Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software
Yaakov Bar-Shalom, X. Rong Li, Thiagalingam Kirubarajan

ISBNs: 0-471-41655-X (Hardback) 0-471-22127-9 (Electronic)

Chapter 7

COMPUTATIONAL ASPECTS OF ESTIMATION

7.1 INTRODUCTION

7.1.1 Implementation of Linear Estimation

This chapter describes briefly some numerical techniques for the efficient ***implementation of the linear estimation techniques*** presented earlier.

The techniques to be discussed are for discrete-time linear systems with state equation

$$x(k+1) = F(k)x(k) + \Gamma(k)v(k) \tag{7.1.1-1}$$

and measurement equation

$$z(k) = H(k)x(k) + w(k) \tag{7.1.1-2}$$

where $v(k)$ and $w(k)$ are the process and measurement noises, assumed to be zero mean, white, mutually uncorrelated, and with covariances $Q(k)$ and $R(k)$, respectively.

The two properties of a covariance matrix

- symmetry and
- positive definiteness

can be lost due to ***round-off errors*** in the course of calculating its propagation equations — the covariance prediction and the covariance update.

The covariance propagation equations are discussed next, and their propensity for causing ***loss of symmetry*** and/or ***loss of positive definiteness*** is examined.

The ***covariance prediction equation***

$$P(k+1|k) = F(k)P(k|k)F(k)' + \Gamma(k)Q(k)\Gamma(k)' \tag{7.1.1-3}$$

can affect only the symmetry of the resulting matrix. A suitable implementation of the products of three matrices will avoid this problem.

More significant numerical problems arise in the ***covariance update equation***, which can be written in the following algebraically equivalent forms:

$$P(k+1|k+1) = [I - W(k+1)H(k+1)]P(k+1|k) \qquad (7.1.1\text{-}4)$$

$$P(k+1|k+1) = P(k+1|k) - W(k+1)S(k+1)W(k+1)' \qquad (7.1.1\text{-}5)$$

$$\begin{aligned} P(k+1|k+1) &= [I - W(k+1)H(k+1)]P(k+1|k) \\ &\cdot[I - W(k+1)H(k+1)]' + W(k+1)R(k+1)W(k+1)' \end{aligned} \qquad (7.1.1\text{-}6)$$

Equation (7.1.1-4) is very sensitive to round-off errors and is bound to lead to loss of symmetry as well as positive definiteness — it is best to avoid it.

Equation (7.1.1-5) will avoid loss of symmetry with a suitable implementation of the last term, which is a product of three matrices (as in the covariance prediction), but it can still lead to loss of positive definiteness due to numerical errors in the subtraction.

Equation (7.1.1-6), the ***Joseph form covariance update***, while computationally more expensive than (7.1.1-5), is less sensitive to round-off errors. With the proper implementation of the products of three matrices, it will preserve symmetry. Furthermore, since the only place it has a subtraction is in the term $I - WH$, which appears "squared," this form of the covariance update has the property of preserving the positive definiteness of the resulting updated covariance.

7.1.2 Outline

The purposes of the techniques to be discussed in this chapter are:

1. Reduction of the computational requirements.
2. Improvement of the numerical accuracy — preservation of the symmetry and the positive definiteness of the state covariance matrix, as well as reduction of its ***condition number***, defined as the logarithm of the ratio of its largest to its smallest eigenvalue in (1.3.6-13).

Section 7.2 presents the ***information filter***, which carries out the recursive computation of the inverse of the covariance matrix. This is an alternative to (7.1.1-5) and is less demanding computationally for systems with dimension of the measurement vector larger than that of the state. This technique also has the advantage that it allows the *start-up of the estimation without an initial estimate.* Specifically, this amounts to having a state estimator with a ***noninformative (diffuse) prior***, similar to the one discussed in Section 2.3.4 for the parameter estimation case.

A technique which carries out the state update with one measurement component at a time — the ***sequential updating*** technique — rather than the entire measurement vector at once, is described in Section 7.3. This implementation of the Kalman filter is less demanding computationally than the standard one.

The technique of ***square-root filtering***, which consists of *sequential updating* and ***factorization of the covariance matrix***, is described in Section 7.4. This approach avoids the numerical problems that can lead, due to round-off errors, to the loss of symmetry and positive definiteness of the state covariance matrix. It is also less expensive computationally than the best standard implementation (7.1.1-6) while it has double the accuracy.

7.1.3 Computational Aspects — Summary of Objectives

Present techniques for implementation of the Kalman filter that are

- more economical computationally and/or
- more robust numerically.

Typical numerical problems with the state covariance matrix:

- Loss of symmetry.
- Loss of positive definiteness.

The algorithms to be discussed:

- Information filter — calculates recursively the inverse covariance matrix and it also allows to start the state estimator without an initial estimate, i.e., with a noniformative prior.
- Sequential processing of measurements in updating the state estimate.
- Square-root filter: a combination of sequential updating with covariance factorization that achieves double precision.

7.2 THE INFORMATION FILTER

7.2.1 Recursions for the Information Matrices

The standard version of the Kalman filter calculates the gain in conjunction with a recursive computation of the state covariance. Starting from $P(k-1|k-1)$, the prediction covariance is obtained as

$$P(k|k-1) = F(k-1)P(k-1|k-1)F(k-1)' + \Gamma(k-1)Q(k-1)\Gamma(k-1)' \tag{7.2.1-1}$$

The gain is

$$W(k) = P(k|k-1)H(k)'[H(k)P(k|k-1)H(k)' + R(k)]^{-1} \tag{7.2.1-2}$$

and the updated state covariance $P(k|k)$ is calculated from

$$\begin{aligned} P(k|k) &= P(k|k-1) - P(k|k-1)H(k)'[H(k)P(k|k-1)H(k)' \\ &\quad + R(k)]^{-1}H(k)P(k|k-1) \end{aligned} \tag{7.2.1-3}$$

which completes one cycle of computations.

The ***information filter*** calculates recursively the *inverses of the covariance matrices*, both for the prediction and the update. The term "information" is used in the sense of the Cramer-Rao lower bound, where the ***(Fisher) information matrix*** is the *inverse of the covariance matrix*.

A recursion from $P(k-1|k-1)^{-1}$ to $P(k|k-1)^{-1}$ to $P(k|k)^{-1}$ and the expression of the filter gain in terms of (one of) the above inverses are presented next.

The update

The update for the information matrix follows immediately from (7.2.1-3) and the matrix inversion lemma (see Subsection 1.3.3) as

$$P(k|k)^{-1} = P(k|k-1)^{-1} + H(k)'R(k)^{-1}H(k) \tag{7.2.1-4}$$

The prediction

Denote the information matrix corresponding to the state prediction *without process noise* as

$$A(k-1)^{-1} \triangleq F(k-1)P(k-1|k-1)F(k-1)' \tag{7.2.1-5}$$

or, since F, as a transition matrix, is invertible,

$$A(k-1) = [F(k-1)^{-1}]'P(k-1|k-1)^{-1}F(k-1)^{-1} \tag{7.2.1-6}$$

The prediction information matrix is

$$\begin{aligned} P(k|k-1)^{-1} &= [F(k-1)P(k-1|k-1)F(k-1)' + \Gamma(k-1) \\ &\quad \cdot Q(k-1)\Gamma(k-1)']^{-1} \\ &= [A(k-1)^{-1} + \Gamma(k-1)Q(k-1)\Gamma(k-1)']^{-1} \end{aligned} \tag{7.2.1-7}$$

and it can be rewritten, again with the matrix inversion lemma, as

$$\begin{aligned} P(k|k-1)^{-1} &= A(k-1) - A(k-1)\Gamma(k-1) \\ &\quad \cdot [\Gamma(k-1)'A(k-1)\Gamma(k-1) + Q(k-1)^{-1}]^{-1}\Gamma(k-1)'A(k-1) \end{aligned} \tag{7.2.1-8}$$

The expression of the gain (7.2.1-2) is rewritten as

$$\begin{aligned} W(k) &= P(k|k-1)H(k)'\Big\{[H(k)P(k|k-1)H(k)'+R(k)]^{-1} \\ &\quad +R(k)^{-1}-R(k)^{-1}\Big\} \\ &= P(k|k-1)H(k)'R(k)^{-1} \\ &\quad +P(k|k-1)H(k)'[H(k)P(k|k-1)H(k)'+R(k)]^{-1} \\ &\quad \cdot\Big\{I-[H(k)P(k|k-1)H(k)'+R(k)]R(k)^{-1}\Big\} \\ &= \Big\{P(k|k-1)-P(k|k-1)H(k)'[H(k)P(k|k-1)H(k)'+R(k)]^{-1} \\ &\quad \cdot H(k)P(k|k-1)\Big\}H(k)'R(k)^{-1} \end{aligned} \tag{7.2.1-9}$$

With the matrix inversion lemma, (7.2.1-9) becomes

$$W(k)=[P(k|k-1)^{-1}+H(k)'R(k)^{-1}H(k)]^{-1}H(k)'R(k)^{-1} \tag{7.2.1-10}$$

which is the sought-after expression of the gain. It can be easily shown that the above is equivalent to the alternate form of the gain:

$$W(k)=P(k|k)H(k)'R(k)^{-1} \tag{7.2.1-11}$$

Duality Between the Covariance and Information Equations

Note the ***duality*** [1] between the covariance and the information propagation equations:

Covariance prediction (7.2.1-1)	$\longleftrightarrow$	Information update (7.2.1-4)
Covariance update (7.2.1-3)	$\longleftrightarrow$	Information prediction (7.2.1-8)

Carrying out in (7.2.1-1), with the time index increased by one, the following replacements

$$P(k+1|k) \rightarrow P(k+1|k+1)^{-1} \tag{7.2.1-12}$$

$$A(k) \rightarrow P(k+1|k) \tag{7.2.1-13}$$

$$\Gamma(k) \rightarrow H(k)' \tag{7.2.1-14}$$

$$Q(k) \rightarrow R(k)^{-1} \tag{7.2.1-15}$$

yield (7.2.1-4).

Similarly, carrying out in (7.2.1-3), again with the time index increased by one, the replacements

$$P(k+1|k)^{-1} \leftarrow P(k+1|k+1) \tag{7.2.1-16}$$

[1] The concept of duality can be illustrated by what the frog said in a restaurant when he got his soup: "Waiter! There is *no* fly in my soup!"

$$A(k) \leftarrow P(k+1|k) \tag{7.2.1-17}$$

$$\Gamma(k) \leftarrow H(k)' \tag{7.2.1-18}$$

$$Q(k)^{-1} \leftarrow R(k) \tag{7.2.1-19}$$

yield (7.2.1-8).

Since (7.2.1-16) to (7.2.1-19) are exactly the inverses of (7.2.1-12) to (7.2.1-15), this proves the duality.

7.2.2 Overview of the Information Filter Algorithm

The sequence of calculations of the information matrices and filter gain for one cycle of the information filter are presented in Fig. 7.2.2-1.

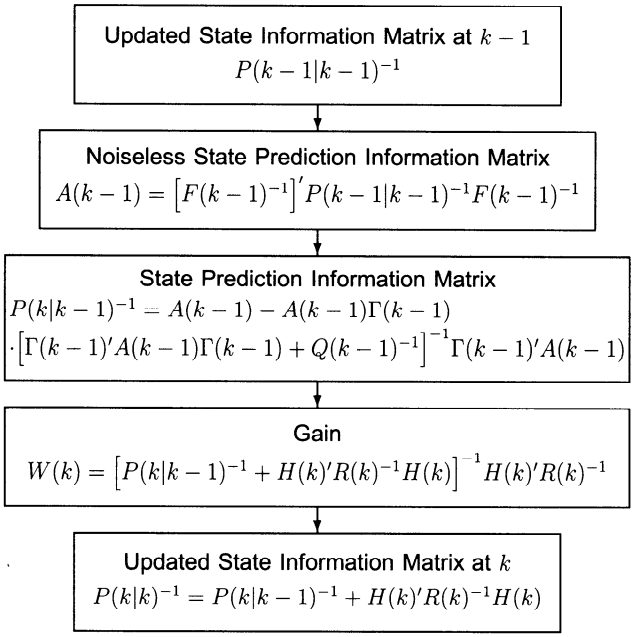

Figure 7.2.2-1: The information matrix and filter gain calculations in the information filter.

The state estimate $\hat{x}$ can be computed in the same manner as in the conventional Kalman filter. Namely, Fig. 7.2.2-1 replaces the right-hand column of computations from Fig. 5.2.4-1. There is an alternative, however, that is discussed in the next subsection.

This implementation of the Kalman filter is advantageous when the dimension n_z of the measurement vector is larger than the dimension n_x of the state. Note that the inversions in the above sequence of calculations are for $n_x \times n_x$ matrices (and Q, which is $n_v \times n_v$, with $n_v \leq n_x$), while the conventional algorithm requires the inversion of the innovation covariance, which is $n_z \times n_z$. The inverse of the measurement noise covariance $R(k)$ is usually simple to obtain because it is, in many cases, diagonal.

7.2.3 Recursion for the Information Filter State

For the situations where there is no initial estimate, one can start the estimation with the initial information matrix as zero, namely,

$$P(0|0)^{-1} = 0 \tag{7.2.3-1}$$

This amounts to a ***noninformative (diffuse) prior*** because of the infinite uncertainty (infinite variance) associated with it.

In this case one can obtain recursions from

$$\hat{y}(k-1|k-1) \triangleq P(k-1|k-1)^{-1}\hat{x}(k-1|k-1) \tag{7.2.3-2}$$

to

$$\hat{y}(k|k-1) \triangleq P(k|k-1)^{-1}\hat{x}(k|k-1) \tag{7.2.3-3}$$

and then to

$$\hat{y}(k|k) \triangleq P(k|k)^{-1}\hat{x}(k|k) \tag{7.2.3-4}$$

Note that, in view of (7.2.3-1), the initial value of $\hat{y}$, to be called the ***information filter state***, is $\hat{y}(0|0) = 0$, regardless of $\hat{x}(0|0)$ (as long as it is finite), i.e., *no initial estimate $\hat{x}(0|0)$ of the system's state is needed.*

The prediction

Using (7.2.3-3), (7.2.1-1) and the matrix inversion lemma, one has

$$\begin{aligned}
\hat{y}(k|k-1) &= P(k|k-1)^{-1}\hat{x}(k|k-1) \\
&= [F(k-1)P(k-1|k-1)F(k-1)' \\
&\quad +\Gamma(k-1)Q(k-1)\Gamma(k-1)']^{-1}F(k-1)\hat{x}(k-1|k-1) \\
&= [F(k-1)']^{-1}[P(k-1|k-1)+F(k-1)^{-1}\Gamma(k-1)Q(k-1) \\
&\quad \cdot\Gamma(k-1)'(F(k-1)')^{-1}]^{-1}F(k-1)^{-1}F(k-1)\hat{x}(k-1|k-1) \\
&= [F(k-1)']^{-1}\{P(k-1|k-1)^{-1}-P(k-1|k-1)^{-1} \\
&\quad \cdot F(k-1)^{-1}\Gamma(k-1)[\Gamma(k-1)'(F(k-1)')^{-1}P(k-1|k-1)^{-1} \\
&\quad \cdot F(k-1)^{-1}\Gamma(k-1)+Q(k-1)^{-1}]^{-1} \\
&\quad \cdot\Gamma(k-1)'(F(k-1)')^{-1}P(k-1|k-1)^{-1}\}\hat{x}(k-1|k-1) \\
&= [F(k-1)']^{-1}\{I-P(k-1|k-1)^{-1}F(k-1)^{-1}\Gamma(k-1) \\
&\quad \cdot[\Gamma(k-1)'(F(k-1)')^{-1}P(k-1|k-1)^{-1}F(k-1)^{-1}\Gamma(k-1) \\
&\quad +Q(k-1)^{-1}]^{-1}\Gamma(k-1)'(F(k-1)')^{-1}\hat{y}(k-1|k-1)
\end{aligned} \tag{7.2.3-5}$$

Using the notation (7.2.1-5), the above can be rewritten as

$$\boxed{\begin{aligned}\hat{y}(k|k-1) = (F(k-1)')^{-1}\{&I - P(k-1|k-1)^{-1}F(k-1)^{-1}\Gamma(k-1)\\ &\cdot[\Gamma(k-1)'A(k-1)\Gamma(k-1) + Q(k-1)^{-1}]^{-1}\\ &\cdot\Gamma(k-1)'(F(k-1)')^{-1}\hat{y}(k-1|k-1)\end{aligned}} \tag{7.2.3-6}$$

The update

Using (7.2.3-4) and (7.2.1-4), one can write

$$\begin{aligned}\hat{y}(k|k) &= P(k|k)^{-1}\hat{x}(k|k)\\ &= [P(k|k-1)^{-1} + H(k)'R(k)^{-1}H(k)]\hat{x}(k|k-1)\\ &\qquad +P(k|k)^{-1}P(k|k)H(k)'R(k)^{-1}[z(k) - H(k)\hat{x}(k|k-1)]\\ &= P(k|k-1)^{-1}\hat{x}(k|k-1) + H(k)'R(k)^{-1}z(k)\end{aligned} \tag{7.2.3-7}$$

which can be rewritten as

$$\boxed{\hat{y}(k|k) = \hat{y}(k|k-1) + H(k)'R(k)^{-1}z(k)} \tag{7.2.3-8}$$

Once $P(k|k)^{-1}$ becomes invertible, i.e., all the uncertainties about the state are finite, one can recover the state estimate from the information filter state as follows:

$$\hat{x}(k|k) = P(k|k)\hat{y}(k|k) \tag{7.2.3-9}$$

7.3 SEQUENTIAL PROCESSING OF MEASUREMENTS

7.3.1 Block vs. Sequential Processing

The standard implementation of the Kalman filter for a vector measurement is by carrying out the state update simultaneously with the entire measurement vector from a given time, i.e., ***block processing***.

If the measurement noise vector components $w_i(k)$, $i = 1, \ldots, n_z$, are uncorrelated, that is,

$$R(k) = E[w(k)w(k)'] = \text{diag}[r_1(k), \ldots, r_{n_z}(k)] \tag{7.3.1-1}$$

then one can carry out the update of the state with *one component of the measurement at a time*, that is, ***sequential processing*** or ***scalar updates***.

If the matrix R is not diagonal, one can apply a linear transformation on the measurement to diagonalize its covariance matrix. This can be done using a LDL′ factorization, to be discussed later in Subsection 7.4.2.

The measurement

$$z(k) = \begin{bmatrix} z_1(k) \\ \vdots \\ z_{n_z}(k) \end{bmatrix} = H(k)x(k) + w(k) = \begin{bmatrix} h_1(k)'x(k) + w_1(k) \\ \vdots \\ h_{n_z}(k)'x(k) + w_{n_z}(k) \end{bmatrix} \quad (7.3.1\text{-}2)$$

will be considered as a *sequence of scalar measurements* $z_i(k)$, $i = 1, \ldots, n_z$.

The uncorrelatedness of the corresponding measurement noises allows the use of the Kalman filter update sequentially for each "scalar measurement" because it implies whiteness for the scalar measurement noise "sequence"

$$w_1(k), \ldots, w_{n_z}(k), w_1(k+1), \ldots, w_{n_z}(k+1), \ldots \quad (7.3.1\text{-}3)$$

A natural application of this is in a ***multisensor*** situation where the measurement noises are independent across sensors.

7.3.2 The Sequential Processing Algorithm

The sequence of *scalar updates* is described below.

Starting from the predicted state to k from $k-1$, denoted as

$$\hat{x}(k|k,0) \triangleq \hat{x}(k|k-1) \quad (7.3.2\text{-}1)$$

with associated covariance

$$P(k|k,0) \triangleq P(k|k-1) \quad (7.3.2\text{-}2)$$

the following sequence of calculations is carried out for $i = 1, \ldots, n_z$:

The scalar innovation corresponding to $z_i(k)$ has variance

$$s(k,i) = h_i(k)'P(k|k,i-1)h_i(k) + r_i(k) \quad (7.3.2\text{-}3)$$

The corresponding gain is

$$W(k,i) = \frac{P(k|k,i-1)h_i(k)}{s(k,i)} \quad (7.3.2\text{-}4)$$

and the updated state is

$$\hat{x}(k|k,i) = \hat{x}(k|k,i-1) + W(k,i)[z_i(k) - h_i(k)'\hat{x}(k|k,i-1)] \quad (7.3.2\text{-}5)$$

with covariance

$$P(k|k,i) = P(k|k,i-1) - W(k,i)h_i(k)'P(k|k,i-1) \quad (7.3.2\text{-}6)$$

which can be rewritten (and should be implemented) as

$$P(k|k,i) = P(k|k,i-1) - \frac{P(k|k,i-1)h_i(k)h_i(k)'P(k|k,i-1)}{h_i(k)'P(k|k,i-1)h_i(k) + r_i(k)} \quad (7.3.2\text{-}7)$$

Note that, in the ith update, $P(k|k,i-1)$ plays the role of a prediction covariance. Finally,

$$\hat{x}(k|k) \triangleq \hat{x}(k|k,n_z) \quad (7.3.2\text{-}8)$$

$$P(k|k) \triangleq P(k|k,n_z) \quad (7.3.2\text{-}9)$$

Note that this approach eliminates the need for the information filter since only scalar inversions are performed here.

If the measurement noise covariance matrix (7.3.1-1) is not diagonal, a transformation that diagonalizes its covariance matrix can be used. An efficient technique to accomplish this is via the LDL′ factorization, to be discussed in detail in Section 7.4. This factorization yields

$$R = LD_RL' \quad (7.3.2\text{-}10)$$

where L is a *lower triangular matrix* and D_R is a *diagonal matrix with positive entries*. Instead of z, one can use the transformed measurement

$$\bar{z} = L^{-1}z \quad (7.3.2\text{-}11)$$

and instead of H one has

$$\bar{H} = L^{-1}H \quad (7.3.2\text{-}12)$$

This then allows the use of the scalar update procedure described above.

A Note on Numerical Accuracy

The covariance update (7.3.2-7) has a subtraction and, therefore, is susceptible to numerical errors, which can cause even the loss of positive definiteness of the covariance matrix. The alternative form, known as the ***Joseph form covariance update*** (see also (5.2.3-18)),

$$\begin{aligned} P(k|k,i) &= [I - W(k,i)h_i(k)']P(k|k,i-1)[I - W(k,i)h_i(k)']' \\ &\quad + W(k,i)r_i(k)W(k,i)' \end{aligned} \quad (7.3.2\text{-}13)$$

avoids this, but is computationally more expensive.

While the above form of the covariance update is better behaved numerically, it is still too sensitive to round-off errors. Further reduction of the effect of numerical errors can be obtained by using the square-root approach, which effectively doubles the numerical precision.

7.4 SQUARE-ROOT FILTERING

7.4.1 The Steps in Square-Root Filtering

The ***square-root filtering*** implementation of the KF carries out the covariance computations for the square root of the state covariance matrix, symbolically written as $P^{1/2}$, where

$$P = P^{1/2}(P^{1/2})' \triangleq \mathcal{P}\mathcal{P}' \tag{7.4.1-1}$$

The ***square root of a matrix*** is not unique, so several approaches are possible. The approach described in the sequel uses the ***LDL' factorization*** of a (positive definite) matrix

$$P = LDL' \tag{7.4.1-2}$$

where L is a ***unit lower triangular matrix***, that is,

$$L_{ii} = 1 \qquad \forall i \tag{7.4.1-3}$$

$$L_{ij} = 0 \qquad i < j \tag{7.4.1-4}$$

and D is a *diagonal matrix with positive elements*.

The matrix square root $\mathcal{P}$ in (7.4.1-1) corresponding to (7.4.1-2) is

$$\mathcal{P} = LD^{1/2} \triangleq \mathcal{L} \tag{7.4.1-5}$$

which is unique, except for the signs of the (diagonal) elements of $D^{1/2}$, which are taken as positive. Note that $\mathcal{L}$ is a ***lower triangular matrix***. Consequently, one has the following factorization

$$P = \mathcal{L}\mathcal{L}' \triangleq \mathcal{L}\mathcal{U} \tag{7.4.1-6}$$

where $\mathcal{U}$ is an ***upper triangular matrix***, which is known as the ***the Cholesky factorization*** .

The square-root filtering algorithm consists of the following steps:

1. Factorization of the initial covariance.
2. Computation of the factors of the predicted state covariance.
3. Computation of the factors of the updated state covariance and computation of the filter gain.

These steps are discussed in the next subsections.

7.4.2 The LDL′ Factorization

Given the positive definite $n \times n$ matrix P, its decomposition, called the ***LDL′ factorization***,

$$P = [p_{ij}] = LDL' \tag{7.4.2-1}$$

where L is a *unit lower triangular matrix* and D is a *diagonal matrix with positive elements*, is obtained as follows [Bierman77]:

For $j = 1, \ldots, n-1$
  $d_j = P_{jj}$
  $L_{jj} = 1$ (all others are zero)
  For $k = j+1, \ldots, n$
    For $i = k, \ldots, n$
      $P_{ik} := P_{ik} - L_{ij}P_{kj}$
    $L_{kj} = \dfrac{P_{kj}}{d_j}$
$d_n = P_{nn}$

Note that since the diagonal terms of P have to be positive (because the matrix was assumed positive definite), the elements of the diagonal matrix D are guaranteed to be positive.

7.4.3 The Predicted State Covariance

The standard equation for the state prediction covariance is

$$P(k|k-1) = F(k-1)P(k-1|k-1)F(k-1)' + \Gamma(k-1)Q(k-1)\Gamma(k-1)' \tag{7.4.3-1}$$

or, in simpler notation, without time arguments

$$\bar{P} = FPF' + \Gamma Q \Gamma' \tag{7.4.3-2}$$

where P denotes the previous updated state covariance and $\bar{P}$ denotes state prediction covariance.

Starting from the factorized form

$$P(k-1|k-1) \triangleq P = LDL' \tag{7.4.3-3}$$

we want to find the new factorized form

$$P(k|k-1) \triangleq \bar{P} = \bar{L}\bar{D}\bar{L}' \tag{7.4.3-4}$$

The brute force method would be to evaluate (7.4.3-2) and find its LDL′ factorization using the method described in Subsection 7.4.2.

A better technique that takes advantage of the existing factorizations to obtain the ***factorized prediction covariance*** is as follows.

It is assumed that Q has been factorized as

$$Q = L_Q D_Q L_Q' \tag{7.4.3-5}$$

It will be shown how the factors of P given in (7.4.3-3) and the factors of Q given in (7.4.3-5) can be used to find the factors of $\bar{P}$.

Then we are looking for $\bar{L}$, $\bar{D}$ such that

$$\bar{P} = \bar{L}\bar{D}\bar{L}' = FLDL'F' + \Gamma L_Q D_Q L_Q' \Gamma' \tag{7.4.3-6}$$

Equation (7.4.3-6) can be rewritten as the product

$$\bar{P} = AA' \tag{7.4.3-7}$$

with the following $n_x \times 2n_x$ matrix

$$A \triangleq \left[FLD^{1/2} \quad \Gamma L_Q D_Q^{1/2} \right] \triangleq \begin{bmatrix} a_1' \\ a_2' \\ \vdots \\ a_{n_x}' \end{bmatrix} \triangleq \text{col}(a_i') \tag{7.4.3-8}$$

where a_i are $2n_x$-vectors.

Using the ***Gram-Schmidt orthogonalization*** procedure (described in Subsection 7.4.6) on the *rows* of the above matrix, one obtains

$$A = \bar{L}V \tag{7.4.3-9}$$

In the above, $\bar{L}$ is a *unit lower triangular matrix* (of dimension $n_x \times n_x$) and V (of dimension $n_x \times 2n_x$) satisfies

$$VV' = \bar{D} \tag{7.4.3-10}$$

where $\bar{D}$ is *diagonal with positive elements* (of dimension $n_x \times n_x$) because V has orthogonal rows.

Thus

$$\bar{P} = AA' = \bar{L}VV'\bar{L}' = \bar{L}\bar{D}\bar{L}' \tag{7.4.3-11}$$

is the sought-after LDL′ factorization of the prediction covariance (7.4.3-4).

Therefore, to obtain the factors $\bar{L}$ and V of the prediction covariance $\bar{P}$, one carries out the Gram-Schmidt orthogonalization of the matrix A in (7.4.3-8), which contains the factors L and D of the previous updated state covariance P and the factors L_Q and D_Q of the process noise covariance Q.

7.4.4 The Filter Gain and the Updated State Covariance

Assuming a scalar measurement update, the implementation of the state update given in (7.3.2-5) requires the calculation of the gain W, which is an n_x-vector. This is obtained together with the factorized form of the updated state covariance as follows.

With the predicted state covariance factorized as $\bar{L}\bar{D}\bar{L}'$, we are looking for the factorized updated covariance

$$P \triangleq LDL' = \bar{L}\bar{D}\bar{L}' - \frac{\bar{L}\bar{D}\bar{L}'hh'\bar{L}\bar{D}\bar{L}'}{h'\bar{L}\bar{D}\bar{L}'h + r} \tag{7.4.4-1}$$

where the time arguments and subscripts have been omitted for simplicity.

Defining

$$f = \bar{L}'h = [f_1 \ \cdots \ f_n]' \tag{7.4.4-2}$$

Equation (7.4.4-1) can be written as

$$LDL' = \bar{L}\left[\bar{D} - \frac{\bar{D}f(\bar{D}f)'}{f'\bar{D}f + r}\right]\bar{L}' \tag{7.4.4-3}$$

With

$$\bar{D} = \text{diag}[\bar{d}_1, \ldots, \bar{d}_n] \tag{7.4.4-4}$$

one has

$$\bar{D}f = [\bar{d}_1 f_1 \ \cdots \ \bar{d}_n f_n]' \tag{7.4.4-5}$$

Denote the *columns* of $\bar{L}$ as $\bar{\ell}_i$ (which are n_x-vectors), that is,

$$\bar{L} = [\bar{\ell}_1 \ \cdots \ \bar{\ell}_n] \tag{7.4.4-6}$$

The ***covariance update and gain calculation*** algorithm consists of the following [Kleinman89]:

1. Initialize

$$\alpha_{n+1} = r \tag{7.4.4-7}$$
$$\alpha_n = r + \bar{d}_n f_n^2 \tag{7.4.4-8}$$
$$\xi = [0 \ \cdots \ 0 \ \ \bar{d}_n f_n]' \tag{7.4.4-9}$$
$$d_n = \frac{\alpha_{n+1}}{\alpha_n}\bar{d}_n \tag{7.4.4-10}$$

2. For $i = n-1, \ldots, 1$

$$\beta = -\frac{f_i}{\alpha_{i+1}} \tag{7.4.4-11}$$
$$\alpha_i = \alpha_{i+1} + \bar{d}_i f_i^2 \tag{7.4.4-12}$$
$$d_i = \frac{\alpha_{i+1}}{\alpha_i}\bar{d}_i \tag{7.4.4-13}$$
$$\ell_i = \bar{\ell}_i + \beta\xi \tag{7.4.4-14}$$
$$\xi := \xi + \bar{\ell}_i \bar{d}_i f_i \tag{7.4.4-15}$$

3. Gain vector for the state update

$$W = \frac{1}{\alpha_1}\xi \tag{7.4.4-16}$$

In the above d_i, $i = 1, \ldots, n_x$, are the elements of the diagonal matrix D and the n_x-vectors ℓ_i, $i = 1, \ldots, n_x$, are the columns of the unit lower diagonal matrix L. These two matrices define the LDL′ factorization (7.4.4-1) of the updated state covariance matrix P.

7.4.5 Overview of the Square-Root Sequential Scalar Update Algorithm

It is assumed that the measurement noise covariance matrix is diagonal. If this is not the case, a Cholesky factorization is performed as in Subsection 7.4.2 and the transformations (7.3.2-10) to (7.3.2-12) are carried out.

Figure 7.4.5-1 describes the sequence of computations for the square-root sequential scalar update algorithm. The last block is repeated for each component of the n_z-dimensional measurement vector.

7.4.6 The Gram-Schmidt Orthogonalization Procedure

Given N-vectors a_i, $i = 1, \ldots, n$, where $N \geq n$, assumed to be linearly independent, let

$$v_i \triangleq a_i - \sum_{j=1}^{i-1} \frac{a_i' v_j}{v_j' v_j} v_j \qquad i = 1, \ldots, n \tag{7.4.6-1}$$

Then the vectors v_i, $i = 1, \ldots, n$, form an orthogonal set. Equation (7.4.6-1) can be rewritten as

$$a_i = v_i + \sum_{j=1}^{i-1} \bar{L}_{ij} v_j \tag{7.4.6-2}$$

where

$$\bar{L}_{ij} \triangleq \frac{a_i' v_j}{v_j' v_j} \tag{7.4.6-3}$$

Then

$$A = \begin{bmatrix} a_1' \\ a_2' \\ \vdots \\ a_n' \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \bar{L}_{21} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \bar{L}_{n1} & \bar{L}_{n2} & \cdots & 1 \end{bmatrix} \begin{bmatrix} v_1' \\ v_2' \\ \vdots \\ v_n' \end{bmatrix} \triangleq \bar{L}V \tag{7.4.6-4}$$

where $\bar{L}$ is a unit lower triangular matrix $(n \times n)$ and

$$V \triangleq \begin{bmatrix} v_1' \\ v_2' \\ \vdots \\ v_n' \end{bmatrix} \triangleq \text{col}(v_i') \tag{7.4.6-5}$$

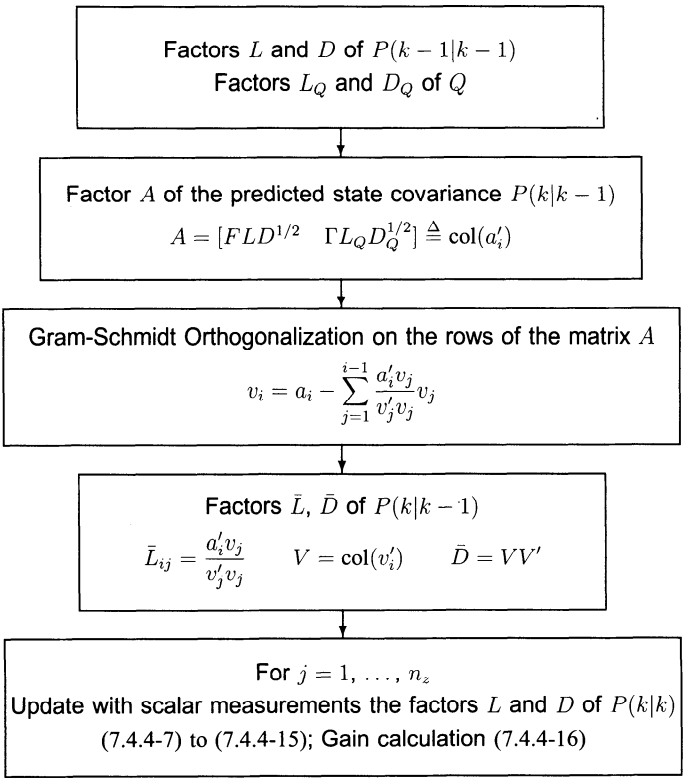

Figure 7.4.5-1: The square-root sequential covariance update algorithm.

Note that the transpose of v_i is the ith *row* of V.

From the orthogonality of the n-vectors v_i, $i = 1, \ldots, n$, it follows that

$$VV' = \bar{D} \tag{7.4.6-6}$$

is a diagonal matrix (also $n \times n$).

Equation (7.4.6-1) is called the ***Gram-Schmidt orthogonalization procedure***.

The so-called modified Gram-Schmidt orthogonalization procedure (e.g., [Bierman77]) has improved numerical properties compared to the standard procedure described above.

Application to the Generation of Correlated Random Variables

This orthogonalization procedure can also be used in the reverse direction to provide a recursion for the ***generation of correlated random variables*** from a standard random number generator that yields independent random variables.

Similarly, one can obtain a simple recursion for the ***generation of a random sequence with exponential autocorrelation*** (see problem 7-1).

These techniques are useful in simulations that require the generation of correlated random variables.

7.5 NOTES AND PROBLEMS

7.5.1 Bibliographical Notes

The most comprehensive documentation of numerical algorithms for linear estimation can be found in [Bierman77, Maybeck79, Maybeck82]. The information filter and its use for estimation start-up without initial estimates is discussed in [Maybeck79]. Applications of the information filter to robotics are discussed in [Mutambara98]. The material presented in this chapter is based in part on private communications from D. L. Kleinman and K. R. Pattipati.

A discussion of the use of parallel computers for the implementation of linear estimation can be found in [O'Halloran88]. Parallel implementation of factorized (square-root) estimation is discussed in [Itzkowitz89].

Several applications of covariance factorization in estimation as well as additional references on this topic can be found in [Baheti90].

7.5.2 Problems

7-1 Generation of a random sequence with exponential autocorrelation. Given a set of independent zero-mean unit-variance random variables, x_i, $i = 1, \ldots,$ find a *recursion*

that yields the zero-mean sequence y_i, $i = 1, \ldots$, with the autocorrelation function

$$E[y_i y_j] = \rho^{|i-j|} \qquad |\rho| < 1$$

Hint: The form of the recursion should be $y_{i+1} = f(y_i, x_{i+1})$.

7-2 Start-up of the filter for the DWNA model without initial estimates. For the DWNA model (Subsection 6.3.2) derive $\hat{x}(2)$ and $P(2|2)$ starting from $P(0|0) = 0$.

7-3 Start-up of the filter for the CWNA model without initial estimates. For the CWNA model (Subsection 6.2.2) derive $\hat{x}(2)$ and $P(2|2)$ starting from $P(0|0) = 0$.

7-4 Start-up of the filter for the DWPA model without initial estimates. For the DWNA model (Subsection 6.3.3) derive $\hat{x}(3)$ and $P(3|3)$ starting from $P(0|0) = 0$.

7-5 Start-up of the filter for the CWPA model without initial estimates. For the DWNA model (Subsection 6.2.3) derive $\hat{x}(3)$ and $P(3|3)$ starting from $P(0|0) = 0$.