CHAPTER 4

State-Space Models

4.1 INTRODUCTION

In our discussion of system descriptions up to this point, we have emphasized and utilized system models that represent the transformation of input signals into output signals. In the case of linear and time-invariant (LTI) models, our focus has been on the impulse response, frequency response and transfer function. Such input-output models do not directly consider the internal behavior of the systems they model.

In this chapter we begin a discussion of system models that considers the internal dynamical behavior of the system as well as the input-output characteristics. Internal behavior can be important for a variety of reasons. For example, in examining issues of stability, a system can be stable from an input-output perspective but hidden internal variables may be unstable, yielding what we would want to think of as unstable system behavior.

We introduce in this chapter an important model description that highlights internal behavior of the system and is specially suited to representing causal systems for real-time applications such as control. Specifically, we introduce state-space models for finite-memory (or lumped) causal systems. These models exist for both continuous-time (CT) and discrete-time (DT) systems, and for nonlinear, time-varying systems — although our focus will be on the LTI case.

Having a state-space model for a causal DT system (similar considerations apply in the CT case) allows us to answer a question that gets asked about such systems in many settings: Given the input value $x[n]$ at some arbitrary time n, how much information do we really need about past inputs, i.e., about $x[k]$ for $k < n$, in order to determine the present output $y[n]$? As the system is causal, we know that having all past $x[k]$ (in addition to $x[n]$) will suffice, but do we actually need this much information? This question addresses the issue of memory in the system, and is a worthwhile question for a variety of reasons.

For example, the answer gives us an idea of the complexity, or number of degrees of freedom, associated with the dynamic behavior of the system. The more information we need about past inputs in order to determine the present output, the richer the variety of possible output behaviors, i.e., the more ways we can be surprised in the absence of information about the past.

Furthermore, in a control application, the answer to the above question suggests the required degree of complexity of the controller, because the controller has to

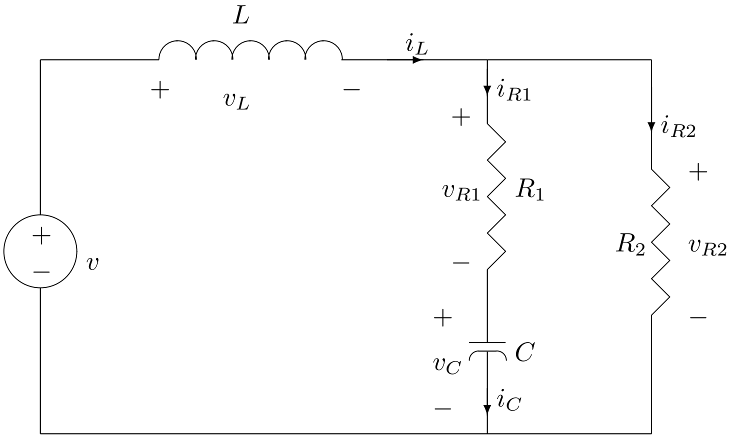

FIGURE 4.1 RLC circuit.

remember enough about the past to determine the effects of present control actions on the response of the system. In addition, for a computer algorithm that acts causally on a data stream, the answer to the above question suggests how much memory will be needed to run the algorithm.

With a state-space description, everything about the past that is relevant to the present and future is summarized in the present state, i.e., in the present values of a set of state variables. The number of state variables, which we refer to as the order of the model, thus indicates the amount of memory or degree of complexity associated with the system or model.

4.2 INPUT-OUTPUT AND INTERNAL DESCRIPTIONS

As a prelude to developing the general form of a state-space model for an LTI system, we present two examples, one in CT and the other in DT.

4.2.1 An RLC circuit

Consider the RLC circuit shown in Figure 4.1. We have labeled all the component voltages and currents in the figure.

The defining equations for the components are:

$$\begin{aligned} L\frac{di_L(t)}{dt} &= v_L(t) \\ C\frac{dv_C(t)}{dt} &= i_C(t) \\ v_{R1}(t) &= R_1 i_{R1}(t) \\ v_{R2}(t) &= R_2 i_{R2}(t)\ , \end{aligned} \tag{4.1}$$

while the voltage source is defined by the condition that its voltage is $v(t)$ regardless of its current $i(t)$. Kirchhoff's voltage and current laws yield

$$\begin{aligned} v(t) &= v_L(t) + v_{R2}(t) \\ v_{R2}(t) &= v_{R1}(t) + v_C(t) \\ i(t) &= i_L(t) \\ i_L(t) &= i_{R1}(t) + i_{R2}(t) \\ i_{R1}(t) &= i_C(t) \ . \end{aligned} \tag{4.2}$$

All these equations together constitute a detailed and explicit representation of the circuit.

Let us take the voltage source $v(t)$ as the input to the circuit; we shall also denote this by $x(t)$, our standard symbol for inputs. Choose any of the circuit voltages or currents as the output — let us choose $v_{R_2}(t)$ for this example, and also denote it by $y(t)$, our standard symbol for outputs. We can then combine (4.1) and (4.2) using, for example, Laplace transforms, in order to obtain a transfer function or a linear constant-coefficient differential equation relating the input and output. The coefficients in the transfer function or differential equation will, of course be functions of the values of the components in the circuit. The resulting transfer function $H(s)$ from input to output is

$$H(s) = \frac{Y(s)}{X(s)} = \frac{\alpha\Big(\frac{R_1}{L}s + \frac{1}{LC}\Big)}{s^2 + \alpha\Big(\frac{1}{R_2C} + \frac{R_1}{L}\Big)s + \alpha\frac{1}{LC}} \tag{4.3}$$

where α denotes the ratio $R_2/(R_1 + R_2)$. The corresponding input-output differential equation is

$$\frac{d^2y(t)}{dt^2} + \alpha\Big(\frac{1}{R_2C} + \frac{R_1}{L}\Big)\frac{dy(t)}{dt} + \alpha\Big(\frac{1}{LC}\Big)y(t) = \alpha\Big(\frac{R_1}{L}\Big)\frac{dx(t)}{dt} + \alpha\Big(\frac{1}{LC}\Big)x(t) \ . \tag{4.4}$$

An important characteristic of a circuit such as in Figure 4.1 is that the behavior for a time interval beginning at some t is completely determined by the input trajectory in that interval as well as the inductor currents and capacitor voltages at time t. Thus, for the specific circuit in Figure 4.1, in determining the response for times $\geq t$, the relevant past history of the system is summarized in $i_L(t)$ and $v_C(t)$. The inductor currents and capacitor voltages in such a circuit at any time t are commonly referred to as state variables, and the particular set of values they take constitutes the state of the system at time t. This state, together with the input from t onwards, are sufficient to completely determine the response at and beyond t.

The concept of state for dynamical systems is an extremely powerful one. For the RLC circuit of Figure 4.1 it motivates us to reduce the full set of equations (4.1) and (4.2) into a set of equations involving just the input, output, and internal variables $i_L(t)$ and $v_C(t)$. Specifically, a description of the desired form can be found by appropriately eliminating the other variables from (4.1) and (4.2), although some attention is required in order to carry out the elimination efficiently. With this, we arrive at a condensed description, written here using matrix notation, and in a format that we shall encounter frequently in this chapter and the next two:

$$\begin{pmatrix} di_L(t)/dt \\ dv_C(t)/dt \end{pmatrix} = \begin{pmatrix} -\alpha R_1/L & -\alpha/L \\ \alpha/C & -1/(R_1+R_2)C \end{pmatrix} \begin{pmatrix} i_L(t) \\ v_C(t) \end{pmatrix} + \begin{pmatrix} 1/L \\ 0 \end{pmatrix} v(t)\ . \tag{4.5}$$

The use of matrix notation is a convenience; we could of course have simply written the above description as two separate but coupled first-order differential equations with constant coefficients.

We shall come to appreciate the properties and advantages of a description in the form of (4.5), referred to as a CT (and, in this case, LTI) state-space form. Its key feature is that it expresses the rates of change of the state variables at any time t as functions (in this case, LTI functions) of their values and those of the input at that same time t.

As we shall see later, the state-space description can be used to solve for the state variables $i_L(t)$ and $v_C(t)$, given the input $v(t)$ and appropriate auxiliary information (specifically, initial conditions on the state variables). Furthermore, knowledge of $i_L(t)$, $v_C(t)$ and $v(t)$ suffices to reconstruct all the other voltages and currents in the circuit at time t. In particular, any output variable can be written in terms of the retained variables. For instance, if the output of interest for this circuit is the voltage $v_{R2}(t)$ across R_2, we can write (again in matrix notation)

$$v_{R2}(t) = \begin{pmatrix} \alpha R_1 & \alpha \end{pmatrix} \begin{pmatrix} i_L(t) \\ v_C(t) \end{pmatrix} + \begin{pmatrix} 0 \end{pmatrix} v(t)\ . \tag{4.6}$$

For this particular example, the output does not involve the input $v(t)$ directly — hence the term $(\,0\,)\ v(t)$ in the above output equation — but in the general case the output equation will involve present values of any inputs in addition to present values of the state variables.

4.2.2 A delay-adder-gain system

For DT systems, the role of state variables is similar to the role discussed in the preceding subsection for CT systems. We illustrate this with the system described by the delay-adder-gain block diagram shown in Figure 4.2.2. The corresponding detailed equations relating the indicated signals are

$$\begin{aligned} q_1[n+1] &= q_2[n] \\ q_2[n+1] &= p[n] \\ p[n] &= x[n] - (1/2)q_1[n] + (3/2)q_2[n] \\ y[n] &= q_2[n] + p[n]\ . \end{aligned} \tag{4.7}$$

The equations in (4.7) can be combined together using, for example, z-transform methods, to obtain the transfer function or linear constant-coefficient difference equation relating input and output:

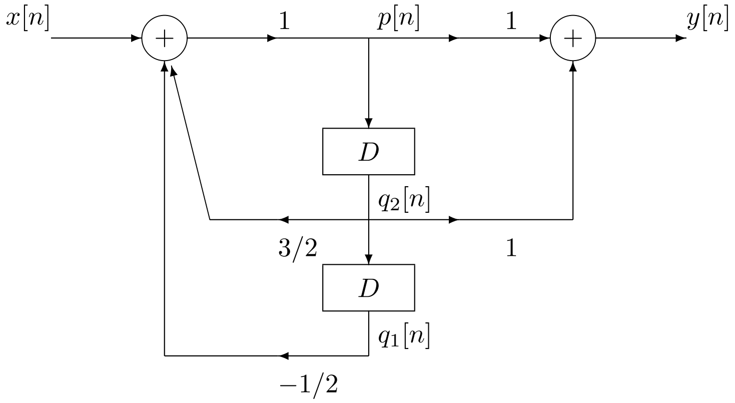

FIGURE 4.2 Delay-adder-gain block diagram.

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 + z^{-1}}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}} \tag{4.8}$$

and

$$y[n] - \frac{3}{2}y[n-1] + \frac{1}{2}y[n-2] = x[n] + x[n-1] \ . \tag{4.9}$$

The response of the system in an interval of time $\geq n$ is completely determined by the input for times $\geq n$ and the values $q_1[n]$ and $q_2[n]$ that are stored at the outputs of the delay elements at time n. Thus, as with the energy storage elements in the circuit of Figure 4.1, the delay elements in the delay-adder-gain system capture the state of the system at any time, i.e., summarize all the past history with respect to how it affects the present and future response of the system. Consequently, we condense (4.7) in terms of only the input, output and state variables to obtain the following matrix equations:

$$\begin{pmatrix} q_1[n+1] \\ q_2[n+1] \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1/2 & 3/2 \end{pmatrix} \begin{pmatrix} q_1[n] \\ q_2[n] \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} x[n] \tag{4.10}$$

$$y[n] = \begin{pmatrix} -1/2 & 5/2 \end{pmatrix} \begin{pmatrix} q_1[n] \\ q_2[n] \end{pmatrix} + (1)x[n] \ . \tag{4.11}$$

In this case it is quite easy to see that, if we are given the values $q_1[n]$ and $q_2[n]$ of the state variables at some time n, and also the input trajectory from n onwards, i.e., $x[n]$ for times $\geq n$, then we can compute the values of the state variables for all times $> n$, and the output for all times $\geq n$. All that is needed is to iteratively apply (4.10) to find $q_1[n+1]$ and $q_2[n+1]$, then $q_1[n+2]$ and $q_2[n+2]$, and so on for increasing time arguments, and to use (4.11) at each time to find the output.

4.3 STATE-SPACE MODELS

As illustrated in Sections 4.2.1 and 4.2.2, it is often natural and convenient, when studying or modeling physical systems, to focus not just on the input and output signals but rather to describe the interaction and time-evolution of several key variables or signals that are associated with the various component processes internal to the system. Assembling the descriptions of these components and their interconnections leads to a description that is richer than an input–output description. In particular, in Sections 4.2.1 and 4.2.2 the description is in terms of the time evolution of variables referred to as the state variables, which completely capture at any time the past history of the system as it affects the present and future response. We turn now to a more formal definition of state-space models in the DT and CT cases, followed by a discussion of two defining characteristics of such models.

4.3.1 DT State-Space Models

A state-space model is built around a set of state variables; the number of state variables in a model or system is referred to as its order. Although we shall later cite examples of distributed or infinite-order systems, we shall only deal with state-space models of finite order, which are also referred to as lumped systems. For an Lth-order model in the DT case, we shall generically denote the values of the L state variables at time n by $q_1[n], q_2[n], \cdots, q_L[n]$. It is convenient to gather these variables into a state vector:

$$\mathbf{q}[n] = \begin{pmatrix} q_1[n] \\ q_2[n] \\ \vdots \\ q_L[n] \end{pmatrix} . \tag{4.12}$$

The value of this vector constitutes the state of the model or system at time n.

A DT LTI state-space model with single (i.e., scalar) input $x[n]$ and single output $y[n]$ takes the following form, written in compact matrix notation:

$$\mathbf{q}[n+1] = \mathbf{A}\mathbf{q}[n] + \mathbf{b}x[n] \ , \tag{4.13}$$

$$y[n] = \mathbf{c}^T\mathbf{q}[n] + \mathtt{d}x[n] \ . \tag{4.14}$$

In (4.13), $\mathbf{A}$ is an $L \times L$ matrix, $\mathbf{b}$ is an $L \times 1$ matrix or column-vector, and $\mathbf{c}^T$ is a $1 \times L$ matrix or row-vector, with the superscript ^{T} denoting transposition of the column vector $\mathbf{c}$ into the desired row vector. The quantity $\mathtt{d}$ is a 1×1 matrix, i.e., a scalar. The entries of all these matrices in the case of an LTI model are numbers or constants or parameters, so they do not vary with n. Note that the model we arrived at in (4.10) and (4.11) of Section 4.2.2 has precisely the above form. We refer to (4.13) as the state evolution equation, and to (4.14) as the output equation. These equations respectively express the next state and the current output at any time as an LTI combination of the current state variables and current input.

Generalizations of the DT LTI State-Space Model. There are various natural generalizations of the above DT LTI single-input, single-output state-space model. A multi-input DT LTI state-space model replaces the single term $\mathbf{b}x[n]$ in (4.13) by a sum of terms, $\mathbf{b}_1x_1[n] + \cdots + \mathbf{b}_Mx_M[n]$, where M is the number of inputs. This corresponds to replacing the scalar input $x[n]$ by an M-component vector $\mathbf{x}[n]$ of inputs, with a corresponding change of $\mathbf{b}$ to a matrix $\mathbf{B}$ of dimension $L \times M$. Similarly, for a multi-output DT LTI state-space model, the single output equation (4.14) is replaced by a collection of such output equations, one for each of the P outputs. Equivalently, the scalar output $y[n]$ is replaced by a P-component vector $\mathbf{y}[n]$ of outputs, with a corresponding change of $\mathbf{c}^T$ and $\mathtt{d}$ to matrices $\mathbf{C}^T$ and $\mathbf{D}$ of dimension $P \times L$ and $P \times M$ respectively.

A linear but time-varying DT state-space model takes the same form as in (4.13) and (4.14) above, except that some or all of the matrix entries are time-varying. A linear but periodically varying model is a special case of this, with matrix entries that all vary periodically with a common period. A nonlinear, time-invariant model expresses $\mathbf{q}[n+1]$ and $\mathbf{y}[n]$ as nonlinear but time-invariant functions of $\mathbf{q}[n]$ and $\mathbf{x}[n]$, rather than as the LTI functions embodied by the matrix expressions on the right-hand-sides of (4.13) and (4.14). A nonlinear, time-varying model expresses $\mathbf{q}[n+1]$ and $\mathbf{y}[n]$ as nonlinear, time-varying functions of $\mathbf{q}[n]$ and $\mathbf{x}[n]$, and one can also define nonlinear, periodically varying models as a particular case in which the time-variations are periodic with a common period.

4.3.2 CT State-Space Models

Continuous-time state-space descriptions take a very similar form to the DT case. We denote the state variables as $q_i(t)$, $i = 1, 2, ..., L$, and the state vector as

$$\mathbf{q}(t) = \begin{pmatrix} q_1(t) \\ q_2(t) \\ \vdots \\ q_L(t) \end{pmatrix} . \tag{4.15}$$

Whereas in the DT case the state evolution equation expresses the state vector at the next time step in terms of the current state vector and input values, in CT the state evolution equation expresses the rates of change (i.e., derivatives) of each of the state variables as functions of the present state and inputs. The general Lth-order CT LTI state-space representation thus takes the form

$$\frac{d\mathbf{q}(t)}{dt} = \dot{\mathbf{q}}(t) = \mathbf{A}\mathbf{q}(t) + \mathbf{b}x(t) \ , \tag{4.16}$$

$$y(t) = \mathbf{c}^T\mathbf{q}(t) + \mathtt{d}x(t) \ , \tag{4.17}$$

where $d\mathbf{q}(t)/dt = \dot{\mathbf{q}}(t)$ denotes the vector whose entries are the derivatives, $dq_i(t)/dt$, of the corresponding entries, $q_i(t)$, of $\mathbf{q}(t)$. Note that the model in (4.5) and (4.6) of Section 4.2.1 is precisely of the above form.

Generalizations to multi-input and multi-output models, and to linear and nonlinear time-varying or periodic models, can be described just as in the case of DT systems, by appropriately relaxing the restrictions on the form of the right-hand sides of (4.16), (4.17). We shall see an example of a nonlinear time-invariant state-space model in Section 1.

4.3.3 Characteristics of State-Space Models

The designations of "state" for $\mathbf{q}[n]$ or $\mathbf{q}(t)$, and of "state-space description" for (4.13), (4.14) and (4.16), (4.17) — or for the various generalizations of these equations — follow from the following two key properties of such models.

State Evolution Property: The state at any initial time, along with the inputs over any interval from that initial time onwards, determine the state over that entire interval. Everything about the past that is relevant to the future state is embodied in the present state.

Instantaneous Output Property: The outputs at any instant can be written in terms of the state and inputs at that same instant.

The state evolution property is what makes state-space models particularly well suited to describing causal systems. In the DT case, the validity of this state evolution property is evident from the state evolution equation (4.13), which allows us to update $\mathbf{q}[n]$ iteratively, going from time n to time $n+1$ using only knowledge of the present state and input. The same argument can also be applied to the generalizations of DT LTI models that we outlined earlier.

The state evolution property should seem intuitively reasonable in the CT case as well. Specifically, knowledge of both the state and the rate of change of the state at any instant allows us to compute the state after a small increment in time. Taking this small step forward, we can re-evaluate the rate of change of the state, and step forward again. A more detailed proof of this property in the general nonlinear and/or time-varying CT case essentially proceeds this way, and is treated in texts that deal with the existence and uniqueness of solutions of differential equations. These more careful treatments also make clear what additional conditions are needed for the state evolution property to hold in the general case. However, the CT LTI case is much simpler, and we shall demonstrate the state evolution property for this class of state-space models in the next chapter, when we show how to explicitly solve for the behavior of such systems.

The instantaneous output property is immediately evident from the output equations (4.14), (4.17). It also holds for the various generalizations of basic single-input, single-output LTI models that we listed earlier.

The two properties above may be considered the defining characteristics of a state-space model. In effect, what we do in setting up a state-space model is to introduce the additional vector of state variables $\mathbf{q}[n]$ or $\mathbf{q}(t)$, to supplement the input variables $x[n]$ or $x(t)$ and output variables $y[n]$ or $y(t)$. This supplementation is done precisely in order to obtain a description that satisfies the two properties above.

Often there are natural choices of state variables suggested directly by the particular context or application. In both DT and CT cases, state variables are related to the "memory" of the system. In many physical situations involving CT models, the state variables are associated with energy storage, because this is what is carried over from the past to the future. Natural state variables for electrical circuits are thus the inductor currents and capacitor voltages, as turned out to be the case in Section 4.2.1. For mechanical systems, natural state variables are the positions and velocities of all the masses in the system (corresponding respectively to potential energy and kinetic energy variables), as we will see in later examples. In the case of a CT integrator-adder-gain block diagram, the natural state variables are associated with the outputs of the integrators, just as in the DT case the natural state variables of a delay-adder-gain model are the outputs of the delay elements, as was the case in the example of Section 4.2.2.

In any of the above contexts, one can choose any alternative set of state variables that together contain exactly the same information. There are also situations in which there is no particularly natural or compelling choice of state variables, but in which it is still possible to define supplementary variables that enable a valid state-space description to be obtained.

Our discussion of the two key properties above — and particularly of the role of the state vector in separating past and future — suggests that state-space models are particularly suited to describing causal systems. In fact, state-space models are almost never used to describe non-causal systems. We shall always assume here, when dealing with state-space models, that they represent causal systems. Although causality is not a central issue in analyzing many aspects of communication or signal processing systems, particularly in non-real-time contexts, it is generally central to simulation and control design for dynamic systems. It is accordingly in such dynamics and control settings that state-space descriptions find their greatest value and use.

4.4 EQUILIBRIA AND LINEARIZATION OF NONLINEAR STATE-SPACE MODELS

An LTI state-space model most commonly arises as an approximate description of the local (or "small-signal") behavior of a nonlinear time-invariant model, for small deviations of its state variables and inputs from a set of constant equilibrium values. In this section we present the conditions that define equilibrium, and describe the role of linearization in obtaining the small-signal model at this equilibrium.

4.4.1 Equilibrium

To make things concrete, consider a DT 3rd-order nonlinear time-invariant state-space system, of the form

$$\begin{aligned} q_1[n+1] &= f_1\Big(q_1[n], q_2[n], q_3[n], x[n]\Big) \\ q_2[n+1] &= f_2\Big(q_1[n], q_2[n], q_3[n], x[n]\Big) \\ q_3[n+1] &= f_3\Big(q_1[n], q_2[n], q_3[n], x[n]\Big) \ , \end{aligned} \tag{4.18}$$

with the output $y[n]$ defined by the equation

$$y[n] = g\Big(q_1[n], q_2[n], q_3[n], x[n]\Big) \ . \tag{4.19}$$

The state evolution functions $f_i(\,\cdot\,)$, for $i = 1, 2, 3$, and the output function $g(\,\cdot\,)$ are all time-invariant nonlinear functions of the three state variables $q_i[n]$ and the input $x[n]$. (Time-invariance of the functions simply means that they combine their arguments in the same way, regardless of the time index n.) The generalization to an Lth-order description should be clear. In vector notation, we can simply write

$$\mathbf{q}[n+1] = \mathbf{f}\Big(\mathbf{q}[n], x[n]\Big) \ , \qquad y[n] = g\Big(\mathbf{q}[n], x[n]\Big) \ , \tag{4.20}$$

where for our 3rd-order case

$$\mathbf{f}(\,\cdot\,) = \begin{bmatrix} f_1(\,\cdot\,) \\ f_2(\,\cdot\,) \\ f_3(\,\cdot\,) \end{bmatrix} \ . \tag{4.21}$$

Suppose now that the input $x[n]$ is constant at the value $\overline{x}$ for all n. The corresponding state equilibrium is a state value $\overline{\mathbf{q}}$ with the property that if $\mathbf{q}[n] = \overline{\mathbf{q}}$ with $x[n] = \overline{x}$, then $\mathbf{q}[n+1] = \overline{\mathbf{q}}$. Equivalently, the point $\overline{\mathbf{q}}$ in the state space is an equilibrium (or equilibrium point) if, with $x[n] \equiv \overline{x}$ for all n and with the system initialized at $\overline{\mathbf{q}}$, the system subsequently remains fixed at $\overline{\mathbf{q}}$. From (4.20), this is equivalent to requiring

$$\overline{\mathbf{q}} = \mathbf{f}(\overline{\mathbf{q}}, \overline{x}) \ . \tag{4.22}$$

The corresponding equilibrium output is

$$\overline{y} = g(\overline{\mathbf{q}}, \overline{x}) \ . \tag{4.23}$$

In defining an equilibrium, no consideration is given to what the system behavior is in the vicinity of the equilibrium point, i.e., of how the system will behave if initialized close to — rather than exactly at — the point $\overline{\mathbf{q}}$. That issue is picked up when one discusses local behavior, and in particular local stability, around the equilibrium.

In the 3rd-order case above, and given $\overline{x}$, we would find the equilibrium by solving the following system of three simultaneous nonlinear equations in three unknowns:

$$\begin{aligned}
\overline{q_1} &= f_1(\overline{q_1}, \overline{q_2}, \overline{q_3}, \overline{x}) \\
\overline{q_2} &= f_2(\overline{q_1}, \overline{q_2}, \overline{q_3}, \overline{x}) \\
\overline{q_3} &= f_3(\overline{q_1}, \overline{q_2}, \overline{q_3}, \overline{x}) \ . \qquad (4.24)
\end{aligned}$$

There is no guarantee in general that an equilibrium exists for the specified constant input $\overline{x}$, and there is no guarantee of a unique equilibrium when an equilibrium does exist.

We can apply the same idea to CT nonlinear time-invariant state-space systems. Again consider the concrete case of a 3rd-order system:

$$\begin{aligned}
\dot{q}_1(t) &= f_1\Big(q_1(t), q_2(t), q_3(t), x(t)\Big) \\
\dot{q}_2(t) &= f_1\Big(q_1(t), q_2(t), q_3(t), x(t)\Big) \\
\dot{q}_3(t) &= f_1\Big(q_1(t), q_2(t), q_3(t), x(t)\Big) \ , \qquad (4.25)
\end{aligned}$$

with

$$y(t) = g\Big(q_1(t), q_2(t), q_3(t), x(t)\Big) \ , \qquad (4.26)$$

or in vector notation,

$$\dot{\mathbf{q}}(t) = \mathbf{f}\Big(\mathbf{q}(t), x(t)\Big) \ , \qquad y(t) = g\Big(\mathbf{q}(t), x(t)\Big) \ . \qquad (4.27)$$

Define the equilibrium $\overline{\mathbf{q}}$ again as a state value that the system does not move from when initialized there, and when the input is fixed at $x(t) = \overline{x}$. In the CT case, what this requires is that the rate of change of the state, namely $\dot{\mathbf{q}}(t)$, is zero at the equilibrium, which yields the condition

$$\mathbf{0} = \mathbf{f}(\overline{\mathbf{q}}, \overline{x}) \ . \qquad (4.28)$$

For the 3rd-order case, this condition takes the form

$$\begin{aligned}
0 &= f_1(\overline{q_1}, \overline{q_2}, \overline{q_3}, \overline{x}) \\
0 &= f_2(\overline{q_1}, \overline{q_2}, \overline{q_3}, \overline{x}) \\
0 &= f_3(\overline{q_1}, \overline{q_2}, \overline{q_3}, \overline{x}) \ , \qquad (4.29)
\end{aligned}$$

which is again a set of three simultaneous nonlinear equations in three unknowns, with possibly no solution for a specified $\overline{x}$, or one solution, or many.

4.4.2 Linearization

We now examine system behavior in the vicinity of an equilibrium. Consider once more the 3rd-order DT nonlinear system (4.18), and suppose that instead of $x[n] \equiv \overline{x}$, we have $x[n]$ perturbed or deviating from this by a value $\widetilde{x}[n]$, so

$$\widetilde{x}[n] = x[n] - \overline{x} \ . \qquad (4.30)$$

The state variables will correspondingly be perturbed from their respective equilibrium values by amounts denoted by

$$\widetilde{q}_i[n] = q_i[n] - \overline{q}_i \tag{4.31}$$

for $i = 1, 2, 3$ (or more generally $i = 1, \cdots, L$), and the output will be perturbed by

$$\widetilde{y}[n] = y[n] - \overline{y}\ . \tag{4.32}$$

Our objective is to find a model that describes the behavior of these various perturbations from equilibrium.

The key to finding a tractable description of the perturbations or deviations from equilibrium is to assume they are small, thereby permitting the use of truncated Taylor series to provide good approximations to the various nonlinear functions. Truncating the Taylor series to first order, i.e., to terms that are linear in the deviations, is referred to as linearization, and produces LTI state-space models in our setting.

To linearize the original DT 3rd-order nonlinear model (4.18), we rewrite the variables appearing in that model in terms of the perturbations, using the quantities defined in (4.30), (4.31), and then expand in Taylor series to first order around the equilibrium values:

$$\begin{aligned}\overline{q}_i + \widetilde{q}_i[n+1] &= f_i\Big(\overline{q}_1 + \widetilde{q}_1[n], \overline{q}_2 + \widetilde{q}_2[n], \overline{q}_3 + \widetilde{q}_3[n], \overline{x} + \widetilde{x}[n]\Big) \qquad \text{for } i = 1, 2, 4 \\ &\approx f_i(\overline{q}_1, \overline{q}_2, \overline{q}_3, \overline{x}) + \frac{\partial f_i}{\partial q_1}\widetilde{q}_1[n] + \frac{\partial f_i}{\partial q_2}\widetilde{q}_2[n] + \frac{\partial f_i}{\partial q_3}\widetilde{q}_3[n] + \frac{\partial f_i}{\partial x}\widetilde{x}[n]\ .\end{aligned} \tag{4.33}$$

All the partial derivatives above are evaluated at the equilibrium values, and are therefore constants, not dependent on the time index n. (Also note that the partial derivatives above are with respect to the continuously variable state and input arguments; there are no "derivatives" taken with respect to n, the discretely varying time index!) The definition of the equilibrium values in (4.24) shows that the term $\overline{q}_i$ on the left of the above set of expressions exactly equals the term $f_i(\overline{q}_1, \overline{q}_2, \overline{q}_3, \overline{x})$ on the right, so what remains is the approximate relation

$$\widetilde{q}_i[n+1] \approx \frac{\partial f_i}{\partial q_1}\widetilde{q}_1[n] + \frac{\partial f_i}{\partial q_2}\widetilde{q}_2[n] + \frac{\partial f_i}{\partial q_3}\widetilde{q}_3[n] + \frac{\partial f_i}{\partial x}\widetilde{x}[n] \tag{4.34}$$

for $i = 1, 2, 3$. Replacing the approximate equality sign ($\approx$) by the equality sign ($=$) in this set of expressions produces what is termed the linearized model at the equilibrium point. This linearized model approximately describes small perturbations away from the equilibrium point.

We may write the linearized model in matrix form:

$$\underbrace{\begin{bmatrix} \widetilde{q}_1[n+1] \\ \widetilde{q}_2[n+1] \\ \widetilde{q}_3[n+1] \end{bmatrix}}_{\widetilde{\mathbf{q}}[n+1]} = \underbrace{\begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \frac{\partial f_1}{\partial q_2} & \frac{\partial f_1}{\partial q_3} \\ \frac{\partial f_2}{\partial q_1} & \frac{\partial f_2}{\partial q_2} & \frac{\partial f_2}{\partial q_3} \\ \frac{\partial f_3}{\partial q_1} & \frac{\partial f_3}{\partial q_2} & \frac{\partial f_3}{\partial q_3} \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} \widetilde{q}_1[n] \\ \widetilde{q}_2[n] \\ \widetilde{q}_3[n] \end{bmatrix}}_{\widetilde{\mathbf{q}}[n]} + \underbrace{\begin{bmatrix} \frac{\partial f_1}{\partial x} \\ \frac{\partial f_2}{\partial x} \\ \frac{\partial f_3}{\partial x} \end{bmatrix}}_{\mathbf{b}} \widetilde{x}[n]\ . \tag{4.35}$$

We have therefore arrived at a standard DT LTI state-space description of the state evolution of our linearized model, with state and input variables that are the respective deviations from equilibrium of the underlying nonlinear model. The corresponding output equation is derived similarly, and takes the form

$$\widetilde{y}[n] = \underbrace{\left[\ \tfrac{\partial g}{\partial q_1}\ \tfrac{\partial g}{\partial q_2}\ \tfrac{\partial g}{\partial q_3}\ \right]}_{\mathbf{c}^T} \widetilde{\mathbf{q}}[n] + \underbrace{\frac{\partial g}{\partial x}}_{\mathtt{d}}\, \widetilde{x}[n]\ . \tag{4.36}$$

The matrix of partial derivatives denoted by $\mathbf{A}$ in (4.35) is also called a Jacobian matrix, and denoted in matrix-vector notation by

$$\mathbf{A} = \left[\frac{\partial \mathbf{f}}{\partial \mathbf{q}}\right]_{\overline{\mathbf{q}},\overline{x}} . \tag{4.37}$$

The entry in its ith row and jth column is the partial derivative $\partial f_i(\,\cdot\,)/\partial q_j$, evaluated at the equilibrium values of the state and input variables. Similarly,

$$\mathbf{b} = \left[\frac{\partial \mathbf{f}}{\partial x}\right]_{\overline{\mathbf{q}},\overline{x}} , \quad \mathbf{c}^T = \left[\frac{\partial g}{\partial \mathbf{q}}\right]_{\overline{\mathbf{q}},\overline{x}} , \quad \mathtt{d} = \left[\frac{\partial g}{\partial x}\right]_{\overline{\mathbf{q}},\overline{x}} . \tag{4.38}$$

The derivation of linearized state-space models in CT follows exactly the same route, except that the CT equilibrium condition is specified by the condition (4.28) rather than (4.22).

EXAMPLE 4.1 A Hoop-and-Beam System

As an example to illustrate the determination of equilibria and linearizations, we consider in this section a nonlinear state-space model for a particular hoop-and-beam system.

The system in Figure 4.3 comprises a beam pivoted at its midpoint, with a hoop that is constrained to maintain contact with the beam but free to roll along it, without slipping. A torque can be applied to the beam, and acts as the control input. Our eventual objective might be to vary the torque in order to bring the hoop to — and maintain it at — a desired position on the beam. We assume that the only measured output that is available for feedback to the controller is the position of the hoop along the beam.

Natural state variables for such a mechanical system are the position and velocity variables associated with each of its degrees of freedom, namely:

- the position $q_1(t)$ of the point of contact of the hoop relative to the center of the beam;
- the angular position $q_2(t)$ of the beam relative to horizontal;
- the translational velocity $q_3(t) = \dot{q}_1(t)$ of the hoop along the beam;
- the angular velocity $q_4(t) = \dot{q}_2(t)$ of the beam.

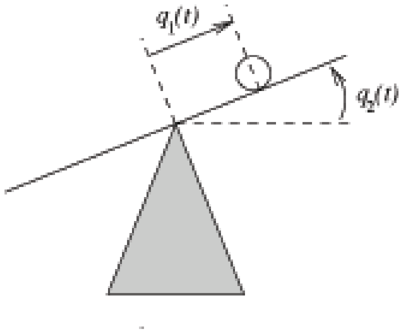

FIGURE 4.3 A hoop rolling on a beam that is free to pivot on its support. The variable $q_1(t)$ is the position of the point of contact of the hoop relative to the center of the beam. The variable $q_2(t)$ is the angle of the beam relative to horizontal.

The measured output is

$$y(t) = q_1(t)\ . \tag{4.39}$$

To specify a state-space model for the system, we express the rate of change of each of these state variables at time t as a function of these variables at t, and as a function of the torque input $x(t)$. We arbitrarily choose the direction of positive torque to be that which would tend to increase the angle $q_2(t)$. The required expressions, which we do not derive here, are most easily obtained using Lagrange's equations of motion, but can also be found by applying the standard and rotational forms of Newton's second law to the system, taking account of the constraint that the hoop rolls without slipping. The resulting nonlinear time-invariant state-space model for the system, with the time argument dropped from the state variables q_i and input x to avoid notational clutter, are:

$$\begin{aligned}
\frac{dq_1}{dt} &= q_3 \\
\frac{dq_2}{dt} &= q_4 \\
\frac{dq_3}{dt} &= \frac{1}{2}\left(q_1 q_4^2 - g\sin(q_2)\right) \\
\frac{dq_4}{dt} &= \frac{mgr\sin(q_2) - mgq_1\cos(q_2) - 2mq_1q_3q_4 + x}{J + mq_1^2}\ .
\end{aligned} \tag{4.40}$$

Here g represents the acceleration due to gravity, m is the mass of the hoop, r is its radius, and J is the moment of inertia of the beam.

Equilibrium values of the model. An equilibrium state of a system is one that can (ideally) be maintained indefinitely without the action of a control input, or more generally with only constant control action. Our control objective might be to design a feedback control system that regulates the hoop-and-beam system to its equilibrium state, with the beam horizontal and the hoop at the center, i.e., with $q_1(t) \equiv 0$ and $q_2(t) \equiv 0$. The possible zero-control equilibrium positions for any CT system described in state-space form can be found by setting the control input and the state derivatives to 0, and then solving for the state variable values.

For the model above, we see that the only zero-control equilibrium position (with the realistic constraint that $-\frac{\pi}{2} < q_2 < \frac{\pi}{2}$) corresponds to a horizontal beam with the hoop at the center, i.e., $q_1 = q_2 = q_3 = q_4 = 0$. If we allow a constant but nonzero control input, it is straightforward to see from (4.40) that it is possible to have an equilibrium state (i.e., unchanging state variables) with a nonzero q_1, but still with q_2, q_3 and q_4 equal to 0.

Linearization for small perturbations. It is generally quite difficult to elucidate in any detail the global or large-signal behavior of a nonlinear model such as (4.40). However, small deviations of the system around an equilibrium, such as might occur in response to small perturbations of the control input from 0, are quite well modeled by a linearized version of the nonlinear model above. As already described in the previous subsection, a linearized model is obtained by approximating all nonlinear terms using first-order Taylor series expansions around the equilibrium. Linearization of a time-invariant model around an equilibrium point always yields a model that is time invariant, as well as being linear. Thus, even though the original nonlinear model may be difficult to work with, the linearized model around an equilibrium point can be analyzed in great detail, using all the methods available to us for LTI systems. Note also that if the original model is in state-space form, the linearization will be in state-space form too, except that its state variables will be the deviations from equilibrium of the original state variables.

Since the equilibrium of interest to us in the hoop-and-beam example corresponds to all state variables being 0, small deviations from this equilibrium correspond to all state variables being small. The linearization is thus easy to obtain without formal expansion into Taylor series. Specifically, as we discard from the nonlinear model (4.40) all terms of higher order than first in any nonlinear combinations of terms, $\sin(q_2)$ gets replaced by q_2, $\cos(q_2)$ gets replaced by 1, and the terms $q_1q_4^2$ and $q_1q_3q_4$ and q_1^2 are eliminated. The result is the following linearized model in state-space form:

$$\begin{aligned}
\frac{dq_1}{dt} &= q_3 \\
\frac{dq_2}{dt} &= q_4 \\
\frac{dq_3}{dt} &= -\frac{g}{2}q_2 \\
\frac{dq_4}{dt} &= \frac{mg(rq_2 - q_1) + x}{J}
\end{aligned} \tag{4.41}$$

This model, along with the defining equation (4.39) for the output (which is already linear and therefore needs no linearization), can be written in the standard matrix form (4.16) and (4.17) for LTI state-space descriptions, with

$$\begin{aligned}
\mathbf{A} &= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -g/2 & 0 & 0 \\ -mg/J & mgr/J & 0 & 0 \end{bmatrix} , \qquad \mathbf{b} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1/J \end{bmatrix} \\
\mathbf{c}^T &= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}
\end{aligned} \tag{4.42}$$

The LTI model is much more tractable than the original nonlinear time-invariant model, and consequently controllers can be designed more systematically and confidently. If the resulting controllers, when applied to the system, manage to ensure that deviations from equilibrium remain small, then our use of the linearized model for design will have been justified.

4.5 STATE-SPACE MODELS FROM INPUT–OUTPUT MODELS

State-space representations can be very naturally and directly generated during the modeling process in a variety of settings, as the examples in Sections 4.2.1 and 4.2.2 suggest. Other — and perhaps more familiar — descriptions can then be derived from them; again, these previous examples showed how input–output descriptions could be obtained from state-space descriptions.

It is also possible to proceed in the reverse direction, constructing state-space descriptions from impulse responses or transfer functions or input–output difference equations, for instance. This is often worthwhile as a prelude to simulation, or filter implementation, or in control design, or simply in order to understand the initial description from another point of view. The following two examples illustrate this reverse process, of synthesizing state-space descriptions from input–output descriptions.

4.5.1 Determining a state-space model from an impulse response or transfer function

Consider the impulse response $h[n]$ of a causal DT LTI system. Causality requires of course that $h[n] = 0$ for $n < 0$. The output $y[n]$ can be related to past and present inputs $x[k]$, $k \leq n$, through the convolution sum

$$y[n] = \sum_{k=-\infty}^{n} h[n-k]\, x[k] \tag{4.43}$$

$$= \Big(\sum_{k=-\infty}^{n-1} h[n-k]\, x[k] \Big) \; + \; h[0]x[n] \; . \tag{4.44}$$

The first term above, namely

$$q[n] = \sum_{k=-\infty}^{n-1} h[n-k]\, x[k] \; , \tag{4.45}$$

represents the effect of the past on the present, at time n, and would therefore seem to have some relation to the notion of a state variable. Updating $q[n]$ to the next time step, we obtain

$$q[n+1] = \sum_{k=-\infty}^{n} h[n+1-k]\, x[k] \; . \tag{4.46}$$

In general, if the impulse response has no special form, the successive values of $q[n]$ have to be recomputed from (4.46) for each n. When we move from n to $n+1$, none of the past inputs $x[k]$ for $k \leq n$, can be discarded, because all of the past will again be needed to compute $q[n+1]$. In other words, the memory of the system is infinite.

However, consider the class of systems for which $h[n]$ has the essentially exponential form

$$h[n] = \beta\, \lambda^{n-1} u[n-1] + \mathtt{d}\, \delta[n] \; , \tag{4.47}$$

where β, λ and $\mathtt{d}$ are constants. The corresponding transfer function is

$$H(z) = \frac{\beta}{z-\lambda} + \mathtt{d} \tag{4.48}$$

(with ROC $|z| > |\lambda|$). What is important about this impulse response is that a time-shifted version of it is simply related to a scaled version of it, because of its DT-exponential form. For this case,

$$q[n] = \beta \sum_{k=-\infty}^{n-1} \lambda^{n-1-k} x[k] \tag{4.49}$$

and

$$q[n+1] = \beta \sum_{k=-\infty}^{n} \lambda^{n-k} x[k] \tag{4.50}$$

$$= \lambda \Big(\beta \sum_{k=-\infty}^{n-1} \lambda^{n-1-k} x[k] \Big) \; + \; \beta x[n]$$

$$= \lambda q[n] + \beta x[n] \; . \tag{4.51}$$

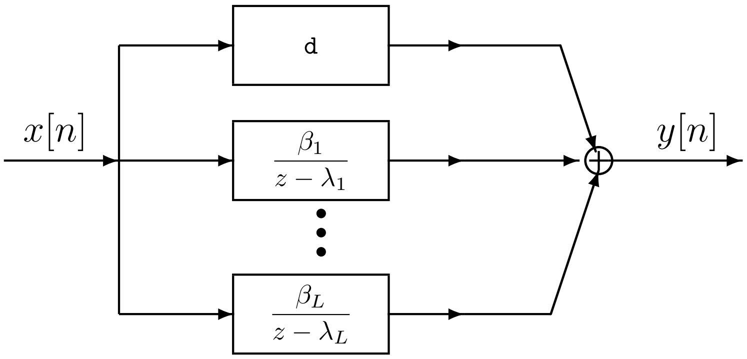

FIGURE 4.4 Decomposition of rational transfer function with distinct poles.

Gathering (4.44) and (4.49) with (4.51) results in a pair of equations that together constitute a state-space description for this system:

$$q[n+1] = \lambda q[n] + \beta x[n] \tag{4.52}$$

$$y[n] = q[n] + \mathtt{d}x[n] \ . \tag{4.53}$$

Let us consider next a similar but higher order system with impulse response:

$$h[n] = (\,\beta_1\lambda_1^{n-1} + \beta_2\lambda_2^{n-1} + \cdots + \beta_L\lambda_L^{n-1}\,)u[n-1] + \mathtt{d}\,\delta[n] \tag{4.54}$$

with the β_i and $\mathtt{d}$ being constants. The corresponding transfer function is

$$H(z) = \Big(\sum_{i=1}^{L} \frac{\beta_i}{z-\lambda_i}\Big) + \mathtt{d} \ . \tag{4.55}$$

By using a partial fraction expansion, the transfer function $H(z)$ of any causal LTI DT system with a rational transfer function can be written in this form, with appropriate choices of the β_i, λ_i, $\mathtt{d}$ and L, provided $H(z)$ has non-repeated — i.e., distinct — poles. Note that although we only treat rational transfer functions $H(z)$ whose numerator and denominator polynomials have real coefficients, the poles of $H(z)$ may include some complex λ_i (and associated β_i), but in each such case its complex conjugate λ_i^* will also be a pole (with associated weighting factor β_i^*), and the sum

$$\beta_i(\lambda_i)^n + \beta_i^*(\lambda_i^*)^n \tag{4.56}$$

will be real.

The block diagram in Figure 4.5.1 shows that this system can be considered as being obtained through the parallel interconnection of subsystems corresponding to the simpler case of (4.47). Motivated by this structure and the treatment of the first-order example, we define a state variable for each of the L subsystems:

$$q_i[n] = \beta_i \sum_{-\infty}^{n-1} \lambda_i^{n-1-k} x[k] \ , \quad i = 1, 2, \ldots, L \ . \tag{4.57}$$

With this, we obtain the following state-evolution equations for the subsystems:

$$q_i[n+1] = \lambda_i q_i[n] + \beta_i x[n] \ , \quad i = 1, 2, \ldots, L \ . \tag{4.58}$$

Also, combining (4.45), (4.53) and (4.54) with the definitions in (4.57), we obtain the output equation

$$y[n] = q_1[n] + q_2[n] + \cdots + q_L[n] + \mathtt{d}\, x[n] \ . \tag{4.59}$$

Equations (4.58) and (4.59) together comprise an Lth-order state-space description of the given system. We can write this state-space description in our standard matrix form (4.13) and (4.14), with

$$\mathbf{A} = \begin{pmatrix} \lambda_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \lambda_2 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & \lambda_L \end{pmatrix} \ , \quad \mathbf{b} = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_L \end{pmatrix} \tag{4.60}$$

$$\mathbf{c}^T = \begin{pmatrix} 1 & 1 & \cdots & \cdots & \cdots & 1 \end{pmatrix} \ . \tag{4.61}$$

The diagonal form of $\mathbf{A}$ in (4.60) reflects the fact that the state evolution equations in this example are decoupled, with each state variable being updated independently according to (4.58). We shall see later how a general description of the form (4.13), (4.14), with a distinct-eigenvalue condition that we shall impose, can actually be transformed to a completely equivalent description in which the new $\mathbf{A}$ matrix is diagonal, as in (4.60). (Note, however, that when there are complex eigenvalues, this diagonal state-space representation will have complex entries.)

4.5.2 Determining a state-space model from an input–output difference equation

Let us examine some ways of representing the following input-output difference equation in state-space form:

$$y[n] + a_1 y[n-1] + a_2 y[n-2] = b_1 x[n-1] + b_2 x[n-2] \ . \tag{4.62}$$

One approach, building on the development in the preceding subsection, is to perform a partial fraction expansion of the 2-pole transfer function associated with this system, and thereby obtain a 2nd-order realization in diagonal form. (If the real coefficients a_1 and a_2 are such that the roots of $z^2 + a_1 z + a_2$ are not real but form a complex conjugate pair, then this diagonal 2nd-order realization will have complex entries.)

For a more direct attempt (and to guarantee a real-valued rather than complex-valued state-space model), consider using as state vector the quantity

$$\mathbf{q}[n] = \begin{pmatrix} y[n-1] \\ y[n-2] \\ x[n-1] \\ x[n-2] \end{pmatrix} \ . \tag{4.63}$$

The corresponding 4th-order state-space model would take the form

$$\begin{aligned}
\mathbf{q}[n+1] = \begin{pmatrix} y[n] \\ y[n-1] \\ x[n] \\ x[n-1] \end{pmatrix} &= \begin{pmatrix} -a_1 & -a_2 & b_1 & b_2 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} y[n-1] \\ y[n-2] \\ x[n-1] \\ x[n-2] \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} x[n] \\
y[n] &= \begin{pmatrix} -a_1 & -a_2 & b_1 & b_2 \end{pmatrix} \begin{pmatrix} y[n-1] \\ y[n-2] \\ x[n-1] \\ x[n-2] \end{pmatrix}
\end{aligned} \tag{4.64}$$

If we are somewhat more careful about our choice of state variables, it is possible to get more economical models. For a 3rd-order model, suppose we pick as state vector

$$\mathbf{q}[n] = \begin{pmatrix} y[n] \\ y[n-1] \\ x[n-1] \end{pmatrix} . \tag{4.65}$$

The corresponding 3rd-order state-space model takes the form

$$\begin{aligned}
\mathbf{q}[n+1] = \begin{pmatrix} y[n+1] \\ y[n] \\ x[n] \end{pmatrix} &= \begin{pmatrix} -a_1 & -a_2 & b_2 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y[n] \\ y[n-1] \\ x[n-1] \end{pmatrix} + \begin{pmatrix} b_1 \\ 0 \\ 1 \end{pmatrix} x[n] \\
y[n] &= \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} y[n] \\ y[n-1] \\ x[n-1] \end{pmatrix}
\end{aligned} \tag{4.66}$$

A still more subtle choice of state variables yields a 2nd-order state-space model by picking

$$\mathbf{q}[n] = \begin{pmatrix} y[n] \\ -a_2 y[n-1] + b_2 x[n-1] \end{pmatrix} . \tag{4.67}$$

The corresponding 2nd-order state-space model takes the form

$$\begin{aligned}
\begin{pmatrix} y[n+1] \\ -a_2 y[n] + b_2 x[n] \end{pmatrix} &= \begin{pmatrix} -a_1 & 1 \\ -a_2 & 0 \end{pmatrix} \begin{pmatrix} y[n] \\ -a_2 y[n-1] + b_2 x[n-1] \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} x[n] \\
y[n] &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} y[n] \\ -a_2 y[n-1] + b_2 x[n-1] \end{pmatrix}
\end{aligned} \tag{4.68}$$

It turns out to be impossible in general to get a state-space description of order lower than 2 in this case. This should not be surprising, in view of the fact that (4.63) is a 2nd-order difference equation, which we know requires two initial conditions in order to solve forwards in time. Notice how, in each of the above cases, we have incorporated the information contained in the original difference equation (4.63) that we started with.

6.011 Introduction to Communication, Control, and Signal Processing
Spring 2010