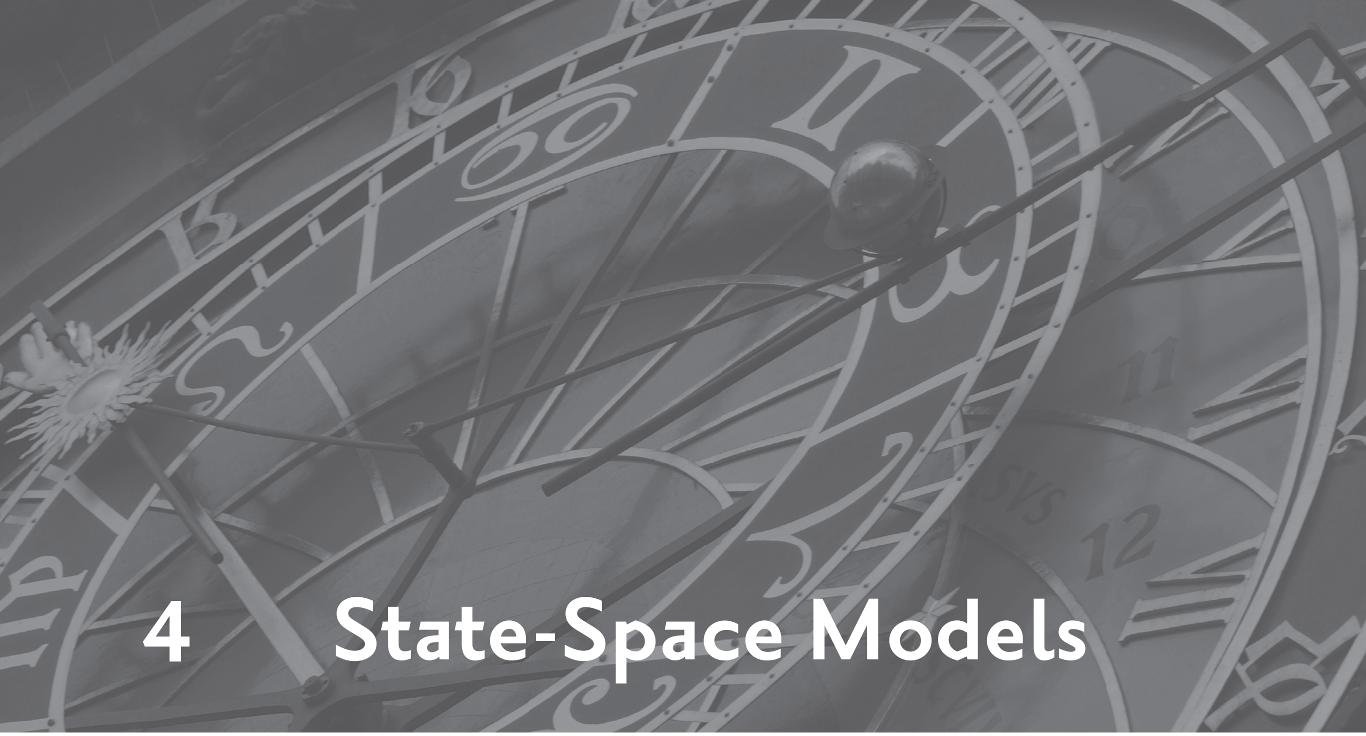
Signals, Systems & Inference

Alan V. Oppenheim & George C. Verghese

Prentice Hall Signal Processing Series | Alan V. Oppenheim, Series Editor

The content of this PDF includes the first 10
pages of Chapter 4 and Chapter 10 respectively.
Oppenheim / Verghese Signals, Systems and Inference, 1/E
L-ENGINEERING AND COMPUTER SCIENCE
ISBN-10: 0133943283 | ISBN-13: 9780133943283
This work is protected by United States copyright laws
and is provided solely for the use of instructors in teaching
their courses and assessing student learning. Dissemination or
sale of any part of this work (including on the World Wide Web)
will destroy the integrity of the work and is not permitted.



4 State-Space Models

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

Internal behavior can be important for a variety of reasons. For instance, in examining issues of stability, a system model can be stable from an input-output perspective, yet internal variables may display unstable behavior. This chapter begins a discussion of system models that display the internal dynamical behavior of the system as well as the input-output characteristics. The discussion is illustrated by numerous examples. The study of such models and their applications continues through Chapters 5 and 6 as well.

4.1 SYSTEM MEMORY

In this chapter we introduce an important model description—the state-space model—that highlights the internal behavior of a system and is especially suited to representing causal systems, particularly for real-time applications such as control. These models arise in both continuous-time (CT) and discrete-time (DT) forms. In general they can be nonlinear and time-varying, although we will focus on the LTI case.

A state-space model for a causal system answers a question asked about such systems in many settings. We pose the question for the causal DT case, though it can also be asked for causal CT systems: given the input value $x[n]$ at some arbitrary time n, how much needs to be known about past values of the input, that is, about $x[k]$ for $k < n$, in order to determine the present output $y[n]$? As the system is causal, having all past values $x[k]$, in addition to $x[n]$, will suffice, but the issue is whether all past $x[k]$ are actually needed.

The above question addresses the issue of memory in the system, and is worthwhile for a variety of reasons. For example, the answer conveys an idea of the complexity, or number of degrees of freedom, associated with the dynamic behavior of the system. The more we need to know about past inputs in order to determine the present output, the richer the variety of possible output behaviors, and the more ways one can be surprised in the absence of knowledge of the past. We will only consider systems with a finite number of degrees of freedom, or with finite-dimensional memory; these are often referred to as lumped systems.

One application in which the above question arises is in implementing a computer algorithm that acts causally on a data stream. Thinking of the algorithm as a system, the answer to the question indicates how much memory will be needed to run the algorithm. In a control application, the answer to the memory question above suggests the required level of complexity for the controller of a given system. The controller has to remember enough about the past to determine the effects of present control actions on the response of the system.

With a state-space description, everything about the past that is relevant to the present and future is summarized in the present values of a finite set of state variables. These values together specify the present state of the system. We are interested in the case of real-valued state variables. The number of state variables, also referred to as the order of the state-space description, indicates the number of degrees of freedom, or the dimension of the memory, associated with the system or model.

4.2 ILLUSTRATIVE EXAMPLES

As a prelude to developing the general form of a state-space model, this section presents in some detail a few CT and DT examples. In addition to illustrating the process of building a state-space model, these examples will suggest how state-space descriptions arise in a variety of contexts. This section may alternatively be read after the more general presentation of state-space models in Section 4.3. Several further examples appear later in the chapter.

To begin, we examine a mechanical system that, despite its simplicity, is rich enough to bring out typical features of a CT state-space model, and serves as a prototype for a variety of other systems.

Example 4.1 Inverted Pendulum

Consider the inverted pendulum shown in Figure 4.1. The pendulum is rigid, with mass m, and can rotate about the pivot at its base, moving in the plane orthogonal to the pivot axis. The distance from the pivot to the center of mass is ℓ, and the pendulum's moment of inertia about the pivot is $\mathcal{I}$. These parameters are all assumed constant.

The line connecting the pivot to the center of mass is at an angle $\theta(t)$ at time t, measured clockwise from the vertical. An external torque is applied to the pendulum around the axis of the pivot. We treat this torque as the input to our system, and denote it by $x(t)$, taken as positive when it acts counterclockwise.

Suppose the system output variable of interest, $y(t)$, is just the pendulum angle, so that $y(t) = \theta(t)$. In a typical control application, one might want to manipulate $x(t)$—in response to measurements that are fed back to the controller—so as to maintain $y(t)$ near the value 0, thus balancing the inverted pendulum vertically.

The external torque is opposed by the torque due to the acceleration g of gravity acting on the mass, which produces a clockwise torque of value $mg\ell\sin(\theta(t))$. Finally, assume a frictional torque that opposes the motion in proportion to the magnitude of the angular velocity. This torque is thus given by $-\beta\dot{\theta}(t)$, where $\dot{\theta}(t) = d\theta(t)/dt$ and β is some nonnegative constant.

Although the inverted pendulum is a simple system in many respects, it captures some essential features of systems that arise in diverse balancing applications, for instance, supporting the body on a human ankle or a mass on a robot joint or wheel axle. There are also control applications in which the pendulum is intended to move in the vicinity of its normal hanging position rather than the inverted position, that is, with $\theta(t) \approx \pi$. One might alternatively want the pendulum to rotate through full circles around the pivot. All of these motions are described by the equations below.

A Conventional Model The rotational form of Newton's law says the rate of change of angular momentum equals the net torque. We can accordingly write

$$\frac{d}{dt}\Big(\mathcal{I}\frac{d\theta(t)}{dt}\Big) = mg\ell\sin(\theta(t)) - \beta\frac{d\theta(t)}{dt} - x(t) . \tag{4.1}$$

Since $\mathcal{I}$ is constant, the preceding expression can be rewritten in a form that is closer to what is typically encountered in an earlier differential equations course:

$$\mathcal{I}\frac{d^2y(t)}{dt^2} + \beta\frac{dy(t)}{dt} - mg\ell\sin(y(t)) = -x(t) , \tag{4.2}$$

which is a single second-order nonlinear differential equation relating the output $y(t)$ to the input $x(t)$.

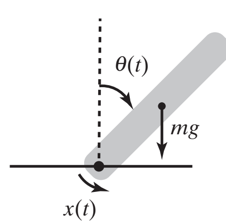


Figure 4.1 Inverted pendulum.

State Variables To get at the notion of state variables, we examine what constitutes the memory of the system at some arbitrary time t_0. Assume the parameters $\mathcal{I}$, m, ℓ, and β are all known, as is the external input $x(t)$ for $t \geq t_0$. The question is, what more needs to be known about the system at t_0 in order to solve for the behavior of the system for $t > t_0$.

Solving Eq. (4.1) for $\theta(t)$ in the interval $t > t_0$ ultimately requires integrating the equation twice, which in turn requires knowledge of the initial position and velocity, $\theta(t_0)$ and $\dot{\theta}(t_0)$ respectively. Another way to recognize the special role of these two variables is by considering the energy of the pendulum at the starting time. The energy is the result of past inputs to the system, and is reflected in the ensuing motion of the system. The potential energy at $t = t_0$ is determined by $\theta(t_0)$ and the kinetic energy by $\dot{\theta}(t_0)$, so these variables are key to understanding the behavior of the system for $t > t_0$.

State-Space Model The above discussion suggests that two natural memory variables of the system at any time t are $q_1(t) = \theta(t)$ and $q_2(t) = \dot{\theta}(t)$. Taking these as candidate state variables, a corresponding state-space description is found by trying to express the rates of change of these variables at time t entirely in terms of the values of these variables and of the input at the same time t. For this simple example, a pair of equations of the desired form can be obtained quite directly. Invoking the definitions of $q_1(t)$ and $q_2(t)$, as well as Eq. (4.1), and still assuming $\mathcal{I}$ is constant, we obtain

$$\frac{dq_1(t)}{dt} = q_2(t) \,, \tag{4.3}$$

$$\frac{dq_2(t)}{dt} = \frac{1}{\mathcal{I}}\Big(mg\ell \sin(q_1(t)) - \beta q_2(t) - x(t)\Big) \,. \tag{4.4}$$

This description comprises a pair of coupled first-order differential equations, driven by the input $x(t)$. These are referred to as the state evolution equations. The corresponding output equation expresses the output $y(t)$ entirely in terms of the values of the state variables and of the input at the same time t; in this case, the output equation is simply

$$y(t) = q_1(t) \,. \tag{4.5}$$

The combination of the state evolution equations and the output equation constitutes a state-space description of the system. The fact that such a description of the system is possible in terms of the candidate state variables $\theta(t)$ and $\dot{\theta}(t)$ confirms these as state variables—the "candidate" label can now be dropped.

Not only does the ordinary differential equation description in Eq. (4.1) or equivalently in Eq. (4.2) suggest what is needed to obtain the state-space model, but the converse is also true: the differential equation in Eq. (4.1), or equivalently in Eq. (4.2), can be obtained from Eqs. (4.3), (4.4), and (4.5).

Some Variations The choice of state variables above is not unique. For instance, the quantities defined by $q_1(t) = \theta(t) + \dot{\theta}(t)$ and $q_2(t) = \theta(t) - \dot{\theta}(t)$ could have functioned equally well. Equations expressing $\dot{q}_1(t)$, $\dot{q}_2(t)$, and $y(t)$ as functions of $q_1(t)$, $q_2(t)$, and $x(t)$ under these new definitions are easily obtained, and yield a different but entirely equivalent state-space representation.

The state-space description obtained above is nonlinear but time-invariant. It is nonlinear because the state variables and input, namely $q_1(t)$, $q_2(t)$, and $x(t)$, are combined nonlinearly in at least one of the functions defining $\dot{q}_1(t)$, $\dot{q}_2(t)$, and $y(t)$—in this case, the function defining $\dot{q}_2(t)$. The description is time-invariant because all the functions defining $\dot{q}_1(t)$, $\dot{q}_2(t)$, and $y(t)$ are time-invariant, that is, they combine their arguments $q_1(t)$, $q_2(t)$, and $x(t)$ according to a prescription that does not depend on time.

For small enough deviations from the fully inverted position, $q_1(t) = \theta(t)$ is small, so $\sin(q_1(t)) \approx q_1(t)$. With this approximation, Eq. (4.4) is replaced by

$$\frac{dq_2(t)}{dt} = \frac{1}{\mathcal{I}}\Big(mg\ell q_1(t) - \beta q_2(t) - x(t)\Big) . \tag{4.6}$$

The function defining $\dot{q}_2(t)$ is now an LTI function of its arguments $q_1(t)$, $q_2(t)$, and $x(t)$, so the resulting state-space model is now also LTI.

For linear models, matrix notation allows a compact representation of the state evolution equations and the output equation. We will use bold lowercase letters for vectors and bold uppercase for matrices. Defining the state vector and its derivative by

$$\mathbf{q}(t) = \begin{bmatrix} q_1(t) \\ q_2(t) \end{bmatrix} , \qquad \dot{\mathbf{q}}(t) = \frac{d\mathbf{q}(t)}{dt} = \begin{bmatrix} \dot{q}_1(t) \\ \dot{q}_2(t) \end{bmatrix} , \tag{4.7}$$

the linear model becomes

$$\begin{aligned} \dot{\mathbf{q}}(t) = \begin{bmatrix} \dot{q}_1(t) \\ \dot{q}_2(t) \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ mg\ell/\mathcal{I} & -\beta/\mathcal{I} \end{bmatrix} \begin{bmatrix} q_1(t) \\ q_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ -1/\mathcal{I} \end{bmatrix} x(t) \\ &= \mathbf{A}\mathbf{q}(t) + \mathbf{b}x(t) , \end{aligned} \tag{4.8}$$

where the definitions of the matrix $\mathbf{A}$ and vector $\mathbf{b}$ should be clear by comparison with the preceding equality. The corresponding output equation can be written as

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} q_1(t) \\ q_2(t) \end{bmatrix} = \mathbf{c}^T\mathbf{q}(t) , \tag{4.9}$$

with $\mathbf{c}^T$ denoting the transpose of a column vector, that is, a row vector. The time invariance of the system is reflected in the fact that the coefficient matrices $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{c}^T$ are constant rather than time-varying.

The ideas in the above example can be generalized to much more elaborate settings. In general, a natural choice of state variables for a mechanical system is the set of position and velocity variables associated with each component mass. For example, in the case of N point masses in three-dimensional space that are interconnected with each other and to rigid supports by massless springs, the natural choice of state variables would be the associated $3N$ position variables and $3N$ velocity variables. If these masses were confined to move in a plane, we would instead have $2N$ position variables and $2N$ velocity variables.

The next example suggests how state-space models arise in describing electrical circuits.

Example 4.2 Electrical Circuit

Consider the resistor-inductor-capacitor (RLC) circuit shown in Figure 4.2. All the component voltages and currents are labeled in the figure.

We begin by listing the characteristics of the various components, which we assume are linear and time-invariant. The defining equations for the inductor,

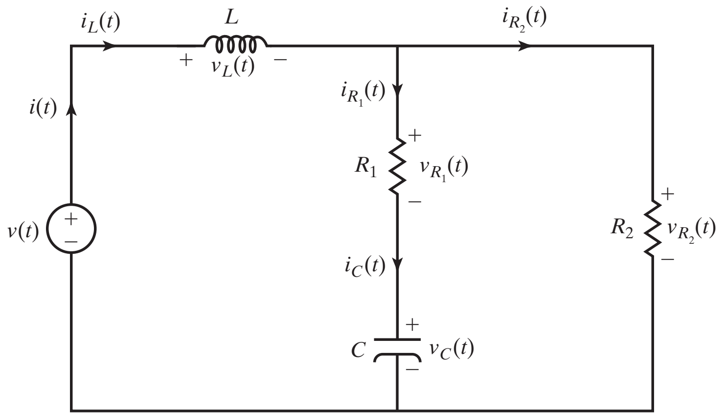


Figure 4.2 RLC circuit.

capacitor, and the two resistors take the form, in each case, of an LTI constraint relating the voltage across the element and the current through it. Specifically, we have

$$\begin{aligned} v_L(t) &= L\frac{di_L(t)}{dt} \\ i_C(t) &= C\frac{dv_C(t)}{dt} \\ v_{R_1}(t) &= R_1 i_{R_1}(t) \\ v_{R_2}(t) &= R_2 i_{R_2}(t)\,. \end{aligned} \tag{4.10}$$

The voltage source is defined by the condition that its voltage is a specified or arbitrary $v(t)$, regardless of the current $i(t)$ that is drawn from it.

The next step is to describe the constraints on these variables that arise from interconnecting the components. The interconnection constraints for an electrical circuit are imposed by Kirchhoff's voltage law (KVL) and Kirchhoff's current law (KCL). Both KVL and KCL produce additional LTI constraints relating the variables associated with the circuit. Here, KVL and KCL yield the following equations:

$$\begin{aligned} v(t) &= v_L(t) + v_{R_2}(t) \\ v_{R_2}(t) &= v_{R_1}(t) + v_C(t) \\ i(t) &= i_L(t) \\ i_L(t) &= i_{R_1}(t) + i_{R_2}(t) \\ i_{R_1}(t) &= i_C(t)\,. \end{aligned} \tag{4.11}$$

Other such KVL and KCL equations can be written for this circuit, but turn out to be consequences of the equations above, rather than new constraints.

Equations (4.10) and (4.11) together represent the individual components in the circuit and their mutual connections. Any set of signals that simultaneously satisfies all these constraint equations constitutes a valid solution—or behavior—of the circuit. Since all the constraints are LTI, it follows that weighted linear combinations or superpositions of behaviors are themselves behaviors of the circuit, and time-shifted behaviors are again behaviors of the circuit, so the circuit itself is LTI.

Input, Output, and State Variables Let us take the source voltage $v(t)$ as the input to the circuit, and also denote this by $x(t)$, our standard symbol for an input. Any of the circuit voltages or currents can be chosen as the output. Choose $v_{R_2}(t)$, for instance, and denote it by $y(t)$, our standard symbol for an output.

As in the preceding example, a good choice of state variables is established by determining what constitutes the memory of the system at any time. Apart from the parameters L, C, R_1, R_2, and the external input $x(t)$ for $t \geq t_0$, we ask what needs to be known about the system at a starting time t_0 in order to solve for the behavior of the system for $t > t_0$.

The existence of the derivatives in the defining expressions in Eq. (4.10) for the inductor and capacitor suggests that at least $i_L(t_0)$ and $v_C(t_0)$ are needed, or quantities equivalent to these. Note that, similarly to what was observed in the previous example, these variables are also associated with energy storage in the system, in this case the energy stored in the inductor and capacitor respectively. We accordingly identify the two natural memory variables of the system at any time t as $q_1(t) = i_L(t)$ and $q_2(t) = v_C(t)$, and these are our candidate state variables.

State-Space Model We now develop a state-space description for the RLC circuit of Figure 4.2 by trying to express the rates of change of the candidate state variables at time t entirely in terms of the values of these variables and of the input at the same time t. This is done by reducing the full set of relations in Eqs. (4.10) and (4.11), eliminating all variables other than the input, output, candidate state variables, and derivatives of the candidate state variables.

This process for the present example is not as transparent as in Example 4.1, and some attention is required in order to carry out the elimination efficiently. A good strategy—and one that generalizes to more complicated circuits—is to express the inductor voltage $v_L(t)$ and capacitor current $i_C(t)$ as functions of just the allowed variables, namely $i_L(t)$, $v_C(t)$, and $x(t) = v(t)$. Once this is accomplished, we make the substitutions

$$v_L(t) = L\frac{di_L(t)}{dt} \qquad \text{and} \qquad i_C(t) = C\frac{dv_C(t)}{dt}\ , \tag{4.12}$$

then rearrange the resulting equations to get the desired expressions for the rates of change of the candidate state variables. Following this procedure, and introducing the definition

$$\alpha = \frac{R_2}{R_1 + R_2} \tag{4.13}$$

for notational convenience, we obtain the desired state evolution equations. These are written below in matrix form, exploiting the fact that these state evolution equations turn out to be linear:

$$\begin{bmatrix} di_L(t)/dt \\ dv_C(t)/dt \end{bmatrix} = \begin{bmatrix} -\alpha R_1/L & -\alpha/L \\ \alpha/C & -1/(R_1+R_2)C \end{bmatrix} \begin{bmatrix} i_L(t) \\ v_C(t) \end{bmatrix} + \begin{bmatrix} 1/L \\ 0 \end{bmatrix} x(t)\ . \tag{4.14}$$

This is of the form

$$\dot{\mathbf{q}}(t) = \mathbf{A}\mathbf{q}(t) + \mathbf{b}x(t)\ , \tag{4.15}$$

where

$$\mathbf{q}(t) = \begin{bmatrix} q_1(t) \\ q_2(t) \end{bmatrix} = \begin{bmatrix} i_L(t) \\ v_C(t) \end{bmatrix} \tag{4.16}$$

and the definitions of the coefficient matrices **A** and **b** are determined by comparison with Eq. (4.14). The fact that these matrices are constant establishes that the description is LTI. The key feature here is that the model expresses the rates of change of the state variables at any time t as constant linear functions of their values and that of the input at the same time instant t.

As we will see in the next chapter, the state evolution equations in Eq. (4.14) can be used to solve for the state variables $i_L(t)$ and $v_C(t)$ for $t > t_0$, given the input $x(t) = v(t)$ for $t \geq t_0$ and the initial conditions on the state variables at time t_0. 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. Having picked the output of interest to be $v_{R_2}(t) = y(t)$, we can write (again in matrix notation)

$$y(t) = v_{R_2}(t) = \begin{bmatrix} \alpha R_1 & \alpha \end{bmatrix} \begin{bmatrix} i_L(t) \\ v_C(t) \end{bmatrix} = \mathbf{c}^T \mathbf{q}(t) \ . \tag{4.17}$$

Input-Output Behavior Transforming Eqs. (4.10) and (4.11) using the bilateral Laplace transform, and noting that differentiation in the time domain maps to multiplication by s in the transform domain, we can solve for the transfer function $H(s)$ of the system from $x(t)$ to $y(t)$. Alternatively, we can obtain the same transfer function from Laplace transformation of the state-space description in Eqs. (4.14) and (4.17). The next chapter presents an explicit formula for this transfer function in terms of the coefficient matrices **A**, **b**, and $\mathbf{c}^T$.

For our RLC example, this transfer function $H(s)$ from input to output is

$$H(s) = \frac{Y(s)}{X(s)} = \frac{\alpha\left(\frac{R_1}{L}s + \frac{1}{LC}\right)}{s^2 + \alpha\left(\frac{1}{R_2C} + \frac{R_1}{L}\right)s + \alpha\frac{1}{LC}} \ . \tag{4.18}$$

The corresponding input-output second-order LTI differential equation is

$$\frac{d^2y(t)}{dt^2} + \alpha\left(\frac{1}{R_2C} + \frac{R_1}{L}\right)\frac{dy(t)}{dt} + \alpha\left(\frac{1}{LC}\right)y(t) = \alpha\left(\frac{R_1}{L}\right)\frac{dx(t)}{dt} + \alpha\left(\frac{1}{LC}\right)x(t) \ . \tag{4.19}$$

The procedure for obtaining a state-space description that is illustrated in Example 4.2 can be used even if some of the circuit components are nonlinear. It can then often be helpful to choose inductor flux rather than current as a state variable, and similarly to choose capacitor charge rather than voltage as a state variable. It is generally the case, just as in the Example 4.2, that the natural state variables in an electrical circuit are the inductor currents or fluxes, and the capacitor voltages or charges. The exceptions occur in degenerate situations, for example where a closed path in the circuit involves only capacitors and voltage sources. In the latter instance, KVL applied to this path shows that the capacitor voltages are not all independent.

State-space models arise naturally in many problems that involve tracking subgroups of some population of objects as they interact in time. For instance, in chemical reaction kinetics the interest is in determining the expected molecule numbers or concentrations of the various interacting chemical constituents as the reaction progresses in continuous time. Another instance involves modeling, in either continuous time or discrete time, the spread of a fashion, opinion, idea, or disease through a human population, or of a software virus through a computer network. The following example develops one such DT model and begins to explore its behavior. Some later examples extend the analysis further.

Example 4.3 Viral Propagation

The DT model presented here captures some essential aspects of viral propagation in a variety of settings. The model is one of a large family of such models, both deterministic and stochastic, that have been widely studied. Though much of the terminology derives from modeling the spread of disease by viruses, the paradigm of viral propagation has been applied to understanding how, for example, malicious software, advertisements, gossip, or cultural memes spread in a population or network.

The deterministic model here tracks three component subpopulations from the nth DT epoch to the $(n+1)$th. Suppose the total population of size P is divided into the following subgroups, or "compartments," at integer time n:

- $s[n] \geq 0$ is the number of susceptibles, currently virus-free but vulnerable to acquiring the virus;
- $i[n] \geq 0$ is the number of infectives, carrying the virus and therefore capable of passing it to the susceptibles by the next epoch; and
- $r[n] \geq 0$ is the number of recovered, no longer carrying the virus and no longer susceptible, because of acquired immunity.

The model below assumes these variables are real-valued rather than integer-valued, which results in substantial simplification of the model, and may be a satisfactory approximation when P is very large.

We assume the birth rate in these three subgroups has the same value β; this is the (deterministic) fractional increase in the population per unit time due to birth. Suppose the death rate is also β, so the total size of the population remains constant at P. Assume $0 \leq \beta < 1$.

Let the rate at which susceptibles become infected be proportional to the concentration of infectives in the general population, hence a rate of the form $\gamma(i[n]/P)$ for some $0 < \gamma \leq 1$. The rate at which infectives move to the recovered compartment is denoted by ρ, with $0 < \rho \leq 1$. We take newborns to be susceptible, even if born to infective or recovered members of the population. Suppose also that newborns are provided immunity at a rate $0 \leq v[n] \leq 1$, for instance by vaccination, moving them directly from the susceptible compartment to the recovered compartment. We consider $v[n]$ to be the control input, and denote it by the alternative symbol $x[n]$.

With the above notation and assumptions, we arrive quite directly at the very simple (and undoubtedly simplistic) model below, for the change in each subpopulation over one time step:

$$\begin{aligned} s[n+1] - s[n] &= -\gamma(i[n]/P)s[n] + \beta(i[n] + r[n]) - \beta P x[n] \\ i[n+1] - i[n] &= \gamma(i[n]/P)s[n] - \rho i[n] - \beta i[n] \\ r[n+1] - r[n] &= \rho i[n] - \beta r[n] + \beta P x[n] \,. \end{aligned} \tag{4.20}$$

A model of this type is commonly referred to as an SIR model, as it comprises susceptible, infective, and recovered populations. We shall assume that the initial conditions, parameters, and control inputs are chosen so as to maintain all subpopulations at nonnegative values throughout the interval of interest. The actual mechanisms of viral spread are of course much more intricate and complicated than captured in this elementary model, and also involve substantial randomness and uncertainty.

If some fraction ϕ of the infectives gets counted at each time epoch, then the aggregate number of infectives reported can be taken as our output $y[n]$, so

$$y[n] = \phi i[n] \,. \tag{4.21}$$

Notice that the expressions in Eq. (4.20) have a very similar form to the CT state evolution equations we arrived at in the earlier two examples. For the DT case, take the rate of change of a variable at time n to be the increment over one time step forward from n. Then Eq. (4.20) expresses the rates of change of the indicated variables at time n as functions of these same variables and the input at time n. It therefore makes sense to think of $s[n]$, $i[n]$, and $r[n]$ as state variables, whose values at time n constitute the state of the system at time n.

The model here is time-invariant because the three expressions that define the rates of change all involve combining the state variables and input at time n according to prescriptions that do not depend on n. The consequence of this feature is that any set of $s[\cdot]$, $i[\cdot]$, and $r[\cdot]$ signals that simultaneously satisfy the model equations will also satisfy the model equations if they are all shifted arbitrarily by the same time offset. However, the model is not linear; it is nonlinear because the first two expressions involve a nonlinear combination of $s[n]$ and $i[n]$, namely their product. The expression in Eq. (4.21) writes the output at time n as a function of the state variables and input at time n—though it happens in this case that only $i[n]$ is needed.

It is conventional in the DT case to rearrange the state evolution equations into a form that expresses the state at time $n+1$ as a function of the state variables and input at time n. Thus Eq. (4.20) would be rewritten as

$$\begin{aligned} s[n+1] &= s[n] - \gamma(i[n]/P)s[n] + \beta(i[n] + r[n]) - \beta P x[n] \\ i[n+1] &= i[n] + \gamma(i[n]/P)s[n] - \rho i[n] - \beta i[n] \\ r[n+1] &= r[n] + \rho i[n] - \beta r[n] + \beta P x[n] \,. \end{aligned} \tag{4.22}$$

In this form, the equations give a simple prescription for obtaining the state at time $n+1$ from the state and input at time n. Summing the three equations also makes clear that for this example

$$s[n+1] + i[n+1] + r[n+1] = s[n] + i[n] + r[n] = P \,. \tag{4.23}$$

Thus, knowing any two of the subgroup populations suffices to determine the third, if P is known. Examining the individual relations in Eqs. (4.20) or (4.22), and noting that the term $i[n] + r[n]$ in the first equation of each set could equivalently have been written as $P - s[n]$, we see that the first two relations in fact only involve the susceptible and infective populations, in addition to the input, and therefore comprise a state evolution description of lower order, namely

$$\begin{aligned} s[n+1] &= s[n] - \gamma(i[n]/P)s[n] + \beta(P - s[n]) - \beta P x[n] \\ i[n+1] &= i[n] + \gamma(i[n]/P)s[n] - \rho i[n] - \beta i[n] \,. \end{aligned} \tag{4.24}$$

Figure 4.3 shows a few state-variable trajectories produced by stepping the model in Eq. (4.24) forward from a particular $s[0]$, fixed at 8000 out of a population (P) of 10,000, using different initial values $i[0]$. Note that in each case the number of

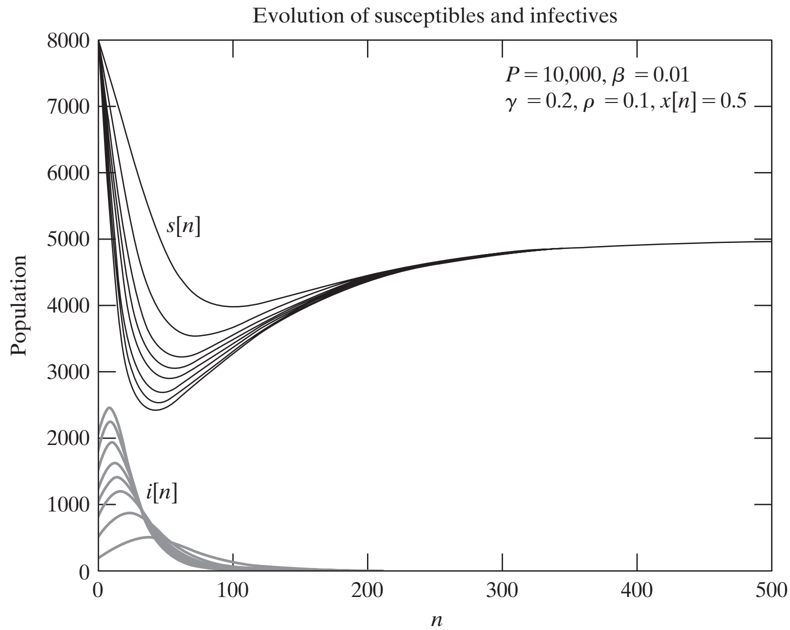


Figure 4.3 Response of SIR model for a particular choice of parameter values and a variety of initial conditions.

infectives, $i[n]$, initially increases from its value at the starting time $n = 0$, before eventually decaying. This initial increase would correspond to "going viral" in the case of a rumor, advertisement, or fashion that spreads through a social network, or to an epidemic in the case of disease propagation. The second equation in Eq. (4.24) shows that $i[n + 1] > i[n]$ precisely when

$$\frac{s[n]}{P} > \frac{\rho + \beta}{\gamma} = \frac{1}{R_0} . \tag{4.25}$$

Here

$$R_0 = \frac{\gamma}{\beta + \rho} \tag{4.26}$$

is a parameter that typically arises in viral propagation models, and is termed the basic reproductive ratio (referring to "reproduction" of infectives, not to population growth). Thus $i[n]$ increases at the next time step whenever the fraction of susceptibles in the population, $s[n]/P$, exceeds the threshold $1/R_0$. As $s[n]/P$ cannot exceed 1, there can be no epidemic if $R_0 \leq 1$. The greater the amount by which R_0 exceeds 1, the fewer the number of susceptibles required in order for an epidemic to occur.

Figure 4.3 also shows that the system in this case, with the immunization rate fixed at $x[n] = 0.5$, reaches a steady state in which there are no infectives. This is termed an infective-free steady state. In Examples 4.8, 4.10, and 5.5, we explore further characteristics of the model in Eq. (4.24). In particular, it will turn out that it is possible—for instance by dropping the immunization rate to $x[n] = 0.2$ while keeping the other parameters as in Figure 4.3—for the attained steady state to have a nonzero number of infectives. This is termed an endemic steady state.

Compartmental models of the sort illustrated in the preceding example are ubiquitous, in both continuous time and discrete time. We conclude this section with another DT example, related to implementation of a filter using certain elementary operations.

Example 4.4 Delay-Adder-Gain System

The block diagram in Figure 4.4 shows a causal DT system obtained by interconnecting delay, adder, and gain elements. A (unit) delay has the property that its output value at any integer time n is the value that was present at its input at time $n-1$; or equivalently, its input value at any time n is the value that will appear at its output at time $n+1$. An adder produces an output that is the sum of its present inputs. A gain element produces an output that is the present input scaled by the gain value. These all correspond to LTI operations on the respective input signals.

Interconnection involves equating, or "connecting," each input of these various elements to a selected output of one of the elements. The result of such an interconnection turns out to be well behaved if every loop has some delay in it, that is, provided there are no delay-free loops. An overall external input $x[n]$ and an overall external output $y[n]$ are also included in Figure 4.4. Such delay-adder-gain systems (and their CT counterparts, which are integrator-adder-gain systems, as in Example 4.5) are widely used in constructing LTI filters that produce a signal $y[\cdot]$ from a signal $x[\cdot]$.

The memory of this system is embodied in the delay elements, so it is natural to consider the outputs of these elements as candidate state variables. Accordingly, we label the outputs of the memory elements in this example as $q_1[n]$ and $q_2[n]$ at time n. For the specific block diagram in Figure 4.4, the detailed component and interconnection equations relating the indicated signals are

$$
\begin{aligned}
q_1[n+1] &= q_2[n] \\
q_2[n+1] &= p[n] \\
p[n] &= x[n] - 0.5q_1[n] + 1.5q_2[n] \\
y[n] &= q_2[n] + p[n] \,.
\end{aligned}
\tag{4.27}
$$

The response of the system for $n \geq n_0$ is completely determined by the external input $x[n]$ for times $n \geq n_0$ and the values $q_1[n_0]$ and $q_2[n_0]$ that are stored at the

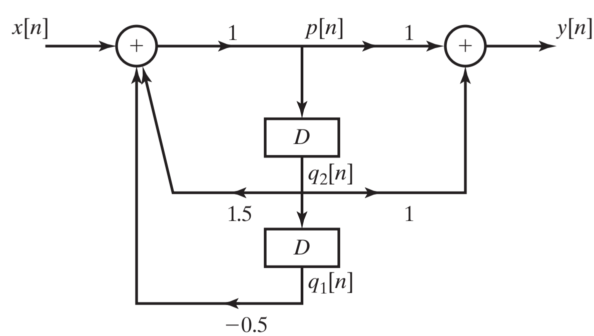


Figure 4.4 Delay-adder-gain block diagram.

outputs of the delay elements at time n_0. The delay elements capture the state of the system at each time step, that is, they summarize all the past history that is relevant to how the present and future inputs to the system determine the present and future response of the system.

The relationships in Eq. (4.27) need to be condensed in order to express the values of the candidate state variables at time $n+1$ in terms of the values of these variables at time n and the value of the external input at the same time instant n. This corresponds to expressing the inputs to all the delay elements at time n in terms of all the delay outputs at time n as well as the external input at this same time. The result for this example is captured in the following matrix equation:

$$\begin{aligned}\mathbf{q}[n+1] = \begin{bmatrix} q_1[n+1] \\ q_2[n+1] \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -0.5 & 1.5 \end{bmatrix} \begin{bmatrix} q_1[n] \\ q_2[n] \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} x[n] \\ &= \mathbf{A}\mathbf{q}[n] + \mathbf{b}x[n] \,. \end{aligned} \tag{4.28}$$

Similarly, the output at time n can be written in terms of the values of the candidate state variables at time n and the value of the external input at the same time instant n:

$$y[n] = \begin{bmatrix} -0.5 & 2.5 \end{bmatrix} \begin{bmatrix} q_1[n] \\ q_2[n] \end{bmatrix} + x[n] = \mathbf{c}^T\mathbf{q}[n] + \mathrm{d}x[n] \,. \tag{4.29}$$

Notice that in this example, unlike in the previous examples, the output $y[n]$ at any time n depends not only on the state variables at time n but also on the input at that time n.

Equations (4.28) and (4.29) establish that $q_1[n]$ and $q_2[n]$ are indeed valid state variables. Specifically, the equations explicitly show that if one is given the values $q_1[n_0]$ and $q_2[n_0]$ of the state variables at some initial time n_0, and also the input trajectory from n_0 onward, that is, $x[n]$ for times $n \geq n_0$, then we can compute the values of the state variables and the output for times $n \geq n_0$. All that is needed is to iteratively apply Eq. (4.28) to find $q_1[n_0+1]$ and $q_2[n_0+1]$, then $q_1[n_0+2]$ and $q_2[n_0+2]$, and so on for increasing time arguments, and to use Eq. (4.29) at each time to find the output.

Transforming the relationships in Eq. (4.27) using the bilateral z-transform, and noting that time-advancing a signal by one step maps to multiplication by z in the transform domain, we can solve for the transfer function $H(z)$ of the system from $x[\cdot]$ to $y[\cdot]$. Alternatively, the same transfer function can be obtained from z-transformation of the state-space description; the next chapter presents an explicit formula for this transfer function in terms of the coefficient matrices $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}^T$, and d. Either way, the resulting transfer function for our example is

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1+z^{-1}}{1-\frac{3}{2}z^{-1}+\frac{1}{2}z^{-2}} \,, \tag{4.30}$$

which corresponds to the following input-output difference equation:

$$y[n] - \frac{3}{2}y[n-1] + \frac{1}{2}y[n-2] = x[n] + x[n-1] \,. \tag{4.31}$$

The development of CT state-space models for integrator-adder-gain systems follows a completely parallel route. Integrators replace the delay elements. Their outputs at time t constitute a natural set of state variables for the system; their values at any starting time t_0 establish the initial conditions for integration over the interval $t \geq t_0$. The state evolution equations result from expressing the inputs to all the integrators at time t in terms of all the integrator outputs at time t as well as the external input at this same time.

4.3 STATE-SPACE MODELS

As illustrated in the examples of the preceding section, 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, the examples in Section 4.2 describe system behavior in terms of the time evolution of a set of state variables that 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 the 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; we mostly limit our discussion to real-valued state variables. The number of state variables in a model or system is referred to as its order. We shall only deal with state-space models of finite order, which are also referred to as lumped models.

For an Lth-order model in the DT case, we generically denote the values of the L real 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

$$\mathbf{q}[n] = \begin{bmatrix} q_1[n] \\ q_2[n] \\ \vdots \\ q_L[n] \end{bmatrix} . \tag{4.32}$$

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

DT LTI State-Space Model A DT LTI state-space model with single or scalar input $x[n]$ and single output $y[n]$ takes the following form, written in compact matrix notation

$$\mathbf{q}[n+1] = \mathbf{A}\mathbf{q}[n] + \mathbf{b}x[n] , \tag{4.33}$$

$$y[n] = \mathbf{c}^T\mathbf{q}[n] + \mathrm{d}x[n] . \tag{4.34}$$

In Eqs. (4.33) and (4.34), $\mathbf{A}$ 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 d is a 1×1 matrix, or a scalar. The entries of all these matrices in the case of an LTI model are numbers, constants, or parameters, so they do not vary with n.

The next value of each state variable and the present value of the output are all expressed as LTI functions of the present state and present input. We refer to Eq. (4.33) as the state evolution equation, and to Eq. (4.34) as the output equation. The model obtained for the delay-adder-gain system in Example 4.4 in the previous section has precisely the above form.

The system in Eqs. (4.33) and (4.34) is termed LTI because of its structure: the next state and current output are LTI functions of the current state and current input. However, this structure also gives rise to a corresponding behavioral sense in which the system is LTI. A particular set of input, state, and output signals—$x[\cdot]$, $\mathbf{q}[\cdot]$, and $y[\cdot]$, respectively—that together satisfy the above state evolution equation and output equation is referred to as a behavior of the DT LTI system. It follows from the linear structure of the above equations that scaling all the signals in a behavior by the same scalar constant again yields a behavior of this system. Also, summing two behaviors again yields a behavior. More generally, a weighted linear combination of behaviors again yields a behavior, so the behaviors of the system have the superposition property. Similarly, it follows from the time invariance of the defining equations that an arbitrary time shift of a behavior—shifting the input, state, and output signals in time by the same amount—again yields a behavior. Thus, the LTI structure of the equations is mirrored by the LTI properties of its solutions or behaviors.

Delay-Adder-Gain Realization A delay-adder-gain system of the form encountered in Example 4.4 can be used to simulate, or "realize," any Lth-order, DT LTI model of the type given in Eqs. (4.33) and (4.34). Key to this is the fact that adders and gains suffice to implement the additions and multiplications associated with the various matrix multiplications in the LTI state-space description.

To set up the simulation, we begin with L delay elements, and label their outputs at time n as $q_j[n]$ for $j = 1, 2, \cdots, L$; the corresponding inputs are then $q_j[n+1]$. The ith row of Eq. (4.33) shows what LTI combination of these $q_j[n]$ and $x[n]$ is required to compute $q_i[n+1]$, for each $i = 1, 2, \cdots, L$. Similarly, Eq. (4.34) shows what LTI combination of the variables is required to compute $y[n]$. Each of these LTI combinations can now be implemented using gains and adders.

The implementation produced by the preceding prescription is not unique: there are multiple ways to implement the linear combinations, depending, for example, on whether there is special structure in the matrices, or on how computation of the various terms in the linear combination is grouped and sequenced. In the case of the system in Example 4.4, for example, starting with the model in Eqs. (4.28) and (4.29) and following the procedure outlined in this paragraph will almost certainly lead to a different realization than the one in Figure 4.4.

Generalizations Although our focus in the DT case will be on the above LTI, single-input, single-output, state-space model, there are various natural generalizations of this description that we mention for completeness. A multi-input DT LTI state-space model replaces the single term $\mathbf{b}x[n]$ in Eq. (4.33) by a sum of terms, $\mathbf{b}_1 x_1[n] + \cdots + \mathbf{b}_M x_M[n]$, 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 quantity in Eq. (4.34) 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 d to matrices $\mathbf{C}^T$ and $\mathbf{D}$ of dimensions $P \times L$ and $P \times M$ respectively.

A linear but time-varying DT state-space model takes the same form as in Eqs. (4.33) and (4.34), 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.

All of the above generalizations can also be simulated or realized by delay-adder-gain systems, except that the gains will need to be time-varying for the case of time-varying systems. For the nonlinear systems described below, more elaborate simulations are needed, involving nonlinear elements or combinations.

A nonlinear, time-invariant, single input, single output model expresses $\mathbf{q}[n+1]$ and $y[n]$ as nonlinear but time-invariant functions of $\mathbf{q}[n]$ and $x[n]$, rather than as the LTI functions embodied by the matrix expressions on the right-hand sides of Eqs. (4.33) and (4.34). Our full and reduced models for viral propagation in Example 4.3 were of this type. A third-order nonlinear time invariant state-space model, for instance, comprises state evolution equations of the form

$$q_1[n+1] = f_1\Big(q_1[n], q_2[n], q_3[n], x[n]\Big)$$
$$q_2[n+1] = f_2\Big(q_1[n], q_2[n], q_3[n], x[n]\Big)$$
$$q_3[n+1] = f_3\Big(q_1[n], q_2[n], q_3[n], x[n]\Big) \tag{4.35}$$

and an output equation of the form

$$y[n] = g\Big(q_1[n], q_2[n], q_3[n], x[n]\Big) , \tag{4.36}$$

where the state evolution functions $f_1(\cdot)$, $f_2(\cdot)$, $f_3(\cdot)$ and the output function $g(\cdot)$ are all time-invariant nonlinear functions of the three state variables $q_1[n]$, $q_2[n]$, $q_3[n]$, and the input $x[n]$. Time invariance here means that the functions combine their arguments in the same way, regardless of the time index n. In vector notation,

$$\mathbf{q}[n+1] = \mathbf{f}\Big(\mathbf{q}[n], x[n]\Big) , \qquad y[n] = g\Big(\mathbf{q}[n], x[n]\Big) , \tag{4.37}$$

where for the third-order case

$$\mathbf{f}(\,\cdot\,) = \begin{bmatrix} f_1(\,\cdot\,) \\ f_2(\,\cdot\,) \\ f_3(\,\cdot\,) \end{bmatrix} . \tag{4.38}$$

The notation for an Lth-order description follows the same pattern.

Finally, a nonlinear, time-varying model expresses $\mathbf{q}[n+1]$ and $y[n]$ as nonlinear, time-varying functions of $\mathbf{q}[n]$ and $x[n]$. In other words, the manner in which the state evolution and output functions combine their arguments can vary with n. For this case, we would write

$$\mathbf{q}[n+1] = \mathbf{f}\Big(\mathbf{q}[n], x[n], n\Big) , \qquad y[n] = g\Big(\mathbf{q}[n], x[n], n\Big) . \tag{4.39}$$

Nonlinear, periodically varying models can also be defined 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. The state variables for an Lth-order system may be denoted as $q_i(t)$, $i = 1, 2, \ldots, L$, and the state vector as

$$\mathbf{q}(t) = \begin{bmatrix} q_1(t) \\ q_2(t) \\ \vdots \\ q_L(t) \end{bmatrix} . \tag{4.40}$$

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 the CT case the state evolution equation expresses the rates of change or derivatives of each of the state variables as functions of the present state and inputs.

CT LTI State-Space Model The general Lth-order CT LTI state-space representation takes the form

$$\frac{d\mathbf{q}(t)}{dt} = \dot{\mathbf{q}}(t) = \mathbf{A}\mathbf{q}(t) + \mathbf{b}x(t) , \tag{4.41}$$

$$y(t) = \mathbf{c}^T\mathbf{q}(t) + \mathrm{d}x(t) , \tag{4.42}$$

where $d\mathbf{q}(t)/dt = \dot{\mathbf{q}}(t)$ denotes the vector whose entries are the derivatives of the corresponding entries of $\mathbf{q}(t)$. The entries of all these matrices are numbers or constants or parameters that do not vary with t. Thus, the rate of change of each state variable and the present value of the output are all expressed as LTI functions of the present state and present input. As in the DT LTI case, the LTI structure of the above system is mirrored by the LTI properties of its solutions or behaviors, a fact that will become explicit in Chapter 5. The models in Eqs. (4.8) and (4.9) of Example 4.1 and Eqs. (4.14) and (4.17) of Example 4.2 are precisely of the above form.

Integrator-Adder-Gain Realization Any CT LTI state-space model of the form in Eqs. (4.41) and (4.42) can be simulated or realized using an integrator-adder-gain system. The approach is entirely analogous to the DT LTI case that was described earlier. We begin with L integrators, labeling their outputs as $q_j(t)$ for $j = 1, 2, \cdots, L$. The inputs of these integrators are then the derivatives $\dot{q}_j(t)$. The ith row of Eq. (4.41) now determines what LTI combination of the $q_j(t)$ and $x(t)$ is required to synthesize $\dot{q}_i(t)$, for each $i = 1, 2, \cdots, L$.

We similarly use Eq. (4.42) to determine what LTI combination of these variables is required to compute $y(t)$. Finally, each of these LTI combinations is implemented using gains and adders. We illustrate this procedure with a specific example below.

Generalizations The basic CT LTI state-space model can be generalized to multi-input and multi-output models, to nonlinear time-invariant models, and to linear and nonlinear time-varying or periodically varying models. These generalizations 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 Eqs. (4.41) and (4.42). The model for the inverted pendulum in Eqs. (4.3), (4.4), and (4.5) in Example 4.1 was nonlinear and time-invariant, of the form

$$\dot{\mathbf{q}}(t) = \mathbf{f}\Big(\mathbf{q}(t), x(t)\Big) , \qquad y(t) = g\Big(\mathbf{q}(t), x(t)\Big) . \tag{4.43}$$

A general nonlinear and time-varying CT state-space model with a single input and single output takes the form

$$\dot{\mathbf{q}}(t) = \mathbf{f}\Big(\mathbf{q}(t), x(t), t\Big) , \qquad y(t) = g\Big(\mathbf{q}(t), x(t), t\Big) . \tag{4.44}$$

Example 4.5 Simulation of Inverted Pendulum for Small Angles

For sufficiently small angular deviations from the fully inverted position for the inverted pendulum considered in Example 4.1, the original nonlinear state-space model simplifies to the LTI state-space model described by Eqs. (4.8) and (4.9). This LTI model is repeated here for convenience, but with the numerical values of a specific pendulum inserted:

$$\begin{aligned}\dot{\mathbf{q}}(t) = \begin{bmatrix} \dot{q}_1(t) \\ \dot{q}_2(t) \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 8 & -2 \end{bmatrix} \begin{bmatrix} q_1(t) \\ q_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix} x(t) \\ &= \mathbf{A}\mathbf{q}(t) + \mathbf{b}x(t) \end{aligned} \tag{4.45}$$

and

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} q_1(t) \\ q_2(t) \end{bmatrix} = \mathbf{c}^T\mathbf{q}(t) . \tag{4.46}$$

To simulate this second-order LTI system using integrators, adders, and gains, we begin with two integrators and denote their outputs at time t by $q_1(t)$ and $q_2(t)$. The inputs to these integrators are then $\dot{q}_1(t)$ and $\dot{q}_2(t)$, respectively, at time t. The right-hand sides of the two expressions in Eq. (4.45) now show how to synthesize $\dot{q}_1(t)$ and $\dot{q}_2(t)$ from particular weighted linear combinations of $q_1(t)$, $q_2(t)$, and $x(t)$. We use gain elements to obtain the appropriate weights, then adders to produce the required weighted linear combinations of $q_1(t)$, $q_2(t)$, and $x(t)$. By feeding these weighted linear combinations to the inputs of the respective integrators, $\dot{q}_1(t)$ and $\dot{q}_2(t)$ are set equal to these expressions. The output $y(t) = q_1(t)$ is directly read from the output of the first integrator. The block diagram in Figure 4.5 shows the resulting simulation.

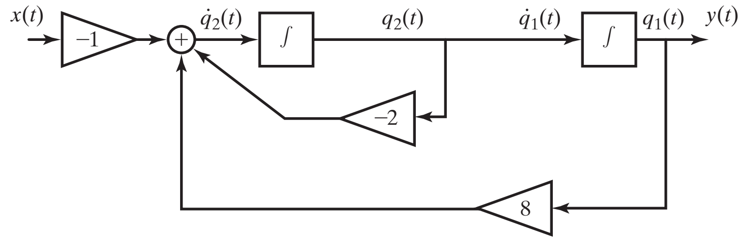


Figure 4.5 Integrator-adder-gain simulation of inverted pendulum for small angular deviations from vertical.

4.3.3 Defining Properties of State-Space Models

The two defining characteristics of state-space models are the following:

- ***State Evolution Property*** The state at any initial time, along with the inputs over any interval from that initial time onward, determine the state trajectory, that is, the state as a function of time, 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 LTI case, the validity of this state evolution property is evident from Eq. (4.33), which allows $\mathbf{q}[n]$ to be updated iteratively, moving 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 general DT state evolution expression in Eq. (4.39).

The state evolution property in the general CT case is more subtle to establish, and actually requires that the function $\mathbf{f}(\mathbf{q}(t), x(t), t)$ defining the rate of change of the state vector satisfy certain mild technical conditions. These conditions are satisfied by all the models of interest to us in this text, so we shall not discuss the conditions further. Instead, we describe how the availability of a CT state-space model enables a simple numerical approximation of the state trajectory at a discrete set of times spaced an interval Δ apart. This numerical algorithm is referred to as the forward-Euler method.

The algorithm begins by using the state and input information at the initial time t_0 to determine the initial rate of change of the state, namely $\mathbf{f}(\mathbf{q}(t_0), x(t_0), t_0)$. As illustrated in Figure 4.6, this initial rate of change is tangent to the state trajectory at t_0. The approximation to the actual trajectory is obtained by stepping forward a time increment Δ along this tangent—the forward-Euler step—to arrive at the estimate

$$\mathbf{q}(t_0 + \Delta) \approx \mathbf{q}(t_0) + \mathbf{f}(\mathbf{q}(t_0), x(t_0), t_0)\Delta \ . \tag{4.47}$$

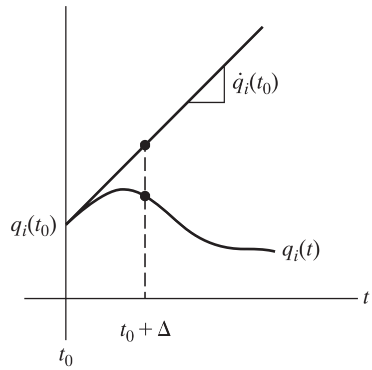


Figure 4.6 Using the CT state evolution equations to obtain the state trajectories over an interval.

This is equivalent to using a first-order Taylor series approximation to the trajectory, or using a forward-difference approximation to $\dot{\mathbf{q}}(t_0)$.

With the estimate of $\mathbf{q}(t_0 + \Delta)$ now available, and knowing the input $x(t_0 + \Delta)$ at time $t_0 + \Delta$, the same procedure can be repeated at this next time instant, thereby getting an approximation to $\mathbf{q}(t_0 + 2\Delta)$. This iteration can be continued over the entire interval of interest. Under the technical conditions alluded to above, the algorithm accumulates an error of order Δ^2 at each time step, and takes T/Δ time steps in an interval of length T, thereby accumulating an error of order $T\Delta$ by the end of the interval. This error can be made arbitrarily small by choosing a sufficiently small Δ.

The forward-Euler algorithm suffices to suggest how a CT state-space description gives rise to the state evolution property. For actual numerical computation, more sophisticated numerical routines would be used, based for example on higher-order Taylor series approximations, and using variable-length time steps for better error control. The CT LTI case is, however, much simpler than the general case. We shall demonstrate the state evolution property for this class of state-space models in detail in the Chapter 5, when we show how to explicitly solve for their behavior.

The instantaneous output property is evident in the LTI case from the output expressions in Eqs. (4.34) and (4.42). It also holds for the various generalizations of basic single-input, single-output LTI models that we listed earlier, most broadly for the output relations in Eqs. (4.39) and (4.44).

The state evolution and instantaneous output properties are the defining characteristics of a state-space model. In setting up a state-space model, we 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 these properties.

Often there are natural choices of state variables suggested directly by the particular context or application. As already noted, and illustrated by the preceding examples 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.

One can always choose any alternative set of state variables that together contain exactly the same information as a given set. 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 noncausal 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 control design and operation for dynamic systems, and this is where state-space descriptions find their greatest value and use.

4.4 STATE-SPACE MODELS FROM LTI 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 Section 4.2 demonstrated. Other—and perhaps more familiar—descriptions can then be derived from them, for instance input-output descriptions.

It is also possible to proceed in the reverse direction, constructing state-space descriptions from transfer functions, unit sample or impulse responses, or input-output difference or differential equations, for instance. This is often worthwhile as a prelude to simulation, filter implementation, in control design, or simply in order to understand the initial description from another point of view. The state variables associated with the resulting state-space descriptions do not necessarily have interesting or physically meaningful interpretations, but still capture the memory of the system.

The following two examples illustrate this reverse process, of synthesizing state-space descriptions from input-output descriptions, for the important case of DT LTI systems. Analogous examples can be constructed for the CT LTI case. The first example below also makes the point that state-space models of varying orders can share the same input-output description, a fact that we will understand better following the structural analysis of LTI systems developed in the next chapter. That structural analysis actually ends up also relating quite closely to the second example in this section.

10 Random Processes

The earlier chapters in this text focused on the effect of linear and time-invariant (LTI) systems on deterministic signals, developing tools for analyzing this class of signals and systems, and using these to understand applications in communication (e.g., AM and FM modulation), control (e.g., stability of feedback systems), and signal processing (e.g., filtering). It is important to develop a comparable understanding and associated tools for treating the effect of LTI systems on signals modeled as the outcome of probabilistic experiments, that is, the class of signals referred to as random signals, alternatively referred to as random processes or stochastic processes. Such signals play a central role in signal and system analysis and design. In this chapter, we define random processes through the associated ensemble of signals, and explore their time-domain properties. Chapter 11 examines their characteristics in the frequency domain. The subsequent chapters use random processes as models for random or uncertain signals that arise in communication, control and signal processing applications, and study a variety of related inference problems involving estimation and hypothesis testing.

10.1 DEFINITION AND EXAMPLES OF A RANDOM PROCESS

In Section 7.3, we defined a random variable X as a function that maps each outcome of a probabilistic experiment to a real number. In a similar manner, a real-valued continuous-time (CT) or discrete-time (DT) random process—$X(t)$ or $X[n]$, respectively—is a function that maps each outcome of a probabilistic experiment to a real CT or DT signal, termed the realization of the random process in that experiment. For any fixed time instant $t = t_0$ or $n = n_0$, the quantities $X(t_0)$ and $X[n_0]$ are simply random variables. The collection of signals that can be produced by the random process is referred to as the ensemble of signals in the random process.

Example 10.1 Random Oscillators

As an example of a random process, consider a warehouse containing N harmonic oscillators, each producing a sinusoidal waveform of some specific amplitude, frequency, and phase. The three parameters may in general differ between oscillators. This collection constitutes the ensemble of signals. The probabilistic experiment that yields a particular signal realization consists of selecting an oscillator according to some probability mass function (PMF) that assigns a probability to each of the numbers from 1 to N, so that the ith oscillator is picked with probability p_i. Associated with each outcome of this experiment is a specific sinusoidal waveform. Before an oscillator is chosen, there is uncertainty about what the amplitude, frequency, and phase of the outcome of the experiment will be, that is, the amplitude A, frequency Φ, and phase Θ are all random variables. Consequently, for this example, we might express the random process as

$$X(t; A, \Phi, \Theta) = A\sin(\Phi t + \Theta) \tag{10.1}$$

where, as in Figure 10.1, we have listed after the semi-colon the parameters that are random variables. As the discussion proceeds, we will typically simplify the notation to refer to $X(t)$ when it is clear which parameters are random variables; so, for example, Eq. (10.1) will alternatively be written as

$$X(t) = A\sin(\Phi t + \Theta)\ . \tag{10.2}$$

The value $X(t_1)$ at some specific time t_1 is also a random variable. In the context of this experiment, knowing the PMF associated with the selection of the numbers 1 to N involved in choosing an oscillator, as well as the specific amplitude, frequency, and phase of each oscillator, we could determine the probability distributions of any of the underlying random variables A, Φ, Θ, or $X(t_1)$ mentioned above.

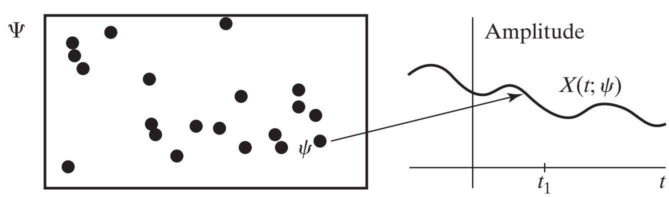


Figure 10.1 A random process.

Throughout this and later chapters, we will consider many examples of random processes. What is important at this point, however, is to develop a good mental picture of what a random process is. A random process is not just one signal but rather an ensemble of signals. This is illustrated schematically in Figure 10.2, for which the outcome of the probabilistic experiment could be any of the four waveforms indicated. Each waveform is deterministic, but the process is probabilistic or random because it is not known *a priori* which waveform will be generated by the probabilistic experiment. Consequently, prior to obtaining the outcome of the probabilistic experiment, many aspects of the signal are unpredictable, since there is uncertainty associated with which signal will be produced. After the experiment, or *a posteriori*, the outcome is totally determined.

If we focus on the values that a CT random process $X(t)$ can take at a particular instant of time, say t_1—that is, if we look down the entire ensemble at a fixed time—what we have is a random variable, namely $X(t_1)$. If we focus on the ensemble of values taken at an arbitrary collection of ℓ fixed time instants $t_1 < t_2 < \cdots < t_\ell$ for some arbitrary positive integer ℓ, we have a set of ℓ jointly distributed random variables $X(t_1), X(t_2), \cdots, X(t_\ell)$, all determined together by the outcome of the underlying probabilistic experiment. From this point of view, a random process can be thought of as a family of jointly distributed random variables indexed by t. A full probabilistic characterization of this collection of random variables would require the joint probability density functions (PDFs) of multiple samples of the signal, taken at arbitrary times:

$$f_{X(t_1),X(t_2),\,\cdots\,,X(t_\ell)}(x_1, x_2, \cdots, x_\ell) \tag{10.3}$$

for all ℓ and all $t_1, t_2, \cdots, t_\ell$.

Correspondingly, a DT random process consists of a collection of random variables $X[n]$ for all integer values of n, with a full probabilistic characterization consisting of the joint PDF

$$f_{X[n_1],X[n_2],\,\cdots\,,X[n_\ell]}(x_1, x_2, \cdots, x_\ell) \tag{10.4}$$

for all ℓ and all integers $n_1, \cdots, n_\ell$.

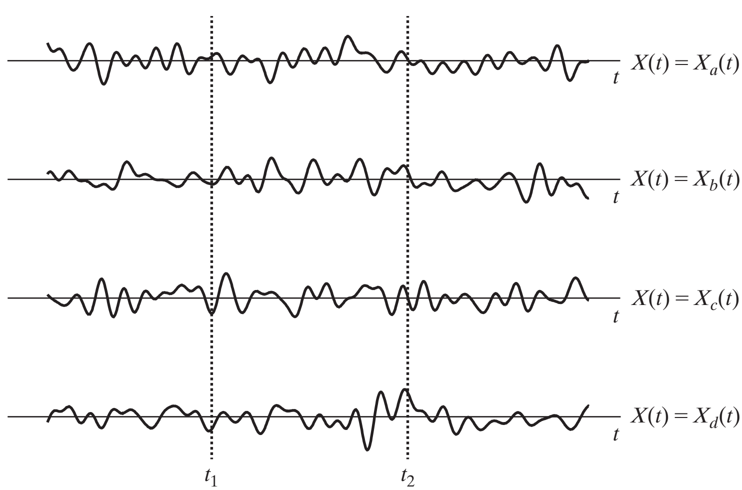


Figure 10.2 Realizations of the random process $X(t)$.

In a general context, it would be impractical to have a full characterization of a random process through Eqs. (10.3) or (10.4). As we will see in Example 10.2 and in other examples in this chapter, in many useful cases the full characterization can be inferred from a simpler probabilistic characterization. Furthermore, for much of what we deal with in this text, a characterization of a random process through first and second moments, as discussed in Section 10.2, is useful and sufficient.

Example 10.2 An Independent Identically Distributed (I.I.D.) Process

Consider a DT random process whose values $X[n]$ may be regarded as independently chosen at each time n from a fixed PDF $f_X(x)$, so the values are independent and identically distributed, thereby yielding what is called an independent identically distributed (i.i.d.) process. Such processes are widely used in modeling and simulation. For example, suppose a particular DT communication channel corrupts a transmitted signal with added noise. If the noise takes on independent values at each time instant, but with characteristics that seem unchanging over the time window of interest, then the noise may be well modeled as an i.i.d. process. It is also easy to generate an i.i.d. process in a simulation environment, provided a random number generator can be arranged to produce samples from a specified PDF. Processes with more complicated dependence across time samples can then be obtained by filtering or other operations on the i.i.d. process, as we will see in this chapter as well as the next.

For an i.i.d. process, we can write the joint PDF as a product of the marginal densities, that is,

$$f_{X[n_1],X[n_2],\cdots X[n_\ell]}(x_1,x_2,\cdots,x_\ell) = f_X(x_1)f_X(x_2)\cdots f_X(x_\ell) \tag{10.5}$$

for any choice of ℓ and $n_1,\cdots,n_\ell$.

An important set of questions that arises as we work with random processes in later chapters of this text is whether, by observing just part of the outcome of a random process, we can determine the complete outcome. The answer will depend on the details of the random process. For the process in Example 10.1, the answer is yes, but in general the answer is no. For some random processes, having observed the outcome in a given time interval might provide sufficient information to know exactly which ensemble member it corresponds to. In other cases this will not be sufficient. Some of these aspects are explored in more detail later, but we conclude this section with two additional examples that further emphasize these points.

Example 10.3 Ensemble of Batteries

Consider a collection of N batteries, with N_i of the batteries having voltage v_i, where v_i is an integer between 1 and 10. The plot in Figure 10.3 indicates the number of batteries with each value v_i. The probabilistic experiment is to choose one of the batteries, with

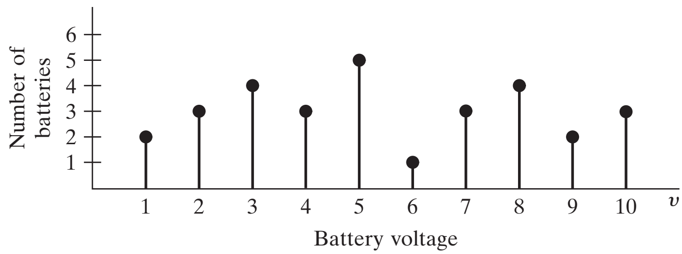


Figure 10.3 Plot of battery voltage distribution for Example 10.3.

the probability of picking any specific one being $\frac{1}{N}$, that is, any one battery is equally likely to be picked. Thus, scaling Figure 10.3 by $\frac{1}{N}$ represents the PMF for the battery voltage obtained as the outcome of the probabilistic experiment. Since the battery voltage is a signal (which in this case happens to be constant with time), this probabilistic experiment generates a random process. In fact, this example is similar to the oscillator example discussed earlier, but with frequency and phase both zero so that only the amplitude is random, and restricted to be an integer.

For this example, observation of $X(t)$ at any one time is sufficient information to determine the outcome for all time.

Example 10.3 is a very simple random process that, together with Example 10.4, helps to visualize some important general concepts of stationarity and ergodicity associated with random processes.

Example 10.4 Ensemble of Coin Tossers

In this example, consider a collection of N people, each independently having written down a long arbitrary string of 1s and 0s, with each entry chosen independently of any other entry in their string (similar to a sequence of independent coin tosses), and with an identical probability of a 1 at each entry. The random process now comprises this ensemble of the strings of 1s and 0s. A realization of the process is obtained by randomly selecting a person (and therefore one of the N strings of 1s and 0s). After selection, the specific ensemble member of the random process is totally determined.

Next, suppose that you are shown only the 10th entry in the selected string. Because of the manner in which the string was generated, it is clearly not possible from that information to determine the 11th entry. Similarly, if the entire past history up to the 10th entry was revealed, it would not be possible to determine the remaining sequence beyond the tenth.

While the entire sequence has been determined in advance by the nature of the experiment, partial observation of a given ensemble member is in general not sufficient to fully specify that member.

Rather than looking at the nth entry of a single ensemble member, we can consider the random variable corresponding to the values from the entire ensemble at the nth entry. Looking down the ensemble at $n = 10$, for example, we would see 1s and 0s in a ratio consistent with the probability of a 1 or 0 being chosen by each individual at $n = 10$.

10.2 FIRST- AND SECOND-MOMENT CHARACTERIZATION OF RANDOM PROCESSES

In the above discussion, we noted that a random process can be thought of as a family of jointly distributed random variables indexed by t or n. However it would in general be extremely difficult or impossible to analytically represent a random process in this way. Fortunately, the most widely used random process models have special structure that permits computation of such a statistical specification. Also, particularly when we are processing our signals with linear systems, we often design the processing or analyze the results by considering only the first and second moments of the process.

The first moment or mean function of a CT random process $X(t)$, which we typically denote as $\mu_X(t)$, is the expected value of the random variable $X(t)$ at each time t, that is,

$$\mu_X(t) = E[X(t)] \, . \tag{10.6}$$

The autocorrelation function and the autocovariance function represent second moments. The autocorrelation function $R_{XX}(t_1, t_2)$ is

$$R_{XX}(t_1, t_2) = E[X(t_1)X(t_2)] \tag{10.7}$$

and the autocovariance function $C_{XX}(t_1, t_2)$ is

$$\begin{aligned} C_{XX}(t_1, t_2) &= E[(X(t_1) - \mu_X(t_1))(X(t_2) - \mu_X(t_2))] \\ &= R_{XX}(t_1, t_2) - \mu_X(t_1)\mu_X(t_2) \, , \end{aligned} \tag{10.8}$$

where t_1 and t_2 are two arbitrary time instants. The word *auto* (which is sometimes dropped to simplify the terminology) refers to the fact that both samples in the correlation function or the covariance function come from the same process.

One case in which the first and second moments actually suffice to completely specify the process is a Gaussian process, defined as a process whose samples are always jointly Gaussian, represented by the generalization of the bivariate Gaussian to many variables.

We can also consider multiple random processes, for example, two processes, $X(\cdot)$ and $Y(\cdot)$. A full stochastic characterization of this requires the PDFs of all possible combinations of samples from $X(\cdot)$ and $Y(\cdot)$. We say that $X(\cdot)$ and $Y(\cdot)$ are independent if every set of samples from $X(\cdot)$ is independent of every set of samples from $Y(\cdot)$, so that the joint PDF factors as follows:

$$\begin{aligned} & f_{X(t_1), \cdots X(t_k), Y(t'_1), \cdots Y(t'_\ell)}(x_1, \cdots, x_k, y_1, \cdots, y_\ell) \\ &= f_{X(t_1), \cdots X(t_k)}(x_1, \cdots, x_k) . f_{Y(t'_1), \cdots Y(t'_\ell)}(y_1, \cdots, y_\ell) \end{aligned} \tag{10.9}$$

for all k, ℓ, and all choices of sample times.

If only first and second moments are of interest, then in addition to the individual first and second moments of $X(\cdot)$ and $Y(\cdot)$, we need to consider the cross-moment functions. Specifically, the cross-correlation function $R_{XY}(t_1,t_2)$ and the cross-covariance function $C_{XY}(t_1,t_2)$ are defined respectively as

$$R_{XY}(t_1,t_2) = E[X(t_1)Y(t_2)], \text{ and} \tag{10.10}$$

$$\begin{aligned} C_{XY}(t_1,t_2) &= E[(X(t_1)-\mu_X(t_1))(Y(t_2)-\mu_Y(t_2))] \\ &= R_{XY}(t_1,t_2)-\mu_X(t_1)\mu_Y(t_2) \end{aligned} \tag{10.11}$$

for arbitrary time t_1,t_2. If $C_{XY}(t_1,t_2)=0$ for all t_1,t_2, we say that the processes $X(\cdot)$ and $Y(\cdot)$ are uncorrelated. Note again that the term *uncorrelated* in its common usage means that the processes have zero covariance rather than zero correlation.

The above discussion carries over to the case of DT random processes, with the exception that now the sampling instants are restricted to integer times. In accordance with our convention of using square brackets $[\cdot]$ around the time argument for DT signals, we will write $\mu_X[n]$ for the mean function of a random process $X[\cdot]$ at time n. Similarly, we will write $R_{XX}[n_1,n_2]$ and $C_{XX}[n_1,n_2]$ for the correlation and covariance functions involving samples at times n_1 and n_2, and $R_{XY}[n_1,n_2]$ and $C_{XY}[n_1,n_2]$ for the cross-moment functions of two random variables $X[\cdot]$ and $Y[\cdot]$ sampled at times n_1 and n_2 respectively.

10.3 STATIONARITY

10.3.1 Strict-Sense Stationarity

In general, we would expect that the joint PDFs associated with the random variables obtained by sampling a random process at an arbitrary number ℓ of arbitrary times will be time-dependent, that is, the joint PDF

$$f_{X(t_1),\cdots,X(t_\ell)}(x_1,\cdots,x_\ell) \tag{10.12}$$

will depend on the specific values of $t_1,\cdots,t_\ell$. If all the joint PDFs remain the same under arbitrary time shifts, so that if

$$f_{X(t_1),\cdots,X(t_\ell)}(x_1,\cdots,x_\ell) = f_{X(t_1+\alpha),\cdots,X(t_\ell+\alpha)}(x_1,\cdots,x_\ell) \tag{10.13}$$

for arbitrary α, then the random process is said to be strict-sense stationary (SSS). Said another way, for an SSS process, the statistics depend only on the relative times at which the samples are taken, not on the absolute times. The processes in Examples 10.2 and 10.3 are SSS. More generally, any i.i.d. process is strict-sense stationary.

10.3.2 Wide-Sense Stationarity

Of particular use is a less restricted type of stationarity. Specifically, if the mean value $\mu_X(t)$ is invariant with time and the autocorrelation $R_{XX}(t_1,t_2)$ or, equivalently, the autocovariance $C_{XX}(t_1,t_2)$ is a function of only the time difference (t_1-t_2), then the process is referred to as wide-sense stationary

(WSS). A process that is SSS is always WSS, but the reverse is not necessarily true. For a WSS random process $X(t)$, we have

$$\mu_X(t) = \mu_X \tag{10.14}$$

$$\begin{aligned} R_{XX}(t_1, t_2) &= R_{XX}(t_1 + \alpha, t_2 + \alpha) \text{ for every } \alpha \\ &= R_{XX}(t_1 - t_2, 0) \\ &= R_{XX}(t_1 - t_2)\,, \end{aligned} \tag{10.15}$$

where the last equality defines a more compact notation since a single argument for the time difference $(t_1 - t_2)$ suffices for a WSS process. Similarly, $C_{XX}(t_1, t_2)$ will be written as $C_{XX}(t_1 - t_2)$ for a WSS process. The time difference $(t_1 - t_2)$ will typically be denoted as τ and referred to as the lag variable for the autocorrelation and autocovariance functions.

For a Gaussian process, that is, a process whose samples are always jointly Gaussian, WSS implies SSS because jointly Gaussian variables are entirely determined by their joint first and second moments.

Two random processes $X(\cdot)$ and $Y(\cdot)$ are referred to as jointly WSS if their first and second moments (including the cross-covariance) are stationary. In this case, we use the notation $R_{XY}(\tau)$ to denote $E[X(t+\tau)Y(t)]$. It is worth noting that an alternative convention sometimes used elsewhere is to define $R_{XY}(\tau)$ as $E[X(t)Y(t+\tau)]$. In our notation, this expectation would be denoted by $R_{XY}(-\tau)$. It is important to take account of what notational convention is being followed when referencing other sources, and you should also be clear about the notational convention used in this text.

Example 10.5 Random Oscillators Revisited

Consider again the harmonic oscillators introduced in Example 10.1:

$$X(t; A, \Theta) = A\cos(\phi_0 t + \Theta) \tag{10.16}$$

where A and Θ are independent random variables, and now the frequency is fixed at some known value denoted by ϕ_0.

If Θ is also fixed at a constant value θ_0, then every outcome is of the form $x(t) = A\cos(\phi_0 t + \theta_0)$, and it is straightforward to see that this process is not WSS (and consequently also not SSS). For instance, if A has a nonzero mean value, $\mu_A \neq 0$, then the expected value of the process, namely $\mu_A \cos(\phi_0 t + \theta_0)$, is time varying. To show that the process is not WSS even when $\mu_A = 0$, we can examine the autocorrelation function. Note that $x(t)$ is fixed at 0 for all values of t for which $\phi_0 t + \theta_0$ is an odd multiple of $\pi/2$, and takes the values $\pm A$ halfway between such points; the correlation between such samples taken π/ϕ_0 apart in time can correspondingly be 0 (in the former case) or $-E[A^2]$ (in the latter). The process is thus not WSS, even when $\mu_A = 0$.

However, if Θ is distributed uniformly in $[-\pi, \pi]$, then

$$\mu_X(t) = \mu_A \int_{-\pi}^{\pi} \frac{1}{2\pi} \cos(\phi_0 t + \theta)\, d\theta = 0\,, \tag{10.17}$$

$$\begin{aligned} C_{XX}(t_1, t_2) &= R_{XX}(t_1, t_2) \\ &= E[A^2]E[\cos(\phi_0 t_1 + \Theta)\cos(\phi_0 t_2 + \Theta)]\,. \end{aligned} \tag{10.18}$$

Equation (10.18) can be evaluated as

$$C_{XX}(t_1,t_2)=\frac{E[A^2]}{2}\int_{-\pi}^{\pi}\frac{1}{2\pi}[\cos(\phi_0(t_2-t_1))+\cos(\phi_0(t_2+t_1)+2\theta)]\,d\theta \tag{10.19}$$

to obtain

$$C_{XX}(t_1,t_2)=\frac{E[A^2]}{2}\cos(\phi_0(t_2-t_1))\,. \tag{10.20}$$

For this restricted case, then, the process is WSS. It can also be shown to be SSS, although this is not totally straightforward to show formally.

For the most part, the random processes that we treat will be WSS. As noted earlier, to simplify notation for a WSS process, we write the correlation function as $R_{XX}(t_1-t_2)$; the argument (t_1-t_2) is often denoted by the lag variable τ at which the correlation is computed. When considering only first and second moments and not the entire PDF or cumulative distribution function (CDF), it will be less important to distinguish between the random process $X(t)$ and a specific realization $x(t)$ of it—so a further notational simplification is introduced by using lowercase letters to denote the random process itself. We shall thus refer to the random process $x(t)$, and—in the case of a WSS process—denote its mean by μ_x and its correlation function $E[x(t+\tau)x(t)]$ by $R_{xx}(\tau)$. Correspondingly, for DT we refer to the random process $x[n]$ and, in the WSS case, denote its mean by μ_x and its correlation function $E[x[n+m]x[n]]$ by $R_{xx}[m]$.

10.3.3 Some Properties of WSS Correlation and Covariance Functions

For real-valued WSS processes $x(t)$ and $y(t)$, the correlation and covariance functions have the following symmetry properties:

$$R_{xx}(\tau)=R_{xx}(-\tau)\,,\qquad C_{xx}(\tau)=C_{xx}(-\tau)\,, \tag{10.21}$$

$$R_{xy}(\tau)=R_{yx}(-\tau)\,,\qquad C_{xy}(\tau)=C_{yx}(-\tau)\,. \tag{10.22}$$

For example, the symmetry in Eq. (10.22) of the cross-correlation function $R_{xy}(\tau)$ follows directly from interchanging the arguments inside the defining expectations:

$$R_{xy}(\tau)=E[x(t)y(t-\tau)] \tag{10.23a}$$

$$=E[y(t-\tau)x(t)] \tag{10.23b}$$

$$=R_{yx}(-\tau)\,. \tag{10.23c}$$

The other properties in Eqs. (10.21) and (10.22) follow in a similar manner.

Equation (10.21) indicates that the autocorrelation and autocovariance functions have even symmetry. Equation (10.22) indicates that for cross-correlation and cross-covariance functions, interchanging the random variables is equivalent to reflecting the function about the τ axis. And of course, Eq. (10.21) is a special case of Eq. (10.22) with $y(t) = x(t)$. Similar properties hold for DT WSS processes.

Another important property of correlation and covariance functions follows from noting that, as discussed in Section 7.7, Eq. (7.63), the correlation coefficient of two random variables has magnitude not exceeding 1. Specifically since the correlation coefficient between $x(t)$ and $x(t+\tau)$ is given by $C_{xx}(\tau)/C_{xx}(0)$, then

$$-1 \leq \frac{C_{xx}(\tau)}{C_{xx}(0)} \leq 1 \, , \tag{10.24}$$

or equivalently,

$$-C_{xx}(0) \leq C_{xx}(\tau) \leq C_{xx}(0) \, . \tag{10.25}$$

Adding μ_x^2 to each term above, we can conclude that

$$-R_{xx}(0) + 2\mu_x^2 \leq R_{xx}(\tau) \leq R_{xx}(0) \, . \tag{10.26}$$

In Chapter 11, we will demonstrate that correlation and covariance functions are characterized by the property that their Fourier transforms are real and nonnegative at all frequencies, because these transforms describe the frequency distribution of the expected power in the random process. The above symmetry constraints and bounds will then follow as natural consequences, but they are worth highlighting here.

We conclude this section with two additional examples. The first, the Bernoulli process, is the more formal name for repeated independent flips of a possibly biased coin. The second example, referred to as the random telegraph wave, is often used as a simplified representation of a random square wave or switch in electronics or communication systems.

Example 10.6 The Bernoulli Process

The Bernoulli process is an example of an i.i.d. DT process with

$$P(x[n] = 1) = p \tag{10.27}$$

$$P(x[n] = -1) = (1 - p) \tag{10.28}$$

and with the value at each time instant n independent of the values at all other time instants. The mean, autocorrelation, and covariance functions are:

$$E\left\{x[n]\right\} = 2p - 1 = \mu_x \tag{10.29}$$

$$E\left\{x[n+m]x[n]\right\} = \begin{cases} 1 & m = 0 \\ (2p-1)^2 & m \neq 0 \end{cases} \tag{10.30}$$

$$C_{xx}[m] = E\{(x[n+m] - \mu_x)(x[n] - \mu_x)\} \tag{10.31}$$

$$= \{1 - (2p-1)^2\}\delta[m] = 4p(1-p)\delta[m] \, . \tag{10.32}$$