CHAPTER 14

Bayesian Filtering for State Estimation of Dynamic Systems

ORGANIZATION OF THE CHAPTER

This chapter focuses on an issue of fundamental importance: the estimation of the hidden state of a dynamic system, given a set of observations.

The chapter is organized as follows:

1. The introductory section, Section 14.1, motivates interest in the study of sequential state estimation.
2. Section 14.2 discusses the notion of a state space and the different ways of modeling it.
3. The celebrated Kalman filter is derived in Section 14.3, which is followed by treatment of square-root implementation of the filter for assured numerical stability in Section 14.4. Section 14.5 derives the extended Kalman filter for dealing with situations where the nonlinearity is of a "mild" sort.
4. Section 14.6 discusses Bayesian filtering, which provides a unifying framework for the state estimation of dynamic systems, at least conceptually; this filtering model includes the Kalman filter as a special case.
5. Section 14.7 presents a description of the cubature rule for direct numerical approximation of the Bayesian filter, paving the way for the description of a new filter, the cubature Kalman filter, which builds on ideas from Kalman filter theory.
6. Section 14.8 addresses another approach for approximating the Bayesian filter; this one is rooted in Monte Carlo simulation. In particular, a detailed treatment of particle filters is presented. A computer experiment comparing the performance of the extended Kalman filter and a particular form of the particle filter is presented in Section 14.9.
7. Section 14.10 discusses the role of Kalman filtering in modeling different parts of the human brain.

The chapter concludes with a summary and discussion in Section 14.11.

14.1 INTRODUCTION

In the neurodynamic systems studied in Chapter 13, the main issue of concern was stability. In this chapter, we consider another important issue: estimation of the state of a dynamic system, given a sequence of observations dependent on the state in some fashion.

The observations take place in discrete time, not for mathematical convenience, but because that is how they arise naturally. Moreover, the state is not only unknown, but also *hidden* from the observer. We may therefore view the state-estimation problem as an inverse problem.

As an illustrative example, consider a dynamically driven multiplayer perceptron with feedback loops from one layer of the network to a preceding one (e.g., from a hidden layer to the input layer). The state of the network could be viewed as a vector made up of all the synaptic weights of the network, arranged in some orderly fashion. What we would like to do is to use sequential state-estimation theory to adjust the weight vector of the network in a supervised manner, given a training sample. This application is discussed in detail in the next chapter. For this application, however, we need a sequential procedure for state estimation, the rationale for which is deferred to that chapter.

The first rigorous treatment of sequential state-estimation theory appeared in Kalman's classic paper, published in 1960. Kalman's exposition was based on two simplifying assumptions for mathematical tractability:

1. The dynamic system is entirely *linear.*
2. The noise processes perturbing the state of the dynamic system and the observables are *additive and Gaussian.*

In making these assumptions, Kalman derived an *optimal estimate* of the unknown state of the system, the computation of which was performed *recursively.* Within its domain of applicability, the Kalman filter has undoubtedly withstood the test of time.

Sequential state-estimation theory remains an active area of research. Much of this research has focused on how to deal with the practical issues of nonlinearity and non-Gaussianity. Under one or both of these conditions, optimal estimation of the state is no longer an option. Rather, we have to settle on the realization of an *approximate estimator.* The challenge is how to derive such an estimator that is both principled and computationally efficient.

14.2 STATE-SPACE MODELS

All dynamic systems share a basic feature: the *state* of the system. We formally define this feature as follows:

> *The state of a stochastic dynamic system is defined as the minimal amount of information about the effects of past inputs applied to the system that is sufficient to completely describe the future behavior of the system.*

Typically, the state is *not* measurable directly. Rather, in an indirect manner, the state makes its effect on the outside world measurable through a set of *observables.* As such, the characterization of an unknown dynamic system is described by a *state-space model*, which embodies a pair of equations:

1. *The system (state) model*, which, formulated as a *first-order Markov chain*, describes the evolution of the state as a function of time, as shown by

$$\mathbf{x}_{n+1} = \mathbf{a}_n(\mathbf{x}_n, \boldsymbol{\omega}_n) \tag{14.1}$$

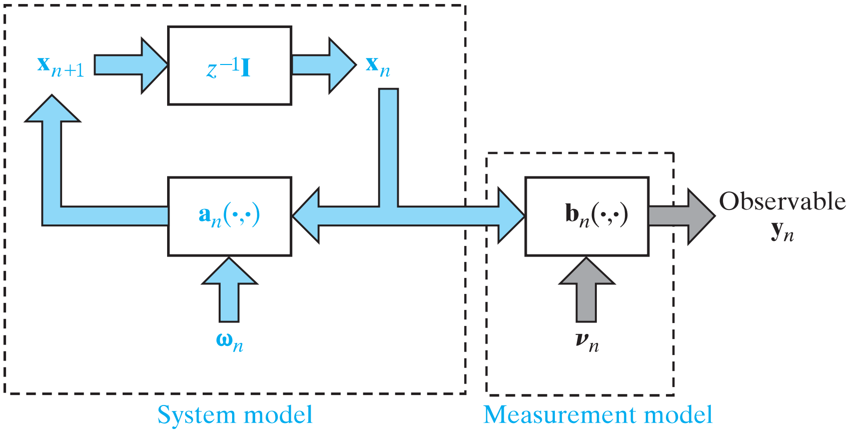

FIGURE 14.1 Generic state-space model of a time-varying, nonlinear dynamic system, where $z^{-1}\mathbf{I}$ denotes a block of unit-time delays.

where n denotes discrete time, the vector $\mathbf{x}_n$ denotes the current value of the state, and $\mathbf{x}_{n+1}$ denotes the subsequent value of the state; the vector $\boldsymbol{\omega}_n$ denotes *dynamic noise*, or *process noise*, and $\mathbf{a}_n(\cdot,\cdot)$ is a vectorial function of its two arguments.

2. *The measurement (observation) model*, which is formulated as

$$\mathbf{y}_n = \mathbf{b}_n(\mathbf{x}_n, \boldsymbol{\nu}_n) \tag{14.2}$$

where the vector $\mathbf{y}_n$ denotes a set of observables, the vector $\boldsymbol{\nu}_n$ denotes *measurement noise*, and $\mathbf{b}_n(\cdot,\cdot)$ denotes another vectorial function.

The subscript n in both $\mathbf{a}_n$ and $\mathbf{b}_n$ is included to cover situations where these two functions are *time varying*. For the state-space model to be of practical value, it must closely describe the underlying physics of the system under study.

Figure 14.1 depicts a signal-flow graph representation of the state-space model defined by Eqs. (14.1) and (14.2), and Fig. 14.2 depicts the state's evolution across time as a *Markov chain*. The time-domain representation of the model depicted in these two figures offers certain attributes:

- mathematical and notational convenience;
- a close relationship of the model to physical reality;
- a meaningful basis of accounting for the statistical behavior of the system.

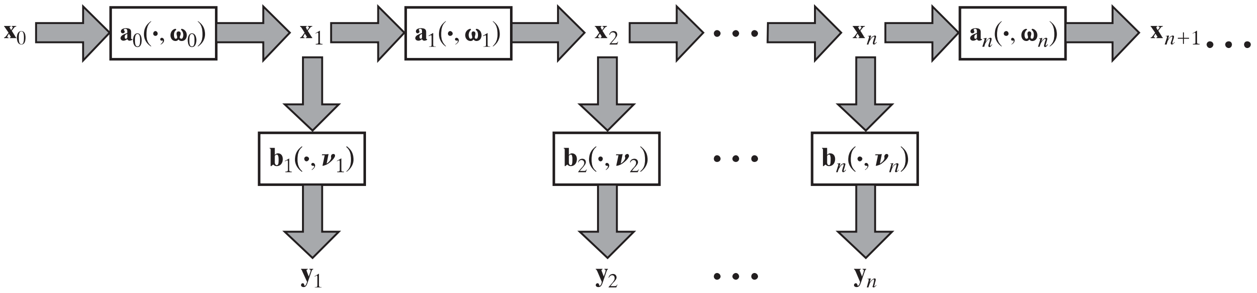
$\mathbf{x}_0$
$\mathbf{a}_0(\cdot, \boldsymbol{\omega}_0)$
$\mathbf{x}_1$
$\mathbf{a}_1(\cdot, \boldsymbol{\omega}_1)$
$\mathbf{x}_2$
• • •
$\mathbf{x}_n$
$\mathbf{a}_n(\cdot, \boldsymbol{\omega}_n)$
$\mathbf{x}_{n+1}$ • • •
$\mathbf{b}_1(\cdot, \boldsymbol{\nu}_1)$
$\mathbf{b}_2(\cdot, \boldsymbol{\nu}_2)$
• • •
$\mathbf{b}_n(\cdot, \boldsymbol{\nu}_n)$
$\mathbf{y}_1$
$\mathbf{y}_2$
• • •
$\mathbf{y}_n$

FIGURE 14.2 Evolution of the state across time, viewed as a first-order Markov chain.

Justifiably, the following assumptions are made:

1. The initial state $\mathbf{x}_0$ is uncorrelated with the dynamic noise $\boldsymbol{\omega}_n$ for all n.
2. The two sources of noise, $\boldsymbol{\omega}_n$ and $\boldsymbol{\nu}_n$, are statistically independent, which means that

$$\mathbb{E}[\boldsymbol{\omega}_n \boldsymbol{\nu}_k^T] = \mathbf{0} \qquad \text{for all } n \text{ and } k \tag{14.3}$$

This equation is a sufficient condition for independence when $\boldsymbol{\omega}_n$ on $\boldsymbol{\nu}_n$ are *jointly Gaussian.*

It is noteworthy that the Markovian model of Fig. 14.2 is fundamentally *different* from the Morkovian model considered in Chapter 12, which covered dynamic programming. Whereas in dynamic programming the state is directly *accessible* to the observer, in sequential state estimation the state is *hidden* from the observer.

Statement of the Sequential State-Estimation Problem

Given an entire record of observations consisting of $\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n$, *compute an estimate of the hidden state* $\mathbf{x}_k$ *that is optimal in some statistical sense, with the estimation being performed in a sequential manner.*

In a way, this statement embodies two systems:

- the unknown dynamic system, whose observable $\mathbf{y}_n$ is a function of the hidden state;
- the sequential state estimator or filter, which exploits the information about the state that is contained in the observables.

In a loose sense, we may view this problem as an "encoding–decoding" problem, with the observables representing an encoded version of the state and the state estimate produced by the filter representing a decoded version of the observables.

In any event, the state-estimation problem is called *prediction* if $k > n$, *filtering* if $k = n$, and *smoothing* if $k < n$. Typically, a smoother is statistically more accurate than both the predictor and filter, as it uses more observables. On the other hand, both prediction and filtering can be performed in real time, whereas smoothing cannot.

Hierarchy of State-Space Models

The mathematical difficulty of solving the state-estimation problem is highly dependent on how the state-space model is actually described, leading to the following hierarchy of models:

1. *Linear, Gaussian model.* In this model, which is the simplest of state-space models, Eqs. (14.1) and (14.2) respectively reduce to

$$\mathbf{x}_{n+1} = \mathbf{A}_{n+1,n}\mathbf{x}_n + \boldsymbol{\omega}_n \tag{14.4}$$

and

$$\mathbf{y}_n = \mathbf{B}_n\mathbf{x}_n + \boldsymbol{\nu}_n \tag{14.5}$$

where $\mathbf{A}_{n+1,n}$ is the *transition matrix* from state $\mathbf{x}_n$ to state $\mathbf{x}_{n+1}$ and $\mathbf{B}_n$ is the *measurement matrix*. The dynamic noise $\boldsymbol{\omega}_n$ and measurement noise $\boldsymbol{\nu}_n$ are both additive and assumed to be *statistically independent zero-mean Gaussian processes*[1] whose covariance matrices are respectively denoted by $\mathbf{Q}_{\omega,n}$ and $\mathbf{Q}_{\nu,n}$. The state-space model defined by Eqs. (14.4) and (14.5) is indeed the model that was used by Kalman to derive his recursive filter, which is mathematically elegant and devoid of any approximation. Kalman filters are discussed in Section 14.3.

2. *Linear, non-Gaussian model.* In this second model, we still use Eqs. (14.4) and (14.5), but the dynamic noise $\boldsymbol{\omega}_n$ and measurement noise $\boldsymbol{\nu}_n$ are now assumed to be additive, statistically independent, *non-Gaussian* processes. The non-Gaussianity of these two processes is therefore the only source of mathematical difficulty. In situations of this kind, we may extend the application of the Kalman filter by using the *Gaussian-sum approximation*, summarized as follows:

> *Any probability density function $p(\mathbf{x})$ describing a multidimensional non-Gaussian vector, represented by the sample value $\mathbf{x}$, can be approximated as closely as desired by the Gaussian-sum formula*
>
> $$p(\mathbf{x}) = \sum_{i=1}^{N} c_i \mathcal{N}(\bar{\mathbf{x}}_i, \boldsymbol{\Sigma}_i) \tag{14.6}$$
>
> *for some integer N and positive scalers c_i, with $\sum_{i=1}^{N} c_i = 1$. The term $\mathcal{N}(\bar{\mathbf{x}}_i, \boldsymbol{\Sigma}_i)$ stands for a Gaussian (normal) density function with mean $\bar{\mathbf{x}}_i$ and covariance matrix $\boldsymbol{\Sigma}_i$* for $i = 1, 2, ..., N$.

The Gaussian sum on the right-hand side of Eq. (14.6) converges uniformly to the given probability density function $p_{\mathbf{X}}(\mathbf{x})$ as the number of terms, N, increases and the covariance matrices $\boldsymbol{\Sigma}_i$ approach zero for all i (Anderson and Moore, 1971). To compute the Gaussian-sum approximation of Eq. (14.6) for a prescribed probability density function $p(\mathbf{x})$, we may, for example, use a procedure based on the *expectation-maximization* (EM) algorithm; this algorithm was described in Chapter 11. Then, having computed this approximation, we may use a bank of Kalman filters to solve the sequential state-estimation problem described by a linear, non-Gaussian model (Alspach and Sorenson, 1972). Note, however, that the terms in a Gaussian-sum model tend to grow exponentially over the course of time, which may therefore require the use of a pruning algorithm.

3. *Nonlinear, Gaussian model.* The third model in the hierarchy of state-space models of increasing complexity is formulated as

$$\mathbf{x}_{n+1} = \mathbf{a}_n(\mathbf{x}_n) + \boldsymbol{\omega}_n \tag{14.7}$$

and

$$\mathbf{y}_n = \mathbf{b}_n(\mathbf{x}_n) + \boldsymbol{\nu}_n \tag{14.8}$$

where the dynamic noise $\boldsymbol{\omega}_n$ and measurement noise $\boldsymbol{\nu}_n$ are both assumed to be additive and Gaussian. This is where we start to experience mathematical difficulty in solving a sequential state-estimation problem. There are basically two radically different approaches for computing an approximate solution to the problem:

(i) *Local approximation.* In this first approach to nonlinear filtering, the *nonlinear function* $\mathbf{a}_n(\cdot)$ in the system model of Eq. (14.7) and the nonlinear function $\mathbf{b}_n(\cdot)$ in the measurement model of Eq. (14.8) are approximated around localized estimates of the state, whereby both equations are *linearized*. The stage is then set for applying the Kalman filter to compute the approximate solution. The extended Kalman filter discussed in Section 14.5 is an example of the local-approximation approach to nonlinear filtering.

(ii) *Global approximation.* In this second approach to nonlinear filtering, the solution is formulated in a *Bayesian estimation framework* in such a way that difficult interpretations inherent to the problem are made mathematically tractable. Particle filters, discussed in Section 14.7, belong to this second approach to nonlinear filtering.

4. *Nonlinear, non-Gaussian model.* This last class of state-space models is described by Eqs. (14.1) and (14.2), where both the system model and the measurement model are nonlinear, and the dynamic noise $\boldsymbol{\omega}_n$ and measurement noise $\boldsymbol{\nu}_n$ are not only non-Gaussian, but may also be nonadditive. In this kind of scenario, particle filters are currently the method of choice, but not necessarily the only method, for solving the sequential state-estimation problem.

14.3 KALMAN FILTERS

The state-space model for the Kalman filter is defined by Eqs. (14.4) and (14.5). This linear Gaussian model is parameterized as follows:

- the transition matrix $\mathbf{A}_{n+1,n}$, which is invertible;
- the measurement matrix $\mathbf{B}_n$, which, in general, is a rectangular matrix;
- the Gaussian dynamic noise $\boldsymbol{\omega}_n$, which is assumed to have zero mean and covariance matrix $\mathbf{Q}_{\omega,n}$;
- the Gaussian measurement noise $\boldsymbol{\nu}_n$, which is assumed to have zero mean and covariance matrix $\mathbf{Q}_{\nu,n}$.

All these parameters are assumed to be known. We are also given the sequence of observables $\{\mathbf{y}_k\}_{k=1}^{n}$. The requirement is to derive an estimate of the state $\mathbf{x}_k$ that is optimized in *the minimum mean-square-error sense.* We will confine the discussion to filtering for which $k = n$, and one-step prediction for which $k = n + 1$.

The Innovations Process

An insightful way of deriving this optimum estimate is to use the so-called *innovations process* associated with the observable $\mathbf{y}_n$, which is defined by

$$\boldsymbol{\alpha}_n = \mathbf{y}_n - \hat{\mathbf{y}}_{n|n-1} \tag{14.9}$$

where $\hat{\mathbf{y}}_{n|n-1}$ is the minimum mean-square-error estimate of $\mathbf{y}_n$, given all the observables up to and including time $n - 1$. In effect, we can say the following:

> *The innovations process $\boldsymbol{\alpha}_n$ is that part of the observable $\mathbf{y}_n$ that is new, since the predictable part of $\mathbf{y}_n$—namely, $\hat{\mathbf{y}}_{n|n-1}$—is completely determined by the sequence $\{\mathbf{y}_k\}_{k=1}^{n-1}$.*

The innovations process has some important properties:

Property 1. The innovations process $\boldsymbol{\alpha}_n$ associated with the observable $\mathbf{y}_n$ is orthogonal to all past observables $\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_{n-1}$, as shown by

$$\mathbb{E}\boldsymbol{\alpha}_n\mathbf{y}_k^T] = \mathbf{0}, \qquad 1 \le k \le n-1 \tag{14.10}$$

Property 2. The innovations process consists of a sequence of random vectors that are orthogonal to each other, as shown by

$$\mathbb{E}[\boldsymbol{\alpha}_n\boldsymbol{\alpha}_k^T] = \mathbf{0}, \qquad 1 \le k \le n-1 \tag{14.11}$$

Property 3. There is a one-to-one correspondence between the sequence of random vectors $\{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n\}$, representing the observed data, and the sequence $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, ..., \boldsymbol{\alpha}_n\}$, representing the innovations process, in that the one sequence may be obtained from the other by means of linear stable operators *without any loss of information.* Thus, we may write

$$\{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n\} \rightleftharpoons \{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, ..., \boldsymbol{\alpha}_n\} \tag{14.12}$$

In light of these properties, we now see why it is easier to work with the innovations process rather than the observables themselves: In general, the observables are *correlated*, whereas the corresponding elements of the innovations processes are *not.*

Covariance Matrix of the Innovations Process

Starting with the initial condition $\mathbf{x}_0$, we may use the system model of Eq. (14.4) to express the state at time k as

$$\mathbf{x}_k = \mathbf{A}_{k,0}\mathbf{x}_0 + \sum_{i=1}^{k-1}\mathbf{A}_{k,i}\boldsymbol{\omega}_i \tag{14.13}$$

Equation (14.13) indicates that the state $\mathbf{x}_k$ is a linear combination of $\mathbf{x}_0$ and $\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, ..., \boldsymbol{\omega}_{k-1}$.

By hypothesis, the measurement noise $\boldsymbol{\nu}_n$ is uncorrelated with both the initial state $\mathbf{x}_0$ and the dynamic noise $\boldsymbol{\omega}_i$. Accordingly, postmultiplying both sides of Eq. (14.13) by $\boldsymbol{\nu}_n^T$ and taking expectations, we obtain

$$\mathbb{E}[\mathbf{x}_k\boldsymbol{\nu}_n^T] = \mathbf{0}, \qquad k, n \ge 0 \tag{14.14}$$

Correspondingly, we find from the measurement equation of Eq. (14.5) that

$$\mathbb{E}[\mathbf{y}_k\boldsymbol{\nu}_k^T] = \mathbf{0}, \qquad 0 \le k \le n-1 \tag{14.15}$$

and

$$\mathbb{E}[\mathbf{y}_k\boldsymbol{\omega}_n^T] = \mathbf{0}, \qquad 0 \le k \le n \tag{14.16}$$

Given the past observations $\mathbf{y}_1, ..., \mathbf{y}_{n-1}$, we also find from the measurement equation of Eq. (14.5) that the minimum mean-square estimate of the current observation $\mathbf{y}_n$ is

$$\hat{\mathbf{y}}_{n|n-1} = \mathbf{B}_n\hat{\mathbf{x}}_{n|n-1} + \hat{\boldsymbol{\nu}}_{n|n-1} \tag{14.17}$$

where $\hat{\boldsymbol{\nu}}_{n|n-1}$ is the corresponding estimate of the measurement noise, given the past observations $\mathbf{y}_1, ..., \mathbf{y}_{n-1}$. The estimate $\hat{\boldsymbol{\nu}}_{n|n-1}$ is zero, since $\boldsymbol{\nu}_n$ is orthogonal to the past observations in light of Eq. (14.15). We may therefore reduce Eq. (14.17) to

$$\hat{\mathbf{y}}_{n|n-1} = \mathbf{B}_n\hat{\mathbf{x}}_{n|n-1} \tag{14.18}$$

Substituting Eqs. (14.5) and (14.18) into Eq. (14.9) and then collecting terms, we obtain

$$\boldsymbol{\alpha}_n = \mathbf{B}_n\boldsymbol{\varepsilon}_{n|n-1} + \boldsymbol{\nu}_n \tag{14.19}$$

where the new term $\boldsymbol{\varepsilon}_{n|n-1}$ is the *state prediction-error vector*, defined by

$$\boldsymbol{\varepsilon}_{n,n-1} = \mathbf{x}_n - \hat{\mathbf{x}}_{n|n-1} \tag{14.20}$$

In Problem 14.1, it is shown that $\boldsymbol{\varepsilon}_{n|n-1}$ is orthogonal to both the dynamic noise $\boldsymbol{\omega}_n$ and the measurement noise $\boldsymbol{\nu}_n$. Then, defining the *covariance matrix of the zero-mean innovations process* $\boldsymbol{\alpha}_n$ as

$$\mathbf{R}_n = \mathbb{E}[\boldsymbol{\alpha}_n\boldsymbol{\alpha}_n^T] \tag{14.21}$$

and using Eq. (14.19), we may readily show that

$$\mathbf{R}_n = \mathbf{B}_n\mathbf{P}_{n|n-1}\mathbf{B}_n^T + \mathbf{Q}_{\nu,n} \tag{14.22}$$

where $\mathbf{Q}_{\nu,n}$ is the covariance matrix of the measurement noise $\boldsymbol{\nu}_n$ and the new term

$$\mathbf{P}_{n|n-1} = \mathbb{E}[\boldsymbol{\varepsilon}_{n|n-1}\boldsymbol{\varepsilon}_{n|n-1}^T] \tag{14.23}$$

is the *prediction-error covariance matrix.* Equation (14.22) is our first entry into the Kalman filtering algorithm.

Estimation of the Filtered State Using the Innovations Process: The predictor-corrector formula

Our next task is to derive the minimum mean-square-error estimate of the state $\mathbf{x}_i$ at some time i, based on the innovations process. To this end, given the innovations sequence $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, ..., \boldsymbol{\alpha}_m$, we first express the corresponding estimate of $\mathbf{x}_i$ as the linear expansion

$$\hat{\mathbf{x}}_{i|n} = \sum_{k=1}^{n}\mathbf{C}_{i,k}\boldsymbol{\alpha}_k \tag{14.24}$$

where $\{\mathbf{C}_{i,k}\}_{k=1}^{n}$ is a set of matrices assuming the role of coefficients of the expansion for time i. The state-prediction error and the innovations process satisfy the following orthogonality condition (see Problem 14.3):

$$\mathbb{E}[\boldsymbol{\varepsilon}_{i|n}\boldsymbol{\alpha}_k^T] = \mathbf{0} \qquad \begin{array}{l}\text{for } k = 1, 2, ..., n\\ \text{and } i \geq n\end{array} \tag{14.25}$$

Hence, substituting Eq. (14.24) into Eq. (14.25) and using the orthogonality property of the innovations process described in Eq. (14.11), we obtain

$$\mathbb{E}[\mathbf{x}_i\boldsymbol{\alpha}_k^T] = \mathbf{C}_{i,k}\mathbf{R}_k$$

where, as defined previously, $\mathbf{R}_k$ is the convariance matrix of the innovations process. Solving this equation for the coefficient matrix $\mathbf{C}_{i,k}$, we thus have

$$\mathbf{C}_{i,k} = \mathbb{E}[\mathbf{x}_i \boldsymbol{\alpha}_k^T]\mathbf{R}_k^{-1}$$

The use of this expression in Eq. (14.24) yields

$$\hat{\mathbf{x}}_{i|n} = \sum_{k=1}^{n} \mathbb{E}[\mathbf{x}_i \boldsymbol{\alpha}_k^T]\mathbf{R}_k^{-1}\boldsymbol{\alpha}_k \tag{14.26}$$

For $i = n$, corresponding to the process of *filtering*, we may use Eq. (14.26) to express the filtered estimate of the state as

$$\begin{aligned}\hat{\mathbf{x}}_{n|n} &= \sum_{k=1}^{n} \mathbb{E}[\mathbf{x}_n \boldsymbol{\alpha}_k^T]\mathbf{R}_k^{-1}\boldsymbol{\alpha}_k \\ &= \sum_{k=1}^{n-1} \mathbb{E}[\mathbf{x}_n \boldsymbol{\alpha}_k^T]\mathbf{R}_k^{-1}\boldsymbol{\alpha}_k + \mathbb{E}[\mathbf{x}_n \boldsymbol{\alpha}_n^T]\mathbf{R}_n^{-1}\boldsymbol{\alpha}_n \end{aligned}\tag{14.27}$$

where, in the second line, the term corresponding to $k = n$ has been isolated from the summation. In order to put Eq. (14.27) into an interpretable form, we first use Eq. (14.26) to write

$$\hat{\mathbf{x}}_{n|n-1} = \sum_{k=1}^{n-1} \mathbb{E}[\mathbf{x}_n \boldsymbol{\alpha}_k^T]\mathbf{R}_k^{-1}\boldsymbol{\alpha}_k \tag{14.28}$$

To simplify the second term in Eq. (14.27), we introduce the following definition:

$$\mathbf{G}_n = \mathbb{E}[\mathbf{x}_n \boldsymbol{\alpha}_n^T]\mathbf{R}_n^{-1} \tag{14.29}$$

Accordingly, we may now express the filtered estimate of the state as the recursion:

$$\hat{\mathbf{x}}_{n|n} = \hat{\mathbf{x}}_{n|n-1} + \mathbf{G}_n\boldsymbol{\alpha}_n \tag{14.30}$$

The two terms comprising the right-hand side of Eq. (14.30) may now be interpreted as follows:

1. The term $\hat{\mathbf{x}}_{n|n-1}$ represents *one-step prediction*: It represents a predicted estimate of the state $\mathbf{x}_n$, given all the observations up to and including time $n - 1$.
2. The product term $\mathbf{G}_n\boldsymbol{\alpha}_n$ represents a *correction term*: The innovations process $\boldsymbol{\alpha}_n$, representing new information brought to the filtering process by the observation $\mathbf{y}_n$, is multiplied by a "gain factor" $\mathbf{G}_n$. For this reason, $\mathbf{G}_n$ is commonly referred to as the *Kalman gain*, in recognition of the pioneering work done by Kalman in his classic 1960 paper.

In light of these two insightful points, Eq. (14.30) is known as the *predictor-corrector formula* in Kalman filter theory.

Computation of the Kalman Gain

In Eq. (14.30), we now have our second equation for the recursive computation of the Kalman filter. However, for this equation to be of practical value, we need a formula for computing the Kalman gain that befits a recursive procedure for estimating the state.

With this objective in mind, we use Eq. (14.19) to write

$$\begin{aligned}\mathbb{E}[\mathbf{x}_n\boldsymbol{\alpha}_n^T] &= \mathbb{E}[\mathbf{x}_n(\mathbf{B}_n\boldsymbol{\varepsilon}_{n|n-1} + \boldsymbol{\nu}_n)^T]\\ &= \mathbb{E}[\mathbf{x}_n\boldsymbol{\varepsilon}_{n|n-1}^T]\mathbf{B}_n^T\end{aligned}$$

where, in the second line, we used the fact that the state $\mathbf{x}_n$ and measurement noise $\boldsymbol{\nu}_n$ are uncorrelated. Next, we note that the state-prediction error vector $\boldsymbol{\varepsilon}_{n|n-1}$ is orthogonal to the state estimate $\hat{\mathbf{x}}_{n|n-1}$ in accordance with the *principle of orthogonality*. Therefore, the expectation of the outer product of $\hat{\mathbf{x}}_{n|n-1}$ and $\boldsymbol{\varepsilon}_{n|n-1}$ is zero, so the expectation $\mathbb{E}[\mathbf{x}_n\boldsymbol{\alpha}_n^T]$ is unaffected if we replace $\mathbf{x}_n$ with $\boldsymbol{\varepsilon}_{n|n-1}$. We may thus write

$$\begin{aligned}\mathbb{E}[\mathbf{x}_n\boldsymbol{\alpha}_n^T] &= \mathbb{E}[\boldsymbol{\varepsilon}_{n|n-1}\boldsymbol{\varepsilon}_{n|n-1}^T]\mathbf{B}_n^T\\ &= \mathbf{P}_{n|n-1}\mathbf{B}_n^T\end{aligned}$$

Therefore, using this formula for the expectation $\mathbb{E}[\mathbf{x}_n\boldsymbol{\alpha}_n^T]$ in Eq. (14.29), we may express the Kalman gain $\mathbf{G}_n$ in terms of the prediction-error covariance matrix $\mathbf{P}_{n|n-1}$ as

$$\mathbf{G}_n = \mathbf{P}_{n|n-1}\mathbf{B}_n^T\mathbf{R}_n^{-1} \tag{14.31}$$

which is the third equation for the recursive computation of the Kalman filter.

Riccati Difference Equation for Updating the Prediction-Error Covariance Matrix

To complete the recursive procedure for computing the Kalman filter, we need a recursive formula to update the prediction-error covariance matrix from one iteration to the next.

To tackle this last step of the state-estimation procedure, we first replace n with $n + 1$ in Eq. (14.20):

$$\boldsymbol{\varepsilon}_{n+1|n} = \mathbf{x}_{n+1} - \hat{\mathbf{x}}_{n+1|n}$$

Next, we find it instructive to express the predicted estimate of the state in terms of its filtered estimate. To this end, replacing n with $n + 1$ in Eq. (14.28) and using Eq. (14.4), we write

$$\begin{aligned}\hat{\mathbf{x}}_{n+1|n} &= \sum_{k=1}^{n}\mathbb{E}[\mathbf{x}_{n+1}\boldsymbol{\alpha}_k^T]\mathbf{R}_k^{-1}\boldsymbol{\alpha}_k\\ &= \sum_{k=1}^{n}\mathbb{E}[(\mathbf{A}_{n+1,n}\mathbf{x}_n + \boldsymbol{\omega}_n)\boldsymbol{\alpha}_k^T]\mathbf{R}_k^{-1}\boldsymbol{\alpha}_k\\ &= \mathbf{A}_{n+1,n}\sum_{k=1}^{n}\mathbb{E}[\mathbf{x}_n\boldsymbol{\alpha}_n^T]\mathbf{R}_k^{-1}\boldsymbol{\alpha}_k\\ &= \mathbf{A}_{n+1,n}\hat{\mathbf{x}}_{n|n}\end{aligned} \tag{14.32}$$

In the third line of Eq. (14.32), we used the fact that the dynamic noise $\boldsymbol{\omega}_n$ is independent of the observations, and therefore the expectation $\mathbb{E}[\boldsymbol{\omega}_n\boldsymbol{\alpha}_k^T]$ is zero; and finally, we used the first line of the defining formula of Eq. (14.27) for the filtered estimate $\hat{\mathbf{x}}_{n|n}$. With the relationship of Eq. (14.32) between the predicted and filtered estimates of the state $\mathbf{x}_n$ at hand, we now use the formula for $\boldsymbol{\varepsilon}_{n+1|n}$ to write

$$\boldsymbol{\varepsilon}_{n+1|n} = \underbrace{(\mathbf{A}_{n+1,n}\mathbf{x}_n + \boldsymbol{\omega}_n)}_{\text{State } \mathbf{x}_{n+1}} - \underbrace{\mathbf{A}_{n+1,n}\hat{\mathbf{x}}_{n|n}}_{\substack{\text{Predicted estimate} \\ \hat{\mathbf{x}}_{n+1|n}}}$$

$$= \mathbf{A}_{n+1,n}(\mathbf{x}_n - \hat{\mathbf{x}}_{n|n}) + \boldsymbol{\omega}_n$$

$$= \mathbf{A}_{n+1,n}\boldsymbol{\varepsilon}_{n|n} + \boldsymbol{\omega}_n \tag{14.33}$$

where the *state-filtering-error vector* is defined by

$$\boldsymbol{\varepsilon}_{n|n} = \mathbf{x}_n - \hat{\mathbf{x}}_{n|n} \tag{14.34}$$

Hence, recognizing that the state-filtering-error vector $\boldsymbol{\varepsilon}_{n|n}$ and the dynamic noise $\boldsymbol{\omega}_n$ are uncorrelated, we may express the prediction-error covariance matrix as

$$\begin{aligned}\mathbf{P}_{n+1|n} &= \mathbb{E}[\boldsymbol{\varepsilon}_{n+1|n}\boldsymbol{\varepsilon}^T_{n+1|n}] \\ &= \mathbf{A}_{n+1,n}\mathbf{P}_{n|n}\mathbf{A}^T_{n+1,n} + \mathbf{Q}_{\omega,n}\end{aligned} \tag{14.35}$$

where $\mathbf{Q}_{\omega,n}$ is the covariance matrix of the dynamic noise $\boldsymbol{\omega}_n$. In Eq. (14.35), we have introduced our last parameter, namely, the *filtering-error covariance matrix*, which is defined by

$$\mathbf{P}_{n|n} = \mathbb{E}[\boldsymbol{\varepsilon}_{n|n}\boldsymbol{\varepsilon}^T_{n|n}] \tag{14.36}$$

To complete the recursion cycle in the Kalman filtering algorithm, we need a formula for computing the filtering-error covariance matrix $\mathbf{P}_{n|n}$. To this end, we first use Eq. (14.30) in Eq. (14.34), obtaining

$$\begin{aligned}\boldsymbol{\varepsilon}_{n|n} &= \mathbf{x}_n - \hat{\mathbf{x}}_{n|n-1} - \mathbf{G}_n\boldsymbol{\alpha}_n \\ &= \boldsymbol{\varepsilon}_{n|n-1} - \mathbf{G}_n\boldsymbol{\alpha}_n\end{aligned}$$

Hence, using the definition of Eq. (14.36), we obtain

$$\begin{aligned}\mathbf{P}_{n|n} &= \mathbb{E}[(\boldsymbol{\varepsilon}_{n|n-1} - \mathbf{G}_n\boldsymbol{\alpha}_n)(\boldsymbol{\varepsilon}_{n|n-1} - \mathbf{G}_n\boldsymbol{\alpha}_n)^T] \\ &= \mathbb{E}[\boldsymbol{\varepsilon}_{n|n-1}\boldsymbol{\varepsilon}^T_{n|n-1}] - \mathbf{G}_n\mathbb{E}[\boldsymbol{\alpha}_n\boldsymbol{\varepsilon}^T_{n|n-1}] - \mathbb{E}[\boldsymbol{\varepsilon}_{n|n-1}\boldsymbol{\alpha}^T_n]\mathbf{G}^T_n + \mathbf{G}_n\mathbb{E}[\boldsymbol{\alpha}_n\boldsymbol{\alpha}^T_n]\mathbf{G}^T_n \\ &= \mathbf{P}_{n|n-1} - \mathbf{G}_n\mathbb{E}[\boldsymbol{\alpha}_n\boldsymbol{\varepsilon}^T_{n|n-1}] - \mathbb{E}[\boldsymbol{\varepsilon}_{n|n-1}\boldsymbol{\alpha}^T_n]\mathbf{G}^T_n + \mathbf{G}_n\mathbf{R}_n\mathbf{G}^T_n\end{aligned} \tag{14.37}$$

Next, we note that since the estimate $\hat{\mathbf{x}}_{n|n-1}$ is orthogonal to the innovations process $\boldsymbol{\alpha}_n$, we have

$$\begin{aligned}\mathbb{E}[\boldsymbol{\varepsilon}_{n|n-1}\boldsymbol{\alpha}^T_n] &= \mathbb{E}[(\mathbf{x}_n - \hat{\mathbf{x}}_{n|n-1})\boldsymbol{\alpha}^T_n] \\ &= \mathbb{E}[\mathbf{x}_n\boldsymbol{\alpha}^T_n]\end{aligned}$$

Similarly,

$$\mathbb{E}[\boldsymbol{\alpha}_n\boldsymbol{\varepsilon}^T_{n|n-1}] = \mathbb{E}[\boldsymbol{\alpha}_n\mathbf{x}^T_n]$$

By using this pair of relationships and the defining formula of Eq. (14.29) for the Kalman gain, it is a straightforward matter to show that

$$\mathbf{G}_n\mathbb{E}[\boldsymbol{\alpha}_n\boldsymbol{\varepsilon}^T_{n|n-1}] = \mathbb{E}[\boldsymbol{\varepsilon}_{n|n-1}\boldsymbol{\alpha}^T_n]\mathbf{G}^T_n = \mathbf{G}_n\mathbf{R}_n\mathbf{G}^T_n$$

Accordingly, Eq. (14.37) is reduced to

$$\mathbf{P}_{n|n} = \mathbf{P}_{n|n-1} - \mathbf{G}_n\mathbf{R}_n\mathbf{G}_n^T$$

Finally, using the formula of Eq. (14.31) for the Kalman gain and invoking the symmetric properties of the covariance matrices $\mathbf{R}_n$ and $\mathbf{P}_{n|n-1}$, we write

$$\mathbf{P}_{n|n} = \mathbf{P}_{n|n-1} - \mathbf{G}_n\mathbf{B}_n\mathbf{P}_{n|n-1} \tag{14.38}$$

Thus, the pair of equations in Eqs. (14.38) and (14.35) provides the means of updating the prediction-error covariance matrix. In particular, Eq. (14.38) is commonly referred to as the discrete form of the *Riccati equation*, which is well known in control theory.

Together with Eq. (14.32), this pair of equations completes the formulation of the Kalman filtering algorithum.

Summary of the Kalman Filter

Table 14.1 presents a summary of the variables and parameters used to formulate the solution of the Kalman filtering problem. The input of the filter is the sequence of observables $\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n$, and the output of the filter is the filtered estimate $\hat{\mathbf{x}}_{n|n}$. The computational procedure is recursive, as summarized in Table 14.2. The summary also includes the *initial conditions* needed to start the recursive computation. Note that the formula for the innovation $\boldsymbol{\alpha}_n$ in Table 14.2 follows from Eqs. (14.9) and (14.18).

The version of the Kalman filter summarized in Table 14.2 is commonly referred to as the *covariance (Kalman) filtering algorithm*.[2] This terminology follows from the fact that the *algorithm propagates the covariance matrix* $\mathbf{P}_{n|n-1}$ across one complete cycle of the recursive computation, where $\mathbf{P}_{n|n-1}$ refers to the prediction.

TABLE 14.1 Summary of the Kalman Variables and Parameters

Variable	Definition	Dimension
$\mathbf{x}_n$	State at time n	M by 1
$\mathbf{y}_n$	Observation at time n	L by 1
$\mathbf{A}_{n+1,n}$	Invertible transition matrix from state at time n to state at time $n+1$	M by M
$\mathbf{B}_n$	Measurement matrix at time n	L by M
$\mathbf{Q}_{\omega,n}$	Covariance matrix of dynamic noise $\boldsymbol{\omega}_n$	M by M
$\mathbf{Q}_{\nu,n}$	Covariance matrix of measurement noise $\boldsymbol{\nu}_n$	L by L
$\hat{\mathbf{x}}_{n\|n-1}$	Predicted estimate of the state at time n, given the observations $\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_{n-1}$	M by 1
$\hat{\mathbf{x}}_{n\|n}$	Filtered estimate of the state at time n, given the observations $\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n$	M by 1
$\mathbf{G}_n$	Kalman gain at time n	M by L
$\boldsymbol{\alpha}_n$	Innovations process at time n	L by 1
$\mathbf{R}_n$	Covariance matrix of the innovations process $\boldsymbol{\alpha}_n$	L by L
$\mathbf{P}_{n\|n-1}$	Prediction-error covariance matrix	M by M
$\mathbf{P}_{n\|n}$	Filtering-error covariance matrix	M by M

TABLE 14.2 Summary of the Kalman Filter Based on Filtered Estimate of the State

Observations = $\{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n\}$

Known parameters

Transition matrix = $\mathbf{A}_{n+1,n}$
Measurement matrix = $\mathbf{B}_n$
Covariance matrix of dynamic noise = $\mathbf{Q}_{\omega,n}$
Covariance matrix of measurement noise = $\mathbf{Q}_{\nu,n}$

Computation: $n = 1, 2, 3, \ldots$

$$\mathbf{G}_n = \mathbf{P}_{n|n-1}\mathbf{B}_n^T[\mathbf{B}_n\mathbf{P}_{n|n-1}\mathbf{B}_n^T + \mathbf{Q}_{\nu,n}]^{-1}$$
$$\boldsymbol{\alpha}_n = \mathbf{y}_n - \mathbf{B}_n\hat{\mathbf{x}}_{n|n-1}$$
$$\hat{\mathbf{x}}_{n|n} = \hat{\mathbf{x}}_{n|n-1} + \mathbf{G}_n\boldsymbol{\alpha}_n$$
$$\hat{\mathbf{x}}_{n+1|n} = \mathbf{A}_{n+1,n}\hat{\mathbf{x}}_{n|n}$$
$$\mathbf{P}_{n|n} = \mathbf{P}_{n|n-1} - \mathbf{G}_n\mathbf{B}_n\mathbf{P}_{n|n-1}$$
$$\mathbf{P}_{n+1|n} = \mathbf{A}_{n+1,n}\mathbf{P}_{n|n}\mathbf{A}_{n+1,n}^T + \mathbf{Q}_{\omega,n}$$

Initial conditions

$$\hat{\mathbf{x}}_{1|0} = \mathbb{E}[\mathbf{x}_1]$$
$$\mathbf{P}_{1,0} = \mathbb{E}[(\mathbf{x}_1 - \mathbb{E}[\mathbf{x}_1])(\mathbf{x}_1 - \mathbb{E}[\mathbf{x}_1])^T] = \boldsymbol{\Pi}_0$$

The matrix $\boldsymbol{\Pi}_0$ is a diagonal matrix with diagonal elements all set equal to δ^{-1}, where δ is a small number.

Figure 14.3 depicts a signal-flow graph of the Kalman filter, where $z^{-1}\mathbf{I}$ represents a bank of unit-time delays. This figure clearly shows that the Kalman filter is a *double-loop feedback system.* One feedback loop, embodying the transition matrix $\mathbf{A}_{n,n-1}$ of the system (state) model, acts as the *predictor*. The second feedback loop, embodying the matrix $\mathbf{B}_n$ of the measurement model, acts as the *corrector*. These two feedback loops work together to generate the filtered estimate of the state $\mathbf{x}_n$—namely, $\hat{\mathbf{x}}_{n|n}$—in response to the observation $\mathbf{y}_n$. It follows, therefore, that the Kalman filter, as depicted in Figure 14.3, is indeed a causal system in that it is capable of operating in real time. In fact, we also have an *overall* feedback loop that encompasses these two feedback loops.

The Kalman gain $\mathbf{G}_n$, central to the operation of the Kalman filter, varies with time n. Thus, we say that the Kalman filter is a *time-varying filter.* This property holds even if the state-space model of the original dynamic system is time invariant.

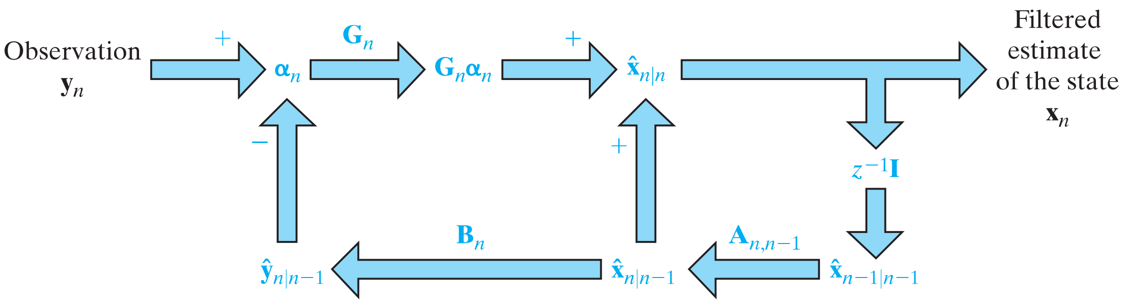

FIGURE 14.3 Signal-flow graph of the Kalman filter, depicting it as a double-loop feedback system.

14.4 THE DIVERGENCE PHENOMENON AND SQUARE-ROOT FILTERING

The covariance filtering algorithm summarized in Table 14.2 is prone to serious numerical difficulties that are well documented in the literature (Kaminski et al., 1971; Bierman and Thornton, 1977).

In practice, numerical difficulties can arise in two basic ways. One way is through *numerical imprecision.* To be specific, the matrix $\mathbf{P}_{n|n}$ is computed as the difference between two nonnegative-definite matrices, as shown in Eq. (14.38). Hence, unless the numerical accuracy employed at every iteration of the algorithm is high enough, there is a possibility that the matrix resulting from this computation will violate the properties of symmetry and nonnegative definiteness. But, according to Eq. (14.36), $\mathbf{P}_{n|n}$ is a covariance matrix and must therefore be nonnegative definite. We thus have a conflicting situation between theory and practice, with the result that the presence of numerical inaccuracies in the computation leads to "unstable" behavior of the Kalman filter. This undesirable behavior of the Kalman filter is commonly referred to as the *divergence phenomenon.*

The divergence phenomenon may also arise in practice in another way. The derivation of the Kalman filter is based on the linear, Gaussian state-space model, described in Eqs. (14.4) and (14.5). Serious deviations of this model from the underlying physics of the dynamic system under study may also contribute to unstable behavior of the algorithm. After all, the algorithm is driven by a real-life sequence of observables, whereas mathematical derivation of the algorithm is based on a hypothesized state-space model. Here, again, we have another conflicting situation between theory and practice, which, in its own way, could lead to divergence of the algorithm.

Given these practical relations, we may now pose the following question:

> *How do we overcome the divergence phenormenon so as to assure stable operation of the Kalman filter in practice?*

A practical answer to this important question is discussed next.

Square-Root Filtering

A mathematically elegant and computationally plausible method of resolving the divergence problem is to use *square-root filtering*. Basically, in this modification of the Kalman filter, we use numerically stable orthogonal transformations at every iteration of the algorithm. Specifically, the matrix $\mathbf{P}_{n|n}$ is propagated in its square-root form by applying the *Cholesky factorization*, according to which we may write

$$\mathbf{P}_{n|n} = \mathbf{P}_{n|n}^{1/2}\mathbf{P}_{n|n}^{T/2} \tag{14.39}$$

where the term $\mathbf{P}_{n|n}^{1/2}$ is reserved for a *lower triangular matrix* and $\mathbf{P}_{n|n}^{T/2}$ is the transposed term. In linear algebra, the *Cholesky factor* $\mathbf{P}_{n|n}^{1/2}$ is commonly referred to as the square root of the matrix $\mathbf{P}_{n,n}$. The very important point to note here is that the matrix product $\mathbf{P}_{n|n}^{1/2}\mathbf{P}_{n|n}^{T/2}$ is *not* likely to become indefinite, because the product of any square matrix and its transpose is always nonnegative definite. Indeed, even in the presence of numerical errors, the matrix conditioning of the Cholesky factor $\mathbf{P}_{n|n}^{1/2}$ is generally better than that of $\mathbf{P}_{n|n}$ itself.

Square-Root Implementation of the Kalman Filter

A lemma in matrix algebra, called the matrix factorization lemma, is pivotal to the derivation of square-root filtering algorithms. Consider any two L-by-M matrices $\mathbf{X}$ and $\mathbf{Y}$ with the dimension $L \leq M$. The *matrix factorization lemma* states the following (Stewart, 1973; Golub and Van Loan, 1996):

> *The matrix equality $\mathbf{X}\mathbf{X}^T = \mathbf{Y}\mathbf{Y}^T$ holds if, and only if, there exists an orthogonal matrix $\boldsymbol{\Theta}$ such that*

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\Theta} \tag{14.40}$$

To prove this lemma, we express the matrix product $\mathbf{Y}\mathbf{Y}^T$ as

$$\begin{aligned}\mathbf{Y}\mathbf{Y}^T &= \mathbf{X}\boldsymbol{\Theta}(\mathbf{X}\boldsymbol{\Theta})^T \\ &= \mathbf{X}\boldsymbol{\Theta}\boldsymbol{\Theta}^T\mathbf{X}^T \\ &= \mathbf{X}\mathbf{X}^T\end{aligned}$$

In the last line of this equation, we invoked the defining property of the orthogonal matrix $\boldsymbol{\Theta}$:

> *The product of an orthogonal matrix with its transpose is equal to the identity matrix.*

As a corollary to this property, we may equivalently write

$$\boldsymbol{\Theta}^{-1} = \boldsymbol{\Theta}^T \tag{14.41}$$

That is, the inverse of an orthogonal matrix is equal to its own transpose.

With the matrix factorization lemma at our disposal, we may now proceed with the derivation of the square-root covariance implementation of the Kalman filter. To begin, we first use Eq. (14.31), defining the gain matrix $\mathbf{G}(n)$, in Eq. (14.38), obtaining

$$\mathbf{P}_{n|n} = \mathbf{P}_{n|n-1} - \mathbf{P}_{n|n-1}\mathbf{B}_n^T\mathbf{R}_n^{-1}\mathbf{B}_n\mathbf{P}_{n|n-1} \tag{14.42}$$

where the matrix $\mathbf{R}_n$ is itself defined by Eq. (14.22), reproduced here for convenience of presentation:

$$\mathbf{R}_n = \mathbf{B}_n\mathbf{P}_{n|n-1}\mathbf{B}_n^T + \mathbf{Q}_{\nu,n}$$

Examining the reformulated Riccati equation of Eq. (14.42), we find that the expression on its right-hand side consists of three distinct matrix terms:

M-by-M matrix: covariance matrix of the predicted state $\mathbf{P}_{n|n-1}$;
L-by-M matrix: measurement matrix $\mathbf{B}_n$ multiplied by $\mathbf{P}_{n|n-1}$;
L-by-L matrix: covariance matrix $\mathbf{R}_n$ of the innovations process.

Keeping in mind the different dimensionalities of these three matrix terms, we may order all three of them in a compatible way in the N-by-N block matrix

$$\begin{aligned}\mathbf{H}_n &= \left[\begin{array}{c|c}\mathbf{R}_n & \mathbf{B}_n\mathbf{P}_{n|n-1} \\ \hline \mathbf{P}_{n|n-1}\mathbf{B}_n^T & \mathbf{P}_{n|n-1}\end{array}\right] \\ &= \left[\begin{array}{c|c}\mathbf{Q}_{\nu,n} + \mathbf{B}_n\mathbf{P}_{n|n-1}\mathbf{B}_n^T & \mathbf{B}_n\mathbf{P}_{n|n-1} \\ \hline \mathbf{P}_{n|n-1}\mathbf{B}_n^T & \mathbf{P}_{n|n-1}\end{array}\right]\end{aligned} \tag{14.43}$$

where, in the second line, we inserted the formula for $\mathbf{R}_n$. The size of the matrix in Eq. (14.43), denoted by N, equals $L + M$. The new block matrix $\mathbf{H}_n$ is nonnegative-definite by definition. We may therefore apply the Cholesky factorization to it, obtaining

$$\mathbf{H}_n = \left[\begin{array}{c|c} \mathbf{Q}^{1/2}_{v,n} & \mathbf{B}_n\mathbf{P}^{1/2}_{n|n-1} \\ \hline \mathbf{O} & \mathbf{P}^{1/2}_{n|n-1} \end{array}\right]\left[\begin{array}{c|c} \mathbf{Q}^{1/2}_{v,n} & \mathbf{O}^T \\ \hline \mathbf{P}^{1/2}_{n|n-1}\mathbf{B}^T_n & \mathbf{P}^{1/2}_{n|n-1} \end{array}\right] \tag{14.44}$$

where $\mathbf{P}^{1/2}_{n|n-1}$ is the square root of the covariance matrix $\mathbf{P}_{n|n-1}$ and $\mathbf{O}$ is a null matrix.

The matrix product on the right-hand side of Eq. (14.44) may be interpreted as the product of matrix $\mathbf{X}_n$, introduced earlier, and its transpose $\mathbf{X}^T_n$. The stage is therefore set for invoking the matrix factorization lemma, according to which the use of Eq. (14.40) yields

$$\underbrace{\left[\begin{array}{c|c} \mathbf{Q}^{1/2}_{v,n} & \mathbf{B}_n\mathbf{P}^{1/2}_{n|n-1} \\ \hline \mathbf{O} & \mathbf{P}^{1/2}_{n|n-1} \end{array}\right]}_{\mathbf{X}_n}\boldsymbol{\Theta}_n = \underbrace{\left[\begin{array}{c|c} \mathbf{Y}_{11,n} & \mathbf{O}^T \\ \hline \mathbf{Y}_{21,n} & \mathbf{Y}_{22,n} \end{array}\right]}_{\mathbf{Y}_n} \tag{14.45}$$

where the matrix $\boldsymbol{\Theta}_n$ is an orthogonal matrix. To be more specific, $\boldsymbol{\Theta}_n$ is an orthogonal matrix that operates on $\mathbf{X}_n$ in such a way that the resulting matrix $\mathbf{Y}_n$ is a *lower triangular matrix*; that is, all the elements of $\mathbf{Y}_n$ above its main diagonal are zero. It is because of this action that the matrix $\boldsymbol{\Theta}_n$ is also referred to as an *orthogonal rotation*. Invoking the orthogonality property of $\boldsymbol{\Theta}_n$, we may expand on Eq. (14.45) by writing

$$\underbrace{\left[\begin{array}{c|c} \mathbf{Q}^{1/2}_{v,n} & \mathbf{B}_n\mathbf{P}^{1/2}_{n|n-1} \\ \hline \mathbf{O} & \mathbf{P}^{1/2}_{n|n-1} \end{array}\right]}_{\mathbf{X}_n}\underbrace{\left[\begin{array}{c|c} \mathbf{Q}^{1/2}_{v,n} & \mathbf{O}^T \\ \hline \mathbf{P}^{1/2}_{n|n-1}\mathbf{B}^T_n & \mathbf{P}^{T/2}_{n|n-1} \end{array}\right]}_{\mathbf{X}^T_n} = \underbrace{\left[\begin{array}{c|c} \mathbf{Y}_{11,n} & \mathbf{O}^T \\ \hline \mathbf{Y}_{21,n} & \mathbf{Y}_{22,n} \end{array}\right]}_{\mathbf{Y}_n}\underbrace{\left[\begin{array}{c|c} \mathbf{Y}^T_{11,n} & \mathbf{Y}^T_{21,n} \\ \hline \mathbf{O}^T & \mathbf{Y}^T_{22,n} \end{array}\right]}_{\mathbf{Y}^T_n} \tag{14.46}$$

Expanding the matrix products $\mathbf{X}_n\mathbf{X}^T_n$ and $\mathbf{Y}_n\mathbf{Y}^T_n$ and then equating corresponding terms in the two sides of Eq. (14.46), we get three identifies:

$$\mathbf{Q}_{v,n} + \mathbf{B}_n\mathbf{P}_{n|n-1}\mathbf{B}^T_n = \mathbf{Y}_{11,n}\mathbf{Y}^T_{11,n} \tag{14.47}$$

$$\mathbf{B}_n\mathbf{P}_{n|n-1} = \mathbf{Y}_{11,n}\mathbf{Y}^T_{21,n} \tag{14.48}$$

$$\mathbf{P}_{n|n-1} = \mathbf{Y}_{21,n}\mathbf{Y}^T_{21,n} + \mathbf{Y}_{22,n}\mathbf{Y}^T_{22,n} \tag{14.49}$$

The left-hand side of Eq. (14.47) is recognized as the covariance matrix $\mathbf{R}_n$, which is factorizable into $\mathbf{R}^{1/2}_n\mathbf{R}^{T/2}_n$. The identity in Eq. (14.47) is therefore satisfied by setting the first unknown as follows:

$$\mathbf{Y}_{11,n} = \mathbf{R}^{1/2}_n \tag{14.50}$$

Next, substituting this value of $\mathbf{Y}_{11,n}$ into the identify in Eq. (14.48) and solving for $\mathbf{Y}_{21,n}$, we find the second unknown:

$$\mathbf{Y}_{21,n} = \mathbf{P}_{n|n-1}\mathbf{B}^T_n\mathbf{R}^{-T/2}_n \tag{14.51}$$

In light of the definition of the Kalman gain $\mathbf{G}_n$, developed previously in Eq. (14.31), we may also express $\mathbf{Y}_{21,n}$ as

$$\mathbf{Y}_{21,n} = \mathbf{G}_n\mathbf{R}^{1/2}_n \tag{14.52}$$

Moreover, substituting the value of $\mathbf{Y}_{21,n}$ given in Eq. (14.51) into Eq. (14.49), solving for the matrix product $\mathbf{Y}_{22,n}\mathbf{Y}_{22,n}^T$, and then using Eq. (14.42), we get

$$\begin{aligned}\mathbf{Y}_{22,n}\mathbf{Y}_{22,n}^T &= \mathbf{P}_{n|n-1} - \mathbf{P}_{n|n-1}\mathbf{B}_n^T\mathbf{R}_n^{-1}\mathbf{B}_n\mathbf{P}_{n|n-1}\\ &= \mathbf{P}_{n|n}\end{aligned}$$

Factorizing the covariance matrix $\mathbf{P}_{n,n}$ into $\mathbf{P}_{n|n}^{1/2}\mathbf{P}_{n|n}^{T/2}$, we find the third unknown:

$$\mathbf{Y}_{22,n} = \mathbf{P}_{n|n}^{1/2} \tag{14.53}$$

With all three nonzero submatrices of $\mathbf{Y}_n$ determined, we may now fill in the unknowns in Eq. (14.45), obtaining

$$\left[\begin{array}{c:c}\mathbf{Q}_{\nu,n}^{1/2} & \mathbf{B}_n\mathbf{P}_{n,n-1}^{1/2}\\ \hdashline \mathbf{O} & \mathbf{P}_{n|n-1}^{1/2}\end{array}\right]\mathbf{\Theta}_n = \left[\begin{array}{c:c}\mathbf{R}_n^{1/2} & \mathbf{O}^T\\ \hdashline \mathbf{G}_n\mathbf{R}_n^{1/2} & \mathbf{P}_{n|n}^{1/2}\end{array}\right] \tag{14.54}$$

In the final solution derived in Eq. (14.54), we may now distinguish between two well-defined arrays of numbers that deserve close scrutiny:

1. *Prearray.* This array of numbers, on the left-hand side of Eq. (14.54), is operated on by the orthogonal rotation $\mathbf{\Theta}_n$, which is designed to *annihilate* the submatrix $\mathbf{B}_n\mathbf{P}_{n|n-1}^{1/2}$, element by element. The measurement matrix $\mathbf{B}_n$ and the covariance matrix of the measurement noise, $\mathbf{Q}_{\nu,n}$, are both given parameters. The square root $\mathbf{P}_{n|n-1}^{1/2}$, being an *old value that is being updated*, is also known. Therefore, the submatrices constituting the prearray are all known at time n.
2. *Postarray.* This second array of numbers, on the right-hand side of Eq. (14.54), is a lower triangular matrix that results from the annihilation performed by the orthogonal rotation on the prearray. In particular, the inclusion of the square root $\mathbf{Q}_{\nu,n}^{1/2}$ in the prearray induces the generation of two useful matrices:
   - the matrix $\mathbf{R}_n^{1/2}$, representing the square root of the covariance matrix of the innovations process $\boldsymbol{\alpha}_n$;
   - the matrix product $\mathbf{G}_n\mathbf{R}_n^{1/2}$, which makes it possible to compute the Kalman gain.

   One other important matrix resulting from computing the postarray is the square root of the filtering-error covariance matrix, $\mathbf{P}_{n|n}^{1/2}$.

With all of this information extracted from the postarray, we are ready to summarize the computations involved in the square-root covariance filtering algorithm, as listed in Table 14.3. A complete *recursion cycle* of the algorithm consists of the transformation of the prearray into the postarray and the computation of updated parameters, which are respectively listed under items 3 and 4 of the table. From this table, it is apparent that the algorithm does indeed propagate the square root of the prediction-error covariance matrix—namely, $\mathbf{P}_{n|n-1}^{1/2}$.

Givens Rotations

Thus far in formulating the square-root covariance filtering algorithm, we have not paid attention to the way in which the orthogonal matrix $\mathbf{\Theta}$ is to be specified, other than to require

TABLE 14.3 Summary of Computations in the Square-Root Filtering Algorithm

1. *Given parameters:*
   Transition matrix: $\mathbf{A}_{n+1,n}$
   Measurement matrix: $\mathbf{B}_n$
   Covariance matrix of measurement noise: $\mathbf{Q}_{\nu,n}$
   Covariance matrix of dynamic noise: $\mathbf{Q}_{\omega,n}$
2. *Old values of parameters to be updated:*
   Predicted estimate of the state: $\hat{\mathbf{x}}_{n|n-1}$
   Square root of the prediction-error covariance matrix: $\mathbf{P}^{1/2}_{n|n-1}$
3. *Orthogonal rotation of the prearray into the postarray:*

$$\left[\begin{array}{c:c} \mathbf{Q}^{1/2}_{\nu,n} & \mathbf{B}_n\mathbf{P}^{1/2}_{n,n-1} \\ \hdashline \mathbf{O} & \mathbf{P}^{1/2}_{n|n-1} \end{array}\right]\boldsymbol{\Theta}_n = \left[\begin{array}{c:c} \mathbf{R}^{1/2}_n & \mathbf{O}^T \\ \hdashline \mathbf{G}_n\mathbf{R}^{1/2}_n & \mathbf{P}^{1/2}_{n|n} \end{array}\right]$$

4. *Updated parameters:*

$$\begin{aligned} \mathbf{G}_n &= [\mathbf{G}_n\mathbf{R}^{1/2}_n][\mathbf{R}^{1/2}_n]^{-1} \\ \boldsymbol{\alpha}_n &= \mathbf{y}_n - \mathbf{B}_n\hat{\mathbf{x}}_{n|n-1} \\ \hat{\mathbf{x}}_{n|n} &= \hat{\mathbf{x}}_{n|n-1} + \mathbf{G}_n\boldsymbol{\alpha}_n \\ \hat{\mathbf{x}}_{n+1|n} &= \mathbf{A}_{n+1|n}\,\hat{\mathbf{x}}_{n|n} \\ \mathbf{P}_{n|n} &= \mathbf{P}^{1/2}_{n|n}[\mathbf{P}^{1/2}_{n|n}]^T \\ \mathbf{P}_{n+1|n} &= [\mathbf{A}_{n+1|n}\mathbf{P}^{1/2}_{n|n} \,\vdots\, \mathbf{Q}^{1/2}_{\omega,n}]\left[\begin{array}{c} \mathbf{P}^{T/2}_{n|n}\mathbf{A}^T_{n+1|n} \\ \hdashline \mathbf{Q}^{T/2}_{\boldsymbol{\omega},n} \end{array}\right] \end{aligned}$$

Notes:

1. Under point 4, all the matrices inside the brackets are extracted from the postarray and known parameters.
2. In writing the updated parameters, we have made use of the corresponding computational formulas of Table 14.2.

that the prearray should be transformed into a lower triangular postarray through a process of annihilations. An elegant way of performing this process is to use *Givens rotations*, the application of which proceeds in a step-by-step manner (Golub and Van Loan, 1996).

Under this procedure, the orthogonal matrix $\boldsymbol{\Theta}$ is expressed as a product of N orthogonal rotation components, as shown by

$$\boldsymbol{\Theta} = \prod_{k=1}^{N}\boldsymbol{\Theta}_k$$

where we have ignored reference to discrete time n to simplify the presentation. The characteristics of each rotation component are as follows:

1. Except for four *strategic elements*, the diagonal elements of $\boldsymbol{\Theta}_k$ are all unity, and the off-diagonal elements are all zero.
2. The subscript k in $\boldsymbol{\Theta}_k$ refers to a *pivotal point*, around which the four strategic elements of $\boldsymbol{\Theta}_k$ are located. As a rule, the pivotal point is always located on the main diagonal of the prearray.
3. Two of the strategic elements of $\boldsymbol{\Theta}_k$ are *cosine parameters*, and the remaining two are *sine parameters*. To add mathematical significance to these cosine and sine

parameters, suppose that the requirement is to annihilate the kl-th element of the prearray, where k refers to row and l refers to column. Then, the corresponding cosine (diagonal) parameters θ_{kk} and θ_{ll} are assigned the same value, but one of the sine (off-diagonal) parameters is assigned a negative value, as shown by the two-by-two matrix

$$\begin{bmatrix} \theta_{kk} & \theta_{kl} \\ \theta_{lk} & \theta_{ll} \end{bmatrix} = \begin{bmatrix} c_k & -s_k \\ s_k & c_k \end{bmatrix} \tag{14.55}$$

All four parameters are real numbers, which is a requirement for satisfying the constraint

$$c_k^2 + s_k^2 = 1 \qquad \text{for all } k \tag{14.56}$$

The following example illustrates the steps involved in the transformation of a prearray into a lower triangular postarray.

EXAMPLE 1. Givens rotations for 3-by-3 prearray

Consider the 3-by-3 prearray $\mathbf{X}$, which is to be transformed into a lower triangular 3-by-3 postarray $\mathbf{Y}$. The transformation will proceed in three steps.

Step 1: For this first step, we write

$$\underbrace{\left[\begin{array}{c:cc} x_{11} & x_{12} & x_{13} \\ \hdashline 0 & x_{22} & x_{23} \\ 0 & x_{32} & x_{33} \end{array}\right]}_{\text{Prearray of step 1}} \underbrace{\left[\begin{array}{c:cc} c_1 & -s_1 & 0 \\ \hdashline s_1 & c_1 & 0 \\ 0 & 0 & 1 \end{array}\right]}_{\substack{\text{1st Givens} \\ \text{rotation}}} = \underbrace{\begin{bmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{bmatrix}}_{\text{Postarray of step 1}} \tag{14.57}$$

where the two zeros in the prearray follow from Eq. (14.54), and

$$u_{12} = -x_{11}s_1 + x_{12}c_1$$

The requirement is to set $u_{12} = 0$, for which the following condition must hold:

$$s_1 = \frac{x_{12}}{x_{11}} c_1$$

Hence, by setting $c_1^2 + s_1^2 = 1$ and solving for c_1 and s_1, we define the first orthogonal rotation used in Eq. (14.57) as

$$\begin{aligned} c_1 &= \frac{x_{11}}{\sqrt{x_{11}^2 + x_{12}^2}} \\ s_1 &= \frac{x_{12}}{\sqrt{x_{11}^2 + x_{12}^2}} \end{aligned} \tag{14.58}$$

Step 2: For this second step, we write

$$\underbrace{\begin{bmatrix} u_{11} & 0 & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{bmatrix}}_{\text{Prearray of step 2}} \underbrace{\begin{bmatrix} c_2 & 0 & -s_2 \\ 0 & 1 & 0 \\ s_2 & 0 & c_2 \end{bmatrix}}_{\substack{\text{2nd Givens} \\ \text{rotation}}} = \underbrace{\begin{bmatrix} v_{11} & 0 & v_{13} \\ v_{21} & v_{22} & v_{23} \\ v_{31} & v_{32} & v_{33} \end{bmatrix}}_{\text{Postarray of step 2}} \tag{14.59}$$