functions, and hence this chapter serves as a necessary introduction to Chaps. 8 and 9.

Practical design methods are discussed in Chaps. 8 and 9. In Chap. 8 the performance criterion is an integral involving a quadratic form both in the state variables and in the control. Before proceeding to specific design procedures, the performance index is given a physical interpretation in terms of a desired response. The remainder of the chapter is devoted to frequency-domain design procedures, with the Kalman equation as the starting point. Approximate methods are discussed on both the Bode and root-locus diagrams.

The design criteria of Chap. 9 is not the integral performance index but the desired closed-loop transfer function. Means by which these two criteria are related serve as an introduction to the design procedure known as the H equivalent method. This design procedure features the introduction of series compensation in addition to state-variable feedback to realize the desired specifications.

Actually, there is no fitting place to end the book. Ideally the reader will consider this book as an introduction to modern control theory and to the more extensive works on the subject, published and yet to come.

two

system representation in state-variable form

2.1 Introduction and outline of chapter

This chapter introduces the concept of the state variable and the various means of representing control systems in state-variable form. Each method of state-variable representation results in a system description in terms of n first-order differential equations, as opposed to the usual nth-order equation. A convenient tool for this new system representation is matrix notation, and thus the early portion of this chapter contains a brief discussion of matrix notation and manipulation.

The use of n first-order differential equations to describe the dynamics of a control system is a radical

departure from the standard block-diagram, transfer-function approach. It is a basic hypothesis of this book that the state-variable description is actually more meaningful than the block-diagram representation. Because the state-variable description is as yet less familiar, a "backdoor" approach to the problem of system representation is used in this chapter. It is assumed that the governing differential equations of the various elements making up the control system are known and that the complete system is represented by a transfer function. This is nothing more than another representation of the differential equations of the system.

With the familiar transfer-function representation as a starting point, several methods of system representation in state-variable form are discussed and compared. The relationship between the conventional and the state-variable approach is emphasized.

2.2 System state and state variables

It is important to stress at the outset that the concept of system state is, first of all, a physical concept. However, it is often convenient to describe the behavior of an actual physical system in terms of a mathematical model. Here this mathematical model is assumed to consist of ordinary differential equations which have a unique solution for all inputs and initial conditions. It is in terms of this mathematical model that the *system state*, or simply *state*, is defined.

> ***Definition 2.2-1*** The *state of a system* at any time t_0 is the minimum set of numbers $x_1(t_0), x_2(t_0), \ldots, x_n(t_0)$ which, along with the input to the system for $t \geq t_0$, is sufficient to determine the behavior of the system for all $t \geq t_0$.

In other words, the state of a system represents the minimum amount of information that we need to know about a system at t_0 such that its future behavior can be determined without reference to the input before t_0.

The idea of state is familiar from a knowledge of the physical world and the means of solving the differential equations used to model the physical world. Consider a ball flying through the air. Intuitively we feel that if we know the ball's position and velocity, we also know its future behavior. It is on this basis that an outfielder positions himself to catch a fly ball. Exactly the same information is needed to solve a differential-equation model of the problem. Consider, for example, the

second-order differential equation

$$\ddot{x} + a\dot{x} + bx = f(t)$$

The solution to this equation may be found as the sum of the forced response, due to $f(t)$, and the natural or unforced response, i.e., the solution of the homogeneous equation. However, the solution is not completely specified until initial conditions are given on x and $\dot{x}$. With reference to the definition, n is 2 in this example, and the initial values of x and $\dot{x}$ are the only two numbers necessary to define completely the solution for all time. Of course, the initial instant is completely arbitrary, so that a knowledge of x and $\dot{x}$ at any time, along with subsequent values of the input, completely describes the system.

In the example above, both x and $\dot{x}$ are functions of time; i.e., they are variables. Since they are variables capable of defining the state of the system, they are designated as state variables. In general, an nth-order system is described by a collection of n state variables. Indeed, this may be regarded as the definition of the state variable: a state variable is one of a set of n variables the knowledge of which is sufficient to describe completely the behavior of the system. In a third-order system, for example, a knowledge of x, $\dot{x}$, and $\ddot{x}$ would be sufficient. Clearly a knowledge of x, $\dot{x}$, and $ax + b\dot{x}$ would not be adequate, since the last expression provides no new information about the system.

Modern control theory is almost completely dependent upon a system representation in n first-order differential equations. It is shown in the following sections that this representation is not unique and, in fact, that there is an infinite number of choices of state variables that can correctly describe the system. Fortunately, only a few of these are in common use—either because they give a mathematical advantage or because they have some relationship to physical reality.

Before proceeding to the various methods of system representation, we must consider the notation employed in modern control theory.

2.3 Vectors and matrices

This section details the notation used throughout the book and defines some of the more basic matrix manipulations. The treatment here is brief, since the rest of this chapter contains examples that amply illustrate the points made in this section.

A matrix is a rectangular array of elements arranged in m rows and n columns which obeys a certain set of manipulations discussed below. The

elements of the matrix may be real or complex numbers or variables of either time or frequency. The matrix **A** is thus

$$\mathbf{A} = [a_{ij}] = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \tag{2.3-1}$$

Equation (2.3-1) indicates the notation that is used throughout this book. The matrix **A** is designated by a boldface capital letter as a shorthand notation for the entire array. When the array itself is given, it is always set off in brackets to emphasize that it is a matrix. The elements of the matrix are indicated by a small letter with a double subscript. Thus the element of **A** that is located in the ith row and the jth column is a_{ij}.

A matrix is classified by the number of its rows and columns; a matrix of m rows and n columns is a matrix of order $m \times n$, or an $m \times n$ matrix. If $m = n$, the matrix is called a *square matrix*. A matrix with n rows and only one column is called a *column matrix* or vector.[1] The shorthand notation for the column vector is a lowercase, boldface letter, as

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} = \text{col } (x_1, x_2, \ldots, x_n)$$

Note that the elements of a vector have only one subscript, to indicate their location in the column. A matrix with one row and n columns is called a *row matrix* or row vector.

The *transpose* of any matrix is formed by interchanging rows and columns of the matrix and is indicated by a superscript T, so that

$$\mathbf{A}^T = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ \cdots & \cdots & \cdots & \cdots \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{bmatrix}$$

[1] The equating of a vector to a column matrix is not exactly correct, since a vector has a fixed geometric meaning while the column matrix is only a representation of a given vector in one coordinate system. However, the usage is common in the literature and is retained here. For a more complete discussion of this subject see Huelsman (1963).

A row matrix may be thought of as the transpose of a column matrix, so that $\mathbf{x}^T$ is the row matrix

$$\mathbf{x}^T = [x_1 \quad x_2 \quad \cdots \quad x_n]$$

Matrices are also classified by their *rank*. The rank of a matrix is the largest number r such that at least one $r \times r$ matrix whose determinant[1] is nonzero may be formed from the original matrix by deleting rows and/or columns. It is obvious that the rank of a matrix may never be larger than the smaller dimension of a matrix, that is, $r \leq$ minimum (m,n). Although there are more elaborate tests, the rank of a matrix can be determined by simply testing all determinants of successively smaller size until a nonzero one is found. Consider, for example, the matrix

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 2 \\ 0 & 1 & 1 \\ 1 & 2 & 3 \end{bmatrix}$$

In this case, the single third-order determinant of the matrix itself is zero. However, every second-order determinant is nonzero, and therefore the matrix has rank 2.

Another type of classification of square matrices is concerned with the elements of the matrix. If $a_{ij} = a_{ji}$ for all i and j, the matrix is called *symmetric*, and we see that $\mathbf{A}^T = \mathbf{A}$. For example the matrix

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 7 \\ 2 & 4 & 13 \\ 7 & 13 & 0 \end{bmatrix}$$

is symmetric. On the other hand, if $a_{ij} = -a_{ji}$ for all i and j, the matrix is called *skew-symmetric*, and we observe that $\mathbf{A}^T = -\mathbf{A}$. The matrix

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & -3 \\ -1 & 0 & -2 \\ 3 & 2 & 0 \end{bmatrix}$$

is skew-symmetric. We note that the diagonal elements a_{ii} of any skew-symmetric matrix must be zero, since $a_{ii} = -a_{ii}$ can be satisfied only by $a_{ii} = 0$.

[1] A knowledge of determinants is assumed. For reference see Hohn (1958).

It should be noted that any square matrix may be written as the sum of a symmetric matrix $\mathbf{A}_s$ and a skew-symmetric matrix $\mathbf{A}_{sk}$, as can be shown as follows. Let

$$\mathbf{A} = \mathbf{A}_s + \mathbf{A}_{sk} \tag{2.3-2}$$

Now take the transpose of both sides,

$$\mathbf{A}^T = \mathbf{A}_s{}^T + \mathbf{A}_{sk}{}^T$$

By the above definitions this becomes

$$\mathbf{A}^T = \mathbf{A}_s - \mathbf{A}_{sk} \tag{2.3-3}$$

If Eqs. (2.3-2) and (2.3-3) are solved simultaneously, we have

$$\mathbf{A}_s = \frac{\mathbf{A} + \mathbf{A}^T}{2}$$

$$\mathbf{A}_{sk} = \frac{\mathbf{A} - \mathbf{A}^T}{2}$$

Therefore the sum of $\mathbf{A}_s$ and $\mathbf{A}_{sk}$ is

$$\mathbf{A}_s + \mathbf{A}_{sk} = \frac{\mathbf{A} + \mathbf{A}^T}{2} + \frac{\mathbf{A} - \mathbf{A}^T}{2} = \mathbf{A}$$

as desired. Consider, for example, the matrix

$$\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 5 & 9 \end{bmatrix}$$

which may be written as

$$\mathbf{A} = \begin{bmatrix} 1 & 4 \\ 4 & 9 \end{bmatrix} + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

If $\mathbf{A}$ is square, and if $a_{ij} = 0$, for $i \neq j$, then $\mathbf{A}$ is called a *diagonal matrix*. In the special case when $\mathbf{A}$ is diagonal and $a_{ii} = 1$ for all i, the matrix is known as the *identity matrix*, designated as $\mathbf{I}$,

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \tag{2.3-4}$$

Closely associated with the definition of a matrix is a matrix algebra governing the allowable manipulations of matrices, which are summarized below.

Equality. Two matrices $\mathbf{A}$ and $\mathbf{B}$ are equal if and only if all the elements of each matrix are equal, i.e., if $a_{ij} = b_{ij}$. Two matrices can be equal only if they are of the same order.

Addition and subtraction. The sum or difference of two matrices $\mathbf{A}$ and $\mathbf{B}$ is a matrix $\mathbf{C}$ whose elements are the sum or difference of the respective elements of $\mathbf{A}$ and $\mathbf{B}$. That is,

$$c_{ij} = a_{ij} \pm b_{ij} \tag{2.3-5}$$

Addition and subtraction are defined only for matrices of the same order. Addition and subtraction are commutative, $\mathbf{A} + \mathbf{B} = \mathbf{B} + \mathbf{A}$, and associative, $\mathbf{A} + (\mathbf{B} + \mathbf{C}) = (\mathbf{A} + \mathbf{B}) + \mathbf{C}$.

Multiplication by a scalar. The product $h\mathbf{A}$ is formed by multiplying each element of the matrix $\mathbf{A}$ by the scalar constant h. Therefore, if $\mathbf{C} = h\mathbf{A}$, then $c_{ij} = ha_{ij}$. Scalar multiplication distributes over vector addition, $h(\mathbf{A} + \mathbf{B}) = h\mathbf{A} + h\mathbf{B}$.

Integration and differentiation of matrices is also treated on this same element basis. For example, the elements of the matrix $d\mathbf{A}(t)/dt$ are $da_{ij}(t)/dt$.

Multiplication. Two matrices may be multiplied only if they are *conformal;* that is, $\mathbf{AB}$ is defined only if the number of columns of $\mathbf{A}$ equals the number of rows of $\mathbf{B}$. If $\mathbf{A}$ is an $n \times r$ matrix and $\mathbf{B}$ is an $r \times m$ matrix, the product $\mathbf{AB} = \mathbf{C}$ is an $n \times m$ matrix with the elements of $\mathbf{C}$ defined as

$$c_{ij} = \sum_{h=1}^{r} a_{ih} b_{hj} \tag{2.3-6}$$

Thus c_{ij} is formed as the sum of the products of the corresponding elements of the ith row of **A** and the jth column of **B**. As an example, consider the following matrix product.

$$\begin{bmatrix} 1 & -2 & 3 \\ 2 & 3 & 1 \\ -1 & 3 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ -2 & 3 \\ 3 & 2 \end{bmatrix} = \begin{bmatrix} 14 & 1 \\ -1 & 13 \\ -1 & 12 \end{bmatrix} \tag{2.3-7}$$

Here the c_{21} term of the resulting matrix is determined as

$$(2)(1) + (3)(-2) + (1)(3) = -1$$

As a consequence of the requirement of conformality, note the following points:

1. An $n \times n$ square matrix times an $n \times n$ square matrix yields an $n \times n$ square matrix.
2. A row vector times an $n \times n$ square matrix yields a $1 \times n$ matrix, or row vector.
3. An $n \times n$ square matrix times a column vector yields an $n \times 1$ column matrix, or column vector.
4. A $1 \times n$ row matrix times an $n \times 1$ column matrix yields a 1×1 matrix. The 1×1 matrix has all the usual properties of a scalar and hence is defined as a scalar.

As a consequence of the definition of matrix multiplication, note also the following points:

1. **AB** is not necessarily equal to **BA**. If **AB** = **BA**, this is purely coincidence. The product **BA** may not even be defined. Consider, for example,

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 5 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{B} = \begin{bmatrix} 2 & 0 \\ 1 & 3 \end{bmatrix}$$

Then

$$\mathbf{AB} = \begin{bmatrix} 4 & 6 \\ 11 & 3 \end{bmatrix} \neq \begin{bmatrix} 2 & 4 \\ 16 & 5 \end{bmatrix} = \mathbf{BA}$$

Because matrix multiplication is not commutative, one must be careful to preserve the order of the matrices when making any manipulation. In addition, one often speaks of pre- or postmultiplication to indicate whether the matrix is multiplied from the right or the left. For example, in the triple matrix product **ABC**, **A** premultiplies **B** while **C** postmultiplies **B**.

2. If **AB** = **0**, it cannot be concluded that either **A** or **B** is identically zero. For example, if

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ 2 & 9 \end{bmatrix}$$

then **AB** = **0** even though neither **A** nor **B** is zero. This is simply another indication of how careful one must be in translating scalar concepts into the world of matrices.

3. If **AB** = **AC**, then **B** and **C** are not necessarily equal. This is an obvious consequence of point 2 above.
4. Multiplication is associative; that is,

$$(\mathbf{AB})\mathbf{C} = \mathbf{A}(\mathbf{BC}) \tag{2.3-8}$$

5. Multiplication is distributive with respect to addition; that is,

$$\mathbf{A}(\mathbf{B} + \mathbf{C}) = \mathbf{AB} + \mathbf{AC} \tag{2.3-9}$$

Inversion. Division is not defined for matrices. The division operation is replaced by matrix inversion for square matrices. The square matrix $\mathbf{A}^{-1}$ is said to be the *inverse* of the square matrix **A** if

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$$

the identity matrix. Not all square matrices have an inverse, since the existence of an inverse requires that the determinant of the matrix be nonzero. If the determinant is nonzero, the matrix has an inverse and is said to be a *nonsingular matrix*. If the determinant is zero, the matrix has no inverse and is said to be *singular*.

Consider the problem of determining the inverse of **A**. If $\mathbf{A}^{-1}$ does exist, the elements of $\mathbf{A}^{-1}$ may be determined in a brute-force manner from the defining equation $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$ by solving these n^2 simultaneous algebraic equations for the elements of $\mathbf{A}^{-1}$. Although this procedure is possible, it is desirable to avoid the direct simultaneous-equation approach. This may be done by defining the *adjoint* of a matrix. The adjoint matrix

adj **A** is related to the matrix **A** through the cofactors of the elements of **A**. The cofactor cof a_{ij} of the element a_{ij} is $(-1)^{i+j}$ times the determinant of the matrix formed by deleting the ith row and the jth column from **A**. The adjoint of the matrix **A** is then defined as the transpose of a matrix whose elements are the cofactors of **A**, or

$$\text{adj } \mathbf{A} = [\text{cof } a_{ij}]^T = \begin{bmatrix} \text{cof } a_{11} & \text{cof } a_{21} & \cdots & \text{cof } a_{n1} \\ \text{cof } a_{12} & \text{cof } a_{22} & \cdots & \text{cof } a_{n2} \\ \cdots & \cdots & \cdots & \cdots \\ \text{cof } a_{1n} & \text{cof } a_{2n} & \cdots & \text{cof } a_{nn} \end{bmatrix} \tag{2.3-10}$$

Then $\mathbf{A}^{-1}$, the inverse of **A**, is

$$\mathbf{A}^{-1} = \frac{\text{adj } \mathbf{A}}{\det \mathbf{A}} \tag{2.3-11}$$

Example 2.3-1 The following example illustrates the steps involved in taking the inverse. Let **A** be given as

$$\mathbf{A} = \begin{bmatrix} 8 & 4 & 2 \\ 2 & 8 & 4 \\ 1 & 2 & 8 \end{bmatrix}$$

The determinant associated with the matrix **A** may be formed by expanding in terms of the minors of the elements of the first row. Thus

$$\det \mathbf{A} = 8(64 - 8) - 4(16 - 4) + 2(4 - 8) = 392 \neq 0$$

and **A** has an inverse. The cofactor matrix is

$$[\text{cof } a_{ij}] = \begin{bmatrix} 56 & -12 & -4 \\ -28 & 62 & -12 \\ 0 & -28 & 56 \end{bmatrix}$$

Here, for example, cof a_{23} was formed as

$$\text{cof } a_{23} = (-1)^5 \det \begin{bmatrix} 8 & 4 \\ 1 & 2 \end{bmatrix} = -12$$

Thus the adjoint matrix is

$$\text{adj } \mathbf{A} = [\text{cof } a_{ij}]^T = \begin{bmatrix} 56 & -28 & 0 \\ -12 & 62 & -28 \\ -4 & -12 & 56 \end{bmatrix}$$

and the inverse $\mathbf{A}^{-1}$ is given by

$$\mathbf{A}^{-1} = \frac{\text{adj } \mathbf{A}}{\det \mathbf{A}} = \begin{bmatrix} \frac{1}{7} & -\frac{1}{14} & 0 \\ -\frac{3}{98} & \frac{31}{196} & -\frac{1}{14} \\ -\frac{1}{98} & -\frac{3}{98} & \frac{1}{7} \end{bmatrix}$$

As a check, the reader may wish to show that $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$.

Before concluding this brief introduction to matrices, the two following identities are given for future reference:

$$(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1} \tag{2.3-12}$$

and

$$(\mathbf{AB})^T = \mathbf{B}^T\mathbf{A}^T \tag{2.3-13}$$

Equation (2.3-12) applies only to cases where **A** and **B** are nonsingular and therefore square and of the same order. Equation (2.3-13) applies in all cases where **A** and **B** are conformal.

The reader is referred to Bellman (1960), Hohn (1958), and Nering (1963) for a more thorough and extensive treatment of the subject of matrices. The object here is not to be definitive but to present only that material which is used in the remainder of the book.

Exercises 2.3 *2.3-1.* Perform the following matrix multiplications.

(*a*) $\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 2 & 0 & 4 \\ 0 & 0 & 1 \end{bmatrix}$ (*b*) $\begin{bmatrix} 2 & 0 & 3 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 10 \\ 5 \\ 4 \end{bmatrix}$

(*c*) $\begin{bmatrix} \frac{1}{2} & \sqrt{3/2} \\ -\sqrt{3/2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{2} & -\sqrt{3/2} \\ \sqrt{3/2} & \frac{1}{2} \end{bmatrix}$

answers:

$$(a)\ \begin{bmatrix}1 & 2 & 2\\ 2 & 4 & 7\end{bmatrix} \qquad (b)\ \begin{bmatrix}32\\ 5\end{bmatrix} \qquad (c)\ \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$$

2.3-2. Find the inverses of the following matrices.

$$(a)\ \begin{bmatrix}1 & 2 & 2\\ 2 & -2 & 1\\ 2 & 1 & -2\end{bmatrix} \qquad (b)\ \begin{bmatrix}0 & 2 & 0\\ 1 & 3 & -1\\ 1 & 0 & 0\end{bmatrix} \qquad (c)\ \begin{bmatrix}a & b\\ c & d\end{bmatrix}$$

answers:

$$(a)\ \frac{1}{9}\begin{bmatrix}1 & 2 & 2\\ 2 & -2 & 1\\ 2 & 1 & -2\end{bmatrix} \qquad (b)\ \frac{1}{2}\begin{bmatrix}0 & 0 & 2\\ 1 & 0 & 0\\ 3 & -2 & 2\end{bmatrix}$$

$$(c)\ \frac{1}{ad-bc}\begin{bmatrix}d & -b\\ -c & a\end{bmatrix}$$

2.3-3. Verify Eqs. (2.3-12) and (2.3-13) for

$$\mathbf{A} = \begin{bmatrix}1 & 1\\ 2 & 1\end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix}1 & 2\\ 0 & 4\end{bmatrix}$$

2.3-4. If $\mathbf{P}$ is symmetric, show that $\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A}$ and $\mathbf{B}^T\mathbf{P}\mathbf{B}$ are symmetric.

2.3-5. Find the rank of the following matrices.

$$(a)\ \begin{bmatrix}1 & -2\\ 1 & -2\end{bmatrix} \qquad (b)\ \begin{bmatrix}1 & 2 & 1\\ 3 & 5 & 1\\ 0 & 0 & 1\end{bmatrix} \qquad (c)\ \begin{bmatrix}0 & 0 & 1\\ 1 & 1 & 0\\ 0 & 0 & 6\end{bmatrix}$$

answers:

(*a*) 1 (*b*) 3 (*c*) 2

2.4 System representation in state-variable form

This section contains a brief introduction to the representation of *n*th-order linear control systems in state-variable form as *n* first-order differential equations and an output expression. The most general form of the system equations considered is

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \qquad \textbf{(AB)}$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) \qquad \textbf{(C)}$$

Here $\mathbf{x}$ is an n-dimensional *state vector*, $\mathbf{u}$ is an r-dimensional *control vector*, $\mathbf{y}$ is an m-dimensional *output vector*, $\mathbf{A}$ is an $n \times n$ *system matrix*, $\mathbf{B}$ is an $n \times r$ *control matrix*, and $\mathbf{C}$ is an $m \times n$ *output matrix*. In expanded form these equations become

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\\ \cdots\\ \dot{x}_n(t)\end{bmatrix} = \begin{bmatrix}a_{11} & a_{12} & \cdots & a_{1n}\\ a_{21} & a_{22} & \cdots & a_{2n}\\ \cdots & \cdots & \cdots & \cdots\\ a_{n1} & a_{n2} & \cdots & a_{nn}\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\\ \cdots\\ x_n(t)\end{bmatrix} + \begin{bmatrix}b_{11} & b_{12} & \cdots & b_{1r}\\ b_{21} & b_{22} & \cdots & b_{2r}\\ \cdots & \cdots & \cdots & \cdots\\ b_{n1} & b_{n2} & \cdots & b_{nr}\end{bmatrix}\begin{bmatrix}u_1(t)\\ u_2(t)\\ \cdots\\ u_r(t)\end{bmatrix} \qquad \textbf{(AB)}$$

$$\begin{bmatrix}y_1(t)\\ y_2(t)\\ \cdots\\ y_m(t)\end{bmatrix} = \begin{bmatrix}c_{11} & c_{12} & \cdots & c_{1n}\\ c_{21} & c_{22} & \cdots & c_{2n}\\ \cdots & \cdots & \cdots & \cdots\\ c_{m1} & c_{m2} & \cdots & c_{mn}\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\\ \cdots\\ x_n(t)\end{bmatrix} \qquad \textbf{(C)}$$

Note that this representation is quite general, as it allows for multiple inputs, r of them, and multiple outputs, m of them. Thus the general system representation of Eqs. (**AB**) and (**C**) is adequate for multiple input–multiple output systems.

Equation (**AB**) is a set of n first-order differential equations and is usually referred to as the *plant equation*, while Eq. (**C**) represents a set of m linear algebraic equations and is referred to as the *output expression*. In its most general form the output expression appears as

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)$$

where the added $\mathbf{D}\mathbf{u}(t)$ term indicates a direct coupling of the input to the output. Since the direct coupling of the input to the output is rare in control systems, where power amplification is generally desired, we shall use the simpler form of Eq. (**C**) as an output expression.

This time-domain state-variable representation of a multiple input–multiple output system is to be contrasted with the frequency-domain

transfer-function approach of classical control theory. In the latter form, the system discussed above would become simply

$$\mathbf{y}(s) = \mathbf{G}(s)\mathbf{u}(s) \tag{2.4-1}$$

Here we have made use of the matrix notation of the preceding section to express the system in a compact form. The matrix $\mathbf{G}(s)$ is known as the transfer-function matrix since each of its elements is a transfer function between an input and output. For example, the $g_{ij}(s)$ element is the transfer function between the jth input and the ith output, or

$$\frac{y_i(s)}{u_j(s)} = g_{ij}(s)$$

It is usual to represent graphically the transfer function by means of a block diagram, as shown in Fig. 2.4-1*a*. For comparison, a block diagram of the state-variable representation of the system is shown in Fig. 2.4-1*b*. This latter diagram is nothing more than a pictorial representation of Eqs. (**AB**) and (**C**).

In order to put the reader on more familiar footing, it is perhaps of value to consider the special case when both the input and output are scalars. The importance of this class of system is indicated by the fact that almost every basic control book treats only the single input–single

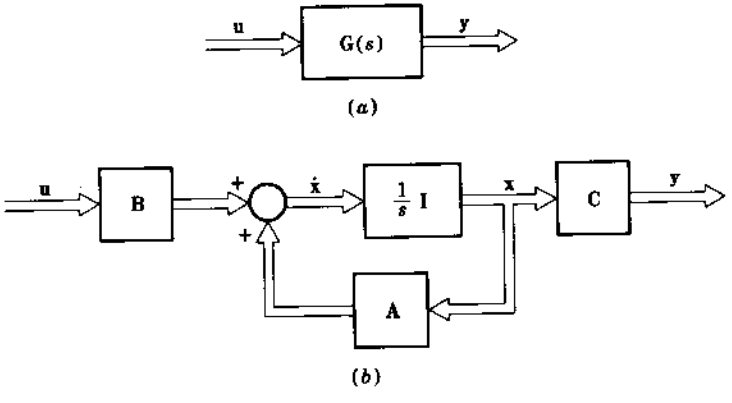

Fig. 2.4-1 Block diagrams of a multiple input–multiple output system. (*a*) Transfer-function representation; (*b*) state-variable representation.

output case. For this case the state-variable representation becomes

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) \tag{Ab}$$
$$y(t) = \mathbf{c}^T\mathbf{x}(t) \tag{c}$$

where the scalars u and y are the input and output, respectively, and $\mathbf{b}$ and $\mathbf{c}$ are n-dimensional vectors. The transfer-function representation is simply

$$\frac{y(s)}{u(s)} = G(s) \tag{2.4-2}$$

where $G(s)$ is the usual scalar transfer function. A graphical representation of this system is shown in Fig. 2.4-2 for both the state-variable and transfer-function forms.

In the classical terminology, Eqs. (**Ab**) and (**c**) represent the fixed-plant portion of the control problem, i.e., the physical object to be controlled, such as the antenna and its associated drive equipment in a radar-positioning system. Therefore $G(s)$ is the uncompensated forward-path transfer function, while the control input $u(t)$ is the classical actuating signal, such as the armature current in a dc positioning system.

In the classical unity-ratio feedback system, $u(t)$ is formed by passing the error signal $r(t) - y(t)$ through a compensation network, as shown in Fig. 2.4-3*a*. This configuration is to be contrasted with the modern con-

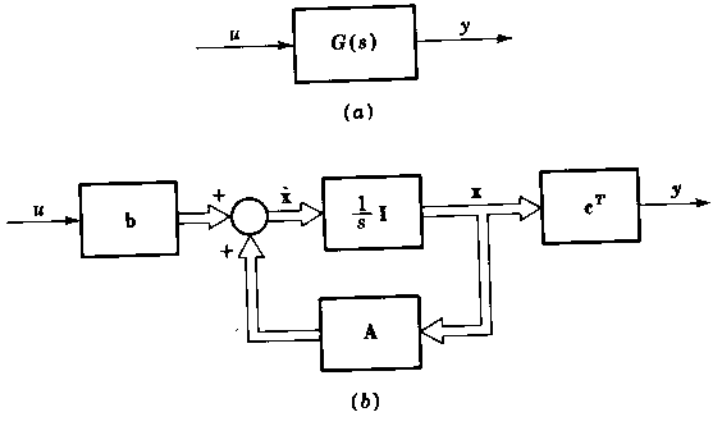

Fig. 2.4-2 Block diagrams of a single input–single output system. (*a*) Transfer-function representation; (*b*) state-variable representation.

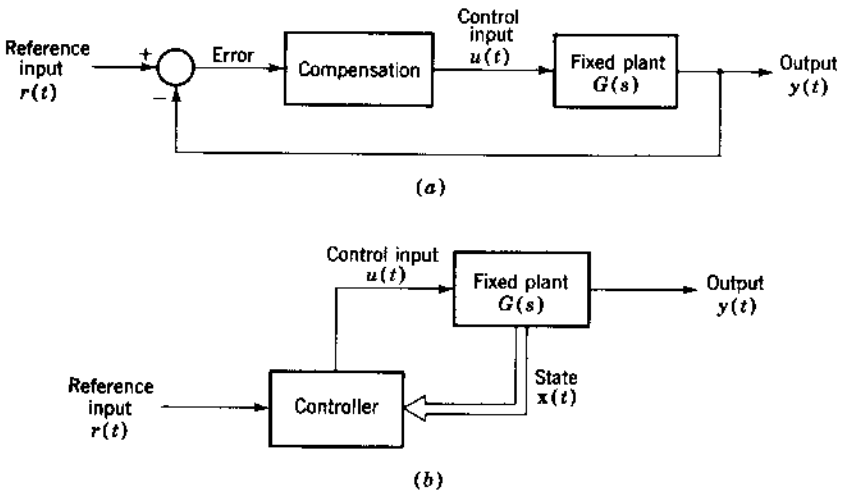

Fig. 2.4-3 Comparison of modern and classical control-system configurations. (*a*) Classical control configuration; (*b*) modern control configuration.

figuration shown in Fig. 2.4-3*b*, where the control signal is determined by the controller from the system state and the reference input. The classical configuration is simply a special case in which the controller consists of an error detector and a compensation network. Chapters 6 to 9 are concerned with the determination of the controller to meet certain performance specifications.

Before continuing our comparison of the state-variable and transfer-function methods of system representation, let us discuss some of the notation, illustrated by the preceding discussion, that is used throughout the text. First, the Laplace transform of a time function, such as $y(t)$, is indicated by simply replacing the time variable t by the frequency variable s. The vector $\mathbf{x}(s)$ therefore indicates a vector whose elements are the Laplace transforms of the respective elements of $\mathbf{x}(t)$, for example, $x_i(s) = \mathfrak{L}[x_i(t)]$. On the other hand $G(t)$ is the impulse response corresponding to the transfer function $G(s)$. The arguments s or t are omitted only when there is no possibility of confusion or where a dual interpretation may be made, as on the block diagrams of Figs. 2.4-1 and 2.4-2. There, the variables may be interpreted as frequency-domain variables [$\mathbf{x}(s)$, for example] and the various blocks as transfer functions, or the variables may be interpreted as time-domain variables [$\mathbf{x}(t)$, for example] and the blocks as linear time-domain operations with s indicating differentiation with respect to time as usual.

The use of the broad arrow on block diagrams, as in Figs. 2.4-1 and 2.4-2, to indicate vector quantities is continued throughout the book as an assistance to the reader in identifying such quantities.

Since Eqs. (**AB**), (**C**), (**Ab**), and (**c**) play such fundamental roles, appearing over and over again in the following development, they are denoted by special symbols throughout the book for the convenience of the reader. Note also that these equation symbols have been selected to assist the reader in remembering the form of the equations.

The most obvious difference between the state-variable and transfer-function approaches is the suppression of the state vector in the transfer-function approach. This is just another way of saying that the transfer-function approach is concerned only with the input-output characteristics of a system while the state-variable approach provides, in addition, a description of the internal behavior of the system.

Two questions naturally arise with regard to these two methods of system representation. First, under what conditions are the approaches equivalent? In other words, when may a system be represented accurately by either of the two approaches? As we shall see shortly, this question could be phrased, "When is a transfer-function representation of a system adequate?"

The second question is closely related to the first, i.e., "In the cases where the two approaches are equivalent, how are they related?" In particular we are interested in knowing how to find the state-variable representation of a system, given its transfer-function representation, or how to find its transfer function, given the state-variable representation. Since the transfer-function method specifies only the input-output relations of the system, there is always a certain amount of arbitrariness in the selection of the state variables of a system specified only by a transfer function. This means that there is, in general, an infinite number of state-variable representations for a given transfer function. The next three sections discuss three of the more common methods of choosing the state variables beginning with a transfer-function representation.

On the other hand, if a state-variable representation of a system is known, the transfer function of the system is completely and uniquely specified. The fact that the state-variable representation uniquely specifies the transfer-function representation while there is an infinite number of state-variable representations for a given transfer function is simply one manifestation of the fact that the state-variable representation is a more complete description of a system. Before discussing how to determine the transfer function from the state-variable representation, let us examine more closely the question of equivalence of the two approaches.

In order to answer the question of equivalence, it is necessary to

define two new concepts known as *controllability* and *observability*. These concepts can be defined and demonstrated only in terms of the state-variable representation of a system. Since the reader is probably more familiar with the transfer-function representation, it is unfortunate that this approach cannot be used. As we shall see, using a transfer function to represent a system presupposes the existence of controllability and observability in the system, and hence there is no meaningful way to discuss the concepts in terms of the transfer-function approach.

The concepts of controllability and observability were first introduced by Kalman (1960). Although these concepts may at first appear too esoteric for an engineer, nevertheless they do play important roles in almost every practical result of modern control theory. The fact that they are necessary to establish the validity of the classical transfer-function methods indicates the importance of these concepts.

The following definitions of controllability and observability are similar to the original definitions of Kalman as presented by Kreindler and Sarachik (1964).

Definition 2.4-1 A system is said to be *controllable* if any initial state $\mathbf{x}(0)$ can be transferred to any final state $\mathbf{x}(t_f)$ in a finite time, $t_f \geq 0$, by some control $\mathbf{u}$.

Controllability implies, as the word itself suggests, the ability of the control input to affect each state variable. Observability, on the other hand, is concerned with the ability of each state variable to influence the output of the system.

Definition 2.4-2 A system is said to be *observable* if every state $\mathbf{x}(0)$ can be exactly determined from measurements of the output $\mathbf{y}$ over a finite interval of time, $0 \leq t \leq t_f$.

Kalman has also defined observability in an alternate fashion by means of a concept known as a *dual system*. The dual of the system described by Eqs. (**AB**) and (**C**) is the system

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

with time running backward. In terms of this dual system, Kalman defines observability by saying that the original system [Eqs. (**AB**) and (**C**)] is observable if the dual system is controllable. Hence Kalman speaks of controllability and observability as dual concepts. We shall make further use of this duality property in Chap. 7 in our discussion of Kalman filters.

Quite often the controllability or observability of a system may be determined by inspection from the state-variable representation of the system or its expanded block-diagram representation. If there are states which are completely decoupled from the input, the system is obviously not controllable. If, on the other hand, some states are decoupled from the output, the system is unobservable.

The idea that some of the states may be uncontrollable or unobservable leads one to consider the possibility of dividing the system into a set of subsystems possessing different controllability and observability properties. Gilbert (1963), in fact, has shown that by proper selection of the state vector, a system may always be partitioned into four possible subsystems:

1. A subsystem S_{co} which is controllable and observable
2. A subsystem S_{cu} which is controllable but unobservable
3. A subsystem S_{uo} which is uncontrollable but observable
4. A subsystem S_{uu} which is uncontrollable and unobservable

This partitioning concept is graphically illustrated in Fig. 2.4-4 by showing the way in which the subsystems are related to the input and output. We see, for example, that the observable but uncontrollable subsystem S_{uo} is unaffected by the input $\mathbf{u}$ but does influence the output $\mathbf{y}$.

From Fig. 2.4-4 it is obvious that the transfer function $\mathbf{G}(s)$, where $\mathbf{y}(s) = \mathbf{G}(s)\mathbf{u}(s)$, is dependent only on the controllable and observable portion of the system S_{co}. In other words, a *transfer function is an accurate*

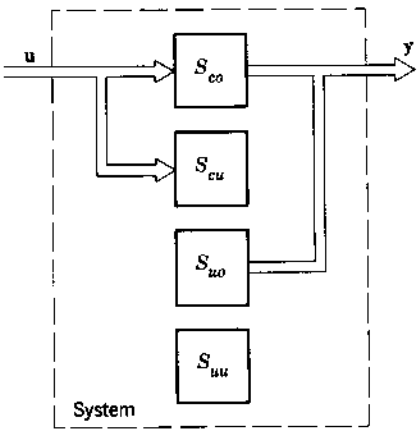

Fig. 2.4-4 Partitioning of a system.

representation of a system if and only if the system is controllable and observable. This is the desired condition of equivalence between the state-variable and transfer-function representations and is one of the primary results of the modern approach to control-system representation.

Example 2.4-1 In order to illustrate this partitioning procedure, let us consider the following state-variable representation of a system.

$$\dot{\mathbf{x}} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} u$$

$$y = [1 \quad 0 \quad 2]\mathbf{x}$$

A block-diagram representation for this system is shown in Fig. 2.4-5 with the various subsystems indicated. The input-output

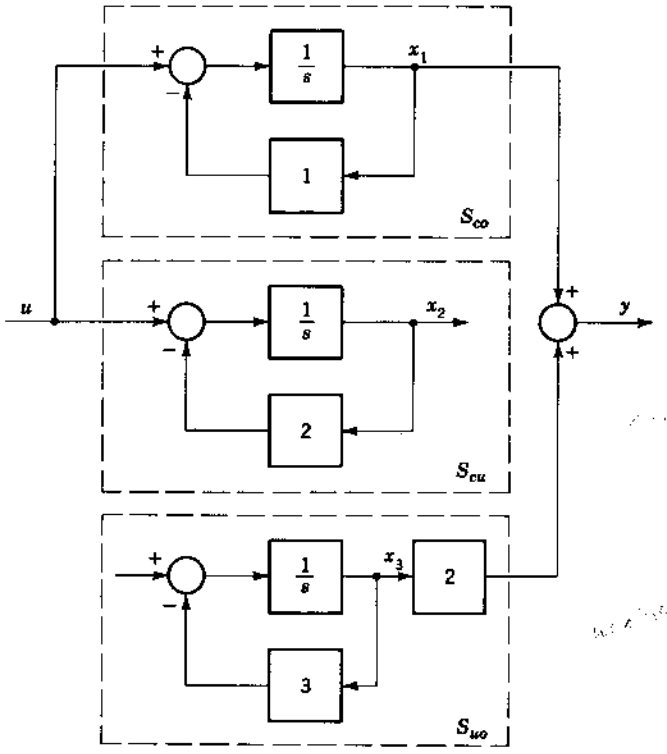

Fig. 2.4-5 Block diagram for Example 2.4-1.

transfer function for this system is obviously

$$\frac{y(s)}{u(s)} = G(s) = \frac{1}{s+1}$$

Hence we see that while the actual system is third-order, the transfer function represents only the first-order controllable and observable subsystem S_{co}.

Unfortunately in order to partition a system as discussed above, it is necessary for the system to be represented in a special and unique manner known as *normal* or *diagonal form*. Thus, in order to use this approach to determine the controllability and observability, in general it is necessary to transform the system from its present representation into this special representation, which is often not a simple task. This suggests the desirability of a more direct test for controllability in terms of a general system representation.

While Kalman has presented a more general partitioning procedure, it still requires a special, although not unique, *canonical* representation of the system. Therefore, it is perhaps justifiable to conclude that partitioning is not a very useful approach to determining controllability and observability of a system, even though it is a very valuable tool in the study of these properties. Gilbert, for example, has used this approach to investigate the controllability and observability of interconnected systems. We shall discuss partitioning further in Sec. 2.6, on normal-form representation. The following direct test can be used to establish the controllability and observability of a system in any state-variable representation.

Theorem 2.4-1 The nth-order multiple input–multiple output system

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$$

is

1. Controllable if and only if the $n \times nr$ composite matrix

$$[\mathbf{B}|\mathbf{AB}|\mathbf{A}^2\mathbf{B}|\cdots|\mathbf{A}^{n-1}\mathbf{B}] \tag{2.4-3}$$

is of rank n;

2. Observable if and only if the $n \times nm$ composite matrix

$$[\mathbf{C}^T|\mathbf{A}^T\mathbf{C}^T|\mathbf{A}^{T2}\mathbf{C}^T| \cdots |\mathbf{A}^{T(n-1)}\mathbf{C}^T] \tag{2.4-4}$$

is of rank n.

In the case of the single input–single output system, this test becomes:

Theorem 2.4-2 The nth-order single input–single output system

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t)$$
$$y(t) = \mathbf{c}^T\mathbf{x}(t)$$

is

1. Controllable if and only if the $n \times n$ composite matrix

$$[\mathbf{b}|\mathbf{A}\mathbf{b}|\mathbf{A}^2\mathbf{b}| \cdots |\mathbf{A}^{n-1}\mathbf{b}] \tag{2.4-5}$$

is nonsingular; i.e., its determinant is nonzero;
2. Observable if and only if the $n \times n$ composite matrix

$$[\mathbf{c}|\mathbf{A}^T\mathbf{c}|\mathbf{A}^{T2}\mathbf{c}| \cdots |\mathbf{A}^{T(n-1)}\mathbf{c}] \tag{2.4-6}$$

is nonsingular.

These theorems are offered without proof because although the proof is not difficult, it involves concepts which have not been introduced in this treatment. The interested reader is directed to Kreindler and Sarachik (1964) for a simple proof.

Example 2.4-2 As illustration of the application of these tests, let us consider again the system of Example 2.4-1. In this case the system is single input, single output and hence Theorem 2.4-2 is applicable. For this system,

$$\mathbf{Ab} = \begin{bmatrix} -1 \\ -2 \\ 0 \end{bmatrix} \quad \text{and} \quad \mathbf{A}^2\mathbf{b} = \begin{bmatrix} 1 \\ 4 \\ 0 \end{bmatrix}$$

and the matrix (2.4-5) becomes

$$[\mathbf{b}|\mathbf{A}\mathbf{b}|\mathbf{A}^2\mathbf{b}] = \begin{bmatrix} 1 & -1 & 1 \\ 1 & -2 & 4 \\ 0 & 0 & 0 \end{bmatrix}$$

It is obvious that the matrix is singular, since it has an all-zero row, and therefore the system is not controllable.

To check observability we form the matrix

$$[\mathbf{c}|\mathbf{A}^T\mathbf{c}|\mathbf{A}^{T2}\mathbf{c}] = \begin{bmatrix} 1 & -1 & 1 \\ 0 & 0 & 0 \\ 2 & -6 & 18 \end{bmatrix}$$

which is also singular. We therefore conclude that the system is neither controllable nor observable, as we had previously seen from the matrix partitioning.

It should be obvious to the reader that the rank of matrix (2.4-3) or (2.4-5) is equal to the number of controllable states, while the rank of matrix (2.4-4) or (2.4-6) is equal to the number of observable states.

All systems treated in this book are assumed to be both controllable and observable. This assumption is made, first of all, because almost all practical systems are controllable and observable. This does not mean that the mathematical models of these systems are always controllable and observable. On the contrary, it is quite possible for the mathematical model to lack these properties, particularly if linearization has been necessary. When this happens, however, the model is not an accurate representation of the physical system, and another model should be sought. We therefore assume that the model has been checked to ensure that the controllability and observability of the physical system have been preserved.

The assumption of controllability and observability also makes possible the meaningful use of transfer functions at various points through the book. In the next three sections, for example, we use the transfer function of a system as the starting point for a state-variable representation of the system. Such an approach is obviously senseless if the system is not controllable and observable.

In many control problems the only knowledge one has of the system is an experimentally obtained transfer function. Without an assumption of controllability and observability, such information would be useless.

In addition, this assumption has been tacitly made in every classical control book where the transfer function was the sole means of system

representation. Also there are very few results of modern control theory which deal with uncontrollable or unobservable systems. The plain fact is that such systems are extremely difficult to treat in any reasonable fashion. Hence the assumption of controllability and observability is neither new nor unreasonable nor unnecessary.

The reader may wonder why the transfer-function approach is used at all since the state-variable representation provides a more complete description of the system. The simple truth is that some of the most practical results of modern control theory can be most compactly and usefully presented in terms of transfer functions. For example, the transfer-function approach is used almost exclusively throughout Chaps. 8 and 9. We shall therefore make use of both the state-variable and transfer-function representations throughout the book in order to present concepts in the simplest and yet most complete form and to interrelate the time-domain approach with the more familiar transform methods.

Having answered the question of equivalence of the state-variable and transfer-function representation, we turn our attention next to the question of how the two approaches are related. Let us consider first the problem of determining the transfer function of a system given the state-variable representation

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \qquad \text{(AB)}$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) \qquad \text{(C)}$$

Since the transfer-function representation is expressed in the frequency domain, we begin by taking the Laplace transform of both equations, assuming as usual in transfer-function determination that the initial conditions on $\mathbf{x}$ are all zero.

$$s\mathbf{x}(s) = \mathbf{A}\mathbf{x}(s) + \mathbf{B}\mathbf{u}(s) \qquad (2.4\text{-}7)$$
$$\mathbf{y}(s) = \mathbf{C}\mathbf{x}(s) \qquad (2.4\text{-}8)$$

Grouping the two $\mathbf{x}(s)$ terms in Eq. (2.4-7), we have

$$(s\mathbf{I} - \mathbf{A})\mathbf{x}(s) = \mathbf{B}\mathbf{u}(s) \qquad (2.4\text{-}9)$$

where the identity matrix has been introduced to allow the indicated factoring. If both sides of this equation are now premultiplied by $(s\mathbf{I} - \mathbf{A})^{-1}$, Eq. (2.4-9) becomes

$$\mathbf{x}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{u}(s)$$

This result may be substituted into Eq. (2.4-8) to obtain

$$\mathbf{y}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{u}(s)$$

Comparing this result with Eq. (2.4-1), we see that the transfer-function matrix $\mathbf{G}(s)$ is given by

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} \qquad (2.4\text{-}10)$$

In the case of the single input–single output system, this result reduces to

$$G(s) = \mathbf{c}^T(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b} \qquad (2.4\text{-}11)$$

The matrix $(s\mathbf{I} - \mathbf{A})^{-1}$ is commonly referred to as the *resolvent matrix* and is designated by $\mathbf{\Phi}(s)$,

$$\mathbf{\Phi}(s) = (s\mathbf{I} - \mathbf{A})^{-1} \qquad (2.4\text{-}12)$$

In terms of this notation, Eqs. (2.4-10) and (2.4-11) become

$$\mathbf{G}(s) = \mathbf{C}\mathbf{\Phi}(s)\mathbf{B} \qquad (2.4\text{-}13)$$

and

$$G(s) = \mathbf{c}^T\mathbf{\Phi}(s)\mathbf{b} \qquad (2.4\text{-}14)$$

Example 2.4-3 Consider the system represented by the equations

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}u$$
$$y = [1 \quad 0]\mathbf{x}$$

In this case, the matrix $(s\mathbf{I} - \mathbf{A})$ becomes

$$(s\mathbf{I} - \mathbf{A}) = \begin{bmatrix} s & -1 \\ 2 & s+3 \end{bmatrix}$$

and its inverse is

$$\mathbf{\Phi}(s) = (s\mathbf{I} - \mathbf{A})^{-1} = \frac{\text{adj } (s\mathbf{I} - \mathbf{A})}{\det (s\mathbf{I} - \mathbf{A})} = \frac{\begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix}}{s^2 + 3s + 2}$$

The transfer function of the system is therefore

$$G(s) = \mathbf{c}^T\mathbf{\Phi}(s)\mathbf{b} = \frac{[1 \quad 0]\begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix}}{s^2+3s+2} = \frac{1}{s^2+3s+2}$$

In the above example, we observe that the determinant of the matrix $(s\mathbf{I} - \mathbf{A})$ is equal to the denominator polynomial of $G(s)$. This is always true for single input–single output systems. We shall discuss this fact further in Sec. 2.6.

Although Eqs. (2.4-13) and (2.4-14) provide a direct method for determining the transfer function of a system from a state-variable representation of the system, it is generally not the most efficient method. This is due to the fact that one must invert the matrix $(s\mathbf{I} - \mathbf{A})$. The inversion of a matrix is never an easy task, and since the elements of the matrix $(s\mathbf{I} - \mathbf{A})$ are functions of s, the job is even more difficult; in addition, it is not easy to program on a computer. Because of this problem, it is often easier to obtain the transfer function by carrying out block-diagram reductions or equivalently signal flow-graph techniques on the block diagram of the state-variable representation.

Consider, for example, the system of Example 2.4-3, which is represented in block-diagram form in Fig. 2.4-6*a*. By reducing the inner loop, we obtain the system of Fig. 2.4-6*b*, which may now be further reduced to obtain the transfer function, as shown in Fig. 2.4-6*c*.

In this simple example, neither the direct method nor the block-diagram approach is difficult to apply. In a more complicated problem, the block-diagram approach could be considerably easier. In fact, for certain types of state-variable representations, the transfer function may often be obtained by inspection using this approach. We shall discuss this feature further in the following sections.

The problem of determining a state-variable representation of a system whose transfer function is known is more complicated than the above problem because an arbitrary choice of state variables must be made. In essence this means that there is an infinite number of state-variable representations for a given system. In the next three sections three of the more common means for choosing the state variables are discussed.

In these sections, we restrict our attention to the single input–single output case. This is done to simplify the discussion of these methods as much as possible and with the feeling that the extension to the general case can be easily made after this special case is mastered.

The problem of state-variable representation of a given transfer

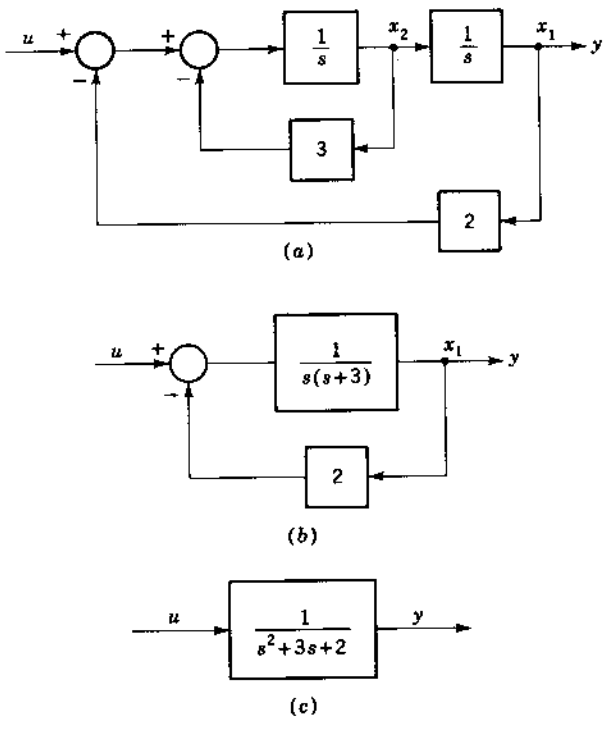

Fig. 2.4-6 (*a*) Original block diagram; (*b*) block diagram after the reduction of the inner loop; (*c*) final block diagram.

function is very closely related to the problem of analog-computer simulation of a transfer function. In fact, the reader familiar with analog-computer simulation will recognize the techniques of the next three sections as nothing more than the *direct*, *parallel*, and *cascade* methods of transfer-function simulation.

The results of this present section provide two valuable checks on any state-variable representation that we may obtain. First, since the system has been assumed to be controllable and observable, it is obvious that the resulting representation must satisfy these assumptions. If this is not true, we have made a mistake in our state-variable representation.

Second, we may always check to see that the transfer function associated with a state-variable representation is the same as the original transfer function. This uniqueness of the transfer-function representation is also helpful in manipulating state-variable representations. Having fixed some of the elements of the state-variable representation, for exam-