Notice that the numerator of the closed-loop transfer function $C(s)/R(s)$ is the product of the transfer functions of the feedforward path. The denominator of $C(s)/R(s)$ is equal to

$$1 + \sum \text{(product of the transfer functions around each loop)}$$

$$= 1 + \left(-G_1G_2H_1 + G_2G_3H_2 + G_1G_2G_3\right)$$

$$= 1 - G_1G_2H_1 + G_2G_3H_2 + G_1G_2G_3$$

(The positive feedback loop yields a negative term in the denominator.)

2–4 MODELING IN STATE SPACE

In this section we shall present introductory material on state-space analysis of control systems.

Modern Control Theory. The modern trend in engineering systems is toward greater complexity, due mainly to the requirements of complex tasks and good accuracy. Complex systems may have multiple inputs and multiple outputs and may be time varying. Because of the necessity of meeting increasingly stringent requirements on the performance of control systems, the increase in system complexity, and easy access to large scale computers, modern control theory, which is a new approach to the analysis and design of complex control systems, has been developed since around 1960. This new approach is based on the concept of state. The concept of state by itself is not new, since it has been in existence for a long time in the field of classical dynamics and other fields.

Modern Control Theory Versus Conventional Control Theory. Modern control theory is contrasted with conventional control theory in that the former is applicable to multiple-input, multiple-output systems, which may be linear or nonlinear, time invariant or time varying, while the latter is applicable only to linear time-invariant single-input, single-output systems. Also, modern control theory is essentially time-domain approach and frequency domain approach (in certain cases such as H-infinity control), while conventional control theory is a complex frequency-domain approach. Before we proceed further, we must define state, state variables, state vector, and state space.

State. The state of a dynamic system is the smallest set of variables (called *state variables*) such that knowledge of these variables at $t = t_0$, together with knowledge of the input for $t \geq t_0$, completely determines the behavior of the system for any time $t \geq t_0$.

Note that the concept of state is by no means limited to physical systems. It is applicable to biological systems, economic systems, social systems, and others.

State Variables. The state variables of a dynamic system are the variables making up the smallest set of variables that determine the state of the dynamic system. If at least n variables $x_1, x_2, \ldots, x_n$ are needed to completely describe the behavior of a dynamic system (so that once the input is given for $t \geq t_0$ and the initial state at $t = t_0$ is specified, the future state of the system is completely determined), then such n variables are a set of state variables.

Note that state variables need not be physically measurable or observable quantities. Variables that do not represent physical quantities and those that are neither measurable nor observable can be chosen as state variables. Such freedom in choosing state variables is an advantage of the state-space methods. Practically, however, it is convenient to choose easily measurable quantities for the state variables, if this is possible at all, because optimal control laws will require the feedback of all state variables with suitable weighting.

State Vector. If n state variables are needed to completely describe the behavior of a given system, then these n state variables can be considered the n components of a vector $\mathbf{x}$. Such a vector is called a *state vector*. A state vector is thus a vector that determines uniquely the system state $\mathbf{x}(t)$ for any time $t \geq t_0$, once the state at $t = t_0$ is given and the input $u(t)$ for $t \geq t_0$ is specified.

State Space. The n-dimensional space whose coordinate axes consist of the x_1 axis, x_2 axis, $\ldots, x_n$ axis, where $x_1, x_2, \ldots, x_n$ are state variables, is called a *state space*. Any state can be represented by a point in the state space.

State-Space Equations. In state-space analysis we are concerned with three types of variables that are involved in the modeling of dynamic systems: input variables, output variables, and state variables. As we shall see in Section 2–5, the state-space representation for a given system is not unique, except that the number of state variables is the same for any of the different state-space representations of the same system.

The dynamic system must involve elements that memorize the values of the input for $t \geq t_1$. Since integrators in a continuous-time control system serve as memory devices, the outputs of such integrators can be considered as the variables that define the internal state of the dynamic system. Thus the outputs of integrators serve as state variables. The number of state variables to completely define the dynamics of the system is equal to the number of integrators involved in the system.

Assume that a multiple-input, multiple-output system involves n integrators. Assume also that there are r inputs $u_1(t), u_2(t), \ldots, u_r(t)$ and m outputs $y_1(t), y_2(t), \ldots, y_m(t)$. Define n outputs of the integrators as state variables: $x_1(t), x_2(t), \ldots, x_n(t)$ Then the system may be described by

$$
\begin{aligned}
\dot{x}_1(t) &= f_1(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t) \\
\dot{x}_2(t) &= f_2(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t) \\
&\cdot \\
&\cdot \\
&\cdot \\
\dot{x}_n(t) &= f_n(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t)
\end{aligned}
\tag{2–8}
$$

The outputs $y_1(t), y_2(t), \ldots, y_m(t)$ of the system may be given by

$$\begin{aligned} y_1(t) &= g_1(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t) \\ y_2(t) &= g_2(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t) \\ &\cdot \\ &\cdot \\ &\cdot \\ y_m(t) &= g_m(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t) \end{aligned} \qquad (2\text{–}9)$$

If we define

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \cdot \\ \cdot \\ \cdot \\ x_n(t) \end{bmatrix}, \qquad \mathbf{f}(\mathbf{x}, \mathbf{u}, t) = \begin{bmatrix} f_1(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t) \\ f_2(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t) \\ \cdot \\ \cdot \\ \cdot \\ f_n(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t) \end{bmatrix},$$

$$\mathbf{y}(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \cdot \\ \cdot \\ \cdot \\ y_m(t) \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}, \mathbf{u}, t) = \begin{bmatrix} g_1(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t) \\ g_2(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t) \\ \cdot \\ \cdot \\ \cdot \\ g_m(x_1, x_2, \ldots, x_n; u_1, u_2, \ldots, u_r; t) \end{bmatrix}, \qquad \mathbf{u}(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \cdot \\ \cdot \\ \cdot \\ u_r(t) \end{bmatrix}$$

then Equations (2–8) and (2–9) become

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \qquad (2\text{–}10)$$

$$\mathbf{y}(t) = \mathbf{g}(\mathbf{x}, \mathbf{u}, t) \qquad (2\text{–}11)$$

where Equation (2–10) is the state equation and Equation (2–11) is the output equation. If vector functions $\mathbf{f}$ and/or $\mathbf{g}$ involve time t explicitly, then the system is called a time-varying system.

If Equations (2–10) and (2–11) are linearized about the operating state, then we have the following linearized state equation and output equation:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) \qquad (2\text{–}12)$$

$$\mathbf{y}(t) = \mathbf{C}(t)\mathbf{x}(t) + \mathbf{D}(t)\mathbf{u}(t) \qquad (2\text{–}13)$$

where $\mathbf{A}(t)$ is called the state matrix, $\mathbf{B}(t)$ the input matrix, $\mathbf{C}(t)$ the output matrix, and $\mathbf{D}(t)$ the direct transmission matrix. (Details of linearization of nonlinear systems about

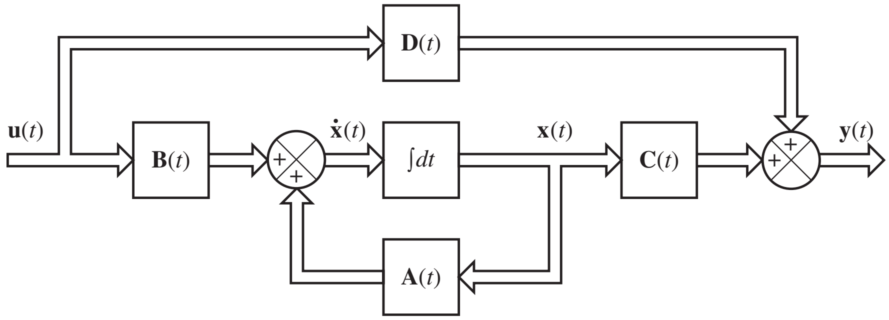

Figure 2–14 Block diagram of the linear, continuous-time control system represented in state space.

the operating state are discussed in Section 2–7.) A block diagram representation of Equations (2–12) and (2–13) is shown in Figure 2–14.

If vector functions **f** and **g** do not involve time t explicitly then the system is called a time-invariant system. In this case, Equations (2–12) and (2–13) can be simplified to

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{2–14}$$

$$\dot{\mathbf{y}}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \tag{2–15}$$

Equation (2–14) is the state equation of the linear, time-invariant system and Equation (2–15) is the output equation for the same system. In this book we shall be concerned mostly with systems described by Equations (2–14) and (2–15).

In what follows we shall present an example for deriving a state equation and output equation.

EXAMPLE 2–2 Consider the mechanical system shown in Figure 2–15. We assume that the system is linear. The external force $u(t)$ is the input to the system, and the displacement $y(t)$ of the mass is the output. The displacement $y(t)$ is measured from the equilibrium position in the absence of the external force. This system is a single-input, single-output system.

From the diagram, the system equation is

$$m\ddot{y} + b\dot{y} + ky = u \tag{2–16}$$

This system is of second order. This means that the system involves two integrators. Let us define state variables $x_1(t)$ and $x_2(t)$ as

$$x_1(t) = y(t)$$

$$x_2(t) = \dot{y}(t)$$

Then we obtain

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = \frac{1}{m}\left(-ky - b\dot{y}\right) + \frac{1}{m}u$$

or

$$\dot{x}_1 = x_2 \tag{2–17}$$

$$\dot{x}_2 = -\frac{k}{m}x_1 - \frac{b}{m}x_2 + \frac{1}{m}u \tag{2–18}$$

The output equation is

$$y = x_1 \tag{2–19}$$

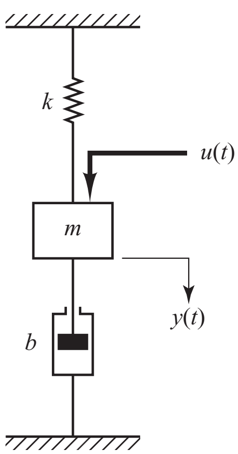

Figure 2–15 Mechanical system.

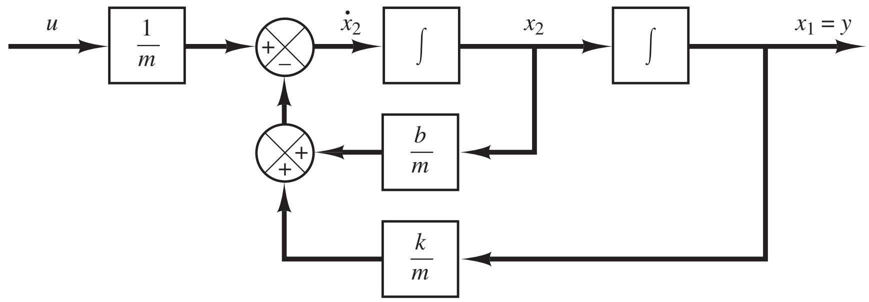

Figure 2–16 Block diagram of the mechanical system shown in Figure 2–15.

In a vector-matrix form, Equations (2–17) and (2–18) can be written as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\dfrac{k}{m} & -\dfrac{b}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \dfrac{1}{m} \end{bmatrix} u \tag{2–20}$$

The output equation, Equation (2–19), can be written as

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{2–21}$$

Equation (2–20) is a state equation and Equation (2–21) is an output equation for the system. They are in the standard form:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$
$$y = \mathbf{C}\mathbf{x} + Du$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -\dfrac{k}{m} & -\dfrac{b}{m} \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 0 \\ \dfrac{1}{m} \end{bmatrix}, \qquad \mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}, \qquad D = 0$$

Figure 2–16 is a block diagram for the system. Notice that the outputs of the integrators are state variables.

Correlation Between Transfer Functions and State-Space Equations. In what follows we shall show how to derive the transfer function of a single-input, single-output system from the state-space equations.

Let us consider the system whose transfer function is given by

$$\frac{Y(s)}{U(s)} = G(s) \tag{2–22}$$

This system may be represented in state space by the following equations:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u \tag{2–23}$$

$$y = \mathbf{C}\mathbf{x} + Du \tag{2–24}$$

where $\mathbf{x}$ is the state vector, u is the input, and y is the output. The Laplace transforms of Equations (2–23) and (2–24) are given by

$$s\mathbf{X}(s) - \mathbf{x}(0) = \mathbf{A}\mathbf{X}(s) + \mathbf{B}U(s) \tag{2–25}$$

$$Y(s) = \mathbf{C}\mathbf{X}(s) + DU(s) \tag{2–26}$$

Since the transfer function was previously defined as the ratio of the Laplace transform of the output to the Laplace transform of the input when the initial conditions were zero, we set $\mathbf{x}(0)$ in Equation (2–25) to be zero. Then we have

$$s\mathbf{X}(s) - \mathbf{A}\mathbf{X}(s) = \mathbf{B}U(s)$$

or

$$(s\mathbf{I} - \mathbf{A})\mathbf{X}(s) = \mathbf{B}U(s)$$

By premultiplying $(s\mathbf{I} - \mathbf{A})^{-1}$ to both sides of this last equation, we obtain

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}U(s) \tag{2–27}$$

By substituting Equation (2–27) into Equation (2–26), we get

$$Y(s) = \left[\mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + D\right]U(s) \tag{2–28}$$

Upon comparing Equation (2–28) with Equation (2–22), we see that

$$G(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + D \tag{2–29}$$

This is the transfer-function expression of the system in terms of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and D.

Note that the right-hand side of Equation (2–29) involves $(s\mathbf{I} - \mathbf{A})^{-1}$. Hence $G(s)$ can be written as

$$G(s) = \frac{Q(s)}{|s\mathbf{I} - \mathbf{A}|}$$

where $Q(s)$ is a polynomial in s. Notice that $|s\mathbf{I} - \mathbf{A}|$ is equal to the characteristic polynomial of $G(s)$. In other words, the eigenvalues of $\mathbf{A}$ are identical to the poles of $G(s)$.

EXAMPLE 2–3 Consider again the mechanical system shown in Figure 2–15. State-space equations for the system are given by Equations (2–20) and (2–21). We shall obtain the transfer function for the system from the state-space equations.

By substituting $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and D into Equation (2–29), we obtain

$$\begin{aligned} G(s) &= \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + D \\ &= [1 \quad 0]\left\{\begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{bmatrix}\right\}^{-1}\begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} + 0 \\ &= [1 \quad 0]\begin{bmatrix} s & -1 \\ \frac{k}{m} & s + \frac{b}{m} \end{bmatrix}^{-1}\begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} \end{aligned}$$

Note that

$$\begin{bmatrix} s & -1 \\ \frac{k}{m} & s + \frac{b}{m} \end{bmatrix}^{-1} = \frac{1}{s^2 + \frac{b}{m}s + \frac{k}{m}} \begin{bmatrix} s + \frac{b}{m} & 1 \\ -\frac{k}{m} & s \end{bmatrix}$$

(Refer to Appendix C for the inverse of the 2 × 2 matrix.)
Thus, we have

$$G(s) = [1 \quad 0] \frac{1}{s^2 + \frac{b}{m}s + \frac{k}{m}} \begin{bmatrix} s + \frac{b}{m} & 1 \\ -\frac{k}{m} & s \end{bmatrix} \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}$$

$$= \frac{1}{ms^2 + bs + k}$$

which is the transfer function of the system. The same transfer function can be obtained from Equation (2–16).

Transfer Matrix. Next, consider a multiple-input, multiple-output system. Assume that there are r inputs $u_1, u_2, \ldots, u_r$, and m outputs $y_1, y_2, \ldots, y_m$. Define

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ \cdot \\ y_m \end{bmatrix}, \qquad \mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ \cdot \\ \cdot \\ u_r \end{bmatrix}$$

The transfer matrix $\mathbf{G}(s)$ relates the output $\mathbf{Y}(s)$ to the input $\mathbf{U}(s)$, or

$$\mathbf{Y}(s) = \mathbf{G}(s)\mathbf{U}(s)$$

where $\mathbf{G}(s)$ is given by

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}$$

[The derivation for this equation is the same as that for Equation (2–29).] Since the input vector $\mathbf{u}$ is r dimensional and the output vector $\mathbf{y}$ is m dimensional, the transfer matrix $\mathbf{G}$(s) is an $m \times r$ matrix.

2–5 STATE-SPACE REPRESENTATION OF SCALAR DIFFERENTIAL EQUATION SYSTEMS

A dynamic system consisting of a finite number of lumped elements may be described by ordinary differential equations in which time is the independent variable. By use of vector-matrix notation, an nth-order differential equation may be expressed by a first-order vector-matrix differential equation. If n elements of the vector are a set of state variables, then the vector-matrix differential equation is a *state* equation. In this section we shall present methods for obtaining state-space representations of continuous-time systems.

State-Space Representation of *n*th-Order Systems of Linear Differential Equations in which the Forcing Function Does Not Involve Derivative Terms. Consider the following *n*th-order system:

$$\overset{(n)}{y} + a_1\overset{(n-1)}{y} + \cdots + a_{n-1}\dot{y} + a_n y = u \tag{2–30}$$

Noting that the knowledge of $y(0), \dot{y}(0), \ldots, \overset{(n-1)}{y}(0)$, together with the input $u(t)$ for $t \geq 0$, determines completely the future behavior of the system, we may take $y(t), \dot{y}(t), \ldots, \overset{(n-1)}{y}(t)$ as a set of n state variables. (Mathematically, such a choice of state variables is quite convenient. Practically, however, because higher-order derivative terms are inaccurate, due to the noise effects inherent in any practical situations, such a choice of the state variables may not be desirable.)

Let us define

$$\begin{aligned} x_1 &= y \\ x_2 &= \dot{y} \\ &\cdot \\ &\cdot \\ &\cdot \\ x_n &= \overset{(n-1)}{y} \end{aligned}$$

Then Equation (2–30) can be written as

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\cdot \\ &\cdot \\ &\cdot \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= -a_n x_1 - \cdots - a_1 x_n + u \end{aligned}$$

or

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u \tag{2–31}$$

where

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ 0 & 0 & 0 & \cdots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \cdots & -a_1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 1 \end{bmatrix}$$

The output can be given by

$$y = [1 \quad 0 \quad \cdots \quad 0] \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix}$$

or

$$y = \mathbf{C}\mathbf{x} \tag{2–32}$$

where

$$\mathbf{C} = [1 \quad 0 \quad \cdots \quad 0]$$

[Note that D in Equation (2–24) is zero.] The first-order differential equation, Equation (2–31), is the state equation, and the algebraic equation, Equation (2–32), is the output equation.

Note that the state-space representation for the transfer function system

$$\frac{Y(s)}{U(s)} = \frac{1}{s^n + a_1 s^{n-1} + \cdots + a_{n-1}s + a_n}$$

is given also by Equations (2–31) and (2–32).

State-Space Representation of *n*th-Order Systems of Linear Differential Equations in which the Forcing Function Involves Derivative Terms. Consider the differential equation system that involves derivatives of the forcing function, such as

$$\overset{(n)}{y} + a_1 \overset{(n-1)}{y} + \cdots + a_{n-1}\dot{y} + a_n y = b_0 \overset{(n)}{u} + b_1 \overset{(n-1)}{u} + \cdots + b_{n-1}\dot{u} + b_n u \tag{2–33}$$

The main problem in defining the state variables for this case lies in the derivative terms of the input u. The state variables must be such that they will eliminate the derivatives of u in the state equation.

One way to obtain a state equation and output equation for this case is to define the following n variables as a set of n state variables:

$$\begin{aligned} x_1 &= y - \beta_0 u \\ x_2 &= \dot{y} - \beta_0 \dot{u} - \beta_1 u = \dot{x}_1 - \beta_1 u \\ x_3 &= \ddot{y} - \beta_0 \ddot{u} - \beta_1 \dot{u} - \beta_2 u = \dot{x}_2 - \beta_2 u \\ &\cdot \\ &\cdot \\ &\cdot \\ x_n &= \overset{(n-1)}{y} - \beta_0 \overset{(n-1)}{u} - \beta_1 \overset{(n-2)}{u} - \cdots - \beta_{n-2}\dot{u} - \beta_{n-1}u = \dot{x}_{n-1} - \beta_{n-1}u \end{aligned} \tag{2–34}$$

where $\beta_0, \beta_1, \beta_2, \ldots, \beta_{n-1}$ are determined from

$$\begin{aligned} \beta_0 &= b_0 \\ \beta_1 &= b_1 - a_1\beta_0 \\ \beta_2 &= b_2 - a_1\beta_1 - a_2\beta_0 \\ \beta_3 &= b_3 - a_1\beta_2 - a_2\beta_1 - a_3\beta_0 \\ &\;\;\vdots \\ \beta_{n-1} &= b_{n-1} - a_1\beta_{n-2} - \cdots - a_{n-2}\beta_1 - a_{n-1}\beta_0 \end{aligned} \tag{2–35}$$

With this choice of state variables the existence and uniqueness of the solution of the state equation is guaranteed. (Note that this is not the only choice of a set of state variables.) With the present choice of state variables, we obtain

$$\begin{aligned} \dot{x}_1 &= x_2 + \beta_1 u \\ \dot{x}_2 &= x_3 + \beta_2 u \\ &\;\;\vdots \\ \dot{x}_{n-1} &= x_n + \beta_{n-1} u \\ \dot{x}_n &= -a_n x_1 - a_{n-1} x_2 - \cdots - a_1 x_n + \beta_n u \end{aligned} \tag{2–36}$$

where β_n is given by

$$\beta_n = b_n - a_1\beta_{n-1} - \cdots - a_{n-1}\beta_1 - a_{n-1}\beta_0$$

[To derive Equation (2–36), see Problem **A–2–6**.] In terms of vector-matrix equations, Equation (2–36) and the output equation can be written as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \cdot \\ \cdot \\ \cdot \\ \dot{x}_{n-1} \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ 0 & 0 & 0 & \cdots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \cdots & -a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_{n-1} \\ x_n \end{bmatrix} + \begin{bmatrix} \beta_1 \\ \beta_2 \\ \cdot \\ \cdot \\ \cdot \\ \beta_{n-1} \\ \beta_n \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} + \beta_0 u$$

or

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u \tag{2–37}$$

$$y = \mathbf{C}\mathbf{x} + Du \tag{2–38}$$

where

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_{n-1} \\ x_n \end{bmatrix}, \qquad \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ 0 & 0 & 0 & \cdots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \cdots & -a_1 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \cdot \\ \cdot \\ \cdot \\ \beta_{n-1} \\ \beta_n \end{bmatrix}, \qquad \mathbf{C} = [1 \quad 0 \quad \cdots \quad 0], \qquad D = \beta_0 = b_0$$

In this state-space representation, matrices **A** and **C** are exactly the same as those for the system of Equation (2–30). The derivatives on the right-hand side of Equation (2–33) affect only the elements of the **B** matrix.

Note that the state-space representation for the transfer function

$$\frac{Y(s)}{U(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_{n-1} s + b_n}{s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n}$$

is given also by Equations (2–37) and (2–38).

There are many ways to obtain state-space representations of systems. Methods for obtaining canonical representations of systems in state space (such as controllable canonical form, observable canonical form, diagonal canonical form, and Jordan canonical form) are presented in Chapter 9.

MATLAB can also be used to obtain state-space representations of systems from transfer-function representations, and vice versa. This subject is presented in Section 2–6.

2–6 TRANSFORMATION OF MATHEMATICAL MODELS WITH MATLAB

MATLAB is quite useful to transform the system model from transfer function to state space, and vice versa. We shall begin our discussion with transformation from transfer function to state space.

Let us write the closed-loop transfer function as

$$\frac{Y(s)}{U(s)} = \frac{\text{numerator polynomial in } s}{\text{denominator polynomial in } s} = \frac{\text{num}}{\text{den}}$$

Once we have this transfer-function expression, the MATLAB command

```
[A,B,C,D] = tf2ss(num,den)
```

will give a state-space representation. It is important to note that the state-space representation for any system is not unique. There are many (infinitely many) state-space representations for the same system. The MATLAB command gives one possible such state-space representation.

Transformation from Transfer Function to State Space Representation. Consider the transfer-function system

$$\frac{Y(s)}{U(s)} = \frac{s}{(s+10)(s^2+4s+16)}$$

$$= \frac{s}{s^3+14s^2+56s+160} \tag{2–39}$$

There are many (infinitely many) possible state-space representations for this system. One possible state-space representation is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -160 & -56 & -14 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ -14 \end{bmatrix} u$$

$$y = [1 \quad 0 \quad 0] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + [0]u$$

Another possible state-space representation (among infinitely many alternatives) is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -14 & -56 & -160 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u \tag{2–40}$$

$$y = [0 \quad 1 \quad 0]\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + [0]u \tag{2–41}$$

MATLAB transforms the transfer function given by Equation (2–39) into the state-space representation given by Equations (2–40) and (2–41). For the example system considered here, MATLAB Program 2–2 will produce matrices **A**, **B**, **C**, and *D*.

MATLAB Program 2–2

```
num = [1    0];
den = [1   14   56   160];
[A,B,C,D] = tf2ss(num,den)

A =

   -14   -56  -160
     1     0     0
     0     1     0

B =

     1
     0
     0

C =

     0     1     0

D =

     0
```

Transformation from State Space Representation to Transfer Function. To obtain the transfer function from state-space equations, use the following command:

[num,den] = ss2tf(A,B,C,D,iu)

iu must be specified for systems with more than one input. For example, if the system has three inputs ($u1$, $u2$, $u3$), then iu must be either 1, 2, or 3, where 1 implies $u1$, 2 implies $u2$, and 3 implies $u3$.

If the system has only one input, then either

[num,den] = ss2tf(A,B,C,D)

or

```
[num,den] = ss2tf(A,B,C,D,1)
```

may be used. For the case where the system has multiple inputs and multiple outputs, see Problem **A–2–12**.

EXAMPLE 2–4 Obtain the transfer function of the system defined by the following state-space equations:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -5 & -25 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 25 \\ -120 \end{bmatrix} u$$

$$y = [1 \quad 0 \quad 0] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

MATLAB Program 2-3 will produce the transfer function for the given system. The transfer function obtained is given by

$$\frac{Y(s)}{U(s)} = \frac{25s + 5}{s^3 + 5s^2 + 25s + 5}$$

MATLAB Program 2–3

```
A = [0   1   0;   0   0   1;   -5   -25   -5];
B = [0; 25; -120];
C = [1   0     0];
D = [0];
[num,den] = ss2tf(A,B,C,D)

num =

   0   0.0000   25.0000  5.0000

den

   1.0000   5.0000   25.0000  5.0000

% ***** The same result can be obtained by entering the following command: *****

[num,den] = ss2tf(A,B,C,D,1)

num =

   0   0.0000   25.0000  5.0000

den =

   1.0000   5.0000   25.0000  5.0000
```

2–7 LINEARIZATION OF NONLINEAR MATHEMATICAL MODELS

Nonlinear Systems. A system is nonlinear if the principle of superposition does not apply. Thus, for a nonlinear system the response to two inputs cannot be calculated by treating one input at a time and adding the results.

Although many physical relationships are often represented by linear equations, in most cases actual relationships are not quite linear. In fact, a careful study of physical systems reveals that even so-called "linear systems" are really linear only in limited operating ranges. In practice, many electromechanical systems, hydraulic systems, pneumatic systems, and so on, involve nonlinear relationships among the variables. For example, the output of a component may saturate for large input signals. There may be a dead space that affects small signals. (The dead space of a component is a small range of input variations to which the component is insensitive.) Square-law nonlinearity may occur in some components. For instance, dampers used in physical systems may be linear for low-velocity operations but may become nonlinear at high velocities, and the damping force may become proportional to the square of the operating velocity.

Linearization of Nonlinear Systems. In control engineering a normal operation of the system may be around an equilibrium point, and the signals may be considered small signals around the equilibrium. (It should be pointed out that there are many exceptions to such a case.) However, if the system operates around an equilibrium point and if the signals involved are small signals, then it is possible to approximate the nonlinear system by a linear system. Such a linear system is equivalent to the nonlinear system considered within a limited operating range. Such a linearized model (linear, time-invariant model) is very important in control engineering.

The linearization procedure to be presented in the following is based on the expansion of nonlinear function into a Taylor series about the operating point and the retention of only the linear term. Because we neglect higher-order terms of the Taylor series expansion, these neglected terms must be small enough; that is, the variables deviate only slightly from the operating condition. (Otherwise, the result will be inaccurate.)

Linear Approximation of Nonlinear Mathematical Models. To obtain a linear mathematical model for a nonlinear system, we assume that the variables deviate only slightly from some operating condition. Consider a system whose input is $x(t)$ and output is $y(t)$. The relationship between $y(t)$ and $x(t)$ is given by

$$y = f(x) \tag{2–42}$$

If the normal operating condition corresponds to $\bar{x}, \bar{y}$, then Equation (2–42) may be expanded into a Taylor series about this point as follows:

$$\begin{aligned} y &= f(x) \\ &= f(\bar{x}) + \frac{df}{dx}(x - \bar{x}) + \frac{1}{2!}\frac{d^2 f}{dx^2}(x - \bar{x})^2 + \cdots \end{aligned} \tag{2–43}$$