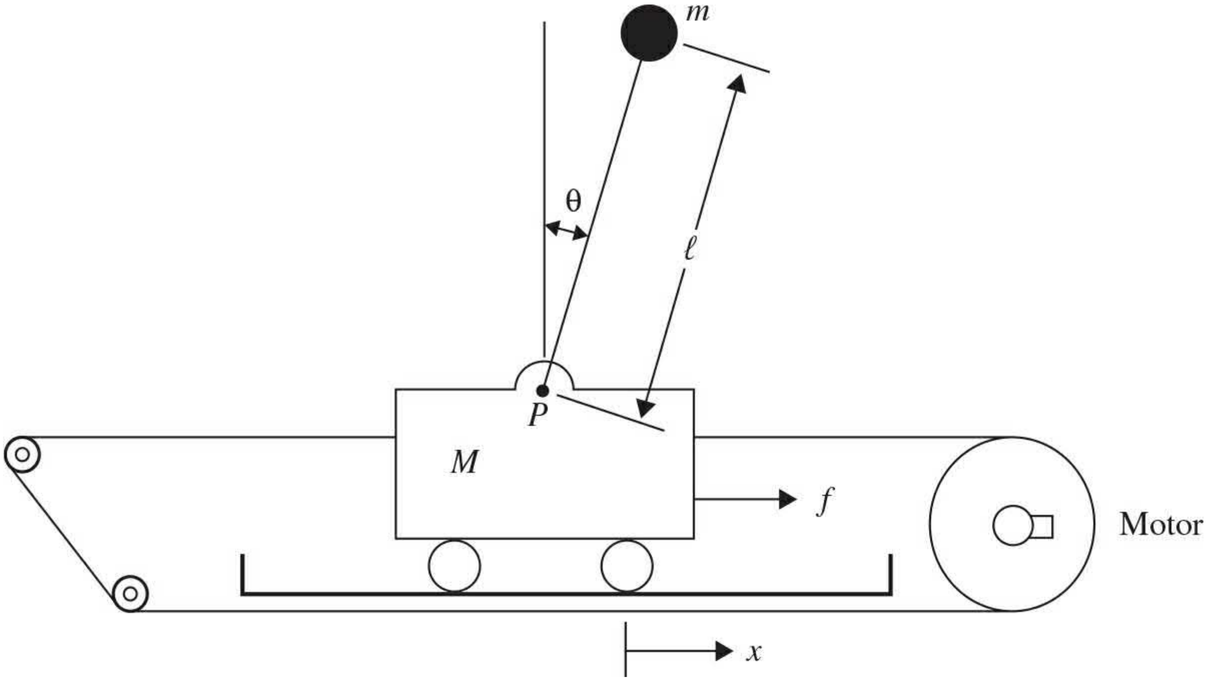


Figure 11P-47

Assuming:

M	Mass of the cart	0.5 kg
m	Mass of the pendulum	0.2 kg
μ	Friction of the cart	0.1 N/m/s
l	Length to pendulum center of mass	0.3 m
I	Inertia of the pendulum	0.006 kg*m^2

(a) Design a PID controller so that the settling time is less than 5 s and the pendulum angle is never more than 0.05 radians from the vertical position.

(b) If the step input is applied to the cart, design a PID controller so that the settling time for *x* and *q* is less than 5 s, the rise time for *x* is less than 0.5 s, and the overshoot of theta is less than 20° (0.35 rad).

11-48. A phase-lock-loop, dc-motor-speed-control system. The block diagram of the system is shown in Fig. 11P-48. The system parameters and transfer functions are given as follows:

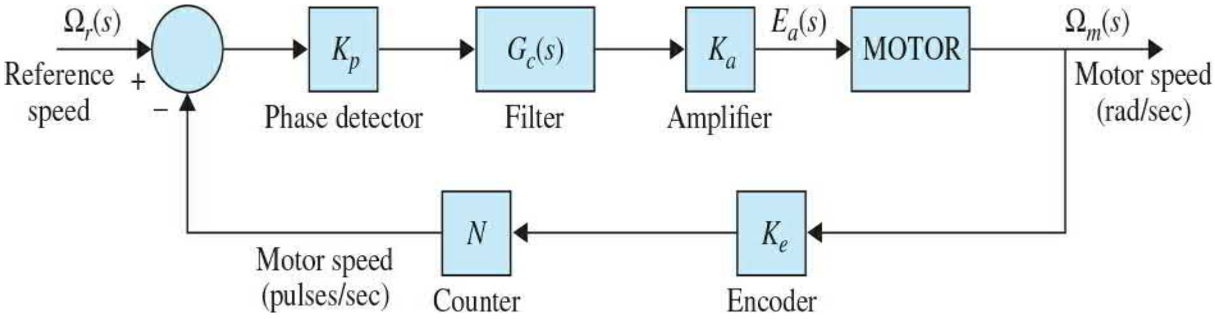


Figure 11P-48

Reference speed command, $\omega_r = 120$ pulse/s

Phase-detector gain, $K_p = 0.06$ V/pulse/s

Amplifier gain, $K_a = 20$

Encoder gain, $K_e = 5.73$ pulse/rad

Counter gain, $N = 1$

Motor transfer function,

$$\frac{\Omega_m(s)}{E_a(s)} = \frac{10}{s(1+0.05s)}$$

(a) Let the filter (controller) transfer function be of the form

$$G_c(s) = \frac{E_o(s)}{E_i(s)} = \frac{1+R_2Cs}{R_1Cs}$$

where $R_1 = 2 \times 10^6$ Ω and $C = 1$ μF. Determine the value of R_2 so that the complex roots of the closed-loop characteristic equation have a maximum relative damping ratio. Sketch the root loci of the characteristic equation for $0 \le R_2 < \infty$. Compute and plot the unit-step responses of the motor speed $f_\omega(t)$ (pulse/s) with the values of R_2 found, when the input is 120 pulse/s. Convert the speed in pulse/s to rpm.

(b) Let the filter transfer function be

$$G_c(s) = \frac{1+aTs}{1+Ts} \quad a > 1$$

where $T = 0.01$. Find a so that the complex roots of the characteristic equation have a maximum relative damping ratio. Compute and plot the unit-step response of the motor speed $f_\omega(t)$ (pulse/s) when the input is 120 pulse/s.

(c) Design the phase-lead controller in the frequency domain so that the phase margin is at least 60°.

11-49. Consider that the controller in the liquid-level control system shown in Fig. 11P-10 is a single-stage phase-lag controller:

$$G_c(s) = \frac{1+aTs}{1+Ts} \quad a < 1$$

(a) For $N = 20$, select the values of a and T so that the two complex roots of the characteristic equation correspond to a relative damping ratio of approximately 0.707. Plot the unit-step response of the output $y(t)$. Find the attributes of the unit-step response. Plot the Bode plot of $G_c(s)G_p(s)$ and determine the phase margin of the designed system.

(b) For $N = 20$, design the phase-lag controller in the frequency domain so that the phase margin is approximately 60°. Plot the unit-step response of the output $y(t)$, and find the attributes of the unit-step response.

11-50. The controlled process of a unity-feedback control system is

$$G_p(s) = \frac{K}{s(s+5)^2}$$

The series controller has the transfer function

$$G_c(s) = \frac{1+aTs}{1+Ts}$$

(a) Design a phase-lead controller ($a > 1$) so that the following performance specifications are satisfied:

Ramp-error constant $K_v = 10$

Maximum overshoot is near minimum

The value of a must not exceed 1000. Plot the unit-step response and give its attributes.

(b) Design a phase-lead controller in the frequency domain so that the following performance specifications are satisfied:

Ramp—error constant $k_v = 10$

Phase margin is near maximum.

The value of a must not exceed 1000.

(c) Design a phase-lag controller ($a < 1$) so that the following performance specifications are satisfied:

Ramp-error constant $K_v = 10$

Maximum overshoot < 1%

Rise time $t_s < 2$ s

Settling time $t_r < 2.5$ s

Find the PM, GM, M_r, and BW of the designed system.

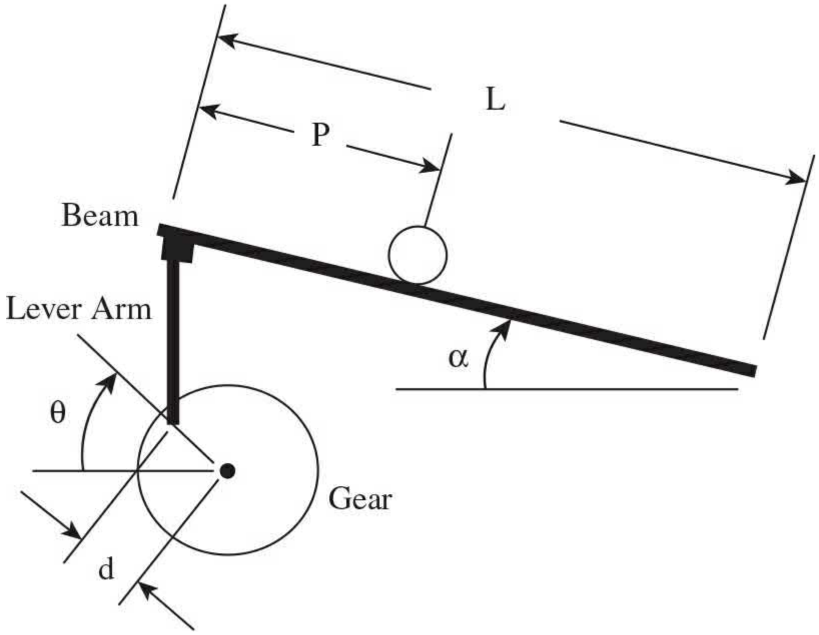
(d) Design the phase-lag controller in the frequency domain so that the following performance specifications are satisfied:

Ramp-error constant $K_v = 10$

Phase margin > 70°

Check the unit-step response attributes of the designed system and compare with those obtained in part (c).

11-51. Figure 11P-51 shows the “beam and ball” system that is described in Prob. 4-11.



Assuming:

$m = 0.1$ kg	mass of the ball
$r = 0.015$	radius of the ball
$d = 0.03$ m	lever arm offset
$g = 9.8$ m/s^2	gravitational acceleration
$L = 1.0$ m	length of the beam
$I = 9.99\text{e} -6$ kgm^2	ball's moment of inertia
P	ball position coordinate
α	beam angle coordinate
θ	servo gear angle

Design a PID controller so that the settling time is less than 3 s and the maximum overshoot is no more than 5 percent.

11-52. The controlled process of a dc-motor control system with unity feedback has the transfer function

$$G_p(s) = \frac{6.087 \times 10^{10}}{s(s^3 + 423.42s^2 + 2.6667 \times 10^6\ s + 4.2342 \times 10^8)}$$

Due to the compliance in the motor shaft, the process transfer function contains two lightly damped poles, which will cause oscillations in the output response. The following performance criteria are to be satisfied:

Maximum overshoot < 1%

Rise time $t_r < 0.15$ s

Settling time $t_s < 0.15$ s

Output response should not have oscillations

Ramp-error constant is not affected

(a) Design a series phase-lead controller,

$$G_c(s) = \frac{1 + aTs}{1 + Ts} \quad a > 1$$

so that all the step-response attributes (except for the oscillations) are satisfied.

(b) To eliminate the oscillations due to the motor shaft compliance, add another stage to the controller with the transfer function

$$G_{c1}(s) = \frac{s^2 + 2\zeta_z \omega_n s + \omega_n^2}{s^2 + 2\zeta_p \omega_n s + \omega_n^2}$$

so that the zeros of $G_{c1}(s)$ will cancel the two complex poles of $G_p(s)$. Set the value of ζ_p so that the two poles of $G_{c1}(s)$ will not have an appreciable effect on the system response. Determine the attributes of the unit-step response to see if all the requirements are satisfied. Plot the unit-step responses of the uncompensated system the compensated system with the phase-lead controller designed in part (b).

11-53. A computer-tape-drive system utilizing a permanent-magnet dc-motor is shown in Fig. 11P-53*a*. The closed-loop system is modeled by the block diagram in Fig. 11P-53*b*. The constant K_L represents the spring constant of the elastic tape, and B_L denotes the viscous-friction coefficient between the tape and the capstans. The system parameters are as follows:

K_t = motor torque contant = 10 oz-in./A

K_b = motor back-emf constant 0.0706 V/rad/s

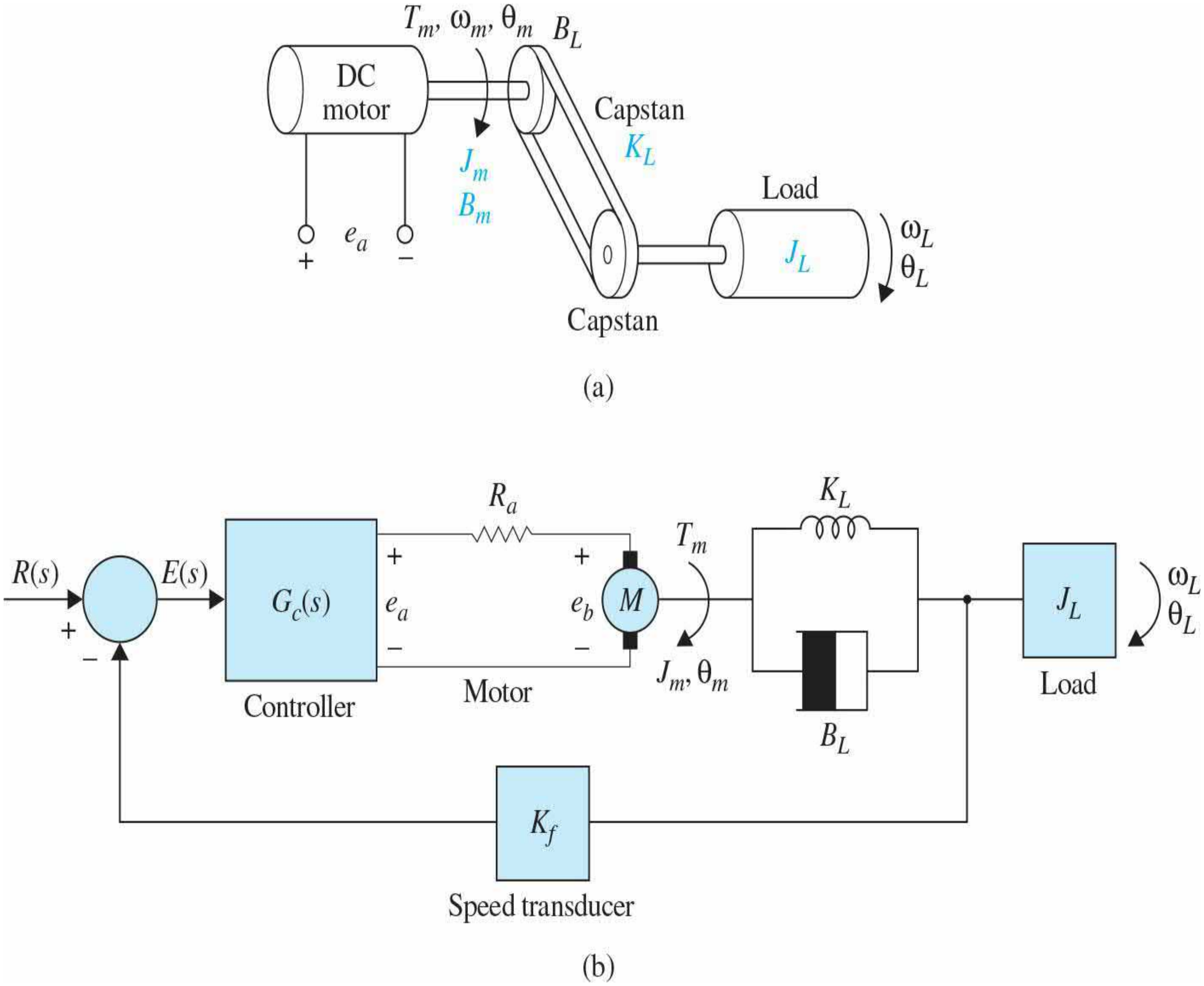
B_m = motor friction coefficient = 3 oz-in./rad/s

$R_a = 0.25\ \Omega\ L_a \cong 0$ H

$K_L = 3000$ oz-in./rad/B_L 10 oz-in./rad/s

$J_L = 6$ oz-in./rad/s^2 K_f 1 V/rad/s

$J_m = 0.05$ oz-in./rad/s^2



(a)

(b)

Figure 11P-53

(a) Write the state equations of the system between e_a and θ_L using θ_L, ω_L, θ_m, and ω_m as state variables and e_a as input. Draw a state diagram using the state equations. Derive the transfer functions:

$$\frac{\Omega_m(s)}{E_a(s)} \quad \text{and} \quad \frac{\Omega_L(s)}{E_a(s)}$$

(b) The objective of the system is to control the speed of the tape, ω_L, accurately. Consider that a PI controller with the transfer function $G_c(s) = K_P + K_I/s$ is to be used. Find the values of K_P and K_I so that the following specifications are satisfied:

Ramp-error constant $K_v = 100$

Rise time < 0.02 s

Settling time < 0.02 s

Maximum overshoot < 1% or at minimun

Plot the unit-step response of $\omega_L(t)$ of the system.

(c) Design the PI controller in the frequency domain. The value of K_I is to be selected as in part (b). Vary the value of K_P and compute the values of PM, GM, M_r, and BW. Find the value of K_P so that PM is maximum. How does this value of K_P compare with the result obtained in part (b)?

11-54. Figure 11P-54 shows the block diagram of a motor-control system that has a flexible shaft between the motor and the load. The transfer function between the motor torque and motor displacement is

$$G_p(s) = \frac{\Theta_m(s)}{T_m(s)} = \frac{J_L s^2 + B_L s + K_L}{s\left[J_m J_L s^3 + (B_m J_L + B_L J_m)s^2 + (K_L J_m + K_L J_L + B_m B_L)s + B_m K_L\right]}$$

where $J_L = 0.01$, $B_L = 0.1$, $K_L = 10$, $J_m = 0.01$, $B_m = 0.1$, and $K = 100$

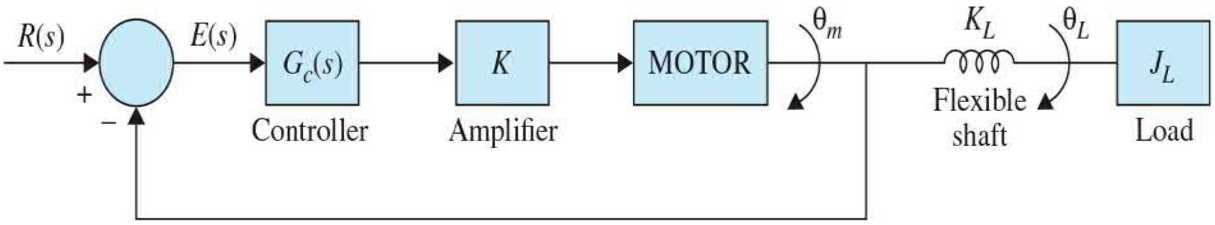


Figure 11P-54

(a) Compute and plot the unit-step response of $\theta_m(t)$. Find the attributes of the unit-step response.

(b) Design a second-order notch controller with the transfer function

$$G_c(s) = \frac{s^2 + 2\zeta_z \omega_n s + \omega_n^2}{s^2 + 2\zeta_p \omega_n s + \omega_n^2}$$

so that its zeros cancel the complex poles of $G_p(s)$. The two poles of $G_c(s)$ should be selected so that they do not affect the steady-state response of the system, and the maximum overshoot is a minimum. Compute the attributes of the unit-step response and plot the response.

(c) Carry out design of the second-order controller in the frequency domain. Plot the Bode diagram of the uncompensated $G_p(s)$, and find the values of PM, GM, M_r, and BW. Set the two zeros of $G_c(s)$ to cancel the two complex poles of $G_p(s)$. Determine the value of ζ_p by determining the amount of attenuation required from the second-order notch controller and using Eq. (11-155). Find the PM, GM, M_r, and BW of the compensated system. How do the frequency-domain design results compare with the results in part (b)?

11-55. The transfer function of the process of a unity-feedback control system is

$$G_p(s) = \frac{500(s+10)}{s(s^2 + 10s + 1000)}$$

(a) Plot the Bode diagram of $G_p(s)$ and determine the PM, GM, M_r, and BW of the uncompensated system. Compute and plot the unit-step response of the system.

(b) Design a series second-order notch controller with the transfer function

$$G_c(s) = \frac{s^2 + 2\zeta_z \omega_n s + \omega_n^2}{s^2 + 2\zeta_p \omega_n s + \omega_n^2}$$

so that its zeros cancel the complex poles of $G_p(s)$. Determine the value of ζ_p using the method outlined in Sec. 11-8-2. Find the PM, GM, M_r, and BW of the designed system. Compute and plot the unit-step response.

(c) Design the series second-order notch controller so that its zeros cancel the complex poles of $G_p(s)$. Determine the value of ζ_p so that the following specifications are satisfied:

Maximum overshoot < 1%

Rise time < 0.4 s

Settling time < 0.5 s

11-56. Design the controllers $G_{cf}(s)$ and $G_c(s)$ for the system shown in Fig. 11P-56 so that the following specifications are satisfied:

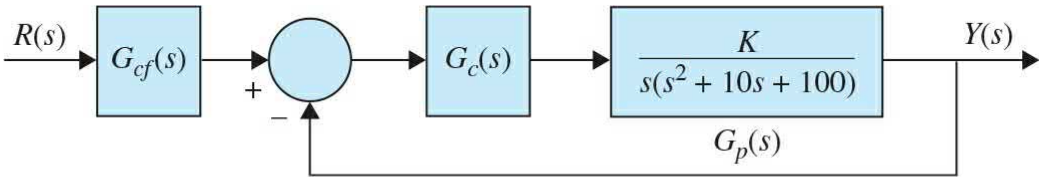


Figure 11P-56

Ramp-error contant $K_v = 50$.

Dominant roots of the characteristic equation at $-5 \pm j5$ approximately

Rise time < 0.01 s

System must be robust when K varies ±20% from the nominal value, with the rise time and overshoot staying within specifications

Compute and plot the unit-step responses to check the design.

11-57. Figure 11P-57 shows the block diagram of a motor-control system. The transfer function of the controlled process is

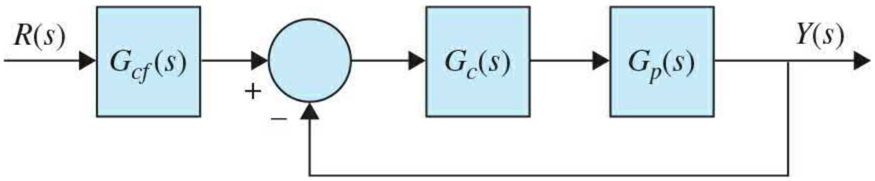


Figure 11P-57

$$G_p(s) = \frac{1000K}{s(s+a)}$$

where K denotes the aggregate of the amplifier gain and motor torque constant, and a is the inverse of the motor time constant. Design the controllers $G_{cf}(s)$ and $G_c(s)$ so that the following performance specifications are satisfied.

Ramp-error constant $K_v = 100$ when $a = 10$

Rise time < 0.3 s

Maximum overshoot < 8%

Dominant characteristic equation roots = $-5 \pm j5$

System must be robust when a varies between 8 and 12.

Compute and plot the unit-step responses to verify the design.

11-58. Figure 11P-58 shows the block diagram of a dc-motor control system with tachometer feedback. Find the values of K and K_t so that the following specifications are satisfied:

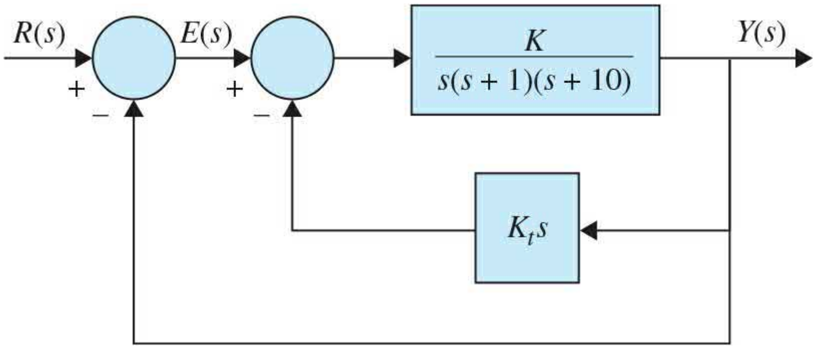


Figure 11P-58

R amp-error constant $K_v = 1$

Dominant characteristic equation roots correspond to a damping ratio of approximately 0.707; if there are two solutions, select the larger value of K.

11-59. Carry out the design with the specifications given in Prob. 11-58 for the system shown in Fig. 11P-59.

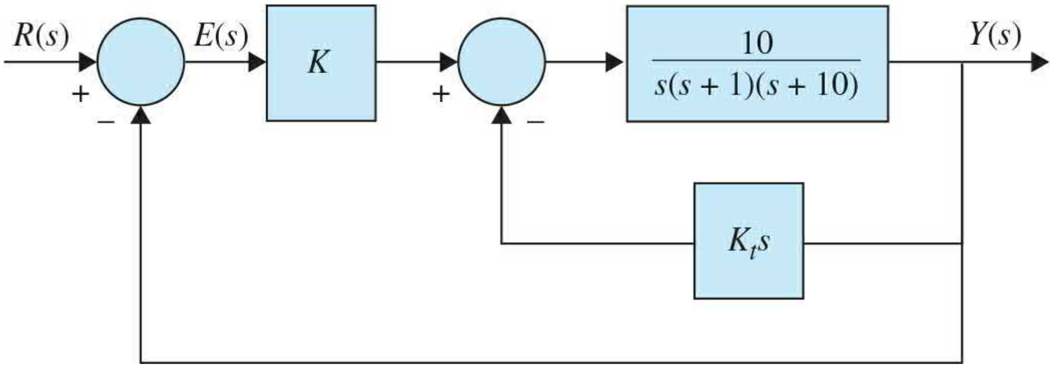


Figure 11P-59

11-60. The block diagram of a control system with a type 2 process is shown in Fig. 11P-60. The system is to be compensated by tachometer feedback and a series controller. Find the values of a, T, K, and K_t so that the following performance specifications are satisfied:

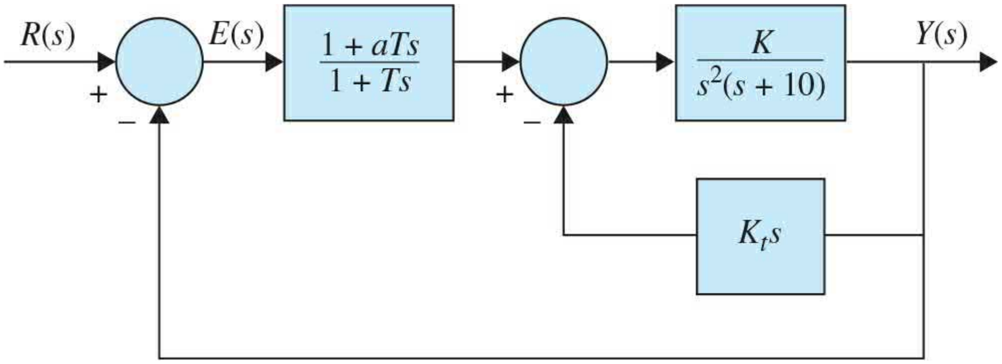


Figure 11P-60

R amp-error constant $K_v = 100$

Dominant characteristic equation roots correspond to a damping ratio of 0.707

11-61. The aircraft-attitude control system described in Sec. 5-8 is modeled by the block diagram shown in Fig. 11P-61. The system parameters are as follows:

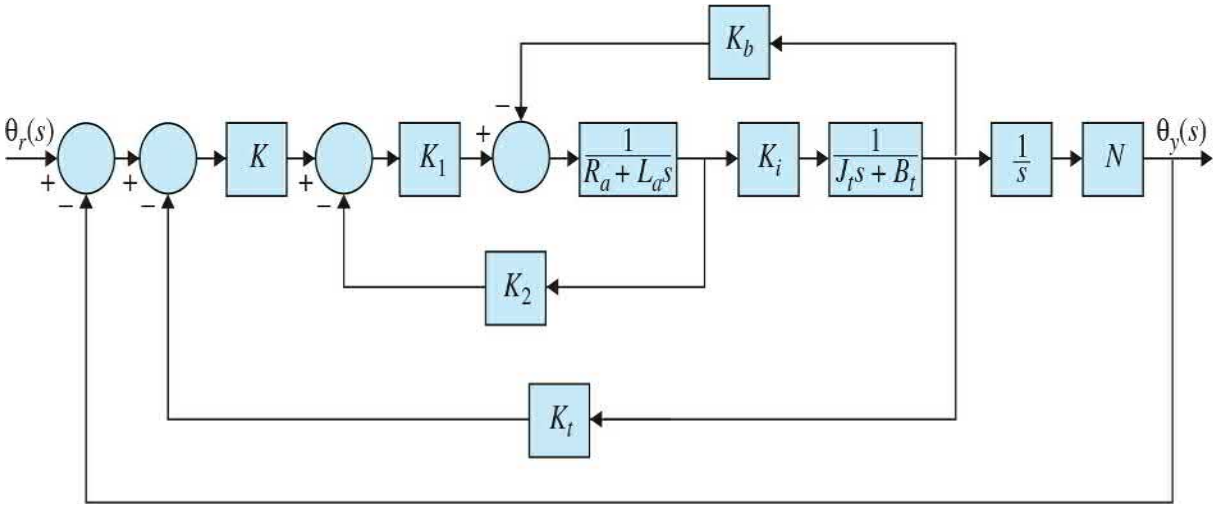


Figure 11P-61

$K = \text{variable}$	$K_s = 1$	$K_1 = 10$	$K_2 = 0.5$	$K_t = \text{variable}$	$R_a = 5$
$L_a = 0.003$	$K_i = 9.0$	$K_b = 0.0636$	$J_m = 0.0001$	$J_L = 0.01$	
$B_m = 0.005$	$B_L = 1.0$	$N = 0.1$			

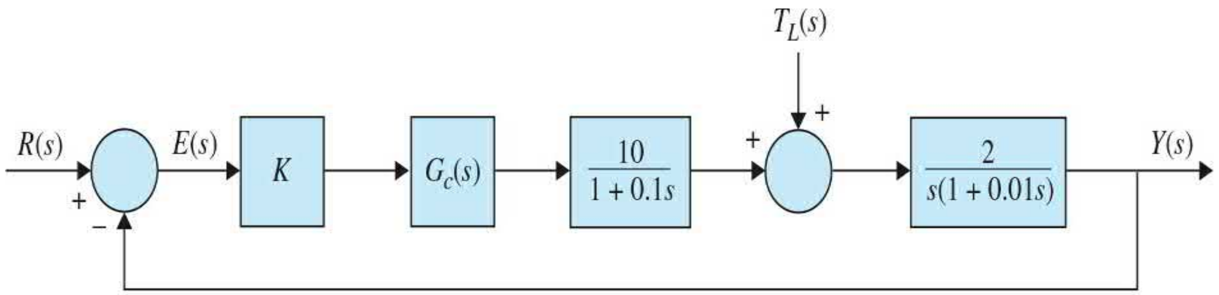
Find the values of K and K_t so that the following specifications are satisfied:

Ramp-error constant $K_v = 100$

Relative damping ratio of the complex roots of the characteristic equation is approximately 0.707

Plot the unit-step response of the designed system. Show that the system performance is extremely insensitive to the value of K. Explain why this is so.

11-62. Figure 11P-62 shows the block diagram of a position-control system with a series controller $G_c(s)$.



controller $G_c(s)$.

Figure 11P-62

(a) Determine the minimum value of the amplifier gain K so that the steady-state value of the output $y(t)$ due to a unit-step torque disturbance is ≤ 0.01.

(b) Show that the uncompensated system is unstable with the minimum value of K determined in part (a). Construct the Bode diagram for the open-loop transfer function $G(s) = Y(s)/E(s)$, and find the values of PM and GM.

(c) Design a single-stage phase-lead controller with the transfer function

$$G_c(s) = \frac{1+aTs}{1+Ts} \quad a > 1$$

so that the phase margin is 30°. Show that this is nearly the highest phase margin that can be achieved with a single-stage phase-lead controller. Find GM, M_r, and BW of the compensated system.

(d) Design a two-stage phase-lead controller using the system arrived at in part (c) as a basis so that the phase margin is 55°. Show that this is the best PM that can be obtained for this system with a two-stage phase-lead controller. Find GM, M_r, and BW of the compensated system.

11-63. The transfer function of the process of a unity-feedback control system is

$$G_c(s) = \frac{60}{s(1+0.2s)(1+0.5s)}$$

Show that, due to the relative high gain, the uncompensated system is unstable.

(a) Design a two-stage phase-lead controller with

$$G_c(s) = \left(\frac{1+aT_1 s}{1+T_1 s}\right)\left(\frac{1+bT_2 s}{1+T_2 s}\right) \quad a>1, \quad b>1$$

so that the phase margin is greater than 60°. Conduct the design by first determining the values of *a* and T_1 to realize a maximum phase margin that can be achieved with a single-stage phase-lead controller. The second stage of the controller is then designed to realize the balance of the 60° phase margin. Determine GM, M_r, and BW of the compensated system. Compute and plot the unit-step response of the compensated system.

(b) Design a single-stage phase-lag controller with

$$G_c(s) = \frac{1+aTs}{1+Ts} \quad a<1$$

so that the phase margin of the compensated system is greater than 60°. Determine GM, M_r, and BW of the compensated system. Compute and plot the unit-step response of the compensated system.

(c) Design a lag-lead controller with $G_c(s)$ as in the equation in part (a). Design the phase-lag portion first by setting the phase margin at 40°. The resulting system is then compensated by the phase-lead portion to achieve a total of 60° of phase margin. Determine GM, M_r, and BW of the compensated system. Compute and plot the unit-step response of the compensated system.

11-64. The block diagram of the steel-rolling system described in Prob. 4-18 is shown in Fig. 11P-64. The transfer function of the process is

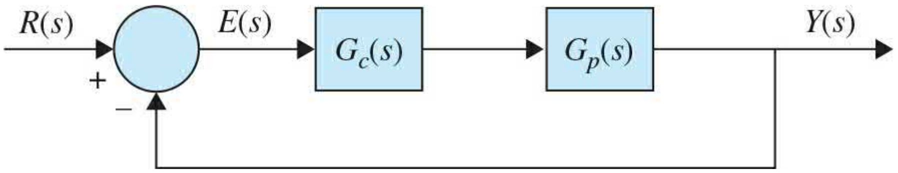


Figure 11P-64

$$G_p(s) = \frac{Y(s)}{E(s)} = \frac{5e^{-0.1s}}{s(1+0.1s)(1+0.5s)} \qquad K_s = 1$$

(a) Approximate the time delay by

$$e^{-0.1s} \frac{1-0.05s}{1+0.05s}$$

Design a series controller of your choice so that the phase margin of the compensated system is at least 60°. Determine GM, M_r, and BW of the compensated system. Compute and plot the unit-step responses of the compensated and the uncompensated systems.

(b) Repeat part (a) without using the approximation of the time delay.

11-65. Human beings breathe in order to provide for gas exchange for the entire body. A respiratory control system is needed to ensure that the body's needs for this gas exchange are adequately met. The criterion of control is adequate ventilation, which ensures satisfactory levels of both oxygen and carbon dioxide in the arterial blood. Respiration is controlled by neural impulses that originate within the lower brain and are transmitted to the chest cavity and diaphragm to govern the rate and tidal volume. One source of signals consists of the chemoreceptors located near the respiratory center, which are sensitive to carbon dioxide and oxygen concentrations. Figure 11P-65 shows the block diagram of a simplified model of the human respiratory control system. The objective is to control the effective ventilation of the lungs so that a satisfactory balance of concentrations of carbon dioxide and oxygen is maintained in the blood circulated at the chemoreceptor.

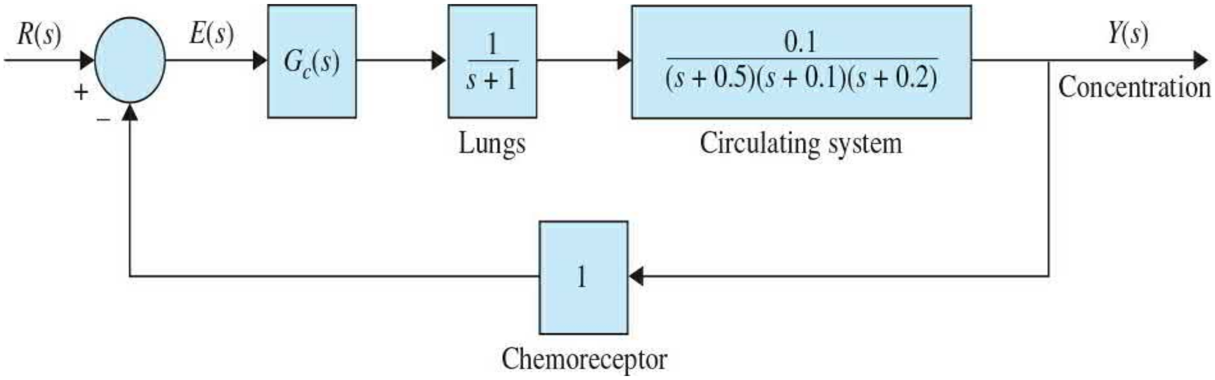


Figure 11P-65

(a) Plot the Bode diagram of the transfer function $G(s) = Y(s)/E(s)$ when $G(s) = 1$. Find the PM and GM. Determine the stability of the system.

(b) Design a PI controller, $G_c(s) = K_P + K_I/s$, so that the following specifications are satisfied:

Ramp-error constant $K_v = 1$

Phase margin is maximized

Plot the unit-step response of the system. Find the attributes of the unit-step response.

(c) Design a PI controller so that the following specifications are satisfied:

Ramp-error constant $K_v = 1$

Maximum overshoot is minimized

Plot the unit-step response of the system. Find the attributes of the unit-step response. Compare the design results in parts (b) and (c).

11-66. The block diagram of a control system with state feedback is shown in Fig. 11P-66. Find the real feedback gains k_1, k_2, and k_3 so that

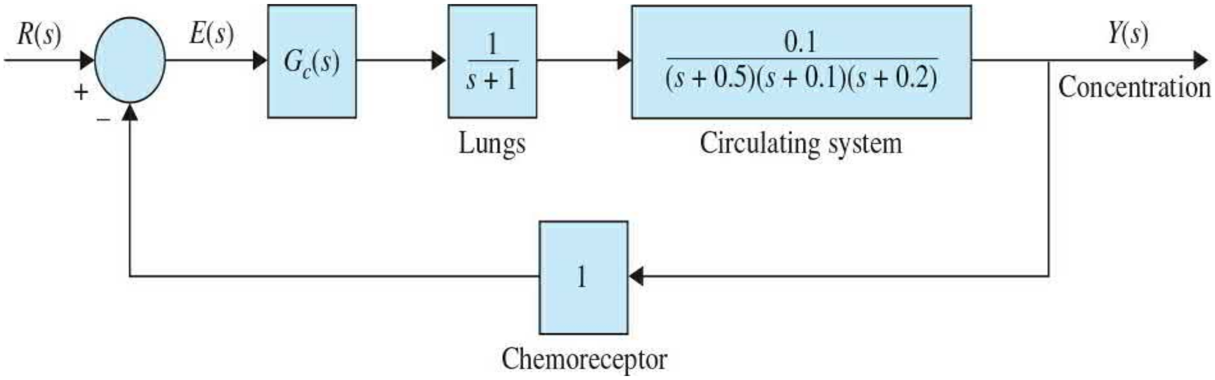


Figure 11P-66

The steady-state error e_{ss} [$e(t)$ is the error signal] due to a step input is zero.

The complex roots of the characteristic equation are at $-1+j$ and $-1-j$.

Find the third root. Can all three roots be arbitrarily assigned while still meeting the steady-state requirement?

11-67. The block diagram of a control system with state feedback is shown in Fig. 11P-67*a*. The feedback gains k_1, k_2, and k_3 are real constants.

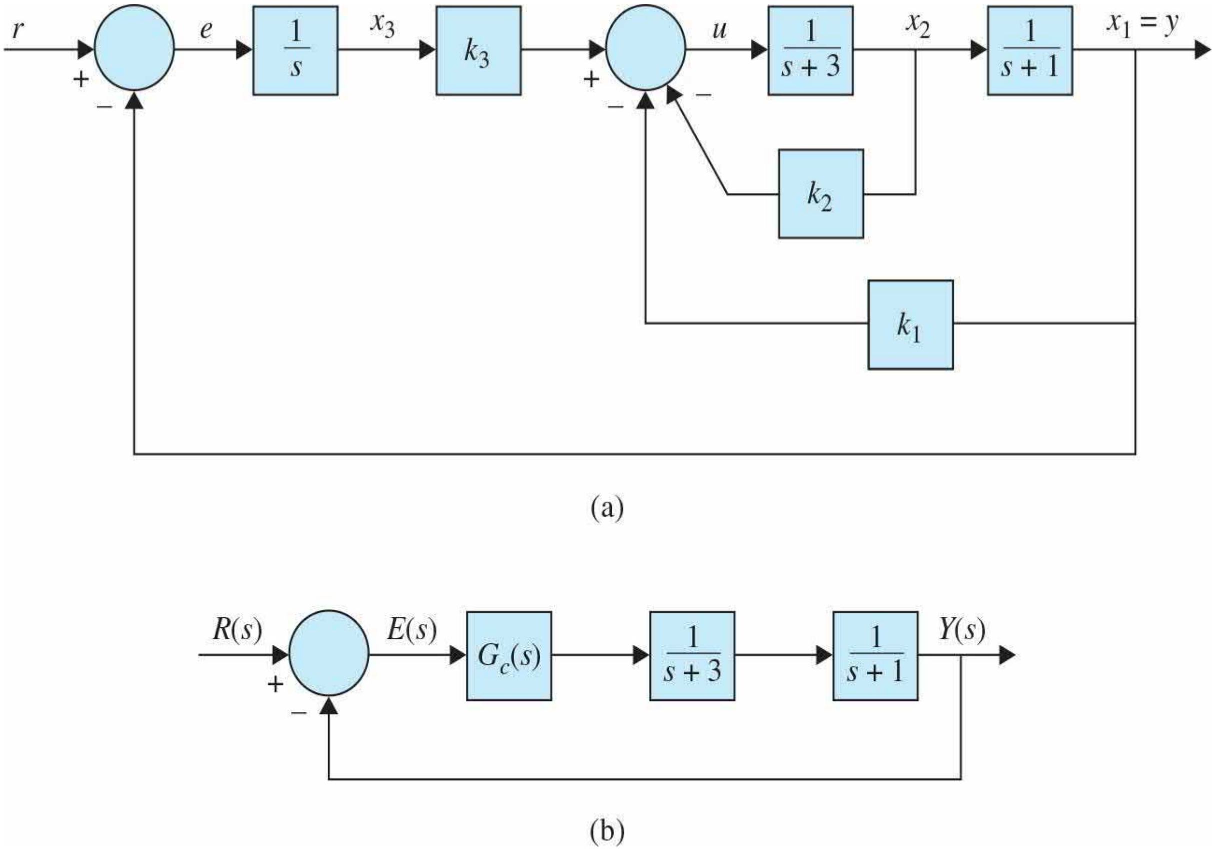


Figure 11P-67

(a) Find the values of the feedback gains so that

The steady-state error e_{ss} [$e(t)$ is the error signal] due to a step input is zero.

The characteristic equation roots are at $-1+j$, $-1-j$, and -10.

(b) Instead of using state feedback, a series controller is implemented, as shown in Fig. 11P-67*b*. Find the transfer function of the controller $G_c(s)$ in terms of k_1, k_2, and k_3 found in part (a) and the other system parameters.

TERM PROJECT

11-68. Develop a proportional, PD, PI, and PID controller for the simple LEGO MINDSTORMS robotic arm discussed in Chap. 8.

PD performance specifications as follows:

Settling time $t_s \leq 0.3$ s

Maximum overshoot ≤ 5 percent

Steady-state error due to unit-ramp input ≤ 0.05

PI performance specifications as follows:

Settling time $t_s \leq 1.5$ s

Rise time $t_r \leq 0.3$ s

Maximum overshoot ≤ 10 percent

Steady-state error due to parabolic input ≤ 0.7

PID performance requirements are as follows:

Rise time $t_r \leq 0.3$ s

Settling time $t_s \leq 1.5$ s

Maximum overshoot ≤ 5 percent

Steady-state error due to parabolic input ≤ 0.7

Please see App. D and the solution manual. (Note: The controller design process is not unique.)

[1] This example has also been solved using the MATLAB SISO design tool. See Example 11-10-1.

[2] For the MATLAB SISO design tool implementation, see Example 11-10-2.

[3] For the MATLAB SISO design tool implementation, see Example 11-10-3.

[4] A comprehensive description of the SISO design tool with examples appears in http://www.mathworks.com/help/control/ug/overview-of-the-siso-design-tool.html.

Index

Please note that index links point to page beginnings from the print edition. Locations are approximate in e-readers, and you may need to page down one or more times after clicking a link to get to the indexed material.

Note: Page numbers followed by *f* indicate figures; those followed by *t* indicate tables.

Table of Contents