5

Sequential State Estimation

The advancement and perfection of mathematics are intimately connected with the prosperity of the State. Napoleon

IN the developments of the previous chapters, estimation concepts are formulated and applied to systems whose measured variables are related to the estimated parameters by *algebraic* equations. The present chapter extends these results to allow estimation of parameters embedded in the model of a *dynamical system*, where the model usually includes both *algebraic* and *differential* equations. We will find that the sequential estimation results of §1.3 and the probability concepts introduced in Chapter 2, developed for estimation of *algebraic* systems, remain valid for estimation of *dynamical* systems upon making the appropriate new interpretations of the matrices involved in the estimation algorithms. In the event that the differential equations have explicitly algebraic solutions, of course, the entire model becomes algebraic equations and the methods of the previous chapters apply immediately (see example 1.7 for instance). On the other hand, we'll find that the sequential estimation results of §1.3 must be extended to properly account for "motion" of the dynamical system between measurement and estimation epochs. We should now note that the words "sequential state estimation" and "filtering" are used synonymously throughout the remainder of the text. The concept of filtering is regularly stated when the time at which an estimate is desired coincides with the last measurement point.[1] In the examples presented in this chapter and in later chapters, sequential state estimation is often used to not only reconstruct state variables but also "filter" noisy measurement processes. Thus,"sequential state estimation" and "filtering" are often interchanged in the literature.

The formulations of the present chapter are developed as natural extensions of the estimation methods of the first two chapters using the differential equation models and notations of Chapter 4. We begin our discussion of sequential state estimation by showing a simple first-order sequential filtering process. Then we will introduce the concept of reconstructing all of the state variables in a dynamical system using Ackermann's formula. Next, the *Kalman filter* is derived for linear systems. We shall see that the filter structure remains unchanged from Ackermann's basic developments; however, the associated gain for the estimator in the Kalman filter is rigourously derived using the probability concepts introduced in Chapter 2. Then, the Kalman filter is expanded to include nonlinear dynamical models, which leads to the development of the *extended Kalman filter*. Formulations are presented for continuous-time measurements and models, discrete-time measurements and models, and discrete-time measurements with continuous-time models. Finally, several advanced topics are shown including: factorization methods, colored-noise Kalman filtering, adaptive filtering, error analysis, Unscented filtering, and robust filtering.

5.1 A Simple First-Order Filter Example

In the estimation formulations developed in the first two chapters, it has been assumed that a specific set of parameters are being estimated; additional data have been allowed, but the parameters being estimated remained unchanged. A more complicated situation arises whenever the set of parameters being estimated is allowed to change during the estimation process. To motivate the discussion, consider real-time estimation of the state of a maneuvering spacecraft. As each subset of observations becomes available, it is desired to obtain an optimal estimate of the state *at that instant* in order to, for example, provide the best current information to base control decisions upon.

In this section we introduce the concept of sequential state estimation by considering a simple first-order example that will be used to motivate the theoretical developments of this chapter. Suppose that a "truth" model is generated using the following first-order differential equation:

$$\dot{x}(t) = F\,x(t), \quad x(t_0) = 1 \tag{5.1a}$$

$$\tilde{y}(t) = H\,x(t) + v(t) \tag{5.1b}$$

Synthetic measurements are created for a 10-second time interval with $F = -1$ and $H = 1$, assuming that $v(t)$ is a zero-mean Gaussian noise process with the standard deviation given by 0.05. The measurements are shown in Figure 5.1.

Suppose now that we wish to estimate $x(t)$ using the available measurements and some dynamic model. In practice the actual "truth" model is unknown (if it were known exactly then we wouldn't need an estimator!). For this example, we will assume that the initial condition is known exactly, but the "modelled" value for F is given by $\bar{F} = -1.5$. Clearly, if we replace F with $\bar{F}$ in eqn. (5.1) and integrate this equation to find an estimate for $x(t)$, we would find that the estimated $x(t)$ is far from the truth. In order to produce better results, we shall use the age-old adage commonly spoken in control of dynamic systems: "when in doubt, use feedback!" Consider the following linear feedback system for the state and output estimates:

$$\dot{\hat{x}}(t) = \bar{F}\,\hat{x}(t) + K[\tilde{y}(t) - \bar{H}\,\hat{x}(t)], \quad \hat{x}(t_0) = 1 \tag{5.2a}$$

$$\hat{y}(t) = \bar{H}\,\hat{x}(t) \tag{5.2b}$$

where $\hat{x}(t)$ denotes the estimate of $x(t)$, K is a constant gain, and $\bar{H} = H = 1$. At this point we do not consider how to determine the value of K, but instead (since

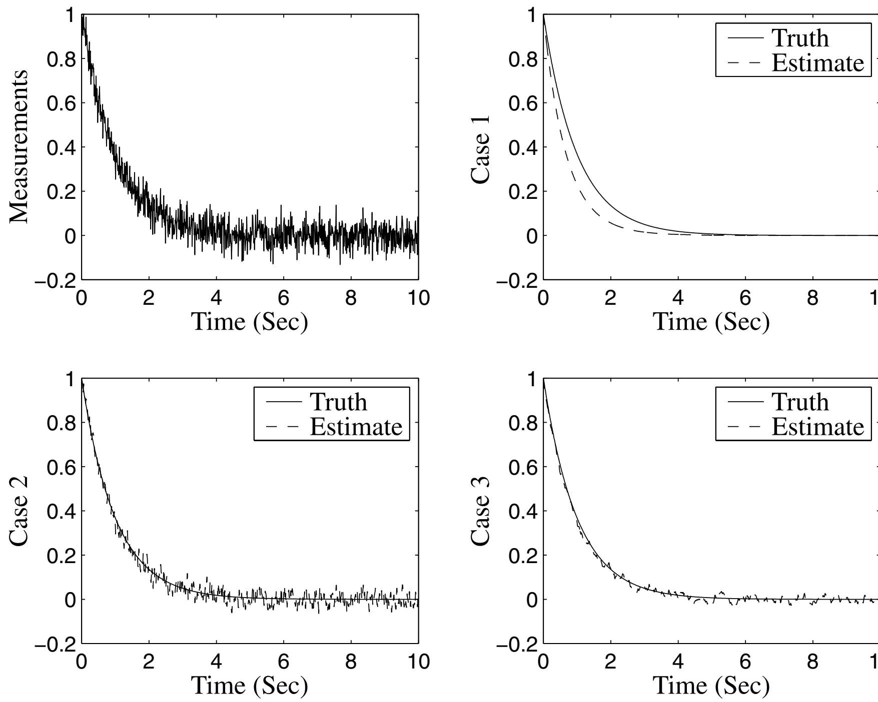

Figure 5.1: First-Order Filter Results

we know the truth) we will pick various values and compare the resulting $\hat{x}(t)$ with the true $x(t)$. Three cases are evaluated: Case 1 ($K = 0.1$), Case 2 ($K = 100$), and Case 3 ($K = 15$). The resulting estimates from each of these cases are shown in Figure 5.1. Clearly for small gains (such as Case 1) the estimates are far from the truth. Also, for large gains (such as Case 2) the estimates are very noisy. Case 3 depicts a gain that closely follows the truth, while at the same time providing filtered estimates.

This simple example illustrates the basic concepts used in state estimation and filtering. We can see from eqn. (5.2) that as the gain (K) decreases, measurements tend to be ignored and the system relies more heavily on the model (which in this case is incorrect leading to erroneous estimates). As the gain increases the estimates rely more on the measurements; however, if the gain is too large then the model tends to be ignored all together, as shown by Case 2. This concept can also be demonstrated using a frequency domain approach. The "filter dynamics" are given by $E = \bar{F} - K\bar{H}$ (here we assume that K is chosen so that the filter dynamics are stable), which is the inverse of the time constant of the system. In the frequency domain, the corner frequency (bandwidth) of the filter is given by $|E|$. As the gain K increases the corner frequency becomes larger, which yields a higher bandwidth in the system, thus allowing more high-frequency noise to enter into the estimate.

Conversely, as the gain K decreases the bandwidth decreases, which allows less noise through the filtered system. An "optimal" gain is one that both closely follows the model while at the same time provides filtered estimates.

5.2 Full-Order Estimators

In the previous section we showed a simple first-order filter. In the present section we expand the previous results to full-order (i.e., n^{th}-order) systems. For the first step we will assume that the plant dynamics $(F,\ B,\ H)$, with $D = 0$, in eqn. (3.11) are known exactly; however, the initial condition $\mathbf{x}(t_0)$ is not known precisely. Expanding eqn. (5.2) for MIMO systems gives (assuming no errors in the plant dynamics)

$$\dot{\hat{\mathbf{x}}} = F\,\hat{\mathbf{x}} + B\,\mathbf{u} + K[\tilde{\mathbf{y}} - H\,\hat{\mathbf{x}}] \tag{5.3a}$$

$$\hat{\mathbf{y}} = H\,\hat{\mathbf{x}} \tag{5.3b}$$

Note that $\mathbf{u}$ is a deterministic quantity (such as a control input). The truth model is given by

$$\dot{\mathbf{x}} = F\,\mathbf{x} + B\,\mathbf{u} \tag{5.4a}$$

$$\mathbf{y} = H\,\mathbf{x} \tag{5.4b}$$

The measurement model follows

$$\tilde{\mathbf{y}} = H\,\mathbf{x} + \mathbf{v} \tag{5.5}$$

where $\mathbf{v}$ is a vector of measurement noise. In order to analyze the estimator's performance we can compute an error representing the difference between the estimated state and the true state:

$$\tilde{\mathbf{x}} \equiv \hat{\mathbf{x}} - \mathbf{x} \tag{5.6}$$

Taking the time derivative of eqn. (5.6) and substituting eqns. (5.3a) and (5.4a) into the resulting expression leads to

$$\dot{\tilde{\mathbf{x}}} = (F - K\,H)\tilde{\mathbf{x}} + K\,\mathbf{v} \tag{5.7}$$

Note that eqn. (5.7) is no longer a function of $\mathbf{u}$. Obviously, we must choose K so that $F - K\,H$ is stable. If the filter dynamics are stable and the measurements errors are negligibly small, then the error will decay to zero and remain there for any initial condition error. It is evident from the $K\,\mathbf{v}$ forcing term in eqn. (5.7) that if the gain K is large then the filter eigenvalues (poles) will be fast, but high-frequency noise can dominate the errors due to the measurements. If the gain K is too small then the errors may take too long to decay toward zero. We must choose K so that $F - K\,H$

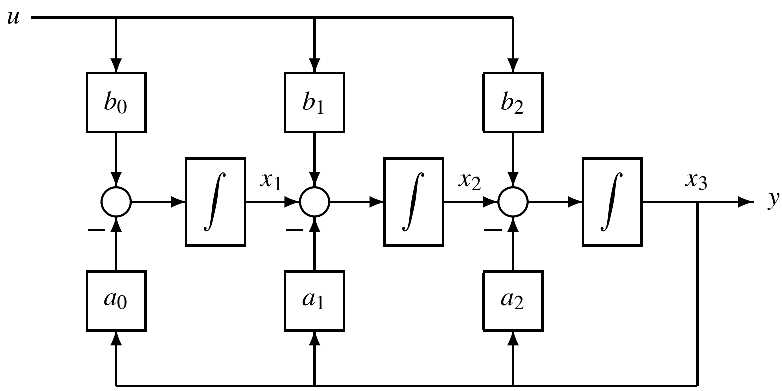

Figure 5.2: Third-Order Observer Canonical Form

is stable with reasonably fast eigenvalues, while at the same time providing filtered state estimates in the estimator.

One method to select K is to define a set of known estimator error-eigenvalue locations, and choose K so that these desired locations are achieved. This "pole-placement" concept is readily applied in the control of dynamic systems. We begin this concept by using the observer canonical form for SISO systems given by eqn. (3.98), which allows for a simple approach to place the estimator eigenvalues:

$$F_o = \begin{bmatrix} 0 & 0 & \cdots & 0 & -a_0 \\ 1 & 0 & \cdots & 0 & -a_1 \\ 0 & 1 & \cdots & 0 & -a_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -a_{n-1} \end{bmatrix} \tag{5.8a}$$

$$B_o = \begin{bmatrix} b_0 & b_1 & \cdots & b_{n-1} \end{bmatrix}^T \tag{5.8b}$$

$$H_o = \begin{bmatrix} 0 & 0 & \cdots & 1 \end{bmatrix} \tag{5.8c}$$

The coefficients of the characteristic equation are given by the last column of F_o.

Consider the third-order case, where the state matrix in eqn. (5.8a) reduces to

$$F_o = \begin{bmatrix} 0 & 0 & -a_0 \\ 1 & 0 & -a_1 \\ 0 & 1 & -a_2 \end{bmatrix} \tag{5.9}$$

Since we have assumed only a single measurement, then K reduces to a 3×1 vector. The estimator closed-loop state matrix $(F_o - K H_o)$ for this case is given by

$$F_o - K H_o = \begin{bmatrix} 0 & 0 & -(a_0 + k_1) \\ 1 & 0 & -(a_1 + k_2) \\ 0 & 1 & -(a_2 + k_3) \end{bmatrix} \tag{5.10}$$

where $K \equiv [k_1\ k_2\ k_3]^T$. A block diagram of this system is shown in Figure 5.2. This shows the advantage of this observer canonical form, since all of the feedback loops come from the output. The characteristic equation associated with the state matrix in eqn. (5.10) is given by

$$s^3 + (a_2 + k_3)\,s^2 + (a_1 + k_2)\,s + (a_0 + k_1) = 0 \tag{5.11}$$

Suppose that we have a desired characteristic equation formed from a set of desired eigenvalues in the estimator, given by

$$d(s) = s^3 + \delta_2\,s^2 + \delta_1\,s + \delta_0 = 0 \tag{5.12}$$

Then the gain matrix K can be obtained by comparing the corresponding coefficients in eqns. (5.11) and (5.12):

$$\begin{aligned} k_1 &= \delta_0 - a_0 \\ k_2 &= \delta_1 - a_1 \\ k_3 &= \delta_2 - a_2 \end{aligned} \tag{5.13}$$

This approach can easily be expanded to higher-order systems; however, this can become quite tedious and numerically inefficient. It would be useful if the gain K can be derived using the matrix F directly, without having to convert F into observer canonical form. Applying the Cayley-Hamilton theorem from eqn. (A.56), which states that every $n \times n$ matrix satisfies its own characteristic equation, to the matrix $E = F - KH$ in eqn. (5.12) leads to

$$d(E) = E^3 + \delta_2\,E^2 + \delta_1\,E + \delta_0\,I = 0 \tag{5.14}$$

Performing the multiplications for E^3 and E^2, and collecting terms gives

$$E^2 = F^2 - KHF - EKH \tag{5.15a}$$

$$E^3 = F^3 - KHF^2 - EKHF - E^2KH \tag{5.15b}$$

Substituting eqn. (5.15) into eqn. (5.14), and again collecting terms gives

$$\begin{aligned} &F^3 + \delta_2\,F^2 + \delta_1\,F + \delta_0\,I \\ &-\delta_1\,KH - \delta_2\,KHF - \delta_2\,EKH - KHF^2 - EKHF - E^2KH = 0 \end{aligned} \tag{5.16}$$

Since the first four terms are defined as $d(F)$, we can rewrite eqn. (5.16) as

$$d(F) = \left[(\delta_1\,K + \delta_2\,EK + E^2K)\ \ (\delta_2\,K + EK)\ \ K\right]\begin{bmatrix} H \\ HF \\ HF^2 \end{bmatrix} \tag{5.17}$$

Therefore, the gain K can be found from

$$K = d(F)\begin{bmatrix} H \\ HF \\ HF^2 \end{bmatrix}^{-1}\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{5.18}$$

This can easily be extended for n^{th}-order systems to give *Ackermann's formula*:

$$\boxed{K = d(F)\begin{bmatrix} H \\ HF \\ HF^2 \\ \vdots \\ HF^{n-1} \end{bmatrix}^{-1}\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \equiv d(F)\mathcal{O}^{-1}\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}} \tag{5.19}$$

where $\mathcal{O}$ is clearly the observability matrix derived in §3.4. Therefore, in order to place the eigenvalues of the estimator state matrix, the original system (F, H) must be observable.

Example 5.1: In this example we will demonstrate the usefulness of eqn. (5.19) to determine the required gain in the estimator for a simple second-order system. Consider the following general system matrices:

$$F = \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix}, \quad H = \begin{bmatrix} h_1 & h_2 \end{bmatrix}$$

where f_{11}, f_{12}, f_{21}, f_{22}, h_1, and h_2 are any real-valued numbers. The gain K is given by $K = \begin{bmatrix} k_1 & k_2 \end{bmatrix}^T$ for this case. The desired characteristic equation of the estimator is given by

$$d(s) = s^2 + \delta_1 s + \delta_0 = 0$$

Computing $\det(sI - F + KH) = 0$ allows us to solve for the gain K by comparing coefficients to the desired characteristic equation. Performing this operation gives

$$\delta_0 = (k_1 h_1 - f_{11})(k_2 h_2 - f_{22}) - (k_1 h_2 - f_{12})(k_2 h_1 - f_{21})$$
$$\delta_1 = k_1 h_1 + k_2 h_2 - f_{11} - f_{22}$$

Solving these two equations for k_1 and k_2 is not trivial (this is left as an exercise for the reader); however, using eqn. (5.19) the solution is straightforward leading to

$$k_1 = \frac{1}{b\,h_1 - a\,h_2}\left[d\,h_1 - c\,h_2 + \delta_1\,(h_1 f_{12} - h_2 f_{11}) - \delta_0 h_2\right]$$
$$k_2 = \frac{1}{b\,h_1 - a\,h_2}\left[g\,h_1 - e\,h_2 + \delta_1\,(h_1 f_{22} - h_2 f_{21}) + \delta_0 h_1\right]$$

where

$$a = h_1 f_{11} + h_2 f_{21}$$
$$b = h_1 f_{12} + h_2 f_{22}$$
$$c = f_{11}^2 + f_{12} f_{21}$$
$$d = f_{11} f_{12} + f_{12} f_{22}$$
$$e = f_{11} f_{21} + f_{21} f_{22}$$
$$g = f_{22}^2 + f_{12} f_{21}$$

Also, as $(b\,h_1 - a\,h_2) \to 0$ the gains k_1 and k_2 approach infinity. This is due to the fact that $(b\,h_1 - a\,h_2)$ is the determinant of the observability matrix. Therefore, as observability slips away the gains must increase in order to "see" the states. This can have a negative effect for noisy systems, as shown in §5.1.

If the system is in observer canonical form, then $h_1 = 0$, $h_2 = 1$, $f_{11} = 0$, and $f_{21} = 1$, and the gain expressions simplify significantly with $a = 1$, $b = f_{22}$, $c = f_{12}$, $d = f_{12}\,f_{22}$, $e = f_{22}$, and $g = f_{22}^2 + f_{12}$. Then the gains are given by

$$
\begin{aligned}
k_1 &= f_{12} + \delta_0 \\
k_2 &= f_{22} + \delta_1
\end{aligned}
$$

which is analogous to the expression shown in eqn. (5.11). This example clearly demonstrates the power of using Ackermann's to determine a gain K to match the desired characteristic equation in an estimator design.

5.2.1 Discrete-Time Estimators

We now will show Ackermann's formula for discrete-time system representations, given by eqn. (3.111). We can simply add a feedback term involving the difference between the measured and estimated output analogous to the continuous-time case; however, this gives an estimate at the current time based on the *previous* measurement (since $\hat{\mathbf{x}}_{k+1}$ will be used in the estimator). In order to provide a current estimate using the current measurement the discrete-time estimator is given by two coupled equations, given by

$$
\boxed{
\begin{aligned}
\hat{\mathbf{x}}_{k+1}^- &= \Phi\,\hat{\mathbf{x}}_k^+ + \Gamma\,\mathbf{u}_k \qquad (5.20a)\\
\hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + K[\tilde{\mathbf{y}}_k - H\,\hat{\mathbf{x}}_k^-] \qquad (5.20b)
\end{aligned}
}
$$

Equation (5.20a) is known as the *prediction* or *propagation* equation, and eqn. (5.20b) is known as the *update* equation. The truth model is given by

$$
\mathbf{x}_{k+1} = \Phi\,\mathbf{x}_k + \Gamma\,\mathbf{u}_k \tag{5.21a}
$$

$$
\mathbf{y}_k = H\,\mathbf{x}_k \tag{5.21b}
$$

A single estimator equation can be derived by simply substituting eqn. (5.20b) into eqn. (5.20a) giving

$$
\hat{\mathbf{x}}_{k+1}^- = \Phi\,\hat{\mathbf{x}}_k^- + \Gamma\,\mathbf{u}_k + \Phi K[\tilde{\mathbf{y}}_k - H\,\hat{\mathbf{x}}_k^-] \tag{5.22}
$$

The error states for the prediction and for the update are defined by

$$
\tilde{\mathbf{x}}_k^- \equiv \hat{\mathbf{x}}_k^- - \mathbf{x}_k \tag{5.23a}
$$

$$
\tilde{\mathbf{x}}_k^+ \equiv \hat{\mathbf{x}}_k^+ - \mathbf{x}_k \tag{5.23b}
$$

Taking one time-step ahead of eqn. (5.23) and substituting eqns. (5.20a) and (5.20b) into the resulting expressions leads to

$$\tilde{\mathbf{x}}_{k+1}^{-} = \Phi[I - KH]\tilde{\mathbf{x}}_k^{-} \tag{5.24a}$$

$$\tilde{\mathbf{x}}_{k+1}^{+} = [I - KH]\Phi\,\tilde{\mathbf{x}}_k^{+} \tag{5.24b}$$

Note that $\Phi[I - KH]$ and $[I - KH]\Phi$ have the same eigenvalues.

The discrete-time desired characteristic equation for the estimator is given by

$$d(z) = z^n + \delta_{n-1}\,z^{n-1} + \cdots + \delta_1\,z + \delta_0 = 0 \tag{5.25}$$

The form for the estimator error in eqn. (5.24b) is similar to the continuous-time case in eqn. (5.7) with H replaced with $H\Phi$. Therefore, Ackermann's formula for the discrete-time case is given by

$$\boxed{K = d(\Phi)\begin{bmatrix} H\Phi \\ H\Phi^2 \\ H\Phi^3 \\ \vdots \\ H\Phi^n \end{bmatrix}^{-1}\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \equiv d(\Phi)\Phi^{-1}\mathcal{O}_d^{-1}\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}} \tag{5.26}$$

where $\mathcal{O}_d$ is the discrete-time observability matrix given in eqn. (3.117). As in the continuous-time case, the discrete-time system must be observable for the inverse in eqn. (5.26) to exist.

The estimator design approach introduced in this section can be tedious and somewhat heuristic for higher-order systems since it is not commonly known where to properly place all the estimator eigenvalues. To overcome this difficulty, we can choose 2 of the n eigenvalues so that a dominant second-order system is produced. The remaining eigenvalues can be chosen to have real parts corresponding to a sufficiently damped response in the estimator.[2] Thus the higher-order estimator will mimic (and can be subsequently analyzed as) a second-order system. Thankfully, there is a better way, as will next be seen in the derivation of the Kalman filter.

5.3 The Discrete-Time Kalman Filter

The estimators derived in §5.2 require a desired characteristic equation in the filter dynamics. The answer to the obvious question "How do we choose the poles of the estimator?" is not trivial. In practice, this usually entails an ad hoc approach until a specified performance level is achieved. The *Kalman filter*[3] provides a rigorous theoretical approach to "place" the poles of the estimator, based upon stochastic processes for the measurement error and model error. As is shown in Chapter 2, we do not know the exact values for these errors; however, we do make some assumptions on the nature of the errors (e.g., a zero-mean Gaussian noise process). Three formulations will be given. The first, described in this section, assumes both discrete-time dynamic models and measurements; the second, described in the next section, assumes both continuous-time dynamic models and measurements; and the third assumes continuous-time dynamic models with discrete-time measurements.

5.3.1 Kalman Filter Derivation

We begin the derivation of the discrete-time Kalman filter assuming that both the model and measurements are available in discrete-time form. Suppose that the initial condition of a state $\mathbf{x}_0$ is unknown (as in §5.2); in addition suppose that the discrete-time model and measurements are corrupted by noise. The "truth" model for this case is given by

$$\boxed{\begin{aligned} \mathbf{x}_{k+1} &= \Phi_k \mathbf{x}_k + \Gamma_k \mathbf{u}_k + \Upsilon_k \mathbf{w}_k \qquad (5.27\text{a}) \\ \tilde{\mathbf{y}}_k &= H_k \mathbf{x}_k + \mathbf{v}_k \qquad (5.27\text{b}) \end{aligned}}$$

where $\mathbf{v}_k$ and $\mathbf{w}_k$ are assumed to be zero-mean Gaussian white-noise processes, which means that the errors are not correlated forward or backward in time so that

$$E\left\{\mathbf{v}_k \mathbf{v}_j^T\right\} = \begin{cases} 0 & k \neq j \\ R_k & k = j \end{cases} \tag{5.28}$$

and

$$E\left\{\mathbf{w}_k \mathbf{w}_j^T\right\} = \begin{cases} 0 & k \neq j \\ Q_k & k = j \end{cases} \tag{5.29}$$

This requirement preserves the block diagonal structure of the covariance and weight matrices introduced in §1.3. We further assume that $\mathbf{v}_k$ and $\mathbf{w}_k$ are *uncorrelated* so that $E\left\{\mathbf{v}_k \mathbf{w}_k^T\right\} = 0$ for all k. The quantity $\mathbf{w}_k$ is a forcing ("process") noise on the system of differential equations.

It is desired to update the current estimate of the state ($\hat{\mathbf{x}}_k$) to obtain ($\hat{\mathbf{x}}_{k+1}$) based upon all $k+1$ measurement subsets. We will still assume that the estimator form given by eqn. (5.20) is valid; however, the gain K can vary in time, so that

$$\boxed{\begin{aligned} \hat{\mathbf{x}}_{k+1}^- &= \Phi_k \hat{\mathbf{x}}_k^+ + \Gamma_k \mathbf{u}_k \qquad (5.30\text{a}) \\ \hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + K_k[\tilde{\mathbf{y}}_k - H_k \hat{\mathbf{x}}_k^-] \qquad (5.30\text{b}) \end{aligned}}$$

Proceeding from the developments of Chapter 2, we define the following error covariances:

$$P_k^- \equiv E\left\{\tilde{\mathbf{x}}_k^- \tilde{\mathbf{x}}_k^{-T}\right\}, \quad P_{k+1}^- \equiv E\left\{\tilde{\mathbf{x}}_{k+1}^- \tilde{\mathbf{x}}_{k+1}^{-T}\right\} \tag{5.31a}$$

$$P_k^+ \equiv E\left\{\tilde{\mathbf{x}}_k^+ \tilde{\mathbf{x}}_k^{+T}\right\}, \quad P_{k+1}^+ \equiv E\left\{\tilde{\mathbf{x}}_{k+1}^+ \tilde{\mathbf{x}}_{k+1}^{+T}\right\} \tag{5.31b}$$

where

$$\tilde{\mathbf{x}}_k^- \equiv \hat{\mathbf{x}}_k^- - \mathbf{x}_k, \quad \tilde{\mathbf{x}}_{k+1}^- \equiv \hat{\mathbf{x}}_{k+1}^- - \mathbf{x}_{k+1} \tag{5.32a}$$

$$\tilde{\mathbf{x}}_k^+ \equiv \hat{\mathbf{x}}_k^+ - \mathbf{x}_k, \quad \tilde{\mathbf{x}}_{k+1}^+ \equiv \hat{\mathbf{x}}_{k+1}^+ - \mathbf{x}_{k+1} \tag{5.32b}$$

are the state errors in the prediction and update, respectively. Our goal is to derive expressions for both P_{k+1}^- and P_{k+1}^+, and also derive an optimal expression for the gain K_k in eqn. (5.30b). Since eqn. (5.30a) is not a direct function of the gain K_k, the expression for P_{k+1}^- is fairly straightforward to derive. Substituting eqns. (5.27a) and (5.30a) into eqn. (5.32a), and using the definition of $\tilde{\mathbf{x}}_k^+$ in eqn. (5.32b) leads to

$$\tilde{\mathbf{x}}_{k+1}^- = \Phi_k \tilde{\mathbf{x}}_k^+ - \Upsilon_k \mathbf{w}_k \tag{5.33}$$

Note that eqn. (5.33) is not a function $\mathbf{u}_k$, since this term represents a known (deterministic) forcing input. Then P_{k+1}^- is given by

$$\begin{aligned} P_{k+1}^- &\equiv E\left\{\tilde{\mathbf{x}}_{k+1}^- \tilde{\mathbf{x}}_{k+1}^{-T}\right\} \\ &= E\left\{\Phi_k \tilde{\mathbf{x}}_k^+ \tilde{\mathbf{x}}_k^{+T} \Phi_k^T\right\} - E\left\{\Phi_k \tilde{\mathbf{x}}_k^+ \mathbf{w}_k^T \Upsilon_k^T\right\} \\ &- E\left\{\Upsilon_k \mathbf{w}_k \tilde{\mathbf{x}}_k^{+T} \Phi_k^T\right\} + E\left\{\Upsilon_k \mathbf{w}_k \mathbf{w}_k^T \Upsilon_k^T\right\} \end{aligned} \tag{5.34}$$

From eqn. (5.27a) we see that $\mathbf{w}_k$ and $\tilde{\mathbf{x}}_k^+$ are uncorrelated since $\tilde{\mathbf{x}}_{k+1}^+$ (not $\tilde{\mathbf{x}}_k^+$) directly depends on $\mathbf{w}_k$. Therefore $E\left\{\tilde{\mathbf{x}}_k^+ \mathbf{w}_k^T\right\} = E\left\{\mathbf{w}_k \tilde{\mathbf{x}}_k^{+T}\right\} = 0$. Using the definitions in eqns. (5.29) and (5.31b), eqn. (5.34) reduces to

$$\boxed{P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T} \tag{5.35}$$

with initial condition given by $P_0^- = E\left\{\tilde{\mathbf{x}}_0^- \tilde{\mathbf{x}}_0^{-T}\right\}$.

Our next step is to develop an optimal expression for P_k^+. Substituting eqn. (5.27b) into eqn. (5.30b), and then substituting the resulting expression into eqn. (5.32b) leads to

$$\tilde{\mathbf{x}}_k^+ = (I - K_k H_k)\hat{\mathbf{x}}_k^- + K_k H_k \mathbf{x}_k + K_k \mathbf{v}_k - \mathbf{x}_k \tag{5.36}$$

From the definition in eqn. (5.32a), eqn. (5.36) reduces to

$$\tilde{\mathbf{x}}_k^+ = (I - K_k H_k)\tilde{\mathbf{x}}_k^- + K_k \mathbf{v}_k \tag{5.37}$$

Then P_k^+ is given by

$$\begin{aligned} P_k^+ &\equiv E\left\{\tilde{\mathbf{x}}_k^+ \tilde{\mathbf{x}}_k^{+T}\right\} \\ &= E\left\{(I - K_k H_k)\tilde{\mathbf{x}}_k^- \tilde{\mathbf{x}}_k^{-T} (I - K_k H_k)^T\right\} \\ &+ E\left\{(I - K_k H_k)\tilde{\mathbf{x}}_k^- \mathbf{v}_k^T K_k^T\right\} \\ &+ E\left\{K_k \mathbf{v}_k \tilde{\mathbf{x}}_k^{-T} (I - K_k H_k)^T\right\} + E\left\{K_k \mathbf{v}_k \mathbf{v}_k^T K_k^T\right\} \end{aligned} \tag{5.38}$$

From eqn. (5.30b) we see that $\mathbf{v}_k$ and $\tilde{\mathbf{x}}_k^-$ are uncorrelated since $\tilde{\mathbf{x}}_k^+$ (not $\tilde{\mathbf{x}}_k^-$) directly depends on $\mathbf{v}_k$. Therefore $E\left\{\tilde{\mathbf{x}}_k^- \mathbf{v}_k^T\right\} = E\left\{\mathbf{v}_k \tilde{\mathbf{x}}_k^{-T}\right\} = 0$. Using the definition in eqns. (5.28) and (5.31a), then eqn. (5.38) reduces to

$$\boxed{P_k^+ = [I - K_k H_k] P_k^- [I - K_k H_k]^T + K_k R_k K_k^T} \tag{5.39}$$

In order to determine the gain K_k we minimize the trace of P_k^+, which is equivalent to minimizing the length of the estimation error vector:

$$\text{minimize} \quad J(K_k) = \text{Tr}(P_k^+) \tag{5.40}$$

Using the helpful trace identities in eqn. (2.37) with symmetric P_k^- and R_k leads to

$$\frac{\partial J}{\partial K_k} = 0 = -2(I - K_k H_k) P_k^- H_k^T + 2 K_k R_k \tag{5.41}$$

Solving eqn. (5.41) for K_k gives

$$\boxed{K_k = P_k^- H_k^T [H_k P_k^- H_k^T + R_k]^{-1}} \tag{5.42}$$

Substituting eqn. (5.42) into eqn. (5.39) yields

$$\begin{aligned} P_k^+ &= P_k^- - K_k H_k P_k^- - P_k^- H_k^T K_k^T + K_k [H_k P_k^- H_k^T + R_k] K_k^T \\ &= P_k^- - K_k H_k P_k^- \end{aligned} \tag{5.43}$$

Therefore

$$\boxed{P_k^+ = [I - K_k H_k] P_k^-} \tag{5.44}$$

Substituting eqn. (5.42) into eqn. (5.44) gives

$$P_k^+ = P_k^- - P_k^- H_k^T [H_k P_k^- H_k^T + R_k]^{-1} H_k P_k^- \tag{5.45}$$

An alternative form for the update P_k^+ is given by using the matrix inversion lemma in eqn. (1.69), which yields

$$P_k^+ = [(P_k^-)^{-1} + H_k^T R_k^{-1} H_k]^{-1} \tag{5.46}$$

Equation (5.45) implies that the update stage of the discrete-time Kalman filter *decreases* the covariance (while the propagation stage in eqn. (5.35) *increases* the covariance).[4] This observation is intuitively consistent since in general more measurements improve the state estimate.

The gain K_k in eqn. (5.42) can also be written as

$$\boxed{K_k = P_k^+ H_k^T R_k^{-1}} \tag{5.47}$$

To prove the identity we manipulate eqn. (5.42) as follows:

$$\begin{aligned} K_k &= P_k^- H_k^T [H_k P_k^- H_k^T + R_k]^{-1} \\ &= P_k^- H_k^T R_k^{-1} R_k [H_k P_k^- H_k^T + R_k]^{-1} \\ &= P_k^- H_k^T R_k^{-1} [I + H_k P_k^- H_k^T R_k^{-1}]^{-1} \end{aligned} \tag{5.48}$$

Equation (5.48) can now be rewritten as

$$K_k [I + H_k P_k^- H_k^T R_k^{-1}] = P_k^- H_k^T R_k^{-1} \tag{5.49}$$

Collecting terms now gives

$$\begin{aligned} K_k &= P_k^- H_k^T R_k^{-1} - K_k H_k P_k^- H_k^T R_k^{-1} \\ &= [I - K_k H_k] P_k^- H_k^T R_k^{-1} \end{aligned} \tag{5.50}$$

Substituting (5.44) into eqn. (5.50) proves the identity in eqn. (5.47).

A further expression can be derived for the state update in eqn. (5.30b). Equation (5.44) can be rearranged as

$$[I - K_k H_k] = P_k^+ \left(P_k^-\right)^{-1} \tag{5.51}$$

Also, the state update in eqn. (5.30b) can be rearranged as

$$\hat{\mathbf{x}}_k^+ = [I - K_k H_k]\hat{\mathbf{x}}_k^- + K_k \tilde{\mathbf{y}}_k \tag{5.52}$$

Substituting eqns. (5.47) and (5.51) into eqn. (5.52) gives

$$\boxed{\hat{\mathbf{x}}_k^+ = P_k^+ \left[\left(P_k^-\right)^{-1} \hat{\mathbf{x}}_k^- + H_k^T R_k^{-1} \tilde{\mathbf{y}}_k\right]} \tag{5.53}$$

Equation (5.53) is not particularly useful since the inverse of P_k^- is required, but its helpfulness will be shown in the derivation of the discrete-time fixed-interval smoother in Chapter 6.

The discrete-time Kalman filter is summarized in Table 5.1. First, initial conditions for the state and error covariance are given. If a measurement is given at the initial time then the state and covariance are updated using eqns. (5.42), (5.30b), and (5.44) with $\hat{\mathbf{x}}_0^- = \hat{\mathbf{x}}_0$ and $P_0^- = P_0$. Then, the state estimate and covariance are propagated to the next time step using eqns. (5.30a) and (5.35). If a measurement isn't given at the initial time then the estimate and covariance are propagated first to the next available measurement point with $\hat{\mathbf{x}}_0^+ = \hat{\mathbf{x}}_0$ and $P_0^+ = P_0$. The process is then repeated sequentially until all measurement times have been used in the filter.

We note that the structure of the discrete-time Kalman filter has the same form as the discrete estimator shown in §5.2.1, but the gain in the Kalman filter has been derived from an optimal probabilistic approach using methods from Chapter 2, namely a minimum variance approach. The propagation stage of the Kalman filter gives a

Table 5.1: Discrete-Time Linear Kalman Filter

Model	$\mathbf{x}_{k+1} = \Phi_k \mathbf{x}_k + \Gamma_k \mathbf{u}_k + \Upsilon_k \mathbf{w}_k, \quad \mathbf{w}_k \sim N(\mathbf{0}, Q_k)$ $\tilde{\mathbf{y}}_k = H_k \mathbf{x}_k + \mathbf{v}_k, \quad \mathbf{v}_k \sim N(\mathbf{0}, R_k)$
Initialize	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$ $P_0 = E\left\{\tilde{\mathbf{x}}(t_0)\,\tilde{\mathbf{x}}^T(t_0)\right\}$
Gain	$K_k = P_k^- H_k^T [H_k P_k^- H_k^T + R_k]^{-1}$
Update	$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k[\tilde{\mathbf{y}}_k - H_k \hat{\mathbf{x}}_k^-]$ $P_k^+ = [I - K_k H_k] P_k^-$
Propagation	$\hat{\mathbf{x}}_{k+1}^- = \Phi_k \hat{\mathbf{x}}_k^+ + \Gamma_k \mathbf{u}_k$ $P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T$

time update through a *prediction* of $\hat{\mathbf{x}}^-$ and covariance P^-. The measurement update stage of the Kalman filter gives a *correction* based on the measurement to yield a new *a posteriori* estimate $\hat{\mathbf{x}}^+$ and covariance P^+.[5] Together these equations form the *predictor-corrector* form of the Kalman filter.

The propagation and measurement update equations can be combined to form the *a priori* recursive form of the Kalman filter. This is accomplished by substituting eqn. (5.30b) into eqn. (5.30a), and substituting eqn. (5.44) into eqn. (5.35), giving

$$\hat{\mathbf{x}}_{k+1} = \Phi_k \hat{\mathbf{x}}_k + \Gamma_k \mathbf{u}_k + \Phi_k K_k [\tilde{\mathbf{y}}_k - H_k \hat{\mathbf{x}}_k] \tag{5.54a}$$

$$K_k = P_k H_k^T [H_k P_k H_k^T + R_k]^{-1} \tag{5.54b}$$

$$P_{k+1} = \Phi_k P_k \Phi_k^T - \Phi_k K_k H_k P_k \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T \tag{5.54c}$$

Equation (5.54c) is known as the *discrete Riccati equation*.

5.3.2 Stability and Joseph's Form

The filter stability can be proved by using Lyapunov's direct method, which is discussed for discrete-time systems in §3.6. We wish to show that the estimation error dynamics, $\tilde{\mathbf{x}}_k \equiv \hat{\mathbf{x}}_k - \mathbf{x}_k$, are stable. For the discrete-time Kalman filter we consider the following candidate Lyapunov function:

$$V(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}_k^T P_k^{-1} \tilde{\mathbf{x}}_k \tag{5.55}$$

Since P_k is required to be positive definite, then clearly its inverse exists and $V(\tilde{\mathbf{x}}) > 0$ for all $\tilde{\mathbf{x}}_k \neq \mathbf{0}$. The increment of $V(\tilde{\mathbf{x}})$ is given by

$$\Delta V(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}_{k+1}^T P_{k+1}^{-1} \tilde{\mathbf{x}}_{k+1} - \tilde{\mathbf{x}}_k^T P_k^{-1} \tilde{\mathbf{x}}_k \tag{5.56}$$

Stability is proven if we can show that $\Delta V(\tilde{\mathbf{x}}) < 0$. Substituting eqns. (5.27a) and (5.54a) into $\tilde{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_{k+1} - \mathbf{x}_{k+1}$, and collecting terms leads to

$$\tilde{\mathbf{x}}_{k+1} = \Phi_k [I - K_k H_k] \tilde{\mathbf{x}}_k + \Phi_k K_k \mathbf{v}_k - \Upsilon_k \mathbf{w}_k \tag{5.57}$$

We only need to consider the homogeneous part of eqn. (5.57) since the matrix $\Phi_k [I - K_k H_k]$ defines the stability of the filter. Substituting $\tilde{\mathbf{x}}_{k+1} = \Phi_k [I - K_k H_k] \tilde{\mathbf{x}}_k$ into eqn. (5.56) gives the following necessary condition for stability:

$$\tilde{\mathbf{x}}_k^T \left\{ [I - K_k H_k]^T \Phi_k^T P_{k+1}^{-1} \Phi_k [I - K_k H_k] - P_k^{-1} \right\} \tilde{\mathbf{x}}_k < 0 \tag{5.58}$$

Therefore, stability is achieved if the matrix within the brackets in eqn. (5.58) can be shown to be negative definite, i.e.,

$$[I - K_k H_k]^T \Phi_k^T P_{k+1}^{-1} \Phi_k [I - K_k H_k] - P_k^{-1} < 0 \tag{5.59}$$

Equation (5.59) can be rewritten as

$$I - P_{k+1} \Phi_k^{-T} [I - K_k H_k]^{-T} P_k^{-1} [I - K_k H_k]^{-1} \Phi_k^{-1} < 0 \tag{5.60}$$

Substituting eqn. (5.39) into eqn. (5.35) gives the following form for P_{k+1}:

$$P_{k+1} = \Phi_k [I - K_k H_k] P_k [I - K_k H_k]^T \Phi_k^T + \Phi_k K_k R K_k^T \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T \tag{5.61}$$

Substituting eqn. (5.61) into eqn. (5.60) gives

$$\begin{aligned} &- [\Phi_k K_k R_k K_k^T \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T] \\ &\times \Phi_k^{-T} [I - K_k H_k]^{-T} P_k^{-1} [I - K_k H_k]^{-1} \Phi_k^{-1} < 0 \end{aligned} \tag{5.62}$$

Since $\Phi_k^{-T} [I - K_k H_k]^{-T} P_k^{-1} [I - K_k H_k]^{-1} \Phi_k^{-1}$ is positive definite, eqn. (5.62) reduces down to

$$-[\Phi_k K_k R_k K_k^T \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T] < 0 \tag{5.63}$$

Clearly if R_k is positive definite and Q_k is at least positive semi-definite then the Lyapunov condition is satisfied and the discrete-time Kalman filter is stable.

In the previous derivations of the discrete-time Kalman filter the covariance matrix P_k must remain positive definite. We now show that if P_k is positive definite then P_{k+1} is also positive definite. Assuming that $Q_k = 0$ without loss in generality, from the recursive Riccati equation in eqn. (5.54c), P_{k+1} will remain positive definite if the following condition is true:

$$P_k > P_k H_k^T [H_k P_k H_k^T + R_k]^{-1} H_k P_k \tag{5.64}$$

Multiplying the left side and right side of eqn. (5.64) by H_k and H_k^T, respectively, gives

$$H_k P_k H_k^T > H_k P_k H_k^T [H_k P_k H_k^T + R_k]^{-1} H_k P_k H_k^T \tag{5.65}$$

Next, we assume that the inverse of $H_k P_k H_k^T$ exists (i.e., the number of measured observations is less than the number of states), which gives the following condition:

$$H_k P_k H_k^T + R_k > H_k P_k H_k^T \tag{5.66}$$

Clearly, if R_k is positive definite, then eqn. (5.66) is satisfied and P_{k+1} will be positive definite. Although this condition is theoretically true, numerical roundoff errors can still make P_{k+1} become negative definite. There are a number of numerical solutions to this problem, which will be further discussed in §5.7.1. One method involves using eqn. (5.39) instead of eqn. (5.44), which is referred to as the *Joseph stabilized version*.[6] This can be shown by substituting $K_k \to K_k + \delta K_k$ and $P_k^+ \to P_k^+ + \delta P_k^+$. Using these definitions eqn. (5.44) can be written as

$$P_k^+ + \delta P_k^+ = [I - K_k H_k - \delta K_k H_k] P_k^- \tag{5.67}$$

Therefore, from the definition of P_k^+ in eqn. (5.44) the perturbation δP_k^+ is given by

$$\boxed{\delta P_k^+ = -\delta K_k H_k P_k^-} \tag{5.68}$$

Equation (5.68) shows a first-order perturbation (i.e., δP_k^+ is a direct function of δK_k), which may produce roundoff errors in a computational algorithm. Substituting $K_k \to K_k + \delta K_k$ into eqn. (5.39) yields

$$\begin{aligned}\delta P^+ = {} & \delta K_k [H_k P_k^- H_k^T + R_k] \delta K_k^T \\ & + \delta K_k [R_k K_k^T - H_k P_k^- (I - K_k H_k)^T] \\ & + [K_k R_k - (I - K_k H_k) P_k^- H_k^T] \delta K_k^T\end{aligned} \tag{5.69}$$

We now will prove that $K_k R_k - (I - K_k H_k) P_k^- H_k^T = 0$. From the definition of P_k^+ in eqn. (5.44) we have

$$K_k R_k - (I - K_k H_k) P_k^- H_k^T = K_k R_k - P_k^+ H_k^T \tag{5.70}$$

Substituting the other definition of the gain K_k from eqn. (5.47) into eqn. (5.70) gives

$$K_k R_k - (I - K_k H_k) P_k^- H_k^T = P_k^+ H_k^T - P_k^+ H_k^T = 0 \tag{5.71}$$

Therefore, eqn. (5.69) reduces to

$$\boxed{\delta P_k^+ = \delta K_k [H_k P_k^- H_k^T + R_k] \delta K_k^T} \tag{5.72}$$

Equation (5.72) shows a second-order perturbation in δK_k, which provides a more robust approach in terms of numerical stability. However, Joseph's stabilized version has more computations than the form given by eqn. (5.44). Hence, a filter designer must trade off computational workload versus potential roundoff errors.

5.3.3 Information Filter and Sequential Processing

The gain K_k in eqn. (5.42) requires an inverse of order R_k, which may cause computational and numerical difficulties for large measurement sets. In order to circumvent these difficulties the *information* form of the Kalman filter can be used. The information matrix (denoted as $\mathcal{P}$) is simply the inverse of the covariance matrix P (i.e., $\mathcal{P} \equiv P^{-1}$). From eqn. (5.46) the update equation for $\mathcal{P}$ is given by

$$\boxed{\mathcal{P}_k^+ = \mathcal{P}_k^- + H_k^T R_k^{-1} H_k} \tag{5.73}$$

The information propagation is given from eqn. (5.35) by using the matrix inversion lemma in eqn. (1.69), which yields

$$\boxed{\mathcal{P}_{k+1}^- = \left[I - \Psi_k \Upsilon_k \left(\Upsilon_k^T \Psi_k \Upsilon_k + Q_k^{-1}\right)^{-1} \Upsilon_k^T\right] \Psi_k} \tag{5.74}$$

where

$$\Psi_k \equiv \Phi_k^{-T} \mathcal{P}_k^+ \Phi_k^{-1} \tag{5.75}$$

The gain can be computed from eqn. (5.47) directly as

$$\boxed{K_k = (\mathcal{P}_k^+)^{-1} H_k^T R_k^{-1}} \tag{5.76}$$

The information form clearly requires inverses of Φ_k and Q_k, which must exist. The inverse of Φ_k exists in most cases, unless a deadbeat response (i.e., a discrete pole at zero) is given in the model. However, Q_k may be zero in some cases, and the information filter cannot be used in this case. Also, if the initial state is known precisely then $P(t_0) = 0$, and the information filter cannot be initialized. Furthermore, the inverse of $\mathcal{P}_k^+$ is required in the gain calculation. The advantage of the information filter is that the largest dimension matrix inverse required is equivalent to the size of the state. Even though more inverses are needed, the information filter may be more computationally efficient than the traditional Kalman filter when the size of the measurement vector is much larger than the size of the state vector.

Another more commonly used approach to handle large measurement vectors in the Kalman filter is to use sequential processing.[4] This procedure involves processing one measurement at a time, repeated in sequence at each sampling instant. The gain and covariance are updated until all measurements at each sampling instant have been processed. The result produces estimates that are equivalent to processing all measurements together at one time instant. The underlying principle of this approach is rooted in the linearity of the Kalman filter update equation, where the rules of superposition in §3.1 apply unequivocally. This approach assumes that the measurements are uncorrelated at each time instant (i.e., R_k is a diagonal matrix). If this is not true then a linear transformation using the methods outlined in §3.1.4 can be used. We perform a linear transformation of the measurement $\tilde{\mathbf{y}}_k$ in eqn. (5.27b), giving a new measurement $\tilde{\mathbf{z}}_k$:

$$\tilde{\mathbf{z}}_k \equiv T_k \tilde{\mathbf{y}}_k = T_k H_k \mathbf{x}_k + T_k \mathbf{v}_k \tag{5.77a}$$

$$\equiv \mathcal{H}_k \mathbf{x}_k + \boldsymbol{\upsilon}_k \tag{5.77b}$$

where

$$\mathcal{H}_k \equiv T_k H_k \tag{5.78a}$$

$$\boldsymbol{\upsilon}_k \equiv T_k \mathbf{v}_k \tag{5.78b}$$

Clearly, $\boldsymbol{\upsilon}_k$ has zero mean and its covariance is given by $\mathcal{R}_k \equiv E\left\{\boldsymbol{\upsilon}_k \boldsymbol{\upsilon}_k^T\right\} = T_k^T R_k T_k$. Reference [7] shows that the eigenvectors of a real symmetric matrix are orthogonal. Therefore, using the results of §3.1.4, if T_k is chosen to be the matrix whose columns are the eigenvectors of R_k, then $\mathcal{R}_k$ is a diagonal matrix with elements given by the eigenvalues of R_k. Note that this decomposition has to be applied at each time instant; however, for many systems the measurement error process is *stationary* so that R_k is constant for all times, denoted simply by R. Therefore, in this case, the decomposition needs to be only performed once, which can significantly reduce the computational load. The Kalman gain and covariance update can now be performed using a sequential procedure, given by

$$\boxed{K_{i_k} = \frac{P^-_{i-1_k} \mathcal{H}^T_{i_k}}{\mathcal{H}_{i_k} P^-_{i-1_k} \mathcal{H}^T_{i_k} + \mathcal{R}_{i_k}}} \tag{5.79a}$$

$$\boxed{P^+_{i_k} = [I - K_{i_k} \mathcal{H}_{i_k}] P^+_{i-1_k}, \quad P^+_{0_k} = P^-_k} \tag{5.79b}$$

where i represents the i^{th} measurement, $\mathcal{R}_i$ is the i^{th} diagonal element of $\mathcal{R}$, and $\mathcal{H}_i$ is the i^{th} row of $\mathcal{H}$. The process continues until all m measurements are processed (i.e., $i = 1, 2, \ldots, m$), with $P^+_k = P^+_{m_k}$. The state update can now be computed using eqn. (5.30b):

$$\boxed{\hat{\mathbf{x}}^+_k = \hat{\mathbf{x}}^-_k + P^+_k \mathcal{H}^T_k \mathcal{R}^{-1}_k [\tilde{\mathbf{z}}_k - \mathcal{H}_k \hat{\mathbf{x}}^-_k]} \tag{5.80}$$

Note that the transformed measurement $\tilde{\mathbf{z}}_k$ is now used in the state update equation.

5.3.4 Steady-State Kalman Filter

The discrete Riccati equation in eqn. (5.54c) requires the propagation of an $n \times n$ matrix. Fortunately for time-invariant systems the error covariance P reaches a steady-state value very quickly. Therefore, a *constant* gain (K) in the filter can be pre-computed using the steady-state covariance, which can significantly reduce the computational burden. Although this approach is suboptimal in the strictest sense, the savings in computations compared to any loss in the estimated state quality makes the fixed-gain Kalman filter attractive in the design of many dynamical systems. The steady-state (autonomous) discrete-time Kalman filter is summarized in Table 5.2.

To determine the steady-state value for P we must solve the *discrete-time algebraic Riccati equation* in Table 5.2. The solution can be derived using the duality between estimation and optimal control theory (discussed in Chapter 8). The nonlinear Riccati equation can be processed using two sets of $n \times n$ matrices, given by

$$\boxed{P_k = S_k Z_k^{-1}} \tag{5.81}$$

Table 5.2: Discrete and Autonomous Linear Kalman Filter

Model	$\mathbf{x}_{k+1} = \Phi\,\mathbf{x}_k + \Gamma\,\mathbf{u}_k + \Upsilon\,\mathbf{w}_k, \quad \mathbf{w}_k \sim N(\mathbf{0}, Q)$ $\tilde{\mathbf{y}}_k = H\,\mathbf{x}_k + \mathbf{v}_k, \quad \mathbf{v}_k \sim N(\mathbf{0}, R)$
Initialize	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$
Gain	$K = P\,H^T[H\,P\,H^T + R]^{-1}$
Covariance	$P = \Phi\,P\,\Phi^T - \Phi\,P\,H^T[H\,P\,H^T + R]^{-1}H\,P\,\Phi^T + \Upsilon\,Q\,\Upsilon^T$
Estimate	$\hat{\mathbf{x}}_{k+1} = \Phi\,\hat{\mathbf{x}}_k + \Gamma\,\mathbf{u}_k + \Phi\,K\,[\tilde{\mathbf{y}}_k - H\,\hat{\mathbf{x}}_k]$

To determine linear equations for S_{k+1} and Z_{k+1} we first rewrite the discrete-time Riccati equation in eqn. (5.54c) using the matrix inversion lemma in eqn. (1.69), which yields

$$P_{k+1} = \Phi\,[\bar{H} + P_k^{-1}]^{-1}\Phi^T + \bar{Q} \tag{5.82}$$

where $\bar{H} \equiv H^T R^{-1} H$ and $\bar{Q} \equiv \Upsilon\,Q\Upsilon^T$. Factoring P_k and multiplying $\bar{Q}$ by an identity gives

$$P_{k+1} = \Phi\,P_k[\bar{H}\,P_k + I]^{-1}\Phi^T + \bar{Q}\,\Phi^{-T}\Phi^T \tag{5.83}$$

Rewriting eqn. (5.83) by factoring $[\bar{H}\,P_k + I]$ gives

$$P_{k+1} = \Big\{\Phi\,P_k + \bar{Q}\,\Phi^{-T}[\bar{H}\,P_k + I]\Big\}[\bar{H}\,P_k + I]^{-1}\Phi^T \tag{5.84}$$

Next collecting P_k terms gives

$$P_{k+1} = \Big\{[\Phi + \bar{Q}\,\Phi^{-T}\bar{H}]P_k + \bar{Q}\,\Phi^{-T}]\Big\}[\bar{H}\,P_k + I]^{-1}\Phi^T \tag{5.85}$$

Substituting eqn. (5.81) into eqn. (5.85) and factoring Z_k yields

$$P_{k+1} = \Big\{[\Phi + \bar{Q}\,\Phi^{-T}\bar{H}]S_k + \bar{Q}\,\Phi^{-T}Z_k]\Big\}Z_k^{-1}[\bar{H}\,S_k Z_k^{-1} + I]^{-1}\Phi^T \tag{5.86}$$

Finally, factoring Z_k^{-1} and Φ^T into the last inverse of eqn. (5.86) gives

$$P_{k+1} = \Big\{[\Phi + \bar{Q}\,\Phi^{-T}\bar{H}]S_k + \bar{Q}\,\Phi^{-T}Z_k]\Big\}[\Phi^{-T}Z_k + \Phi^{-T}\bar{H}\,S_k]^{-1} \tag{5.87}$$

Using a one-time step ahead of eqn. (5.81) yields the following relationship:

$$\begin{bmatrix} Z_{k+1} \\ S_{k+1} \end{bmatrix} = \mathcal{H}\begin{bmatrix} Z_k \\ S_k \end{bmatrix} \tag{5.88}$$

where the *Hamiltonian matrix* is defined as

$$\boxed{\mathcal{H} \equiv \begin{bmatrix} \Phi^{-T} & \Phi^{-T} H^T R^{-1} H \\ \Upsilon\, Q \Upsilon^T \Phi^{-T} & \Phi + \Upsilon\, Q \Upsilon^T \Phi^{-T} H^T R^{-1} H \end{bmatrix}} \tag{5.89}$$

We will now show that if λ is an eigenvalue of $\mathcal{H}$, then λ^{-1} is also an eigenvalue of $\mathcal{H}$ (i.e., $\mathcal{H}$ is a *symplectic* matrix[8]). The eigenvalues of $\mathcal{H}$ are determined by taking the determinant of the following equation and setting the resultant to zero:

$$\lambda I - \mathcal{H} = \begin{bmatrix} \lambda I - \Phi^{-T} & -\Phi^{-T} \bar{H} \\ -\bar{Q}\Phi^{-T} & \lambda I - \Phi - \bar{Q}\Phi^{-T}\bar{H} \end{bmatrix} \tag{5.90}$$

Next we multiply the right side of eqn. (5.90) by the following matrix:

$$\bar{H}_I \equiv \begin{bmatrix} I & -\bar{H} \\ 0 & I \end{bmatrix} \tag{5.91}$$

Since $\det(\bar{H}_I) = 1$ see Appendix A hen the determinant of eqn. (5.90) is given by

$$\det(\lambda I - \mathcal{H}) = \det \begin{bmatrix} \lambda I - \Phi^{-T} & -\lambda \bar{H} \\ -\bar{Q}\Phi^{-T} & \lambda I - \Phi \end{bmatrix} = 0 \tag{5.92}$$

Next we use the following identity for square matrices A, B, C, and D:

$$\det \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \det(D)\det(A - B\, D^{-1} C) \tag{5.93}$$

assuming that D^{-1} exists. This leads to

$$\det(\lambda I - \Phi)\det\left[(\lambda\Phi^T - I) - \bar{H}(I - \lambda^{-1}\Phi)^{-1}\bar{Q}\right] = 0 \tag{5.94}$$

where $\det(A\,B) = \det(A)\det(B)$ was used to factor out the term Φ^{-T}. Next, we factor the term $(\lambda\Phi^T - I)$ from the second term and multiply both sides of the resultant equation by λ^{-n}, where n is the order of Φ, to find

$$\alpha(\lambda)\alpha(\lambda^{-1})\det\left[I + (\lambda\Phi^T - I)^{-1}\bar{H}(\lambda^{-1}\Phi - I)^{-1}\bar{Q}\right] = 0 \tag{5.95}$$

where $\alpha(\lambda) \equiv \det(\lambda I - \Phi)$. Since both $\bar{H}$ and $\bar{Q}$ are symmetric matrices, they can be factored into $\bar{H} = \Xi^T\,\Xi$ and $\bar{Q} = \Theta^T\Theta$. Then using the identity $\det(I + A\,B) = \det(I + B\,A)$, with $A = (\lambda\Phi^T - I)^{-1}\,\Xi^T$, gives

$$\alpha(\lambda)\alpha(\lambda^{-1})\det\left[I + \Xi(\lambda^{-1}\Phi - I)^{-1}\Theta^T\Theta\,(\lambda\Phi^T - I)^{-1}\,\Xi^T\right] = 0 \tag{5.96}$$

Therefore, if λ is replaced by λ^{-1}, the result in eqn. (5.96) remains unchanged since the determinant of a matrix is equal to the determinant of its transpose. Thus the eigenvalues can be arranged in a diagonal matrix given by

$$\mathcal{H}_\Lambda = \begin{bmatrix} \Lambda & 0 \\ 0 & \Lambda^{-1} \end{bmatrix} \tag{5.97}$$

where Λ is a diagonal matrix of the n eigenvalues outside of the unit circle. Assuming that the eigenvalues are distinct, we can perform a linear state transformation, as shown in §3.1.4, such that

$$\mathcal{H}_\Lambda = W^{-1}\mathcal{H}\,W \tag{5.98}$$

where W is the matrix of eigenvectors, which can be represented in block form as

$$W = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \tag{5.99}$$

At steady-state the unstable eigenvalues (Λ) will dominate the response of P_k. Using only the unstable eigenvalues we can partition eqn. (5.98) as

$$\begin{bmatrix} W_{11} \\ W_{21} \end{bmatrix} \Lambda = \mathcal{H} \begin{bmatrix} W_{11} \\ W_{21} \end{bmatrix} \tag{5.100}$$

If we make the analogy that $Z \to W_{11}$ and $S \to W_{21}$ from eqn. (5.88), then the steady-state solution for P with $k \to k+1$ is given by

$$\boxed{P = [W_{21}\Lambda][W_{11}\Lambda]^{-1} = W_{21}W_{11}^{-1}} \tag{5.101}$$

Therefore, the gain K in Table 5.2 can be computed off-line and remains constant. This can significantly reduce the on-board computational load on a computer.

Vaughan[9] has shown that a nonrecursive solution for P_k is given by

$$P_k = [W_{21} + W_{22}Y_k][W_{11} + W_{12}Y_k]^{-1} \tag{5.102}$$

where

$$Y_k = \Lambda^{-k} X \Lambda^{-k} \tag{5.103a}$$

$$X = -[W_{22} - P_0 W_{12}]^{-1}[W_{21} - P_0 W_{11}] \tag{5.103b}$$

The steady-state solution for P can be found by letting $k \to \infty$, which leads directly to eqn. (5.101).

5.3.5 Correlated Measurement and Process Noise

The derivations thus far have assumed that the measurement error is uncorrelated with the process noise (state error). In this section the correlated Kalman filter is derived. This correlation can be written mathematically by

$$\boxed{E\left\{\mathbf{w}_{k-1}\mathbf{v}_k^T\right\} = S_k} \tag{5.104}$$

Before proceeding, we must first explain why we wish to investigate the correlation between $\mathbf{w}_{k-1}$ and $\mathbf{v}_k$, not between $\mathbf{w}_k$ and $\mathbf{v}_k$. This is mainly due to the fact that the measurement at time t_k will be dependent on the state, deterministic input, and process noise at time t_{k-1}, as shown by eqn. (5.27). This is extremely useful for the correspondence between a sampled continuous-time system, since it represents correlation between the process noise over a sample period and the measurement at the end of the period.[5] Note that S_k is not a symmetric matrix in this case.

Equations (5.33) and (5.37) will be used to derive the filter equations. Clearly, when eqn. (5.33) is substituted into eqn. (5.37) at time t_k, the covariance update P_k^- in eqn. (5.35) remains unchanged since $E\left\{\mathbf{w}_k\mathbf{v}_k^T\right\} = E\left\{\mathbf{v}_k\mathbf{w}_k^T\right\} = 0$ from the assumptions in this section. However, the terms $E\left\{\tilde{\mathbf{x}}_k^-\mathbf{v}_k^T\right\}$ and $E\left\{\mathbf{v}_k\tilde{\mathbf{x}}_k^{-T}\right\}$ in eqn. (5.38) are no longer zero in this case. Performing the expectation for the previous expression gives

$$\begin{aligned} E\left\{\tilde{\mathbf{x}}_k^-\mathbf{v}_k^T\right\} &= E\left\{(\Phi_{k-1}\tilde{\mathbf{x}}_{k-1}^+ - \Upsilon_{k-1}\mathbf{w}_{k-1})\,\mathbf{v}_k^T\right\} \\ &= -\Upsilon_{k-1}S_k \end{aligned} \tag{5.105}$$

This is due to the fact that $\tilde{\mathbf{x}}_{k-1}^+$ is uncorrelated with $\mathbf{v}_k$. Therefore eqn. (5.38) becomes

$$\begin{aligned} P_k^+ &= [I - K_kH_k]P_k^-[I - K_kH_k]^T + K_kR_kK_k^T \\ &\quad - [I - K_kH_k]\Upsilon_{k-1}S_kK_k^T - K_kS_k^T\Upsilon_{k-1}^T[I - K_kH_k]^T \end{aligned} \tag{5.106}$$

This expression is valid for any gain K_k. To determine this gain we again minimize the trace of P_k^+, which leads to

$$\boxed{K_k = [P_k^-H_k^T + \Upsilon_{k-1}S_k][H_kP_k^-H_k^T + R_k + H_k\Upsilon_{k-1}S_k + S_k^T\Upsilon_{k-1}^TH_k^T]^{-1}} \tag{5.107}$$

Note that if $S_k = 0$ then the gain reduces to the standard form given in eqn. (5.42). Substituting eqn. (5.107) into eqn. (5.106), after some algebraic manipulations, yields

$$\boxed{P_k^+ = [I - K_kH_k]P_k^- - K_kS_k^T\Upsilon_{k-1}^T} \tag{5.108}$$

This again reduces to the standard form of the covariance update in eqn. (5.44) if $S_k = 0$. A summary of the correlated discrete-time Kalman filter is given in Table 5.3

An excellent example of the usefulness of the correlated Kalman filter is an aircraft flying through a field of random turbulence.[4] The effect of turbulence in the aircraft's acceleration are complex, but can easily be modelled as process noise on $\mathbf{w}_{k-1}$. Since any sensor mounted on an aircraft is also corrupted by turbulence, the measurement error $\mathbf{v}_k$ is correlated with the process noise $\mathbf{w}_{k-1}$. Hence, the filter formulation presented in this section can be used directly to estimate aircraft state quantities in the face of turbulence disturbances.

Table 5.3: Correlated Discrete-Time Linear Kalman Filter

Model	$\mathbf{x}_{k+1} = \Phi_k \mathbf{x}_k + \Gamma_k \mathbf{u}_k + \Upsilon_k \mathbf{w}_k, \quad \mathbf{w}_k \sim N(\mathbf{0}, Q_k)$ $\tilde{\mathbf{y}}_k = H_k \mathbf{x}_k + \mathbf{v}_k, \quad \mathbf{v}_k \sim N(\mathbf{0}, R_k)$ $E\left\{\mathbf{w}_{k-1}\mathbf{v}_k^T\right\} = S_k$
Initialize	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$ $P_0 = E\left\{\tilde{\mathbf{x}}(t_0)\,\tilde{\mathbf{x}}^T(t_0)\right\}$
Gain	$K_k = [P_k^- H_k^T + \Upsilon_{k-1} S_k]$ $\times[H_k P_k^- H_k^T + R_k + H_k \Upsilon_{k-1} S_k + S_k^T \Upsilon_{k-1}^T H_k^T]^{-1}$
Update	$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k[\tilde{\mathbf{y}}_k - H_k \hat{\mathbf{x}}_k^-]$ $P_k^+ = [I - K_k H_k] P_k^- - K_k S_k^T \Upsilon_{k-1}^T$
Propagation	$\hat{\mathbf{x}}_{k+1}^- = \Phi_k \hat{\mathbf{x}}_k^+ + \Gamma_k \mathbf{u}_k$ $P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T$

5.3.6 Orthogonality Principle

One of the interesting aspects of the Kalman filter is the orthogonality of the estimate and its error,[1] which is stated mathematically as

$$\boxed{E\left\{\hat{\mathbf{x}}_k^+ \tilde{\mathbf{x}}_k^{+T}\right\} = 0} \tag{5.109}$$

This states that the estimate is uncorrelated from its error. To prove eqn. (5.109) set the time step to $k = 1$, and substitute eqn. (5.33) into eqn. (5.37), which gives

$$\tilde{\mathbf{x}}_1^+ = (\Phi_0 - K_1 H_1 \Phi_0)\,\tilde{\mathbf{x}}_0^+ + (K_1 H_1 - I)\,\Upsilon_0 \mathbf{w}_0 + K_1 \mathbf{v}_1 \tag{5.110}$$

Next, substituting eqn. (5.27a) into eqn. (5.27b), and then substituting the resultant into eqn. (5.30b) leads to the following state estimate update:

$$\hat{\mathbf{x}}_1^+ = \Phi_0 \hat{\mathbf{x}}_0^+ + \Gamma_0 \mathbf{u}_0 + K_1\left(H_1 \Upsilon_0 \mathbf{w}_0 + \mathbf{v}_1 - H_1 \Phi_0 \tilde{\mathbf{x}}_0^+\right) \tag{5.111}$$

Since the initial conditions are uncorrelated, then $E\left\{\hat{\mathbf{x}}_0^+ \tilde{\mathbf{x}}_0^{+T}\right\} = 0$, and we have

$$\begin{aligned} E\left\{\hat{\mathbf{x}}_1^+ \tilde{\mathbf{x}}_1^{+T}\right\} &= K_1 H_1 \Upsilon_0 Q_0 \Upsilon_0^T \left(H_1^T K_1^T - I\right) \\ &\quad + K_1 H_1 \Phi_0 P_0^+ \left(\Phi_0 H_1^T K_1^T - \Phi_0^T\right) + K_1 R_1 K_1^T \end{aligned} \tag{5.112}$$

Collecting terms yields

$$E\left\{\hat{\mathbf{x}}_1^+\tilde{\mathbf{x}}_1^{+T}\right\} = -K_1H_1\left(\Phi_0P_0^+\Phi_0^T + \Upsilon_0Q_0\Upsilon_0^T\right) + K_1H_1\left(\Phi_0P_0^+\Phi_0^T + \Upsilon_0Q_0\Upsilon_0^T\right)H_1^TK_1^T + K_1R_1K_1^T \tag{5.113}$$

Using eqn. (5.35) in eqn. (5.113) gives

$$E\left\{\hat{\mathbf{x}}_1^+\tilde{\mathbf{x}}_1^{+T}\right\} = K_1H_1P_1^-\left(H_1^TK_1^T - I\right) + K_1R_1K_1^T \tag{5.114}$$

Next, using the definition of P_1^+ from eqn. (5.44) in eqn. (5.114) gives

$$E\left\{\hat{\mathbf{x}}_1^+\tilde{\mathbf{x}}_1^{+T}\right\} = -K_1H_1P_1^+ + K_1R_1K_1^T \tag{5.115}$$

Then substituting the gain K_1 from eqn. (5.47) into eqn. (5.115) yields

$$E\left\{\hat{\mathbf{x}}_1^+\tilde{\mathbf{x}}_1^{+T}\right\} = -P_1^+H_1^TR^{-1}H_1P_1^+ + P_1^+H_1^TR^{-1}H_1P_1^+ = 0 \tag{5.116}$$

The process is then repeated for the $k = 2$ case, and by induction the identity in eqn. (5.109) is proven. At first glance the Orthogonality Principle does not seem to have any practical value, but as we shall see it is extremely important in the derivation of the linear quadratic-Gaussian controller of §8.6.

Example 5.2: In this simple example the discrete-time Kalman filter is used to estimate a scalar state for a time-invariant system, whose truth model follows

$$\begin{aligned} x_{k+1} &= \phi\, x_k + \gamma\, u_k + w_k \\ \tilde{y}_k &= h\, x_k + v_k \end{aligned}$$

where the random errors are assumed to be stationary noise processes with $w_k \sim N(0,q)$ and $v_k \sim N(0,r)$. Since the filter dynamics converge rapidly in this case we will use the steady-state Kalman filter, given in Table 5.2. The steady-state covariance equation gives the following second-order polynomial equation:

$$h^2p^2 + (r - \phi^2 r - h^2 q)\,p - q\,r = 0$$

The closed-form solution for even this simple system is difficult to intuitively visualize; however, some simple forms can be given for two special cases. Consider the perfect-measurement case where $r = 0$, which simply yields $p = q$. Then the gain K in Table 5.2 is simply given by $1/h$, and the state estimate is given by

$$\hat{x}_{k+1} = \frac{\phi}{h}\,\tilde{y}_k + \gamma\, u_k$$

Note that the current state estimate $\hat{x}_{k+1}$ does not depend on the previous state estimate $\hat{x}_k$ in this case. This is due to the fact that with $r = 0$, the measurements are assumed perfect and the dynamics model can be ignored, which intuitively makes sense. Next, we consider the perfect-model case when $q = 0$, which simply yields $p = 0$. The gain is zero in this case and the state estimate is given by

$$\hat{x}_{k+1} = \phi\,\hat{x}_k + \gamma\,u_k$$

In this case the measurement is completely ignored, which again intuitively makes sense since the model is perfect with no errors.

Example 5.3: In this example the single axis attitude estimation problem using angle-attitude measurements and rate information from gyros is shown. We will demonstrate the power of the Kalman filter to update both the attitude-angle estimates and gyro drift rate. Angle measurements are corrupted with noise, which can be filtered by using rate information. However, all gyros inherently drift over time, which degrades the rate information over time. Two error sources are generally present in gyros.[10] The first is a short-term component of instability referred to as *random drift*, and the second is a random walk component referred to as *drift rate ramp*. The effects of both of these noise sources on the uncertainty of the gyro outputs can be compensated using a Kalman filter with attitude measurements. The attitude rate $\dot{\theta}$ is assumed to be related to the gyro output $\tilde{\omega}$ by

$$\dot{\theta} = \tilde{\omega} - \beta - \eta_v$$

where β is the gyro drift rate, and η_v is a zero-mean Gaussian white-noise process with variance given by σ_v^2. The drift rate is modelled by a random walk process, given by

$$\dot{\beta} = \eta_u$$

where η_u is a zero-mean Gaussian white-noise process with variance given by σ_u^2. The parameters σ_v^2 and σ_u^2 can be experimentally obtained using frequency response data from the gyro outputs. The estimated states clearly follow

$$\dot{\hat{\theta}} = \tilde{\omega} - \hat{\beta}$$
$$\dot{\hat{\beta}} = 0$$

Assuming a constant sampling interval in the gyro output, the discrete-time error propagation is given by[11]

$$\begin{bmatrix} \theta_{k+1} - \hat{\theta}_{k+1} \\ \beta_{k+1} - \hat{\beta}_{k+1} \end{bmatrix} = \Phi \begin{bmatrix} \theta_k - \hat{\theta}_k \\ \beta_k - \hat{\beta}_k \end{bmatrix} + \begin{bmatrix} p_k \\ q_k \end{bmatrix}$$

where the state transition matrix is given by

$$\Phi = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix}$$

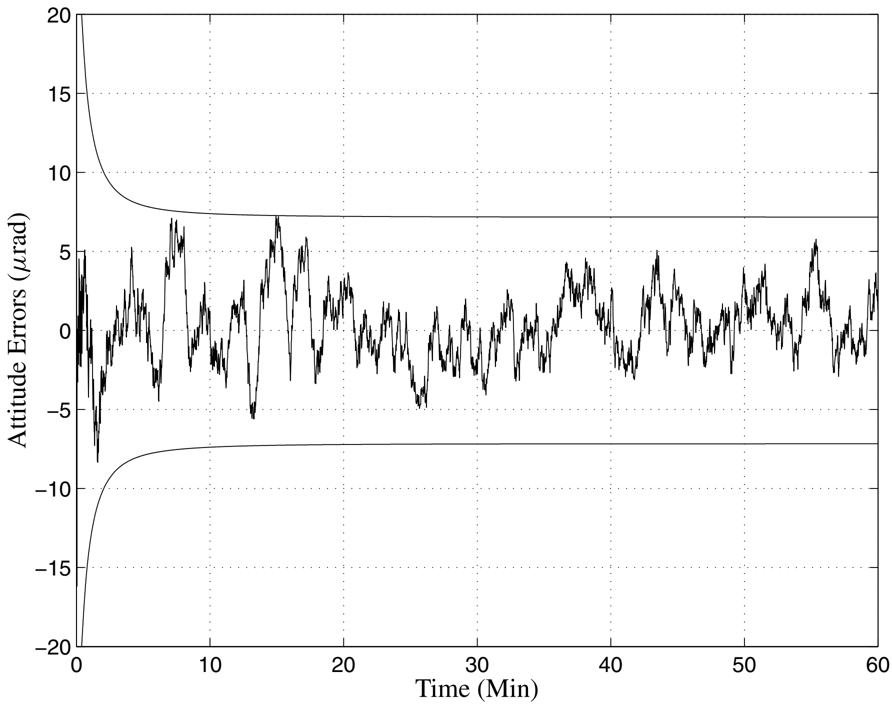

Figure 5.3: Kalman Filter Attitude Error and Bounds

where $\Delta t = t_{k+1} - t_k$ is the sampling interval, and

$$p_k = \int_{t_k}^{t_{k+1}} \left[-\eta_v(\tau) - (t_{k+1} - \tau)\eta_u(\tau) \right] d\tau$$

$$q_k = \int_{t_k}^{t_{k+1}} \eta_u(\tau)\, d\tau$$

The process noise covariance matrix Q can be computed as

$$Q = \begin{bmatrix} E\left\{p_k^2\right\} & E\left\{p_k q_k\right\} \\ E\left\{q_k p_k\right\} & E\left\{q_k^2\right\} \end{bmatrix}$$

$$= \begin{bmatrix} \sigma_v^2 \Delta t + \frac{1}{3}\sigma_u^2 \Delta t^3 & -\frac{1}{2}\sigma_u^2 \Delta t^2 \\ -\frac{1}{2}\sigma_u^2 \Delta t^2 & \sigma_u^2 \Delta t \end{bmatrix}$$

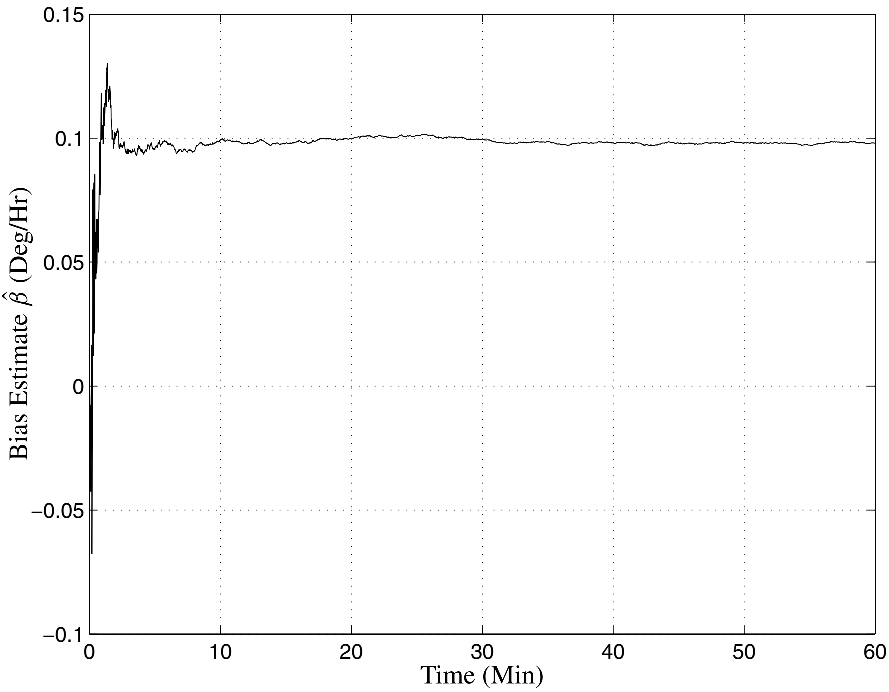

Figure 5.4: Kalman Filter Gyro Bias Estimate

which is independent of k since the sampling interval is assumed to be constant. The attitude-angle measurement is modelled by

$$\tilde{y}_k = \theta_k + v_k$$

where v_k is a zero-mean Gaussian white-noise process with variance given by $R = \sigma_n^2$. The discrete-time system used in the Kalman filter can now be written as

$$\begin{aligned}\mathbf{x}_{k+1} &= \Phi\,\mathbf{x}_k + \Gamma\,\tilde{\omega} + \mathbf{w}_k \\ \tilde{y}_k &= H\,\mathbf{x}_k + v_k\end{aligned}$$

where $\mathbf{x} = \left[\theta\ \beta\right]^T$, $\Gamma = \left[\Delta t\ 0\right]^T$, $H = \left[1\ 0\right]$, and $E\left\{\mathbf{w}_k \mathbf{w}_k^T\right\} = Q$. We should note that the input to this system involves a measurement ($\tilde{\omega}$), which is counterintuitive but valid in the Kalman filter form and poses no problems in the estimation process. The discrete-time Kalman filter shown in Table 5.1 can now be applied to this system. Synthetic measurements are created using a true constant angle-rate given by $\dot{\theta} = 0.0011$ rad/sec and a sampling rate of 1 second. The noise parameters are given by $\sigma_n = 17 \times 10^{-6}$ rad, $\sigma_u = \sqrt{10} \times 10^{-10}$ rad/sec$^{3/2}$, and $\sigma_v = \sqrt{10} \times 10^{-7}$ rad/sec$^{1/2}$. The initial bias β_0 is given as 0.1 deg/hr, and the initial covariance matrix is set to $P_0 = \text{diag}\left[1 \times 10^{-4}\ \ 1 \times 10^{-12}\right]$. A plot of the attitude-angle error and 3σ bounds is shown in Figure 5.3. Clearly, the Kalman filter provides filtered estimates and the theoretical 3σ bounds do indeed bound the errors. A steady-state Kalman filter using the algebraic Riccati equation in Table 5.2 can also be used, which yields nearly identical results as the time-varying case. At steady-state the theoretical 3σ bound is given by 7.18 μrad. A plot of the estimated bias is shown in Figure 5.4. Clearly, the Kalman filter estimates the bias well. This example demonstrates the usefulness of the Kalman filter by fusing two sensors to produce estimates that are better than each sensor alone.

5.4 The Continuous-Time Kalman Filter

In this section the Kalman filter is derived using continuous-time models and measurements. The continuous-time Kalman filter is not widely used in practice due to the extensive use of digital computers in today's time; however, the derivation does provide some unique perspectives that are especially useful for small sampling intervals (i.e., well below Nyquist's limit). Two approaches are shown, which yield the same Kalman filter structure. The first uses the continuous-time structure directly, while the second uses the discrete-time formulation described in §5.4.1 to derive the corresponding continuous-time form.

5.4.1 Kalman Filter Derivation in Continuous Time

In this section the Kalman filter is derived directly from continuous-time models and measurements. Consider the following truth model:

$$\boxed{\begin{aligned} \dot{\mathbf{x}}(t) &= F(t)\,\mathbf{x}(t) + B(t)\,\mathbf{u}(t) + G(t)\,\mathbf{w}(t) \qquad (5.117a)\\ \tilde{\mathbf{y}}(t) &= H(t)\,\mathbf{x}(t) + \mathbf{v}(t) \qquad (5.117b) \end{aligned}}$$

where $\mathbf{w}(t)$ and $\mathbf{v}(t)$ are zero-mean Gaussian noise processes with covariances given by

$$E\left\{\mathbf{w}(t)\,\mathbf{w}^T(\tau)\right\} = Q(t)\,\delta(t-\tau) \tag{5.118a}$$

$$E\left\{\mathbf{v}(t)\,\mathbf{v}^T(\tau)\right\} = R(t)\,\delta(t-\tau) \tag{5.118b}$$

$$E\left\{\mathbf{v}(t)\,\mathbf{w}^T(\tau)\right\} = 0 \tag{5.118c}$$

Equation (5.118c) implies that $\mathbf{v}(t)$ and $\mathbf{w}(t)$ are uncorrelated. Also, the control input $\mathbf{u}(t)$ is a deterministic quantity. The Kalman filter structure for the state and output estimate is given by

$$\boxed{\begin{aligned}\dot{\hat{\mathbf{x}}}(t) &= F(t)\,\hat{\mathbf{x}}(t) + B(t)\,\mathbf{u}(t) + K(t)[\tilde{\mathbf{y}}(t) - H(t)\,\hat{\mathbf{x}}(t)] \\ \hat{\mathbf{y}}(t) &= H(t)\,\hat{\mathbf{x}}(t)\end{aligned}} \tag{5.119a, 5.119b}$$

Defining the state error $\tilde{\mathbf{x}}(t) = \hat{\mathbf{x}}(t) - \mathbf{x}(t)$ and using eqns. (5.117) and (5.119) leads to

$$\dot{\tilde{\mathbf{x}}}(t) = E(t)\,\tilde{\mathbf{x}}(t) + \mathbf{z}(t) \tag{5.120}$$

where

$$E(t) = F(t) - K(t)\,H(t) \tag{5.121}$$

$$\mathbf{z}(t) = -G(t)\,\mathbf{w}(t) + K(t)\,\mathbf{v}(t) \tag{5.122}$$

Note that $\mathbf{u}(t)$ cancels in the error state. Since $\mathbf{v}(t)$ and $\mathbf{w}(t)$ are uncorrelated, we have

$$E\left\{\mathbf{z}(t)\,\mathbf{z}^T(\tau)\right\} = \left[G(t)\,Q(t)\,G^T(t) + K(t)\,R(t)\,K^T(t)\right]\delta(t-\tau) \tag{5.123}$$

Using the matrix exponential solution in eqn. (3.53) gives

$$\tilde{\mathbf{x}}(t) = \Phi(t,t_0)\,\tilde{\mathbf{x}}(t_0) + \int_{t_0}^{t} \Phi(t,\tau)\,\mathbf{z}(\tau)\,d\tau \tag{5.124}$$

The state error-covariance is defined by

$$P(t) \equiv E\left\{\tilde{\mathbf{x}}(t)\,\tilde{\mathbf{x}}^T(t)\right\} \tag{5.125}$$

Substituting eqn. (5.124) into eqn. (5.125), assuming that $\mathbf{z}(t)$ and $\tilde{\mathbf{x}}(t_0)$ are uncorrelated, leads to

$$\begin{aligned} P(t) &= \Phi(t,t_0)\,P(t_0)\,\Phi^T(t,t_0) \\ &\quad + \int_{t_0}^{t} \Phi(t,\tau)\left[G(\tau)\,Q(\tau)\,G^T(\tau) + K(\tau)\,R(\tau)\,K^T(\tau)\right]\Phi^T(t,\tau)\,d\tau \end{aligned} \tag{5.126}$$

Taking the time derivative of eqn. (5.126) gives

$$\begin{aligned} \dot{P}(t) &= \frac{\partial\Phi(t,t_0)}{\partial t}P(t_0)\Phi^T(t,t_0) + \Phi(t,t_0)P(t_0)\frac{\partial\Phi^T(t,t_0)}{\partial t} \\ &\quad + \int_{t_0}^{t} \frac{\partial\Phi(t,\tau)}{\partial t}\left[G(\tau)\,Q(\tau)\,G^T(\tau) + K(\tau)\,R(\tau)\,K^T(\tau)\right]\Phi^T(t,\tau)\,d\tau \\ &\quad + \int_{t_0}^{t} \Phi(t,\tau)\left[G(\tau)\,Q(\tau)\,G^T(\tau) + K(\tau)\,R(\tau)\,K^T(\tau)\right]\frac{\partial\Phi^T(t,\tau)}{\partial t}\,d\tau \\ &\quad + \Phi(t,t)\left[G(t)\,Q(t)\,G^T(t) + K(t)\,R(t)\,K^T(t)\right]\Phi^T(t,t) \end{aligned} \tag{5.127}$$

Using the properties of the matrix exponential in eqns. (3.17a) and (3.19) leads to

$$\begin{aligned}
\dot{P}(t) &= E(t)\,\Phi(t,t_0)\,P(t_0)\Phi^T(t,t_0) + \Phi(t,t_0)P(t_0)\,\Phi^T(t,t_0)\,E^T(t) \\
&\quad + E(t)\int_{t_0}^{t} \Phi(t,\tau)\left[G(\tau)\,Q(\tau)\,G^T(\tau) + K(\tau)\,R(\tau)\,K^T(\tau)\right]\Phi^T(t,\tau)\,d\tau \\
&\quad + \int_{t_0}^{t} \Phi(t,\tau)\left[G(\tau)\,Q(\tau)\,G^T(\tau) + K(\tau)\,R(\tau)\,K^T(\tau)\right]\Phi^T(t,\tau)\,d\tau\,E^T(t) \\
&\quad + G(t)\,Q(t)\,G^T(t) + K(t)\,R(t)\,K^T(t)
\end{aligned} \tag{5.128}$$

Using eqns. (5.121) and (5.126) in eqn. (5.128) simplifies the expression for $\dot{P}(t)$ significantly to

$$\begin{aligned}
\dot{P}(t) &= [F(t) - K(t)\,H(t)]\,P(t) + P(t)\,[F(t) - K(t)\,H(t)]^T \\
&\quad + G(t)\,Q(t)\,G^T(t) + K(t)\,R(t)\,K^T(t)
\end{aligned} \tag{5.129}$$

In order to determine the gain $K(t)$ we minimize the trace of $\dot{P}(t)$:

$$\text{minimize} \quad J[K(t)] = \text{Tr}[\dot{P}(t)] \tag{5.130}$$

The necessary conditions lead to

$$\frac{\partial J}{\partial K(t)} = 0 = 2K(t)\,R(t) - 2P(t)\,H^T(t) \tag{5.131}$$

Solving eqn. (5.131) for $K(t)$ gives

$$\boxed{K(t) = P(t)\,H^T(t)\,R^{-1}(t)} \tag{5.132}$$

Note the similarity of the gain $K(t)$ to the discrete-time case given in eqn. (5.47). Substituting eqn. (5.132) into eqn. (5.129) gives

$$\boxed{\begin{aligned}
\dot{P}(t) &= F(t)\,P(t) + P(t)\,F^T(t) \\
&\quad - P(t)\,H^T(t)\,R^{-1}(t)H(t)\,P(t) + G(t)\,Q(t)\,G^T(t)
\end{aligned}} \tag{5.133}$$

Equation (5.133) is known as the *continuous Riccati equation*.

A summary of the continuous-time Kalman filter is given in Table 5.4 First, initial conditions for the state and error covariances are given. Then, the gain $K(t)$ is computed using eqn. (5.132) with the initial covariance value. Next, the covariance in eqn. (5.133) and state estimate in eqn. (5.119a) are numerically integrated forward in time using the continuous-time measurement $\tilde{\mathbf{y}}(t)$ and known input $\mathbf{u}(t)$. The integration of the state estimate and covariance continues until the final measurement time is reached.

Table 5.4: Continuous-Time Linear Kalman Filter

Model	$\dot{\mathbf{x}}(t) = F(t)\,\mathbf{x}(t) + B(t)\,\mathbf{u}(t) + G(t)\,\mathbf{w}(t),\ \mathbf{w}(t) \sim N(\mathbf{0}, Q(t))$ $\tilde{\mathbf{y}}(t) = H(t)\,\mathbf{x}(t) + \mathbf{v}(t),\ \mathbf{v}(t) \sim N(\mathbf{0}, R(t))$
Initialize	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$ $P_0 = E\left\{\tilde{\mathbf{x}}(t_0)\,\tilde{\mathbf{x}}^T(t_0)\right\}$
Gain	$K(t) = P(t)\,H^T(t)\,R^{-1}(t)$
Covariance	$\dot{P}(t) = F(t)\,P(t) + P(t)\,F^T(t)$ $-P(t)\,H^T(t)\,R^{-1}(t)H(t)\,P(t) + G(t)\,Q(t)\,G^T(t)$
Estimate	$\dot{\hat{\mathbf{x}}}(t) = F(t)\,\hat{\mathbf{x}}(t) + B(t)\,\mathbf{u}(t)$ $+K(t)[\tilde{\mathbf{y}}(t) - H(t)\,\hat{\mathbf{x}}(t)]$

5.4.2 Kalman Filter Derivation from Discrete Time

The continuous-time Kalman filter can also be derived from the discrete-time version of §5.4.1. We must first find relationships between the discrete-time covariance matrices, Q_k and R_k, and continuous-time covariance matrices, $Q(t)$ and $R(t)$. From eqn. (5.118a) and from the theory of discrete-time systems in §3.5 we can write

$$\begin{aligned}
&\Upsilon_k E\left\{\mathbf{w}_k \mathbf{w}_k^T\right\} \Upsilon_k^T = \Upsilon_k Q_k \Upsilon_k^T \\
&= E\left\{\left[\int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau)\, G(\tau)\, \mathbf{w}(\tau)\, d\tau\right]\left[\int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \varsigma)\, G(\varsigma)\, \mathbf{w}(\varsigma)\, d\varsigma\right]^T\right\} \\
&= \int_{t_k}^{t_{k+1}}\int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau)\, G(\tau)\, E\left\{\mathbf{w}(\tau)\, \mathbf{w}^T(\varsigma)\right\} G^T(\varsigma)\Phi^T(t_{k+1}, \varsigma)\, d\tau\, d\varsigma
\end{aligned} \tag{5.134}$$

Substituting eqn. (5.118a) into eqn. (5.134) and using the property of the Dirac delta function leads to

$$\Upsilon_k Q_k \Upsilon_k^T = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau)\, G(\tau)\, Q(\tau) G^T(\tau)\Phi^T(t_{k+1}, \tau)\, d\tau \tag{5.135}$$

The integral in eqn. (5.135) is difficult to evaluate even for simple systems. However, we are only interested in the first-order terms, since in the limit as $\Delta t \to 0$ higher-order terms vanish. Therefore, for small Δt we have $\Phi \approx (I + \Delta t\, F)$, and integrating over the small Δt simply yields

$$\boxed{\Upsilon_k Q_k \Upsilon_k^T = \Delta t\, G(t)\, Q(t)\, G^T(t)} \tag{5.136}$$

where eqn. (5.118a) has been used, and terms of order Δt^2 and higher have been dropped. We should note here that the matrix Q_k is a covariance matrix; however, the matrix $Q(t)$ is a *spectral density matrix*.[1, 12] Multiplying $Q(t)$ by the delta function converts it into a covariance matrix.

The integral in eqn. (5.135) may be difficult to evaluate for complex systems. Fortunately, a numerical solution is given by van Loan[13, 14] for fixed-parameter systems, which includes a constant sampling interval and time invariant state and covariance matrices. First, the following $2n \times 2n$ matrix is formed:

$$\mathcal{A} = \begin{bmatrix} -F & G\,Q\,G^T \\ 0 & F^T \end{bmatrix} \Delta t \tag{5.137}$$

where Δt is the constant sampling interval, F is the constant continuous-time state matrix, and Q is the constant continuous-time process noise covariance. Then, the matrix exponential of eqn. (5.137) is computed:

$$\mathcal{B} = e^{\mathcal{A}} \equiv \begin{bmatrix} \mathcal{B}_{11} & \mathcal{B}_{12} \\ 0 & \mathcal{B}_{22} \end{bmatrix} = \begin{bmatrix} \mathcal{B}_{11} & \Phi^{-1}\mathcal{Q} \\ 0 & \Phi^T \end{bmatrix} \tag{5.138}$$

where Φ is the state transition matrix of F and $\mathcal{Q} = \Upsilon\, Q_k \Upsilon^T$ (note, this matrix is constant, but we maintain the subscript k in Q_k to distinguish Q_k from the continuous-time equivalent). An efficient numerical solution of eqn. (5.138) is given by using the series approach in eqn. (3.25). The state transition matrix is then given by

$$\boxed{\Phi = \mathcal{B}_{22}^T} \tag{5.139}$$

Also, the discrete-time process noise covariance is given by

$$\boxed{\mathcal{Q} = \Phi\, \mathcal{B}_{12}} \tag{5.140}$$

If the sampling interval is "small" enough, then eqn. (5.136) is a good approximation for the solution given by eqn. (5.140).

The relationship between the discrete measurement covariance and continuous measurement covariance is not as obvious as the process noise covariance case. Consider the following linear model:

$$\tilde{y}_k = x + v_k \tag{5.141}$$

where an estimate of x is desired. Suppose that the time interval Δt is broken into equal samples, denoted by δ. Using the principles of Chapter 1, the estimate of x, denoted by $\hat{x}$ for m measurement samples over the interval Δt, is given by

$$\hat{x} = \frac{1}{m}\sum_{j=1}^{m} \tilde{y}_j \tag{5.142}$$

The relationship between the discrete-time process v_k and the continuous-time process must surely involve the sampling interval. We consider the following relationship:

$$E\left\{v_k v_j^T\right\} = \begin{cases} 0 & k \neq j \\ \delta^d R & k = j \end{cases} \tag{5.143}$$

for some value of d. Then the estimate error-variance is given by

$$E\left\{(x-\hat{x})^2\right\} = \frac{\delta^d R}{m} \tag{5.144}$$

The limit $m \to \infty$, $\delta \to 0$, and $m\delta \to \Delta t$ gives

$$E\left\{(x-\hat{x})^2\right\} = \begin{cases} 0 & d < -1 \\ \infty & d > -1 \\ \dfrac{R}{\Delta t} & d = -1 \end{cases} \tag{5.145}$$

Therefore, if the continuous model $\tilde{y}(t) = x + v(t)$ is to be meaningful in the sense that the error-variance is nonzero but finite, we must choose $d = -1$.[15] Toward this end in the sampling process, the continuous-time measurement process must be averaged over the sampling interval Δt in order to determine the equivalent discrete sample (where **x** is approximated as a constant over the interval).[14] Then we have

$$\begin{aligned} \tilde{\mathbf{y}}_k &= \frac{1}{\Delta t}\int_{t_k}^{t_{k+1}} \tilde{\mathbf{y}}(t)\, dt = \frac{1}{\Delta t}\int_{t_k}^{t_{k+1}} [H(t)\mathbf{x}(t) + \mathbf{v}(t)]\, dt \\ &\approx H_k \mathbf{x}_k + \frac{1}{\Delta t}\int_{t_k}^{t_{k+1}} \mathbf{v}(t)\, dt \end{aligned} \tag{5.146}$$

Therefore, the discrete-to-continuous equivalence can be found by solving the following equation:

$$E\left\{\mathbf{v}_k \mathbf{v}_k^T\right\} \equiv R_k = \frac{1}{\Delta t^2}\int_{t_k}^{t_{k+1}}\int_{t_k}^{t_{k+1}} E\left\{\mathbf{v}(\tau)\,\mathbf{v}^T(\varsigma)\right\}\, d\tau\, d\varsigma \tag{5.147}$$

Substituting eqn. (5.118b) into eqn. (5.147) and using the property of the Dirac delta function leads to

$$\boxed{R_k = \frac{R(t)}{\Delta t}} \tag{5.148}$$

The implication of this relationship is that the discrete-time covariance approaches infinity in the continuous representation. This may be counterintuitive at first, but as shown in eqn. (5.145) the inverse time dependence of the discrete-time covariance and the continuous-time equivalent is the *only* relationship that yields a well-behaved process.

To derive the continuous-time Kalman filter we start with the discrete-time version summarized in eqn. (5.54):

$$\hat{\mathbf{x}}_{k+1} = \Phi_k \hat{\mathbf{x}}_k + \Gamma_k \mathbf{u}_k + \Phi K_k [\tilde{\mathbf{y}}_k - H_k \hat{\mathbf{x}}_k] \tag{5.149a}$$

$$K_k = P_k H_k^T [H_k P_k H_k^T + R_k]^{-1} \tag{5.149b}$$

$$P_{k+1} = \Phi_k P_k \Phi_k^T - \Phi_k K_k H_k P_k \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T \tag{5.149c}$$

Then, using the first-order approximation $\Phi = (I + \Delta t\, F)$ and the relationship in eqn. (5.136) gives the following discrete-time covariance update:

$$\begin{aligned} P_{k+1} = {} & [I + \Delta t\, F(t)] P_k [I + \Delta t\, F(t)]^T + \Delta t\, G(t)\, Q(t)\, G^T(t) \\ & - [I + \Delta t\, F(t)] K_k H_k P_k [I + \Delta t\, F(t)]^T \end{aligned} \tag{5.150}$$

Dividing eqn. (5.150) by Δt and collecting terms yields

$$\begin{aligned} \frac{P_{k+1} - P_k}{\Delta t} = {} & F(t)\, P_k + P_k F^T(t) + \Delta t\, F(t)\, P_k F^T(t) \\ & - F(t)\, K_k H_k P_k - K_k H_k P_k F^T(t) - \frac{1}{\Delta t} K_k H_k P_k \\ & - \Delta t\, F(t)\, K_k H_k P_k F^T(t) + G(t)\, Q(t)\, G^T(t) \end{aligned} \tag{5.151}$$

From the definition of the gain K_k in eqn. (5.149b) and using the relationship in eqn. (5.148) we have

$$\begin{aligned} K_k &= P_k H_k^T \left[H_k P_k H_k^T + \frac{R(t)}{\Delta t} \right]^{-1} \\ &= \Delta t\, P_k H_k^T [\Delta t\, H_k P_k H_k^T + R(t)]^{-1} \end{aligned} \tag{5.152}$$

Therefore the limiting condition on K_k gives

$$\lim_{\Delta t \to 0} K_k = 0 \tag{5.153}$$

However when K_k is divided by Δt we have

$$\lim_{\Delta t \to 0} \frac{K_k}{\Delta t} = P(t)\, H^T(t)\, R^{-1}(t) \tag{5.154}$$

Hence in the limit as $\Delta t \to 0$ eqn. (5.151) reduces exactly to the continuous-time covariance propagation in Table 5.4.

Using the first-order approximations of $\Gamma = \Delta t\, B$ and $\Phi = (I + \Delta t\, F)$, the state estimate in eqn. (5.149a) becomes

$$\hat{\mathbf{x}}_{k+1} = [I + \Delta t\, F(t)]\hat{\mathbf{x}}_k + \Delta t\, B(t)\, \mathbf{u}_k + [I + \Delta t\, F(t)] K_k [\tilde{\mathbf{y}}_k - H_k \hat{\mathbf{x}}_k] \tag{5.155}$$

Dividing both sides of eqn. (5.155) by Δt and collecting terms leads to

$$\frac{\hat{\mathbf{x}}_{k+1} - \hat{\mathbf{x}}_k}{\Delta t} = F(t)\, \hat{\mathbf{x}}_k + B(t)\, \mathbf{u}_k + \left[\frac{K_k}{\Delta t} + F(t)\, K_k \right] [\tilde{\mathbf{y}}_k - H_k \hat{\mathbf{x}}_k] \tag{5.156}$$

Hence, using eqns. (5.153) and (5.154), in the limit as $\Delta t \to 0$ eqn. (5.156) reduces exactly to the continuous-time estimate propagation in Table 5.4.

5.4.3 Stability

The filter stability can be proved by using Lyapunov's direct method, which is discussed for continuous-time systems in §3.6. We wish to show that the estimation error dynamics, $\tilde{\mathbf{x}}(t) \equiv \hat{\mathbf{x}}(t) - \mathbf{x}(t)$, are stable. For the continuous-time Kalman filter we consider the following candidate Lyapunov function:

$$V[\tilde{\mathbf{x}}(t)] = \tilde{\mathbf{x}}^T(t)\, P^{-1}(t)\, \tilde{\mathbf{x}}(t) \tag{5.157}$$

Since $P(t)$ is required to be positive definite, then clearly its inverse exists and $V[\tilde{\mathbf{x}}(t)] > 0$ for all $\tilde{\mathbf{x}}(t) \neq \mathbf{0}$. We now need to determine an expression for $\dot{P}^{-1}(t)$ to evaluate the time derivative of eqn. (5.157). This is accomplished by taking the time derivative of $P(t)\, P^{-1}(t) = I$, which gives

$$\frac{d}{dt}\left[P(t)\, P^{-1}(t)\right] = \dot{P}(t)\, P^{-1}(t) + P(t)\, \dot{P}^{-1}(t) = 0 \tag{5.158}$$

Solving eqn. (5.158) for $\dot{P}^{-1}(t)$ gives

$$\dot{P}^{-1}(t) = -P^{-1}(t)\, \dot{P}(t)\, P^{-1}(t) \tag{5.159}$$

Substituting eqn. (5.133) into eqn. (5.159) gives

$$\begin{aligned}\dot{P}^{-1}(t) = &-P^{-1}(t)\, F(t) - F^T(t)\, P^{-1}(t) + H^T(t)\, R^{-1}(t)\, H(t) \\ &- P^{-1}(t)\, G(t)\, Q(t)\, G^T(t)\, P^{-1}(t)\end{aligned} \tag{5.160}$$

Taking the time derivative of eqn. (5.157) yields

$$\dot{V}[\tilde{\mathbf{x}}(t)] = \dot{\tilde{\mathbf{x}}}^T(t)\, P^{-1}(t)\, \tilde{\mathbf{x}}(t) + \tilde{\mathbf{x}}^T(t)\, P^{-1}(t)\, \dot{\tilde{\mathbf{x}}}(t) + \tilde{\mathbf{x}}^T(t)\, \dot{P}^{-1}(t)\, \tilde{\mathbf{x}}(t) \tag{5.161}$$

The continuous-time error dynamics are given by eqn. (5.120). Analogous to the discrete-time case the matrix $F(t) - K(t)\; H(t)$ defines the stability of the filter for the continuous-time case. Substituting $\dot{\tilde{\mathbf{x}}}(t) = [F(t) - K(t)\; H(t)]\tilde{\mathbf{x}}(t)$ and the inverse covariance propagation of eqn. (5.160) into eqn. (5.161), and simplifying leads to

$$\dot{V}[\tilde{\mathbf{x}}(t)] = -\tilde{\mathbf{x}}^T(t)\left[H^T(t)\, R^{-1}(t)\, H(t) + P^{-1}(t)\, G(t)\, Q(t)\, G^T(t)\, P^{-1}(t)\right]\tilde{\mathbf{x}}(t) \tag{5.162}$$

Clearly if $R(t)$ is positive definite and $Q(t)$ is at least positive semi-definite then the Lyapunov condition is satisfied and the continuous-time Kalman filter is stable.

5.4.4 Steady-State Kalman Filter

The continuous Riccati equation in eqn. (5.133) requires $n(n+1)/2$ nonlinear equations to be integrated numerically (normally an $n \times n$ matrix equation requires n^2 integrations, but we use the fact that $P(t)$ is symmetric to significantly reduce this number). Fortunately, analogous to the discrete-time case, for time-invariant systems

Table 5.5: Continuous and Autonomous Linear Kalman Filter

Model	$\dot{\mathbf{x}}(t) = F\,\mathbf{x}(t) + B\,\mathbf{u}(t) + G\,\mathbf{w}(t),\ \mathbf{w}(t) \sim N(\mathbf{0}, Q)$ $\tilde{\mathbf{y}}(t) = H\,\mathbf{x}(t) + \mathbf{v}(t),\ \mathbf{v}(t) \sim N(\mathbf{0}, R)$
Initialize	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$
Gain	$K = P\,H^T R^{-1}$
Covariance	$F\,P + P\,F^T - P\,H^T R^{-1} H\,P + G\,Q\,G^T = 0$
Estimate	$\dot{\hat{\mathbf{x}}}(t) = F\,\hat{\mathbf{x}}(t) + B\,\mathbf{u}(t) + K[\tilde{\mathbf{y}}(t) - H\,\hat{\mathbf{x}}(t)]$

the error covariance P reaches a steady-state value very quickly. The steady-state continuous-time Kalman filter is summarized in Table 5.5.

To determine the steady-state value for P we must solve the *continuous-time algebraic Riccati equation* in Table 5.5. A sufficient condition for the existence of a steady-state solution is complete observability.[3] Also, the solution is unique if complete controllability exists.[1] These conditions also hold true for the discrete-time Riccati equation in §5.3.4. The continuous-time Riccati equation is a nonlinear differential equation, but it can be transformed into two coupled linear differential equations. This is accomplished by writing P as a product of two matrices:[16]

$$\boxed{P(t) = S(t)\,Z^{-1}(t)} \tag{5.163}$$

or $P(t)\,Z(t) = S(t)$. Differentiating this equation leads to

$$\dot{P}(t)\,Z(t) + P(t)\dot{Z}(t) = \dot{S}(t) \tag{5.164}$$

Substituting eqn. (5.133) into eqn. (5.164) and collecting terms gives

$$\begin{aligned} &P(t)[F^T Z(t) - H^T R^{-1} H\,S(t) + \dot{Z}(t)] \\ &+ [G\,Q\,G^T Z(t) + F\,S(t) - \dot{S}(t)] = 0 \end{aligned} \tag{5.165}$$

Therefore, the following two matrix differential equations must be true to satisfy eqn. (5.165):

$$\dot{Z}(t) = -F^T Z(t) + H^T R^{-1} H\,S(t) \tag{5.166a}$$

$$\dot{S}(t) = G\,Q\,G^T Z(t) + F\,S(t) \tag{5.166b}$$

In order to satisfy eqn. (5.163), initial conditions of $Z(t_0) = I$ and $S(t_0) = P(t_0)$ can be used. Separating the columns of the $Z(t)$ and $S(t)$ gives

$$\begin{bmatrix} \dot{\mathbf{z}}_i(t) \\ \dot{\mathbf{s}}_i(t) \end{bmatrix} = \mathcal{H} \begin{bmatrix} \mathbf{z}_i(t) \\ \mathbf{s}_i(t) \end{bmatrix} \tag{5.167}$$

where $\mathbf{z}_i(t)$ and $\mathbf{s}_i(t)$ are the i^{th} columns of $Z(t)$ and $S(t)$, respectively, and $\mathcal{H}$ is the *Hamiltonian matrix* defined by

$$\mathcal{H} \equiv \begin{bmatrix} -F^T & H^T R^{-1} H \\ G\,Q\,G^T & F \end{bmatrix} \tag{5.168}$$

It can be shown that if λ is an eigenvalue of $\mathcal{H}$, then $-\lambda$ is also an eigenvalue of $\mathcal{H}$, which is left as an exercise for the reader. Thus the eigenvalues can be arranged in a diagonal matrix given by

$$\mathcal{H}_\Lambda = \begin{bmatrix} \Lambda & 0 \\ 0 & -\Lambda \end{bmatrix} \tag{5.169}$$

where Λ is a diagonal matrix of the n eigenvalues in the right half-plane. Assuming that the eigenvalues are distinct, we can perform a linear state transformation, as shown in §3.1.4, such that

$$\mathcal{H}_\Lambda = W^{-1} \mathcal{H}\, W \tag{5.170}$$

where W is the matrix of eigenvectors, which can be represented in block form as

$$W = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \tag{5.171}$$

The solutions for $\mathbf{z}_i(t)$ and $\mathbf{s}_i(t)$ can be found in terms of their eigensystems:

$$\mathbf{z}_i(t) = \mathbf{w}_1 e^{\lambda t} \tag{5.172a}$$

$$\mathbf{s}_i(t) = \mathbf{w}_2 e^{\lambda t} \tag{5.172b}$$

where $\mathbf{w}_1$ and $\mathbf{w}_2$ are eigenvectors that satisfy

$$(\lambda I - \mathcal{H}) \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{bmatrix} = \mathbf{0} \tag{5.173}$$

Going forward in time the unstable eigenvalues dominate, so that

$$\mathbf{z}_i(t) \to W_{11}\, e^{\Lambda t}\, \mathbf{c}_i \tag{5.174a}$$

$$\mathbf{s}_i(t) \to W_{21}\, e^{\Lambda t}\, \mathbf{c}_i \tag{5.174b}$$

where $\mathbf{c}_i$ is an arbitrary constant, and W_{11} and W_{21} are the eigenvectors associated with the unstable eigenvalues. Then from eqn. (5.163) it follows that at steady-state, we have

$$\boxed{P = W_{21} W_{11}^{-1}} \tag{5.175}$$

This requires an inverse of an $n \times n$ matrix.

Vaughan[17] has also shown that a solution for $P(t)$ is given by

$$P(t) = [W_{21} + W_{22} Y(t)][W_{11} + W_{12} Y(t)]^{-1} \tag{5.176}$$

where

$$Y(t) = e^{-\Lambda t} X e^{-\Lambda t} \tag{5.177a}$$

$$X = -[W_{22} - F\, W_{12}]^{-1} [W_{21} - F\, W_{11}] \tag{5.177b}$$

The steady-state solution for P can be found from

$$P = \lim_{t \to \infty} P(t) = W_{21} W_{11}^{-1} \tag{5.178}$$

This result is identical to the steady-state solution derived independently by MacFarlane[18] and Potter,[19] which has been shown previously. Therefore, the gain K in eqn. (5.132) can be computed off-line and remains constant. As in the discrete-time case, this can significantly reduce the on-board computational load on a computer. As a final note, the steady-state solution for the Riccati equation can also be found using a *Schur decomposition*,[20, 21] which is more computationally efficient and more stable than the eigenvector approach. The interested reader is encouraged to pursue this approach, which is more widely used today.

Example 5.4: In this example a simple first-order system is analyzed. The truth model is given by

$$\begin{aligned} \dot{x}(t) &= f\, x(t) + w(t) \\ y(t) &= x(t) + v(t) \end{aligned}$$

where f is a constant, and the variances of $w(t)$ and $v(t)$ are given by q and r, respectively. The first step involves solving the scalar version of the Riccati equation given in eqn. (5.133):

$$\dot{p}(t) = 2\, f\, p(t) - r^{-1} p(t)^2 + q, \quad p(t_0) = p_0 \tag{5.179}$$

To accomplish this task we use the approach given by eqns. (5.163) and (5.166). The Hamiltonian system is given by

$$\begin{bmatrix} \dot{z}(t) \\ \dot{s}(t) \end{bmatrix} = \begin{bmatrix} -f & r^{-1} \\ q & f \end{bmatrix} \begin{bmatrix} z(t) \\ s(t) \end{bmatrix}, \quad \begin{bmatrix} z(t_0) \\ s(t_0) \end{bmatrix} = \begin{bmatrix} 1 \\ p_0 \end{bmatrix}$$

The characteristic equation of this system is given by $s^2 - (f^2 + r^{-1} q) = 0$, which means the solutions for $z(t)$ and $s(t)$ involve hyperbolic functions. We assume that the solutions are given by

$$\begin{aligned} z(t) &= \cosh(a\, t) + c_1 \sinh(a\, t) \\ s(t) &= p_0 \cosh(a\, t) + c_2 \sinh(a\, t) \end{aligned}$$

where $a = \sqrt{f^2 + r^{-1} q}$, and c_1 and c_2 are constants. The assumed solutions obviously satisfy the initial condition requirements. To determine the other constants we take time derivatives of $z(t)$ and $s(t)$ and compare them to the Hamiltonian system, which gives

$$c_1 = \frac{p_0 r^{-1} - f}{a}, \quad c_2 = \frac{p_0 f + q}{a}$$

Hence, using eqn. (5.163) the solution for $p(t)$ is given by

$$p(t) = \frac{p_0 a + (p_0 f + q)\tanh(a\,t)}{a + (p_0 r^{-1} - f)\tanh(a\,t)}$$

Clearly, even for this simple first-order system the solution to the Riccati equation involves complicated functions. Analytical solutions are extremely difficult (if not impossible!) to determine for higher-order systems, so numerical procedures are typically required to integrate the Riccati differential equation. The steady-state value for $p(t)$ is given by noting that as $t \to \infty$ the hyperbolic tangent function approaches one, so that

$$\lim_{t\to\infty} p(t) \equiv p = \frac{(a+f)\,p_0 + q}{r^{-1}p_0 + a - f} = r\,(a+f)$$

The steady-state value is independent of p_0, which is intuitively correct. This result is verified by solving the algebraic Riccati equation in Table 5.5. Hence, the continuous-time Kalman filter equations are given by

$$\begin{aligned}\dot{\hat{x}}(t) &= -a\,\hat{x}(t) + (a+f)\,\tilde{y}(t)\\ \hat{y}(t) &= \hat{x}(t)\end{aligned}$$

Note that the filter dynamics are always stable. Also, when $q = 0$ the solution for the steady-state gain is given by zero, and the measurements are completely ignored in the state estimate. Furthermore, the individual values for r and q are irrelevant; only their ratio is important in the filter design. In fact, one of the most arduous tasks in the Kalman filter design is the proper selection of q, which is often not well known. For some systems the filter designer may choose to select the gain K directly (often by trial and error), if the process noise covariance is not well known.

In the preceding example the final form of the steady-state estimator for the state takes the form of a first-order low-pass filter. In the Laplace domain the transfer function from the measured input to the state estimate output is given by

$$\frac{\hat{X}(s)}{\tilde{Y}(s)} = \frac{a+f}{s+a} \tag{5.180}$$

The time constant of this system is given by $1/a$. When q is large or r is small the time constant for the filter approaches zero, so that more high-frequency information is allowed into the state estimate by the filter (i.e., the bandwidth increases). The converse to this statement is also true. When q is small or r is large the time constant for the filter approaches a large value, so that less high-frequency information is allowed into the state estimate by the filter (i.e., the bandwidth decreases). This clearly demonstrates the relationship between the Kalman filter and frequency domain.

The design of the optimal gain using frequency domain methods is known as *Wiener* filtering*.[22] The Wiener filter obtains the best estimates by analyzing time series in the frequency domain using the Fourier transform. The Wiener and Kalman approach can be shown to be identical for the optimal steady-state filter.[5] Unfortunately, Wiener filters are difficult to derive for systems that involve time-varying models or MIMO models, which the Kalman filter handles with ease. Therefore, although a brief introduction of the Wiener filter is given here, we choose not to fully derive the appropriate Wiener (more commonly known as the Wiener-Hopf[5, 14]) filter equation. Still, Wiener filtering is widely used today for many applications in signal processing (e.g., digital image processing). The interested reader is encouraged to pursue Wiener filtering in the open literature.

5.4.5 Correlated Measurement and Process Noise

In this section the correlated Kalman filter for continuous-time models and measurements is derived. The procedure to derive the results of §5.3.5 can also be applied to the continuous-time case. However, an easier approach can be used.[1, 5] We consider the following correlation between the process and measurement noise:

$$\boxed{E\left\{\mathbf{w}(t)\,\mathbf{v}^T(t)\right\} = S(t)\,\delta(t-\tau)} \tag{5.181}$$

Next consider adding zero to the right-hand side of equation eqn. (5.117a), so that

$$\begin{aligned}\dot{\mathbf{x}}(t) &= F(t)\,\mathbf{x}(t) + B(t)\,\mathbf{u}(t) + G(t)\,\mathbf{w}(t) \\ &\quad + \mathcal{D}(t)[\tilde{\mathbf{y}}(t) - H(t)\,\mathbf{x}(t) - \mathbf{v}(t)]\end{aligned} \tag{5.182a}$$

$$\begin{aligned}&= [F(t) - \mathcal{D}(t)\,H(t)]\mathbf{x}(t) + B(t)\,\mathbf{u}(t) \\ &\quad + \mathcal{D}(t)\,\tilde{\mathbf{y}}(t) + [G(t)\,\mathbf{w}(t) - \mathcal{D}(t)\,\mathbf{v}(t)]\end{aligned} \tag{5.182b}$$

where $\mathcal{D}(t)$ is a nonzero matrix. The new process noise for this system is given by $G(t)\,\mathbf{w}(t) - \mathcal{D}(t)\,\mathbf{v}(t) \equiv \boldsymbol{\upsilon}(t)$, which has zero-mean and covariance, so

$$\begin{aligned}E\left\{\boldsymbol{\upsilon}(t)\,\boldsymbol{\upsilon}^T(\tau)\right\} = \Big[&G(t)\,Q(t)\,G^T(t) + \mathcal{D}(t)\,R(t)\,\mathcal{D}^T(t) \\ &-\mathcal{D}(t)\,S(t)\,G^T(t) - G(t)\,S^T(t)\,\mathcal{D}^T(t)\Big]\,\delta(t-\tau)\end{aligned} \tag{5.183}$$

Any $\mathcal{D}(t)$ can be chosen since eqn. (5.182) will always be true. We choose $\mathcal{D}(t)$ so that $\boldsymbol{\upsilon}(t)$ and $\mathbf{v}(t)$ are uncorrelated. Specifically, if we choose

$$\mathcal{D}(t) = G(t)\,S^T(t)\,R^{-1}(t) \tag{5.184}$$

*Norbert Wiener developed this approach in response to some of the very practical technological problems to improve radar communication that arose during World War II.

Table 5.6: Correlated Continuous-Time Linear Kalman Filter

Model	$\dot{\mathbf{x}}(t) = F(t)\,\mathbf{x}(t) + B(t)\,\mathbf{u}(t) + G(t)\,\mathbf{w}(t),\ \mathbf{w}(t) \sim N(\mathbf{0}, Q)$ $\tilde{\mathbf{y}}(t) = H(t)\,\mathbf{x}(t) + \mathbf{v}(t),\ \mathbf{v}(t) \sim N(\mathbf{0}, R)$ $E\left\{\mathbf{w}(t)\,\mathbf{v}^T(t)\right\} = S(t)\,\delta(t-\tau)$
Initialize	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$ $P_0 = E\left\{\tilde{\mathbf{x}}(t_0)\,\tilde{\mathbf{x}}^T(t_0)\right\}$
Gain	$K(t) = \left[P(t)\,H^T(t) + G(t)\,S^T(t)\right]R^{-1}(t)$
Covariance	$\dot{P}(t) = F(t)\,P(t) + P(t)\,F^T(t)$ $-K(t)\,R(t)\,K^T(t) + G(t)\,Q(t)\,G^T(t)$
Estimate	$\dot{\hat{\mathbf{x}}}(t) = F(t)\,\hat{\mathbf{x}}(t) + B(t)\,\mathbf{u}(t)$ $+K(t)[\tilde{\mathbf{y}}(t) - H(t)\,\hat{\mathbf{x}}(t)]$

then

$$E\left\{\boldsymbol{\upsilon}(t)\,\mathbf{v}^T(\tau)\right\} = [G(t)S^T(t) - \mathcal{D}(t)\,R(t)]\delta(t-\tau) = 0 \tag{5.185}$$

Hence the covariance of the new process noise $\boldsymbol{\upsilon}(t)$ is given by

$$E\left\{\boldsymbol{\upsilon}(t)\,\boldsymbol{\upsilon}^T(\tau)\right\} = G(t)\left[Q(t) - S^T(t)\,R^{-1}(t)\,S(t)\right]G^T(t)\,\delta(t-\tau) \tag{5.186}$$

The derivation procedure of §5.4.1 can now be applied to eqn. (5.182b). The results are summarized in Table 5.6. Note that a nonzero $S(t)$ produces a smaller covariance than the uncorrelated case, which is due to the additional information provided by the cross-correlation between $\mathbf{w}(t)$ and $\mathbf{v}(t)$. Also, when $S(t) = 0$, i.e., $\mathbf{w}(t)$ and $\mathbf{v}(t)$ are uncorrelated, the correlated Kalman filter reduces exactly to the standard Kalman filter given in Table 5.4.

5.5 The Continuous-Discrete Kalman Filter

Most physical dynamical systems involve continuous-time models and discrete-time measurements taken from a digital signal processor. Therefore, the system model and measurement model are given by

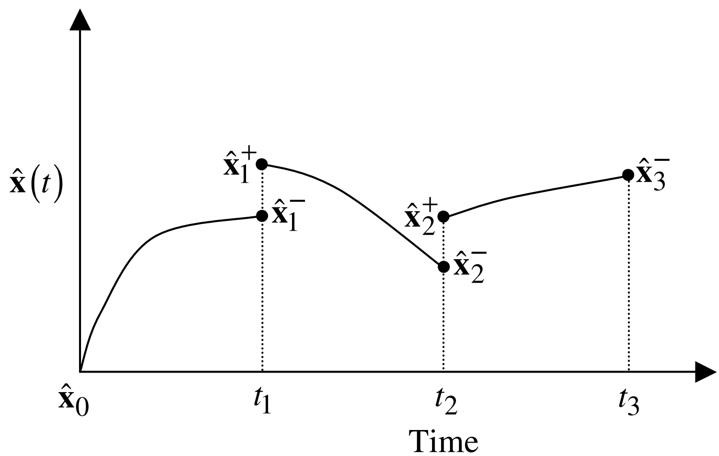

Figure 5.5: Mechanism for the Continuous-Discrete Kalman Filter

$$\boxed{\begin{aligned}\dot{\mathbf{x}}(t) &= F(t)\,\mathbf{x}(t) + B(t)\,\mathbf{u}(t) + G(t)\,\mathbf{w}(t) \\ \tilde{\mathbf{y}}_k &= H_k\mathbf{x}_k + \mathbf{v}_k\end{aligned}} \qquad \begin{aligned}(5.187\text{a}) \\ (5.187\text{b})\end{aligned}$$

where the continuous-time covariance of $\mathbf{w}(t)$ is given by eqn. (5.118a) and the discrete-time covariance of $\mathbf{v}_k$ is given by eqn. (5.28).

The extension of the Kalman filter for this case is very straightforward. The mechanism of the filter approach for this case is illustrated in Figure 5.5. The state estimate model is propagated forward in time until a measurement occurs, given at time t_1. Then a discrete-time state update occurs, which updates the final value of the propagated state $\hat{\mathbf{x}}_1^-$ to the new state $\hat{\mathbf{x}}_1^+$. Finally this state is then used as the initial condition to propagate the state estimate model to time t_2. The scheme continues forward in time, updating the state when a measurement occurs.

A summary of the continuous-discrete Kalman filter is given in Table 5.7. Note that the continuous-time propagation model equation does not involve the measurement directly. Hence, the covariance propagation follows a continuous-time Lyapunov differential equation, which is a linear equation. When a measurement occurs both the state and the covariance are updated using the standard discrete-time updates. Also, if the state and measurement models are autonomous, and the measurements sampling interval is constant and well below Nyquist's limit, then a steady-state covariance expression can be found (this is left as an exercise for the reader).

We should note that the sample times of the measurements need not occur in regular intervals. In fact different measurement sets can be spread out over various time intervals. Whenever a measurement occurs then an update is invoked. The measurement set at that time may involve only one measurement or multiple measurements. The real beauty of the continuous-discrete Kalman filter is that it can handle different scattered measurement sets quite easily.

Table 5.7: Continuous-Discrete Kalman Filter

Model	$\dot{\mathbf{x}}(t) = F(t)\mathbf{x}(t) + B(t)\mathbf{u}(t) + G(t)\mathbf{w}(t),\ \mathbf{w}(t) \sim N(\mathbf{0}, Q(t))$ $\tilde{\mathbf{y}}_k = H_k\mathbf{x}_k + \mathbf{v}_k,\quad \mathbf{v}_k \sim N(\mathbf{0}, R_k)$
Initialize	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$ $P_0 = E\left\{\tilde{\mathbf{x}}(t_0)\,\tilde{\mathbf{x}}^T(t_0)\right\}$
Gain	$K_k = P_k^- H_k^T[H_k P_k^- H_k^T + R_k]^{-1}$
Update	$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k[\tilde{\mathbf{y}}_k - H_k\hat{\mathbf{x}}_k^-]$ $P_k^+ = [I - K_k H_k]P_k^-$
Propagation	$\dot{\hat{\mathbf{x}}}(t) = F(t)\,\hat{\mathbf{x}}(t) + B(t)\,\mathbf{u}(t)$ $\dot{P}(t) = F(t)\,P(t) + P(t)\,F^T(t) + G(t)\,Q(t)\,G^T(t)$

5.6 Extended Kalman Filter

A large class of estimation problems involve nonlinear models. For several reasons, state estimation for nonlinear systems is considerably more difficult and admits a wider variety of solutions than the linear problem.[1] A vast majority of nonlinear models are given in continuous-time. Therefore, we first consider the following common nonlinear truth model with continuous-time measurements:

$$\boxed{\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) + G(t)\,\mathbf{w}(t) \qquad (5.188\text{a})\\ \tilde{\mathbf{y}}(t) &= \mathbf{h}(\mathbf{x}(t), t) + \mathbf{v}(t) \qquad (5.188\text{b})\end{aligned}}$$

where $\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t)$ and $\mathbf{h}(\mathbf{x}(t), t)$ are assumed to be continuously differentiable, and $\mathbf{w}(t)$ and $\mathbf{v}(t)$ follow exactly from §5.4.1. The problem with this nonlinear model is that a Gaussian input does not necessarily produce a Gaussian output (unlike the linear case). Some of these problems are seen by considering the simple nonlinear and stochastic function

$$y(t) = \sin(t) + v(t) \tag{5.189}$$

The top plot of Figure 5.6 shows $y(t)$ with a Gaussian input ($\sigma = 1$), as a function of normalized time in degrees (360 degrees is equivalent to 2π seconds). Clearly, the probability density function of $v(t)$ is altered as it is transmitted through the nonlinear element. The exact probability density function can be determined using a transformation of variables[14, 23] (see Appendix B). But for small angles the output

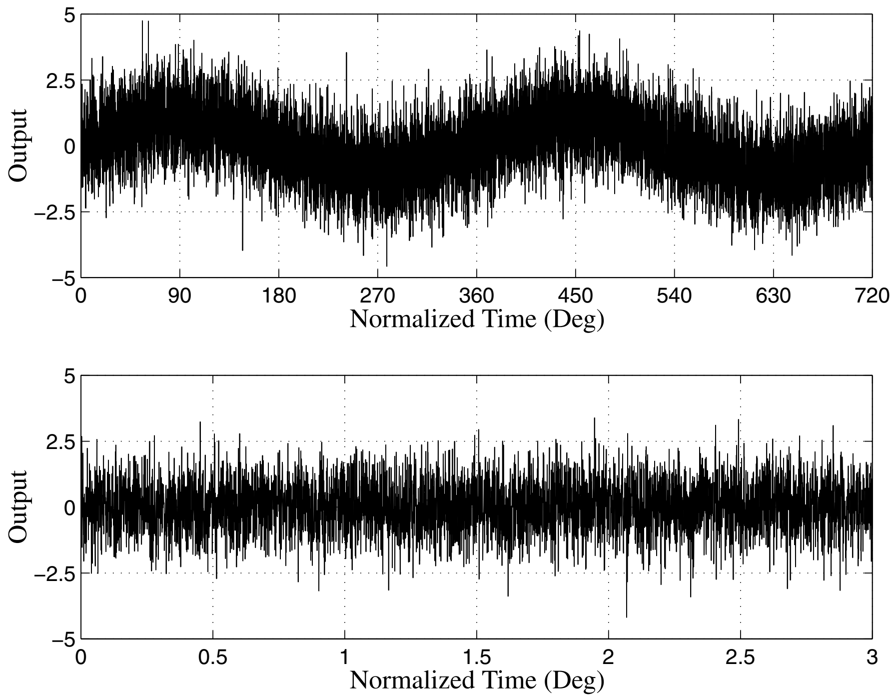

Figure 5.6: Stochastic Nonlinear Example

is *approximately* Gaussian, as shown by the bottom plot of Figure 5.6, where $\sin(t)$ can be approximated by t for small t. Also, $E\left\{y^2(t)\right\} \approx 1$ since terms in t^2 are second-order in nature, which can be ignored. This approach can be used to derive a Kalman filter using nonlinear models.

There are many possible ways to produce a linearized version of the Kalman filter.[1, 24] We will consider the most common approach, which is the *extended Kalman filter*. The extended Kalman filter, though not precisely "optimum," has been successfully applied to many nonlinear systems over the past many years. The fundamental concept of this filter involves the notion that the true state is sufficiently close to the estimated state. Therefore, the error dynamics can be represented fairly accurately by a linearized first-order Taylor series expansion. Consider the first-order expansion of $\mathbf{f}(\mathbf{x}(t),\, \mathbf{u}(t),\, t)$ about some nominal state $\bar{\mathbf{x}}(t)$:

$$\mathbf{f}(\mathbf{x}(t),\, \mathbf{u}(t),\, t) \cong \mathbf{f}(\bar{\mathbf{x}}(t),\, \mathbf{u}(t),\, t) + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\bar{\mathbf{x}}(t)} [\mathbf{x}(t) - \bar{\mathbf{x}}(t)] \tag{5.190}$$

where $\bar{\mathbf{x}}(t)$ is close to $\mathbf{x}(t)$. Also, the output in eqn. (5.188b) can also be expanded using

$$\mathbf{h}(\mathbf{x}(t),\, t) \cong \mathbf{h}(\bar{\mathbf{x}}(t),\, t) + \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\bar{\mathbf{x}}(t)} [\mathbf{x}(t) - \bar{\mathbf{x}}(t)] \tag{5.191}$$

Table 5.8: Continuous-Time Extended Kalman Filter

Model	$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) + G(t)\,\mathbf{w}(t),\ \ \mathbf{w}(t) \sim N(\mathbf{0}, Q(t))$ $\tilde{\mathbf{y}}(t) = \mathbf{h}(\mathbf{x}(t), t) + \mathbf{v}(t),\ \ \mathbf{v}(t) \sim N(\mathbf{0}, R(t))$
Initialize	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$ $P_0 = E\left\{\tilde{\mathbf{x}}(t_0)\,\tilde{\mathbf{x}}^T(t_0)\right\}$
Gain	$K(t) = P(t)\,H^T(\hat{\mathbf{x}}(t), t)\,R^{-1}(t)$
Covariance	$\dot{P}(t) = F(\hat{\mathbf{x}}(t), t)\,P(t) + P(t)\,F^T(\hat{\mathbf{x}}(t), t)$ $-P(t)\,H^T(\hat{\mathbf{x}}(t), t)\,R^{-1}(t)H(\hat{\mathbf{x}}(t), t)\,P(t) + G(t)\,Q(t)\,G^T(t)$ $F(\hat{\mathbf{x}}(t), t) \equiv \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right\|_{\hat{\mathbf{x}}(t)}, \quad H(\hat{\mathbf{x}}(t), t) \equiv \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right\|_{\hat{\mathbf{x}}(t)}$
Estimate	$\dot{\hat{\mathbf{x}}}(t) = \mathbf{f}(\hat{\mathbf{x}}(t), \mathbf{u}(t), t) + K(t)[\tilde{\mathbf{y}}(t) - \mathbf{h}(\hat{\mathbf{x}}(t), t)]$

In the extended Kalman filter, the current estimate (i.e., conditional mean) is used for the nominal state estimate, so that $\bar{\mathbf{x}}(t) = \hat{\mathbf{x}}(t)$. Taking the expectation of both sides of eqns. (5.190) and (5.191), with $\bar{\mathbf{x}}(t) = \hat{\mathbf{x}}(t)$, gives

$$E\{\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t)\} = \mathbf{f}(\hat{\mathbf{x}}(t), \mathbf{u}(t), t) \tag{5.192a}$$

$$E\{\mathbf{h}(\mathbf{x}(t), t)\} = \mathbf{h}(\hat{\mathbf{x}}(t), t) \tag{5.192b}$$

Therefore, the extended Kalman filter structure for the state and output estimate is given by

$$\boxed{\dot{\hat{\mathbf{x}}}(t) = \mathbf{f}(\hat{\mathbf{x}}(t), \mathbf{u}(t), t) + K(t)[\tilde{\mathbf{y}}(t) - \mathbf{h}(\hat{\mathbf{x}}(t), t)]} \tag{5.193a}$$

$$\boxed{\hat{\mathbf{y}}(t) = \mathbf{h}(\hat{\mathbf{x}}(t), t)} \tag{5.193b}$$

Substituting eqns. (5.190) and (5.191), with $\bar{\mathbf{x}}(t) = \hat{\mathbf{x}}(t)$, into eqn. (5.193a), and using eqn. (5.188) leads to

$$\dot{\tilde{\mathbf{x}}}(t) = \left[F(\hat{\mathbf{x}}(t), t) - K(t)\,H(\hat{\mathbf{x}}(t), t)\right]\tilde{\mathbf{x}}(t) - G(t)\,\mathbf{w}(t) + K(t)\,\mathbf{v}(t) \tag{5.194}$$

where $\tilde{\mathbf{x}}(t) = \hat{\mathbf{x}}(t) - \mathbf{x}(t)$ and

$$\boxed{F(\hat{\mathbf{x}}(t), t) \equiv \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}(t)}, \quad H(\hat{\mathbf{x}}(t), t) \equiv \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}(t)}} \tag{5.195}$$

Equation (5.194) has the same structure as eqn. (5.120). Hence the covariance expression given by eqn. (5.133) can be used with $F(t)$ replaced by $F(\hat{\mathbf{x}}(t), t)$ and $H(t)$

replaced by $H(\hat{\mathbf{x}}(t), t)$. A summary of the continuous-time extended Kalman filter is given in Table 5.8. The matrices $F(\hat{\mathbf{x}}(t), t)$ and $H(\hat{\mathbf{x}}(t), t)$ will not be constant in general. Therefore, a steady-state gain cannot be found, which may significantly increase the computational burden since $n(n+1)/2$ nonlinear equations need to be integrated to determine $P(t)$.

Another approach involves linearizing about the nominal (*a priori*) state vector $\bar{\mathbf{x}}(t)$ instead of the current estimate $\hat{\mathbf{x}}(t)$. In this case taking the expectation of both sides of eqns. (5.190) and (5.191) gives

$$E\{\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t)\} = \mathbf{f}(\bar{\mathbf{x}}(t), \mathbf{u}(t), t) + F(\bar{\mathbf{x}}(t), t)[\hat{\mathbf{x}}(t) - \bar{\mathbf{x}}(t)] \tag{5.196a}$$

$$E\{\mathbf{h}(\mathbf{x}(t), t)\} = \mathbf{h}(\bar{\mathbf{x}}(t), t) + H(\bar{\mathbf{x}}(t), t)[\hat{\mathbf{x}}(t) - \bar{\mathbf{x}}(t)] \tag{5.196b}$$

Therefore, the Kalman filter structure for the state and output estimate is given by

$$\boxed{\begin{aligned}\dot{\hat{\mathbf{x}}}(t) &= \mathbf{f}(\bar{\mathbf{x}}(t), \mathbf{u}(t), t) + F(\bar{\mathbf{x}}(t), t)[\hat{\mathbf{x}}(t) - \bar{\mathbf{x}}(t)] \\ &\quad + K(t)\left\{\tilde{\mathbf{y}}(t) - \mathbf{h}(\bar{\mathbf{x}}(t), t) - H(\bar{\mathbf{x}}(t), t)[\hat{\mathbf{x}}(t) - \bar{\mathbf{x}}(t)]\right\}\end{aligned}} \tag{5.197a}$$

$$\boxed{\hat{\mathbf{y}}(t) = \mathbf{h}(\bar{\mathbf{x}}(t), t) + H(\bar{\mathbf{x}}(t), t)[\hat{\mathbf{x}}(t) - \bar{\mathbf{x}}(t)]} \tag{5.197b}$$

The covariance equation follows the form given in Table 5.8, with the partials evaluated at the nominal state instead of the current estimate. These equations form the *linearized Kalman filter*. In general, the linearized Kalman filter is less accurate than the extended Kalman filter since $\bar{\mathbf{x}}(t)$ is usually not as close to the truth as is $\hat{\mathbf{x}}(t)$.[1] However since the nominal state is known *a priori* the gain $K(t)$ can be pre-computed and stored, which reduces the on-line computational burden.

A summary of the continuous-discrete extended Kalman filter is given in Table 5.9. The approach used in the extended Kalman filter assumes that the true state is "close" to the estimated state. This restriction can prove to be especially damaging for highly nonlinear applications with large initial condition errors. Proving convergence in the extended Kalman filter is difficult (if not impossible!) even for simple systems where the initial condition is not well known. Even so, the extended Kalman filter is widely used in practice, and is often robust to initial condition errors, which can be often verified through simulation.

The current estimate in the extended Kalman filter can be improved by applying local iterations to repeatedly calculate $\hat{\mathbf{x}}_k^+$, P_k^+, and K_k, each time linearizing about the most recent estimate.[1, 23] This approach is known as the *iterated extended Kalman filter*. The iterations are given by

$$\boxed{\begin{aligned}\hat{\mathbf{x}}_{k_i}^+ &= \hat{\mathbf{x}}_k^- + K_{k_i}\left[\tilde{\mathbf{y}}_k - \mathbf{h}(\hat{\mathbf{x}}_{k_i}^+) - H_k(\hat{\mathbf{x}}_{k_i}^+)\left(\hat{\mathbf{x}}_k^- - \hat{\mathbf{x}}_{k_i}^+\right)\right] && (5.198a)\\ K_{k_i} &= P_k^- H_k^T(\hat{\mathbf{x}}_{k_i}^+)\left[H_k(\hat{\mathbf{x}}_{k_i}^+) P_k^- H_k^T(\hat{\mathbf{x}}_{k_i}^+) + R_k\right]^{-1} && (5.198b)\\ P_{k_i}^+ &= \left[I - K_{k_i} H_k(\hat{\mathbf{x}}_{k_i}^+)\right] P_k^- && (5.198c)\end{aligned}}$$

with $\hat{\mathbf{x}}_{k_0}^+ = \hat{\mathbf{x}}_k^-$. The iterations are continued until the estimate is no longer improved. The reference trajectory over $[t_{k-1}, t_k)$ can also be improved once the measurement

Table 5.9: Continuous-Discrete Extended Kalman Filter

Model	$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) + G(t)\,\mathbf{w}(t),\ \mathbf{w}(t) \sim N(\mathbf{0}, Q(t))$ $\tilde{\mathbf{y}}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k,\ \mathbf{v}_k \sim N(\mathbf{0}, R_k)$
Initialize	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$ $P_0 = E\left\{\tilde{\mathbf{x}}(t_0)\,\tilde{\mathbf{x}}^T(t_0)\right\}$
Gain	$K_k = P_k^- H_k^T(\hat{\mathbf{x}}_k^-)[H_k(\hat{\mathbf{x}}_k^-)P_k^- H_k^T(\hat{\mathbf{x}}_k^-) + R_k]^{-1}$ $H_k(\hat{\mathbf{x}}_k^-) \equiv \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right\|_{\hat{\mathbf{x}}_k^-}$
Update	$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k[\tilde{\mathbf{y}}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-)]$ $P_k^+ = [I - K_k H_k(\hat{\mathbf{x}}_k^-)]P_k^-$
Propagation	$\dot{\hat{\mathbf{x}}}(t) = \mathbf{f}(\hat{\mathbf{x}}(t), \mathbf{u}(t), t)$ $\dot{P}(t) = F(\hat{\mathbf{x}}(t), t)\,P(t) + P(t)\,F^T(\hat{\mathbf{x}}(t), t) + G(t)\,Q(t)\,G^T(t)$ $F(\hat{\mathbf{x}}(t), t) \equiv \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right\|_{\hat{\mathbf{x}}(t)}$

$\tilde{\mathbf{y}}_k$ is taken. This is accomplished by applying a nonlinear smoother (see §6.1.3) backward to time t_{k-1}. This approach is known as an *iterated linearized filter-smoother*.[23, 24] The algorithm can also be iterated globally, having processed all measurements, by applying a smoother back to time t_0.[24]

Example 5.5: In this example we will demonstrate the usefulness of the extended Kalman filter to estimate the states of Van der Pol's equation, given by

$$m\,\ddot{x} + 2\,c\,(x^2 - 1)\,\dot{x} + k\,x = 0$$

where m, c, and k have positive values. This equation induces a limit cycle that is sustained by periodically releasing energy into and absorbing energy from the environment, through the damping term.[25] The system can be represented in first-order form by defining the following state vector $\mathbf{x} = \begin{bmatrix} x & \dot{x} \end{bmatrix}^T$:

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= -2\,(c/m)(x_1^2 - 1)\,x_2 - (k/m)\,x_1
\end{aligned}$$

The measurement output is position, so that $H = \begin{bmatrix} 1 & 0 \end{bmatrix}$. Synthetic states are generated using $m = c = k = 1$, with an initial condition of $\mathbf{x}_0 = \begin{bmatrix} 1 & 0 \end{bmatrix}^T$. The measure-

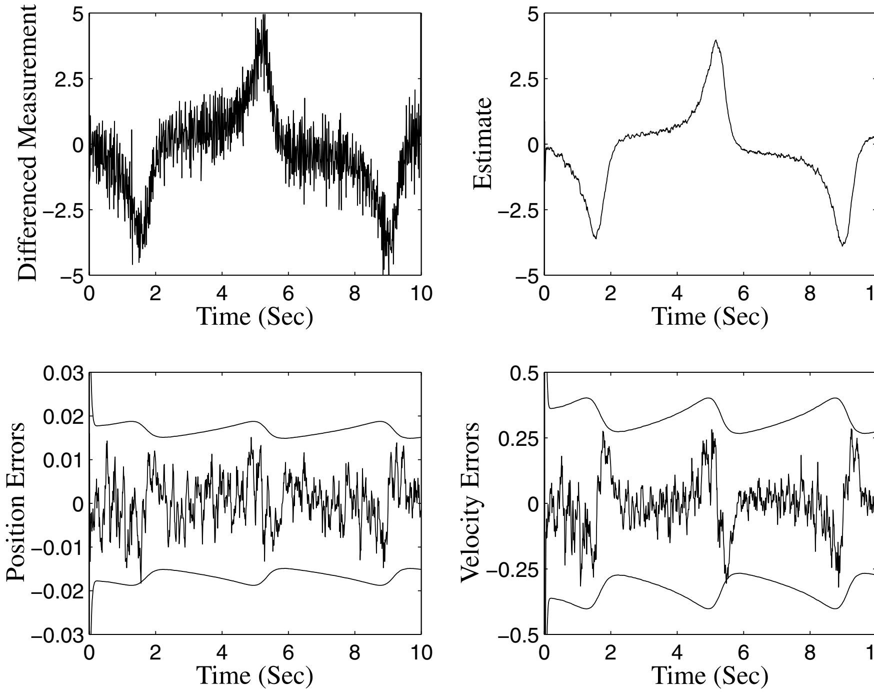

Figure 5.7: Extended Kalman Filter Results for Van der Pol's Equation

ments are sampled at $\Delta t = 0.01$-second intervals with a measurement-error standard deviation of $\sigma = 0.01$. The linearized model and G matrix used in the extended Kalman filter are given by

$$F = \begin{bmatrix} 0 & 1 \\ -4\,(c/m)\,\hat{x}_1\hat{x}_2 - (k/m) & -2\,(c/m)\,(\hat{x}_1^2 - 1) \end{bmatrix}, \quad G = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Note that no process noise (i.e., no error) is introduced into the first state. This is due to the fact that the first state is a kinematical relationship that is correct in theory and in practice (i.e., velocity is always the derivative of position). In the extended Kalman filter the model parameters are assumed to be given by $m = 1$, $c = 1.5$, and $k = 1.2$, which introduces errors in the assumed system, compared to the true system. The initial covariance is chosen to be $P_0 = 1000\,I$. The scalar $q \equiv Q(t)$ in the extended Kalman filter is then tuned until reasonable state estimates are achieved (this tuning process is often required in the design of a Kalman filter). The answer to the question "what are reasonable estimates?" is often left to the design engineer. Since for this simulation the truth is known, we can compare our estimates with the truth to tune q. It was found that $q = 0.2$ results in good estimates. The adaptive methods of §5.7.4 can also be employed to help determine q using measurement residuals.

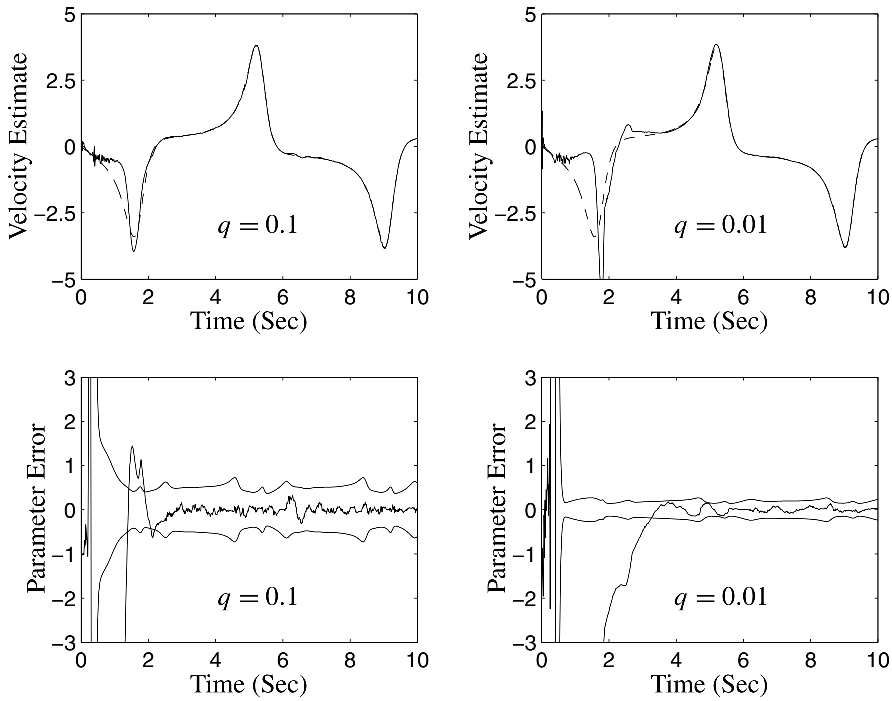

Figure 5.8: Extended Kalman Filter Parameter Identification Results

When first confronted with the position measurements, one may naturally choose to take a numerical finite-difference to derive a velocity estimate. The top left plot of Figure 5.7 shows the result of this approach (with the truth overlapped in the plot). Clearly the result is very noisy. The top right plot of Figure 5.7 shows the velocity estimate using the tuned extended Kalman filter. Clearly the state estimate is closer to the truth than using a numerical finite-difference approach. The bottom plots of Figure 5.7 show the state errors (estimate minus truth) with 3σ boundaries. The boundaries do provide a bound for the estimate errors. We should note that the estimate error does not look Gaussian. This is due to the fact that the process noise is in fact modelling errors in this example. However, the extended Kalman filter still works well even for this case. This example shows the power of the extended Kalman filter to provide accurate estimates for a highly nonlinear system.

Example 5.6: We next show the power of using the Kalman filter to estimate model parameters online. We will now assume that the damping coefficient c is unknown. This parameter can be estimated by appending the state vector of the assumed model in the extended Kalman filter. A common approach assumes a random-walk process, so that $\dot{\hat{c}} \equiv \dot{\hat{x}}_3 = 0$. The linearized model is now given by

$$F = \begin{bmatrix} 0 & 1 & 0 \\ -4(\hat{x}_3/m)\hat{x}_1\hat{x}_2 - (k/m) & -2\,(\hat{x}_3/m)(\hat{x}_1^2 - 1) & -(2/m)(\hat{x}_1^2 - 1)\hat{x}_2 \\ 0 & 0 & 0 \end{bmatrix}$$

In this case we assume that the model structure with $m = 1$ and $k = 1$ are known perfectly. Our objective is to find the parameter c, where the true value is $c = 1$. Therefore, the matrix G is assumed to be given by $G = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$. The same measurements as before are used in this simulation. Also, the initial condition for the parameter estimate is set to zero ($\hat{c}(t_0) = 0$). Results using two different values for q are shown in Figure 5.8. The top plots show the estimated velocity states, while the bottom plots show the parameter error-states. When $q = 0.1$ the filter converges fairly rapidly as opposed to the case when $q = 0.01$. However, the estimate for c is more accurate using $q = 0.01$, since the covariance is smaller than the $q = 0.1$ case. Intuitively this makes sense since a smaller q relies more on the model, which implies better knowledge that leads to more accurate estimates. However, a price is paid in convergence, which may be a cause for concern if the model estimate is needed in an online control algorithm. This shows the classic tradeoff between convergence and accuracy when using the Kalman filter to identify model parameters.

5.7 Advanced Topics

In this section we will show some advanced topics used in the Kalman filter. As in previous chapters we encourage the interested reader to pursue these topics further in the references provided. These topics include: factorization methods, colored-noise Kalman filtering, consistency of the Kalman filter, adaptive filtering, error analysis, Unscented filtering, and robust filtering.

5.7.1 Factorization Methods

The linear and autonomous Kalman filter has been shown to be theoretically stable using Lyapunov's direct method (i.e., the estimates will not diverge from the true values), and provides accurate estimates under properly defined conditions. However, the numerical stability of the extended Kalman filter must be properly addressed before on-board implementation. Many factors affect filter stability for this case. One common problem is in the error covariance update and propagation, which may become semi-definite or even negative definite, chiefly due to computational instabilities. A measure of the potential for difficulty in an ill-conditioned matrix can be found by using the *condition number* (see Appendix A). This problem may be overcome by using the Joseph form shown in §5.3.2. Other methods described here decompose the covariance matrix P into better conditioned matrices, which attempt to overcome finite-word length computation errors. We should note that the methods described here do not increase the performance of the Kalman filter in theory. These methods are strictly used to provide a better conditioned Kalman filter in practice (i.e., in a computational sense).

Square Root Information Filter

The first method is based upon a square root factorization of P, given by

$$P = S\,S^T \tag{5.199}$$

One nice property of this factorization is that P is always positive semi-definite even if S is not. Unfortunately, the matrix S is not unique. The original idea for the square root filter is attributed to James E. Potter, and was developed only one year after Kalman's original paper.[26] So, the problem of computational stability was known from the onset. An estimator based on this approach was used extensively in the Apollo navigation system. The square root formulation requires half the significant digits of the standard covariance formulation.[4] Instead of the factorization shown in eqn. (5.199), we show a more robust approach by decomposing the inverse of P. This algorithm is known as the *Square Root Information Filter* (SRIF). The equations are described without derivation. We refer the readers to Refs. [23] and [27], which provide a thorough treatise on square root filtering. The SRIF uses the inverse of eqn. (5.199):

$$\boxed{\begin{aligned} \mathcal{P}_k^+ &\equiv (P_k^+)^{-1} = \mathcal{S}_k^{+T}\,\mathcal{S}_k^+ && (5.200\text{a})\\ \mathcal{P}_k^- &\equiv (P_k^-)^{-1} = \mathcal{S}_k^{-T}\,\mathcal{S}_k^- && (5.200\text{b})\end{aligned}}$$

where $\mathcal{S} \equiv S^{-1}$. A square root decomposition of the inverse measurement covariance and an eigenvalue decomposition of the process noise covariance is also used in the SRIF:

$$\boxed{\begin{aligned} R_k^{-1} &= \mathcal{V}_k^T\,\mathcal{V}_k && (5.201\text{a})\\ Q_k &= Z_k\,E_k\,Z_k^T && (5.201\text{b})\end{aligned}}$$

where $\mathcal{V}_k$ is the inverse of the matrix V_k in $R = V_k\,V_k^T$. The matrix Z_k is an $s \times s$ (where s is the dimension of the matrix Q_k) orthogonal matrix, and E_k is an $s \times s$ diagonal matrix of the eigenvalues of Q_k. Next, the following $(n+m) \times n$ matrix is formed:

$$\tilde{\mathcal{S}}_k^+ \equiv \begin{bmatrix} \mathcal{S}_k^- \\ \mathcal{V}_k\,H_k \end{bmatrix} \tag{5.202}$$

It can be shown that when a QR decomposition (see §1.6.1) of $\tilde{\mathcal{S}}_k^+$ is taken, then the updated matrix $\mathcal{S}_k^+$ can be extracted from

$$\boxed{\mathcal{Q}_k^T \tilde{\mathcal{S}}_k^+ = \begin{bmatrix} \mathcal{S}_k^+ \\ 0_{m\times n} \end{bmatrix}} \tag{5.203}$$

where $\mathcal{Q}_k$ is the orthogonal matrix from the QR decomposition of $\tilde{\mathcal{S}}_k^+$. In the SRIF the state is not explicitly estimated. Instead the following quantities are used:

$$\hat{\boldsymbol{\alpha}}_k^+ \equiv \mathcal{S}_k^+ \hat{\mathbf{x}}_k^+ \tag{5.204a}$$
$$\hat{\boldsymbol{\alpha}}_k^- \equiv \mathcal{S}_k^- \hat{\mathbf{x}}_k^- \tag{5.204b}$$

Note the updated and propagated state can easily be found by taking the inverse of eqn. (5.204). The update equation is given by

$$\boxed{\begin{bmatrix} \hat{\boldsymbol{\alpha}}_k^+ \\ \boldsymbol{\beta}_k \end{bmatrix} = \mathcal{Q}_k^T \begin{bmatrix} \hat{\boldsymbol{\alpha}}_k^- \\ \tilde{\mathbf{y}}_k \end{bmatrix}} \tag{5.205}$$

where $\boldsymbol{\beta}_k$ is an $m \times 1$ vector, which is the residual after processing the measurement, that is not required in the SRIF calculations. The following $n \times s$ matrix is now defined:

$$\Xi_k \equiv \Upsilon_k Z_k \tag{5.206}$$

where Υ_k is defined in the discrete-time Kalman filter (see Table 5.1). Let $\Xi_k(i)$ denote the i^{th} column of Ξ_k and $E_k(i,i)$ denote the i^{th} diagonal value of the matrix E_k. The propagated values are given by a set of s iterations:

for $i = 1$

$$\mathbf{a} = \mathcal{S}_k^+ \Phi_k^{-1} \Xi_k(1) \tag{5.207a}$$
$$b = \left[\mathbf{a}^T \mathbf{a} + 1/E_k(1,1)\right]^{-1} \tag{5.207b}$$
$$c = \left[1 + \sqrt{b/E_k(1,1)}\right]^{-1} \tag{5.207c}$$
$$\mathbf{d}^T = b\, \mathbf{a}^T \mathcal{S}_k^+ \Phi_k^{-1} \tag{5.207d}$$
$$\hat{\boldsymbol{\alpha}}_{k+1}^- = \hat{\boldsymbol{\alpha}}_k^+ - b\, c\, \mathbf{a}\mathbf{a}^T \hat{\boldsymbol{\alpha}}_k^+ \tag{5.207e}$$
$$\mathcal{S}_{k+1}^- = \mathcal{S}_k^+ \Phi_k^{-1} - c\, \mathbf{a}\, \mathbf{d}^T \tag{5.207f}$$

for $i > 1$

$$\mathbf{a} = \mathcal{S}_{k+1}^{-} \Xi_k(i) \tag{5.208a}$$

$$b = \left[\mathbf{a}^T \mathbf{a} + 1/E_k(i,i)\right]^{-1} \tag{5.208b}$$

$$c = \left[1 + \sqrt{b/E_k(i,i)}\right]^{-1} \tag{5.208c}$$

$$\mathbf{d}^T = b\,\mathbf{a}^T \mathcal{S}_{k+1}^{-} \tag{5.208d}$$

$$\hat{\boldsymbol{\alpha}}_{k+1}^{-} \leftarrow \hat{\boldsymbol{\alpha}}_{k+1}^{-} - b\,c\,\mathbf{a}\mathbf{a}^T \hat{\boldsymbol{\alpha}}_{k+1}^{-} \tag{5.208e}$$

$$\mathcal{S}_{k+1}^{-} \leftarrow \mathcal{S}_{k+1}^{-} - c\,\mathbf{a}\,\mathbf{d}^T \tag{5.208f}$$

where Φ_k is the state matrix defined in the Kalman filter, and $\leftarrow$ denotes replacement. If a control input is present, then this can be added to $\hat{\boldsymbol{\alpha}}_{k+1}^{-}$ after the final iteration, with $\hat{\boldsymbol{\alpha}}_{k+1}^{-} \leftarrow \hat{\boldsymbol{\alpha}}_{k+1}^{-} + \mathcal{S}_{k+1}^{-} \Gamma_k \mathbf{u}_k$.

U-D Filter

A typically more computationally efficient algorithm than the square root approach is given by the *U-D* filter.[28] The derivation is based on the sequential processing approach presented in §5.3.3. The *U-D* filter factors the covariance matrix using

$$P_{i_k}^{-} = U_{i_k}^{-} D_{i_k}^{-} U_{i_k}^{-T} = \left[U_{i_k}^{-}\left(D_{i_k}^{-}\right)^{1/2}\right]\left[U_{i_k}^{-}\left(D_{i_k}^{-}\right)^{1/2}\right]^T \equiv S_{i_k}^{-} S_{i_k}^{-T} \tag{5.209}$$

where $U_{i_k}^{-}$ is a unitary (with ones along the diagonal) upper triangular matrix and $D_{i_k}^{-}$ is a diagonal matrix. The main advantage of this approach is that the factorization is accomplished without taking square roots.[7] This leads to a formulation that approaches the standard Kalman filter in computational effort. The gain matrix, covariance propagation, and update are given in terms of these matrices. Using the factorization in eqn. (5.209) on the covariance update in eqn. (5.79b) leads to

$$P_{i_k}^{+} = U_{i_k}^{+} D_{i_k}^{+} U_{i_k}^{+T} = U_{i_k}^{-}\left[D_{i_k}^{-} - \frac{1}{\alpha_{i_k}} \mathbf{e}_{i_k} \mathbf{e}_{i_k}^T\right] U_{i_k}^{-T} \tag{5.210}$$

where

$$\alpha_{i_k} \equiv \mathcal{H}_{i_k} P_{i_k}^{-} \mathcal{H}_{i_k}^T + \mathcal{R}_{i_k} \tag{5.211a}$$

$$\mathbf{e}_{i_k} \equiv D_{i_k}^{-} U_{i_k}^{-T} \mathcal{H}_{i_k}^T \tag{5.211b}$$

Since the bracketed term in eqn. (5.210) is also symmetric, it can be factored into

$$\left[D_{i_k}^{-} - \frac{1}{\alpha_{i_k}} \mathbf{e}_{i_k} \mathbf{e}_{i_k}^T\right] = L_{i_k}^{-} \mathcal{E}_{i_k}^{-} L_{i_k}^{-T} \tag{5.212}$$

where $L_{i_k}^-$ is a unitary upper triangular matrix and $\mathcal{E}_{i_k}^-$ is a diagonal matrix. Therefore, eqn. (5.210) is given by

$$U_{i_k}^+ D_{i_k}^+ U_{i_k}^{+T} = \left[U_{i_k}^- L_{i_k}^-\right] \mathcal{E}_{i_k}^- \left[U_{i_k}^- L_{i_k}^-\right]^T \tag{5.213}$$

Since the matrix $\left[U_{i_k}^- L_{i_k}^-\right]$ is upper triangular and $\mathcal{E}_{i_k}^-$ is diagonal then the update matrices are simply given by

$$\boxed{\begin{aligned} U_{i_k}^+ &= U_{i-1_k}^+ L_{i_k}^-, \quad U_{0_k}^+ = U_k^- && (5.214\text{a}) \\ D_{i_k}^+ &= \mathcal{E}_{i_k}^-, \quad D_{0_k}^+ = D_k^- && (5.214\text{b}) \end{aligned}}$$

The covariance update is given in terms of the factorized matrices U_k^+ and D_k^+ instead of using P_k^- directly, which leads to a more computationally stable algorithm. Also, filter gain is given by

$$\boxed{K_{i_k} = \frac{1}{\alpha_{i_k}} U_{i_k}^- \mathbf{e}_{i_k}} \tag{5.215}$$

The propagated values for U_{k+1}^- and D_{k+1}^- are computed by first defining the following variables:

$$W_{k+1}^- \equiv \left[\Phi_k U_k^+ \;\; \Xi_k\right] \tag{5.216a}$$

$$\tilde{D}_{k+1}^- \equiv \begin{bmatrix} D_k^+ & 0_{n\times s} \\ 0_{s\times n} & E_k \end{bmatrix} \tag{5.216b}$$

where Ξ_k is given by eqn. (5.206) and E_k is given by eqn. (5.201b). The matrix W_{k+1}^{-T} is partitioned into $(n+s)$ column vectors as

$$\left[\mathbf{w}(1)\ \mathbf{w}(2)\ \cdots\ \mathbf{w}(n)\right] = W_{k+1}^{-T} \tag{5.217}$$

First, the matrix U_{k+1}^- is initialized to be an $n\times n$ identity matrix and the matrix D_{k+1}^- is initialized to be an $n\times n$ matrix of zeros. Then, the following iterations are performed for $i = n, n-1, \ldots, 1$ to determine the upper triangular elements of U_{k+1}^- and the diagonal elements of D_{k+1}^-:

$$\boxed{\begin{aligned} \mathbf{c}(i) &= \tilde{D}_{k+1}^- \mathbf{w}(i) && (5.218\text{a}) \\ D_{k+1}^-(i,i) &= \mathbf{w}^T(i)\,\mathbf{c}(i) && (5.218\text{b}) \\ \mathbf{d}(i) &= \mathbf{c}(i)/D_{k+1}^-(i,i) && (5.218\text{c}) \\ U_{k+1}^-(j,i) &= \mathbf{w}^T(j)\,\mathbf{d}(i), \quad j = 1, 2, \ldots, i-1 && (5.218\text{d}) \\ \mathbf{w}(j) &\leftarrow \mathbf{w}(j) - U_{k+1}^-(j,i)\,\mathbf{w}(i), \quad j = 1, 2, \ldots, i-1 && (5.218\text{e}) \end{aligned}}$$

On the last iteration, for $i = 1$, only the first two equations in eqn. (5.218) need to be processed. The state propagation still follows eqn. (5.30a).

Finding a tractable solution for any numerical issues in the propagation equation is often problem dependent, and often relies on other factors such as computational load. In general, the factorization algorithms presented in this section should always be employed if the computational load is not burdensome. The SRIF algorithm is less computationally efficient than the *U-D* filter, but the SRIF is computationally competitive if the number of measurements is large (see Ref. [23] for more details). With the rapid progress in computer technology today the methods shown in this section have nearly become obsolete. Still, they should be employed as a first step to investigate any anomalous behaviors in the Kalman filter, especially in nonlinear systems.

5.7.2 Colored-Noise Kalman Filtering

A critical assumption required in the derivation of the Kalman filter in §5.3 is that both the process and measurement noise are represented by zero-mean Gaussian white-noise processes. If this assumption is invalid then the filter may be suboptimal and even produce biased estimates. An example of this scenario involves spacecraft attitude determination using three-axis magnetometers (TAMs). The TAM sensor measurement error itself can adequately be represented by a white-noise process, but the errors in the actual Earth's magnetic field cannot be modelled by a white-noise process. These errors appear in the actual measurement equation.[29] For many spacecraft missions the state errors introduced from the colored measurement process may not cause any concerns; however, other systems may require the need to provide increased accuracy for colored (non-white) errors. Fortunately an exact Kalman filter can still be designed by using *shaping filters* that are driven by zero-mean white-noise processes. However, this is generally at the expense of increased complexity in the filter. Still for many systems a suboptimal filter should have its performance compared with that of the optimal filter.[30]

In this section colored-noise filters are designed for both the process noise and measurement noise. Only discrete-time systems are discussed here since the extension to continuous-time models is fairly straightforward. We first consider the case of a colored process noise. Consider the discrete-time autonomous system given in Table 5.2. Next, we assume that the process noise vector $\mathbf{w}_k$ is not white, but is uncorrelated with the initial condition and measurement noise. A shaping filter for $\mathbf{w}_k$ is given by[5, 30]

$$\chi_{k+1} = \Psi\,\chi_k + \mathcal{V}\,\omega_k \tag{5.219a}$$

$$\mathbf{w}_k = \mathcal{H}\,\chi_k + \mathcal{D}\,\omega_k \tag{5.219b}$$

where χ_k is the shaping filter state, and ω_k is a zero-mean Gaussian white-noise process with covariance given by $\mathcal{Q}$ (in general we can assume that $\mathcal{Q} = I$ and use $\mathcal{D}$ in the filter design to yield identical results for any general covariance matrix).

The system matrices Ψ, $\mathcal{V}$, $\mathcal{H}$, and $\mathcal{D}$ are used to "shape" the process noise into a colored-noise process. The augmented system that includes the state $\mathbf{x}_k$ is given by

$$\boxed{\begin{aligned} \begin{bmatrix} \mathbf{x}_{k+1} \\ \chi_{k+1} \end{bmatrix} &= \begin{bmatrix} \Phi & \Upsilon \mathcal{H} \\ 0 & \Psi \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ \chi_k \end{bmatrix} + \begin{bmatrix} \Gamma \\ 0 \end{bmatrix} \mathbf{u}_k + \begin{bmatrix} \Upsilon \mathcal{D} \\ \mathcal{V} \end{bmatrix} \boldsymbol{\omega}_k && (5.220a) \\ \tilde{\mathbf{y}}_k &= \begin{bmatrix} H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ \chi_k \end{bmatrix} + \mathbf{v}_k && (5.220b) \end{aligned}}$$

The discrete-time Kalman filter in Table 5.2 can now be employed on the augmented system given in eqn. (5.220). Clearly the new system order is equal to the order of the original system plus the order of the shaping filter, which increases the complexity of the filter design. However, better performance may be possible if the shaping filter can adequately "model" the colored-noise process.

We now consider the case of colored measurement noise, where the measurement noise $\mathbf{v}_k$ is modelled by the following shaping filter:

$$\chi_{k+1} = \Psi \chi_k + \mathcal{V} \boldsymbol{\omega}_k \tag{5.221a}$$
$$\mathbf{v}_k = \mathcal{H} \chi_k + \mathcal{D} \boldsymbol{\omega}_k + \boldsymbol{\nu}_k \tag{5.221b}$$

where $\boldsymbol{\omega}_k$ and $\boldsymbol{\nu}_k$ are both zero-mean Gaussian white-noise processes with covariances given by $\mathcal{Q}$ and $\mathcal{R}$, respectively. Assuming that $\boldsymbol{\omega}_k$ and $\boldsymbol{\nu}_k$ are uncorrelated, the new measurement noise covariance is given by

$$\begin{aligned} R &\equiv E\left\{ (\mathcal{D}\boldsymbol{\omega}_k + \boldsymbol{\nu}_k)(\mathcal{D}\boldsymbol{\omega}_k + \boldsymbol{\nu}_k)^T \right\} \\ &= \mathcal{D}\mathcal{Q}\mathcal{D}^T + \mathcal{R} \end{aligned} \tag{5.222}$$

The augmented system that includes the state $\mathbf{x}_k$ is given by

$$\boxed{\begin{aligned} \begin{bmatrix} \mathbf{x}_{k+1} \\ \chi_{k+1} \end{bmatrix} &= \begin{bmatrix} \Phi & 0 \\ 0 & \Psi \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ \chi_k \end{bmatrix} + \begin{bmatrix} \Gamma \\ 0 \end{bmatrix} \mathbf{u}_k + \begin{bmatrix} \Upsilon & 0 \\ 0 & \mathcal{V} \end{bmatrix} \begin{bmatrix} \mathbf{w}_k \\ \boldsymbol{\omega}_k \end{bmatrix} && (5.223a) \\ \tilde{\mathbf{y}}_k &= \begin{bmatrix} H & \mathcal{H} \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ \chi_k \end{bmatrix} + \mathcal{D}\boldsymbol{\omega}_k + \boldsymbol{\nu}_k && (5.223b) \end{aligned}}$$

Assuming that $\mathbf{w}_k$ and $\boldsymbol{\omega}_k$ are uncorrelated, the new process noise covariance matrix is given by

$$\boxed{E\left\{ \begin{bmatrix} \mathbf{w}_k \\ \boldsymbol{\omega}_k \end{bmatrix} \begin{bmatrix} \mathbf{w}_k^T & \boldsymbol{\omega}_k^T \end{bmatrix} \right\} = \begin{bmatrix} Q & 0 \\ 0 & \mathcal{Q} \end{bmatrix}} \tag{5.224}$$

However for the augmented system in eqn. (5.223), the new process noise and measurement noise are now correlated. This correlation is given by

$$\boxed{S \equiv E\left\{ (\mathcal{D}\boldsymbol{\omega}_k + \boldsymbol{\nu}_k) \begin{bmatrix} \mathbf{w}_k^T & \boldsymbol{\omega}_k^T \end{bmatrix} \right\} = \begin{bmatrix} 0 & \mathcal{D}\mathcal{Q} \end{bmatrix}} \tag{5.225}$$

Therefore, the correlated Kalman filter in Table 5.3 should be employed in this case. However, for many practical systems $\mathcal{D} = 0$ so that the standard Kalman filter in Table 5.2 can be used.

As in the colored process noise case the state vector for the colored measurement noise case can also be augmented by the shaping filter state. However, an alternative to this augmentation is possible if the shaping filter can be generated by the following expression:[5]

$$\boxed{\begin{aligned}\chi_{k+1} &= \Psi\,\chi_k + \mathcal{V}\,\omega_k \qquad (5.226a)\\ \mathbf{v}_k &= \chi_k \qquad (5.226b)\end{aligned}}$$

Note that the order of the shaping filter is the same as the dimension of the measurement noise vector. Next we define the following derived measurement:

$$\boxed{\tilde{\gamma}_{k+1} \equiv \tilde{\mathbf{y}}_{k+1} - \Psi\,\tilde{\mathbf{y}}_k - H\,\Gamma\,\mathbf{u}_k} \tag{5.227}$$

Substituting eqn. (5.27b) into eqn. (5.227) gives

$$\tilde{\gamma}_{k+1} = H\,\mathbf{x}_{k+1} + \mathbf{v}_{k+1} - \Psi\,H\,\mathbf{x}_k - \Psi\,\mathbf{v}_k - H\,\Gamma\,\mathbf{u}_k \tag{5.228}$$

Finally substituting eqns. (5.27a) and (5.226) into eqn. (5.228), and collecting terms yields

$$\tilde{\gamma}_{k+1} = \mathcal{H}\,\mathbf{x}_k + \mathcal{V}\,\omega_k + H\,\Upsilon\,\mathbf{w}_k \tag{5.229}$$

where

$$\mathcal{H} \equiv H\,\Phi - \Psi\,H \tag{5.230}$$

Assuming that ω_k and $\mathbf{w}_k$ are uncorrelated, the new measurement noise covariance is given by

$$\boxed{\begin{aligned}R &\equiv E\left\{(\mathcal{V}\,\omega_k + H\,\Upsilon\,\mathbf{w}_k)\,(\mathcal{V}\,\omega_k + H\,\Upsilon\,\mathbf{w}_k)^T\right\}\\ &= \mathcal{V}\,Q\,\mathcal{V}^T + H\,\Upsilon\,Q\,\Upsilon^T\,H^T\end{aligned}} \tag{5.231}$$

However, the new process noise and measurement noise are correlated with

$$\boxed{S \equiv E\left\{(\mathcal{V}\,\omega_k + H\,\Upsilon\,\mathbf{w}_k)\,\mathbf{w}_k^T\right\} = H\,\Upsilon\,Q} \tag{5.232}$$

For this case the correlation S is rarely zero, so the correlated Kalman filter in Table 5.3 needs to be employed. However, the order of the system does not increase, which leads to a computational efficient routine, assuming that the colored measurement noise can be adequately modelled by eqn. (5.226).

Example 5.7: In this example a colored-noise filter will be designed using the longitudinal short-period dynamics of an aircraft. The approximate dynamical equations are given by a harmonic oscillator model:[5, 31]

$$\begin{bmatrix}\dot{\theta}(t)\\ \ddot{\theta}(t)\end{bmatrix} = \begin{bmatrix}0 & 1\\ -\omega_n^2 & -2\,\zeta\,\omega_n\end{bmatrix}\begin{bmatrix}\theta(t)\\ \dot{\theta}(t)\end{bmatrix} + \begin{bmatrix}0\\ 1\end{bmatrix} w(t)$$

$$\tilde{y}(t) = \theta(t) + v(t)$$

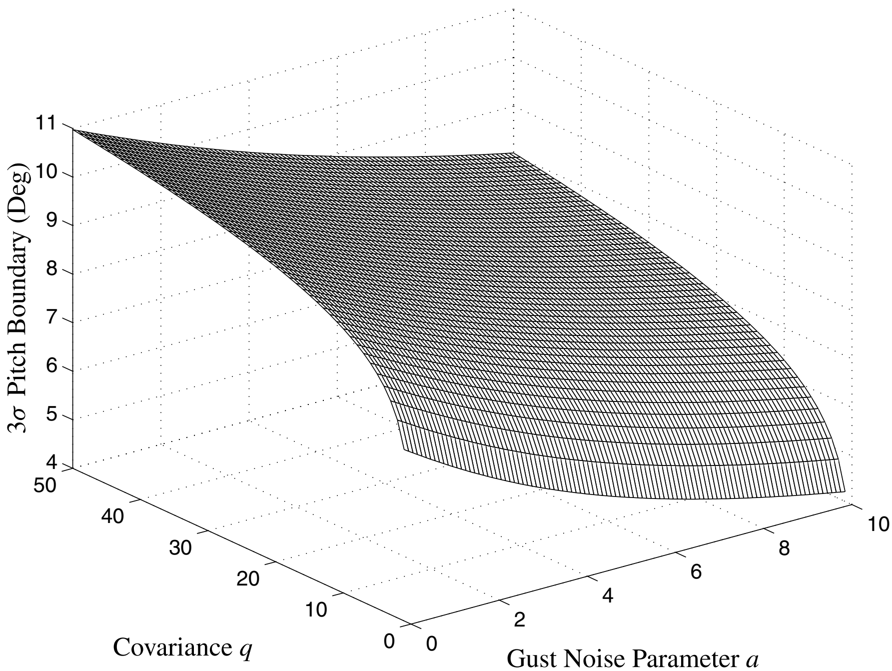

Figure 5.9: Colored-Noise Covariance Analysis

where $\theta(t)$ is the pitch angle, and ω_n and ζ are the short-period natural frequency and damping ratio, respectively. The process noise $w(t)$ now represents a wind gust input that is not white. This gust noise can be approximated by a first-order shaping filter, given by

$$\dot{\chi}(t) = -a\,\chi(t) + \omega(t)$$
$$w(t) = \chi(t)$$

where $\omega(t)$ is a zero-mean Gaussian white-noise process with variance q, and a dictates the "edge" of the gust profile. A larger value of a produces a shaper-edged gust. Also, the takeoff and landing performance of an aircraft can be shown to be a function of the wing loading. Aircraft designed for minimum runway requirements such as short-takeoff-and-landing aircraft will have low wing loadings compared with conventional transport aircraft and, therefore, should be more responsive to wind gusts.[31]

Augmenting the aircraft model by the shaping filter gives the following Kalman filter model-form:

$$\begin{bmatrix} \dot{\theta}(t) \\ \ddot{\theta}(t) \\ \chi(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -\omega_n^2 & -2\zeta\,\omega_n & 1 \\ 0 & 0 & -a \end{bmatrix} \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \\ \chi(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega(t)$$

$$\tilde{y}(t) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \\ \chi(t) \end{bmatrix} + v(t)$$

A discrete-time version of this model can easily be derived with a known sampling rate. As an example of the performance tradeoffs in the colored-noise Kalman filter we will consider the case where $\omega_n = 1$ rad/sec and $\zeta = \sqrt{2}/2$, and the standard deviation of the measurement noise process is given by 1 degree. A plot of the 3σ bound for $\theta(t)$, derived using the steady-state covariance equation in Table 5.2, with various values of a and p is shown in Figure 5.9. Clearly as q decreases more accurate pitch estimates are provided by the Kalman filter, which intuitively makes sense since the magnitude of the wind gust is smaller. As a increases better estimates are also provided. This is due to the effect of the gust edge on the observability of the pitch motion. As a increases more pitch motion from the wind gust is prevalent.

5.7.3 Consistency of the Kalman Filter

As discussed in example 5.5, a tuning process is usually required in the Kalman filter to achieve reasonable state estimates. In this section we show methods that can help answer the question: "what are reasonable estimates?" In practice the truth is never known, but there are still checks available to the design engineer that can (at the very least) provide mechanisms to show that a Kalman filter is not performing in an optimal fashion. For example, several tests can be applied to check the *consistency* of the Kalman filter from the desired characteristics of the measurement residuals. These include: the normalized error square (NES) test, the autocorrelation test, and the normalized mean error (NME) test.[32]

Suppose that some discrete error process $\mathbf{e}_k$ with dimension $m \times 1$ is known to be a zero-mean Gaussian white-noise process with covariance given by E_k. This process may be the state error or the measurement residual in the Kalman filter. Define the following NES:

$$\epsilon_k \equiv \mathbf{e}_k^T E_k^{-1} \mathbf{e}_k \tag{5.233}$$

The NES can be shown to have a chi-square distribution with n degrees of freedom (see Appendix B). A suitable check for the NES is to numerically show that the following condition is met with some level of confidence:

$$E\{\epsilon_k\} = m \tag{5.234}$$

This can be accomplished by using *Hypothesis testing*, which incorporates a degree of plausibility specified by a confidence interval.[33] A 95% confidence interval is most commonly used in practice, which is specified using $100(1-\alpha)$, where $\alpha = 0.05$ in this case. In practice a *two-sided probability region* is used (cutting off both 2.5% tails). Suppose that M Monte Carlo runs are taken, and the following average NES is computed:

$$\boxed{\bar{\epsilon}_k = \frac{1}{M}\sum_{i=1}^{M}\epsilon_k(i) = \frac{1}{M}\sum_{i=1}^{M}\mathbf{e}_k^T(i)\,E_k^{-1}(i)\,\mathbf{e}_k(i)} \tag{5.235}$$

where $\epsilon_k(i)$ denotes the i^{th} run at time t_k. Then $M\bar{\epsilon}_k$ will have a chi-square density with Mm degrees of freedom.[32] This condition can be checked using a *chi-square test*. The hypothesis is accepted if the following condition is satisfied:

$$\bar{\epsilon}_k \in [\zeta_1, \zeta_2] \tag{5.236}$$

where ζ_1 and ζ_2 are derived from the tail probabilities of the chi-square density. For example, for $m = 2$ and $M = 100$, using eqn. (B.51), we have $\chi^2_{Mm}(0.025) = 162$ and $\chi^2_{Mm}(0.975) = 241$. This gives $\zeta_1 = \chi^2_{Mm}(0.025)/M = 1.62$ and $\zeta_2 = \chi^2_{Mm}(0.975)/M = 2.41$.

Another test for consistency is given by a *test for whiteness*. This is accomplished by using the following sample autocorrelation:[32]

$$\boxed{\bar{\rho}_{k,j} = \frac{1}{\sqrt{m}}\sum_{i=1}^{M}\mathbf{e}_k^T(i)\left[\sum_{i=1}^{M}\mathbf{e}_k(i)\mathbf{e}_k^T(i)\sum_{i=1}^{M}\mathbf{e}_k(j)\mathbf{e}_k^T(j)\right]^{-1/2}\mathbf{e}_k(j)} \tag{5.237}$$

For M large enough, $\bar{\rho}_{k,j}$ for $k \neq j$ is zero mean with variance given by $1/M$. A normal approximation can now be used with the central limit theorem.[33] With a 95% acceptance interval we have

$$\bar{\rho}_{k,j} \in \left[-\frac{1.96}{\sqrt{M}}, \frac{1.96}{\sqrt{M}}\right] \tag{5.238}$$

The hypothesis is accepted if eqn. (5.238) is satisfied.

The final consistency test is given by the NME for the j^{th} element of $\mathbf{e}_k$:

$$\boxed{[\bar{\boldsymbol{\mu}}_k]_j = \frac{1}{M}\sum_{i=1}^{M}\frac{[\mathbf{e}_k]_j}{\sqrt{[E_k]_{jj}}}, \quad j = 1, 2, \ldots, m} \tag{5.239}$$

Then, since the variance of $[\bar{\boldsymbol{\mu}}_k]_j$ is $1/M$, for a 95% acceptance interval we have

$$[\bar{\boldsymbol{\mu}}_k]_j \in \left[-\frac{1.96}{\sqrt{M}}, \frac{1.96}{\sqrt{M}}\right] \tag{5.240}$$

The hypothesis is accepted if eqn. (5.240) is satisfied.

The NES, autocorrelation, and NME tests can all be performed with a single run using N data points, which is useful when a set of data cannot be collected more than once. From our example of $m = 2$ with $M = 1$, the two-sided 95% confidence interval is [0.05, 7.38], which is much wider than the $M = 100$ case. This illustrates the variability reduction with multiple runs. A low variability test statistic, which can be executed in real time, can be developed using a time-average approach. The time-average NES is given by

$$\boxed{\bar{\epsilon} = \frac{1}{N}\sum_{k=1}^{N} \mathbf{e}_k^T E_k^{-1} \mathbf{e}_k} \tag{5.241}$$

If $\mathbf{e}_k$ is a zero-mean, white noise process, then $N\bar{\epsilon}$ has a chi-square density distribution with Nm degrees of freedom. The whiteness test for $\mathbf{e}_k$ that are j steps apart, from a single run is derived by computing the time-average autocorrelation:

$$\boxed{\bar{\rho}_j = \frac{1}{\sqrt{n}}\sum_{k=1}^{N} \mathbf{e}_k^T \mathbf{e}_{k+j} \left[\sum_{k=1}^{N} \mathbf{e}_k^T \mathbf{e}_k \sum_{k=1}^{N} \mathbf{e}_{k+j}^T \mathbf{e}_{k+j}\right]^{-1/2}} \tag{5.242}$$

For N large enough, $\bar{\rho}_j$ is zero mean with variance given by $1/N$. With a 95% acceptance interval we have

$$\bar{\rho}_j \in \left[-\frac{1.96}{\sqrt{N}}, \frac{1.96}{\sqrt{N}}\right] \tag{5.243}$$

The hypothesis is accepted if eqn. (5.243) is satisfied. These tests can be applied to the Kalman filter residuals or the state errors through simulated runs to check the necessary consistency for filter optimality. If these tests are not satisfied then the Kalman filter is not running optimally, and the design needs to be investigated to identify the source of the problem for the particular system.

Example 5.8: In this example single run consistency tests will be performed on the residual between a scalar measurement and the estimated output of a Kalman filter. The discrete-time system for this example is given by

$$\mathbf{x}_{k+1} = \begin{bmatrix} 0.9999 & 0.0099 \\ -0.0296 & 0.9703 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} 0 \\ 0.01 \end{bmatrix} w_k$$
$$\tilde{y}_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_k + v_k$$

where the true covariances of w_k and r_k are given by $q = 10$ and $r = 0.01$, respectively. The initial condition is given by $\mathbf{x}_0 = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$. A steady-state Kalman filter shown in Table 5.2 is executed for various values of assumed q with 1001 synthetic measurements. The single run consistency checks involving the time-average NES and autocorrelation tests are performed on the last 500 points, which is well after the filter has converged. With $N = 500$ the two-sided 95% region for the NES test is [0.88, 1.125], and the 95% upper limit for the autocorrelation test is $1.96/\sqrt{500} = 0.0877$.

The true state is always generated using $q = 10$, and the same measurement set is used for the consistency tests. Various values of assumed q in the Kalman filter,

Table 5.10: Results of the Kalman Filter Consistency Tests

q	$\bar{\epsilon}$	$\lvert\bar{\rho}_1\rvert$
0.1	1.9334	0.4752
0.5	1.3501	0.2408
1	1.2065	0.1463
10	1.0367	0.0015
20	1.0231	0.0100
100	1.0006	0.0224
1000	0.9817	0.0424
1×10^4	0.9372	0.0888
1×10^5	0.8607	0.1739

ranging from 0.1 to 1×10^5, are tested. For the consistency tests involving the measurement residual we use $\mathbf{e}_k = \tilde{\mathbf{y}}_k - H_k \mathbf{x}_k$, where for our case $\mathbf{e}_k$, $\tilde{\mathbf{y}}_k$ are scalars and $H_k = \begin{bmatrix} 1 & 0 \end{bmatrix}$. The covariance of $\mathbf{e}_k$, denoted by E_k, can be shown to be given by

$$E_k = H_k^T P_k H_k + R_k$$

which is used in the NES test. Table 5.10 gives numerical values for the computed NES and autocorrelation values. The NES values are outside the region when q is larger than about 1×10^5 or smaller than about 1. The autocorrelation is computed using a one time-step ahead sample. Table 5.10 shows that the autocorrelation test gives about the same level of confidence as the NES test. From both a theoretical and practical point of view the best results are obtained with an autocorrelation near zero and an NES close to one. Table 5.10 indicates that these conditions are met with q values of 10 or 20. Therefore, since the true value of q is 10, we can conclude the consistency tests provide a good means to find q.

5.7.4 Adaptive Filtering

The results of §5.7.3 can be used to manually tune the Kalman filter. In this section a common approach used to automatically identify the process noise and measurement-error noise covariances is shown. The theoretical aspects of the Kalman filter for linear systems are very sound, derived from a rigorous analysis. In practice "tuning" a Kalman filter can be arduous and very time-consuming. Usually, the measurement-error covariance is fairly well known, derived from statistical inferences of the hardware sensing device. However, the process noise covariance is usually not well known and is often derived from experiences gained by the design engineer based on intimate knowledge of the particular system. The approach is based on "residual whitening."[34, 35] The approach presented in this section is applicable to time-invariant systems with stationary noise processes only. Consider the following discrete-time residual equation:

$$\boxed{\begin{aligned}\mathbf{e}_k &\equiv \tilde{\mathbf{y}}_k - H\,\hat{\mathbf{x}}_k^- \\ &= -H\,\tilde{\mathbf{x}}_k^- + \mathbf{v}_k\end{aligned}} \tag{5.244}$$

where eqn. (5.32a) and (5.27b) have been used in eqn. (5.244). The following autocorrelation function matrix can be computed:

$$C_i = \begin{cases} H\,E\left\{\tilde{\mathbf{x}}_k^- \tilde{\mathbf{x}}_{k-i}^{-T}\right\} H^T - H\,E\left\{\tilde{\mathbf{x}}_k^- \mathbf{v}_{k-i}^{-T}\right\} & i > 0 \\ H\,P\,H^T + R & i = 0 \end{cases} \tag{5.245}$$

where $C_i \equiv E\left\{\mathbf{e}_k \mathbf{e}_{k-i}^T\right\}$, and P is the steady-state covariance obtained from

$$P = \Phi\,[(I - K\,H)\,P\,(I - K\,H) + K\,R\,K^T]\,\Phi^T + \Upsilon\,Q\,\Upsilon^T \tag{5.246}$$

Note the use of a suboptimal gain K in eqn. (5.246), but an optimal Q and R.[35] Substituting eqn. (5.37) into eqn. (5.33) leads to

$$\tilde{\mathbf{x}}_k^- = \Phi\,(I - K\,H)\,\tilde{\mathbf{x}}_{k-1}^- + \Phi\,K\,\mathbf{v}_{k-1} - \Upsilon\,\mathbf{w}_{k-1} \tag{5.247}$$

Carrying eqn. (5.247) i steps back yields

$$\begin{aligned}\tilde{\mathbf{x}}_k^- &= [\Phi\,(I - K\,H)]^i\,\tilde{\mathbf{x}}_{k-i}^- + \sum_{j=1}^{i}[\Phi\,(I - K\,H)]^{j-1}\,\Phi\,K\,\mathbf{v}_{k-j} \\ &\quad - \sum_{j=1}^{i}[\Phi\,(I - K\,H)]^{j-1}\,\Upsilon\,\mathbf{w}_{k-j}\end{aligned} \tag{5.248}$$

Then, the following expectations are easily given:

$$E\left\{\tilde{\mathbf{x}}_k^- \tilde{\mathbf{x}}_{k-i}^{-T}\right\} = [\Phi\,(I - K\,H)]^i\,P \tag{5.249a}$$

$$E\left\{\tilde{\mathbf{x}}_k^- \mathbf{v}_{k-i}^{-T}\right\} = [\Phi\,(I - K\,H)]^{i-1}\,\Phi\,K\,R \tag{5.249b}$$

Hence, substituting eqn. (5.249) into eqn. (5.245), the autocorrelation is now given by

$$\boxed{C_i = \begin{cases} H\,[\Phi\,(I - K\,H)]^{i-1}\,\Phi\,[P\,H^T - K\,C_0] & i > 0 \\ H\,P\,H^T + R & i = 0 \end{cases}} \tag{5.250}$$

where the definition of C_0 is used to simplify the resulting substitution process leading to eqn. (5.250). Note that if the optimal gain K is used, given by eqn. (5.50), then $C_i = 0$ for $i \neq 0$.

A test for whiteness can now be computed based on the autocorrelation matrix. Note that if $\mathbf{e}_k$ is a white-noise process, then $C_i = 0$ for $i \neq 0$, which means that the filter is performing in an optimal fashion. An estimate of C_i is given by

$$\boxed{\hat{C}_i = \frac{1}{N}\sum_{j=i}^{N} \mathbf{e}_j \mathbf{e}_{j-i}^T} \tag{5.251}$$

where N is sufficiently large. The estimate for C_i is biased, which can be removed by dividing by $N-i$ instead of N, but the original form may be preferable to an unbiased estimate since less mean-square error is given. The diagonal elements of C_i are of particular interest. These can be normalized by their zero-lag elements leading to the following autocorrelation coefficients:

$$[\rho_i]_{jj} \equiv \frac{[\hat{C}_i]_{jj}}{[\hat{C}_0]_{jj}} \tag{5.252}$$

where the subscript jj denotes a diagonal element of $\hat{C}$. The numbered values for $[\rho_i]_{jj}$ range between 0 and 1. A 95% confidence interval on $[\rho_i]_{jj}$ for $i \neq 0$ is given by

$$\boxed{|[\rho_i]_{jj}| \leq 1.96/N^{1/2}} \tag{5.253}$$

Therefore, if less than 5% of the values of $[\rho_i]_{jj}$ exceed the threshold given by eqn. (5.253), then the j^{th} residual is a white-noise process.

Our first goal is to determine an estimate for $Z \equiv P\,H^T$. Writing out the autocorrelation matrix in eqn. (5.250) for $i > 0$ gives

$$\begin{aligned}
C_1 &= H\,\Phi\,P\,H^T - H\,\Phi\,K\,C_0 \\
C_2 &= H\,\Phi^2 P\,H^T - H\,\Phi\,K\,C_1 - H\,\Phi^2 K\,C_0 \\
&\;\vdots \\
C_n &= H\,\Phi^n P\,H^T - H\,\Phi\,K\,C_{n-1} - \cdots - H\,\Phi^n K\,C_0
\end{aligned} \tag{5.254}$$

Using the methods of Chapter 1 the following least-squares estimate for Z is obtained:

$$\hat{Z} = (M^T M)^{-1} M^T \begin{bmatrix} \hat{C}_1 + H\,\Phi\,K\,\hat{C}_0 \\ \hat{C}_2 + H\,\Phi\,K\,\hat{C}_1 + H\,\Phi^2 K\,\hat{C}_0 \\ \vdots \\ \hat{C}_n + H\,\Phi\,K\,\hat{C}_{n-1} + \cdots + H\,\Phi^n K\,\hat{C}_0 \end{bmatrix} \tag{5.255}$$

where M is the product of the observability matrix in eqn. (3.117) and the transition matrix Φ, i.e., $M \equiv \mathcal{O}_d \Phi$. Note that the dynamical system must be observable in order for the inverse in eqn. (5.255) to exist. Therefore, using eqn. (5.250) an estimate for R is given by

$$\boxed{\hat{R} = \hat{C}_0 - H\,\hat{Z}} \tag{5.256}$$

Determining an estimate for Q is not as straightforward as the R case. If the number of unknown elements of Q is $n\times m$ or less, then a unique solution is possible. We first rewrite eqn. (5.246) as

$$P = \Phi\, P\Phi^T + \Omega + \Upsilon\, Q\, \Upsilon^T \tag{5.257}$$

where

$$\Omega \equiv \Phi\,[K\, C_0 K^T - P\, H^T K^T - K\, H\, P]\,\Phi^T \tag{5.258}$$

Substituting back for P n times on the right-hand side of eqn. (5.257) yields

$$\sum_{j=0}^{i-1} \Phi^j \Upsilon\, Q\, \Upsilon^T (\Phi^j)^T = P - \Phi^i P\,(\Phi^i)^T - \sum_{j=0}^{i-1} \Phi^j \Omega\,(\Phi^j)^T, \quad i = 1, 2, \ldots, n \tag{5.259}$$

Multiplying the left-hand side of eqn. (5.259) by H and multiplying the right-hand side by $(\Phi^{-i})^T H^T$, and using estimated quantities leads to

$$\boxed{\begin{aligned}\sum_{j=0}^{i-1} H\Phi^j \Upsilon\, \hat{Q}\, \Upsilon^T (\Phi^{j-i})^T H^T &= \hat{Z}^T (\Phi^{-i})^T H^T - H\,\Phi^i \hat{Z} \\ &\quad - \sum_{j=0}^{i-1} H\,\Phi^j \hat{\Omega}\,(\Phi^{j-i})^T H^T, \quad i = 1, 2, \ldots, n\end{aligned}} \tag{5.260}$$

where

$$\hat{\Omega} \equiv \Phi\,[K\, \hat{C}_0 K^T - \hat{Z}\, K^T - K\, \hat{Z}^T]\,\Phi^T \tag{5.261}$$

Once the right-hand side of eqn. (5.260) has been evaluated then $\hat{Q}$ can be extracted. Note that the equations for the elements of $\hat{Q}$ are not linearly independent, and one has to choose a linearly independent subset of these equations.[34]

If the number of unknown elements of Q is greater than $n \times m$, then a unique solution is not possible. To overcome this case, the gain K can be estimated directly, which is denoted by K^*. Then the optimal covariance P^* follows

$$P^* = \Phi(P^* - K^* H\, P^*)\Phi^T + \Upsilon\, Q\, \Upsilon^T \tag{5.262}$$

Defining $\delta P = P^* - P$ and using eqns. (5.246) and (5.262) yields[35]

$$\begin{aligned}\delta P = \Phi\Big[&\delta P - (P\, H^T + \delta P\, H^T)\,(C_0 + H\,\delta P\, H^T)^{-1}(H\, P + H\,\delta P) \\ &+ K\, H\, P + P\, H^T K^T - K\, C_0\, K^T\Big]\,\Phi^T\end{aligned} \tag{5.263}$$

where $C_0 = H\, P\, H^T + R$ is used to eliminate R. An optimal estimate for δP, denoted by $\delta\hat{P}$ is obtained by using $\hat{C}_0$ from eqn. (5.251) and $\hat{Z}$ from eqn. (5.255), so that

$$\begin{aligned}\delta\hat{P} = \Phi\Big[&\delta P - (\hat{Z} + \delta P\, H^T)\,(\hat{C}_0 + H\,\delta P\, H^T)^{-1}(\hat{Z}^T + H\,\delta P) \\ &+ K\,\hat{Z}^T + \hat{Z}\, K^T - K\,\hat{C}_0\, K^T\Big]\,\Phi^T\end{aligned} \tag{5.264}$$

which can now be solved for $\delta\hat{P}$. The optimal gain is given by

$$\begin{aligned} K^* &= P^* H^T [H\, P^* H^T + R]^{-1} \\ &= \left[(P + \delta P)\, H^T\right]\left[H\, P\, H^T + H\, \delta P\, H^T + R\right]^{-1} \\ &= \left[P\, H^T + \delta P\, H^T\right]\left[C_0 + H\, \delta P\, H^T\right]^{-1} \end{aligned} \tag{5.265}$$

Therefore the estimate of the optimal gain is given by

$$\boxed{\hat{K}^* = \left[\hat{Z} + \delta\hat{P}\, H^T\right]\left[\hat{C}_0 + H\, \delta\hat{P}\, H^T\right]^{-1}} \tag{5.266}$$

For batch-type applications, local iterations on the estimates $\hat{C}_0$, $\hat{Z}$, $\delta\hat{P}$, and $\hat{K}^*$ are possible on the same set of N measurements, which could improve these estimates, where the residual sequence becomes increasingly more white.[35] Also, care must be given when estimating for the gain directly since no guarantees can be made about the stability of the resulting filter. Reference [34] provides an example involving an inertial navigation problem to estimate components of the matrices Q and R. Asymptotic convergence of the estimates toward their true values has been shown in this example. Other adaptive methods, such as covariance matching, can be found in Refs. [4] and [35].

5.7.5 Error Analysis

The optimality of the Kalman filter hinges on many factors. First, although precise knowledge of the process noise and measurements inputs is not required, we must have accurate knowledge of their respective covariance values. When these covariances are not well known then the methods in §5.7.4 can be applied to estimate them online. Also, errors in the assumed model may be present. Determining these errors is a formidable (and nearly impossible!) task. This section shows an analysis on how the error-covariance of the nominal system is changed with the aforementioned errors. This new covariance can be used to assess the performance of the nominal Kalman filter given bounds on the model and noises quantities, which may provide insight to filter performance and sensitivity to various errors. The development in this section is based on continuous-time models and measurements. Also, in this section we eliminate the explicit dependence on time for notational brevity. Consider the following nominal system, which will be used to derive the Kalman filter:

$$\dot{\bar{\mathbf{x}}} = \bar{F}\, \bar{\mathbf{x}} + B\, \mathbf{u} + \bar{G}\, \bar{\mathbf{w}} \tag{5.267a}$$

$$\tilde{\mathbf{y}} = \bar{H}\, \bar{\mathbf{x}} + \bar{\mathbf{v}} \tag{5.267b}$$

where $\bar{F}$, $\bar{G}$, and $\bar{H}$ are the nominal model matrices (note we assume that the control input and its associated input matrix are known exactly). The Kalman filter for this system is given by

$$\boxed{\dot{\hat{\bar{\mathbf{x}}}} = \bar{F}\, \hat{\bar{\mathbf{x}}} + B\, \mathbf{u} + \bar{K}[\tilde{\mathbf{y}} - \bar{H}\, \hat{\bar{\mathbf{x}}}]} \tag{5.268}$$

with

$$\bar{K} = \bar{P}\,\bar{H}^T\,\bar{R}^{-1} \tag{5.269a}$$

$$\dot{\bar{P}} = \bar{F}\,\bar{P} + \bar{P}\,\bar{F}^T - \bar{P}\,\bar{H}^T\,\bar{R}^{-1}\bar{H}\,\bar{P} + \bar{G}\,\bar{Q}\,\bar{G}^T \tag{5.269b}$$

where $\bar{Q}$ and $\bar{R}$ are the nominal process noise and measurement noise covariances, respectively.

The actual system is given by

$$\dot{\mathbf{x}} = F\,\mathbf{x} + B\,\mathbf{u} + G\,\mathbf{w} \tag{5.270a}$$

$$\tilde{\mathbf{y}} = H\,\mathbf{x} + \mathbf{v} \tag{5.270b}$$

We now define the following variables: $\tilde{\bar{\mathbf{x}}} \equiv \mathbf{x} - \hat{\bar{\mathbf{x}}}$, $\Delta F \equiv F - \bar{F}$, and $\Delta H \equiv H - \bar{H}$, where $\tilde{\bar{\mathbf{x}}}$ is the error between the truth and the estimate using the assumed nominal model. Taking the time derivative of $\tilde{\bar{\mathbf{x}}}$ yields

$$\dot{\tilde{\bar{\mathbf{x}}}} = (\bar{F} - \bar{K}\,\bar{H})\,\tilde{\bar{\mathbf{x}}} + (\Delta F - \bar{K}\,\Delta H)\mathbf{x} + G\,\mathbf{w} - \bar{K}\,\mathbf{v} \tag{5.271}$$

The covariance of $\tilde{\bar{\mathbf{x}}}$ can be shown to be given by[36, 37]

$$\boxed{P_{\tilde{\mathbf{x}}} = V_{\tilde{\mathbf{x}}} + \boldsymbol{\mu}_{\tilde{\mathbf{x}}}\boldsymbol{\mu}_{\tilde{\mathbf{x}}}^T} \tag{5.272}$$

where

$$\boxed{\begin{aligned} \dot{V}_{\tilde{\mathbf{x}}} &= (\bar{F} - \bar{K}\,\bar{H})\,V_{\tilde{\mathbf{x}}} + V_{\tilde{\mathbf{x}}}(\bar{F} - \bar{K}\,\bar{H})^T + V^T(\Delta F - \bar{K}\,\Delta H)^T \\ &\quad + (\Delta F - \bar{K}\,\Delta H)\,V + G\,Q\,G^T + \bar{K}\,R\,\bar{K}^T \end{aligned}} \tag{5.273}$$

The matrix V is determined from

$$\boxed{\begin{aligned} \dot{V} &= F\,V + V\,(\bar{F} - \bar{K}\,\bar{H})^T + V_{\mathbf{x}}(\Delta F - \bar{K}\,\Delta H)^T + G\,Q\,G^T & (5.274a)\\ \dot{V}_{\mathbf{x}} &= F\,V_{\mathbf{x}} + V_{\mathbf{x}}F^T + G\,Q\,G^T & (5.274b) \end{aligned}}$$

The mean of the estimation error $\boldsymbol{\mu}_{\tilde{\mathbf{x}}}$ is determined from

$$\boxed{\begin{aligned} \dot{\boldsymbol{\mu}}_{\tilde{\mathbf{x}}} &= (\bar{F} - \bar{K}\,\bar{H})\,\boldsymbol{\mu}_{\tilde{\mathbf{x}}} + (\Delta F - \bar{K}\,\Delta H)\,\boldsymbol{\mu}_{\mathbf{x}} & (5.275a)\\ \dot{\boldsymbol{\mu}}_{\mathbf{x}} &= F\,\boldsymbol{\mu}_{\mathbf{x}} & (5.275b) \end{aligned}}$$

where $\boldsymbol{\mu}_{\mathbf{x}}$ is the system mean. The initial conditions for the differential equations are left to the discretion of the filter designer.

The procedure to determine $P_{\tilde{\mathbf{x}}}$ is as follows. First, compute $\boldsymbol{\mu}_{\mathbf{x}}$ and $V_{\mathbf{x}}$ using eqns. (5.275b) and (5.274b), respectively. Note these variables require knowledge of the true system matrices. Then compute $\boldsymbol{\mu}_{\tilde{\mathbf{x}}}$ and V using eqns. (5.275a) and (5.274a), respectively. Next, compute $V_{\tilde{\mathbf{x}}}$ using eqn. (5.273), and finally compute $P_{\tilde{\mathbf{x}}}$ using eqn. (5.272). Note that if $\Delta F - \bar{K}\,\Delta H = 0$, then both V and $V_{\mathbf{x}}$ do not need to be computed. A more useful quantity involves rewriting eqn. (5.272) as

$$\boxed{P_{\tilde{\mathbf{x}}} = \bar{P} + \Delta V_{\tilde{\mathbf{x}}} + \boldsymbol{\mu}_{\tilde{\mathbf{x}}}\boldsymbol{\mu}_{\tilde{\mathbf{x}}}^T} \tag{5.276}$$

where $(\Delta V_{\tilde{\mathbf{x}}} + \boldsymbol{\mu}_{\tilde{\mathbf{x}}}\boldsymbol{\mu}_{\tilde{\mathbf{x}}}^T)$ is now the covariance difference between total error-covariance and the nominal error-covariance. The quantity $\Delta V_{\tilde{\mathbf{x}}}$ can be found from

$$\boxed{\begin{aligned}\Delta V_{\tilde{\mathbf{x}}} &= (\bar{F} - \bar{K}\,\bar{H})\,\Delta V_{\tilde{\mathbf{x}}} + V_{\tilde{\mathbf{x}}}(\bar{F} - \bar{K}\,\bar{H})^T + V^T(\Delta F - \bar{K}\,\Delta H)^T \\ &\quad + (\Delta F - \bar{K}\,\Delta H)\,V + (G\,Q\,G^T - \bar{G}\,\bar{Q}\,\bar{G}^T) + \bar{K}\,(R - \bar{R})\,\bar{K}^T\end{aligned}} \tag{5.277}$$

Under steady-state conditions, for time-invariant stable systems, both $\boldsymbol{\mu}_{\mathbf{x}}$ and $\boldsymbol{\mu}_{\tilde{\mathbf{x}}}$ are zero. Also, eqns. (5.274b) and (5.272) can be found using an algebraic Lyapunov equation, which has the same form as given by eqn. (3.125). Other forms can be given in which the system estimation error is separated into optimum and non-optimum error components.[36, 37]

5.7.6 Unscented Filtering

The problem of filtering using nonlinear dynamic and/or measurement models is inherently more difficult than for the case of linear models. The extended Kalman filter in §5.6 typically works well only in the region where the first-order Taylor-series linearization adequately approximates the nonlinear probability distribution. The primary area of concern for this application is during the initialization stage, where the estimated initial state may be far from the true state. This may lead to instabilities in the extended Kalman filter. To overcome these instabilities a Kalman filter can be used based upon including second-order terms in the Taylor-series.[1, 35] Improved performance can be achieved in many cases, but at the expense of an increased computational burden. Maybeck[35] also suggests that a first-order filter with bias correction terms, without altering the covariance and gain expressions, may be generated to obtain the essential benefits of second-order filtering with the computational penalty of additional second-moment calculations. An exact nonlinear filter has been developed by Daum,[38] which reduces to the standard Kalman filter in linear systems. However, Daum's theory may be difficult to implement on practical systems due to the nature of the requirement to solve a partial differential equation (known as the Fokker-Planck equation). Therefore, the standard form of the extended Kalman filter has remained the most popular method for nonlinear estimation to this day, and other designs are investigated only when the performance of the standard form is not sufficient.

In this section a new approach that has been developed by Julier, Uhlmann, and Durrant-Whyte[39, 40] is shown as an alternative to the extended Kalman filter. This approach, which they called the *Unscented filter* (UF), typically involves more computations than the extended Kalman filter, but has several advantages, including: 1) the expected error is lower than the extended Kalman filter, 2) the new filter can be applied to non-differentiable functions, 3) the new filter avoids the derivation of Jacobian matrices, and 4) the new filter is valid to higher-order expansions than the standard extended Kalman filter. The Unscented filter works on the premise that with a fixed number of parameters it should be easier to approximate a Gaussian distribution than to approximate an arbitrary nonlinear function. The filter presented in Ref. [39] is derived for discrete-time nonlinear equations, where the system model is given by

$$\boxed{\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k, \mathbf{w}_k, \mathbf{u}_k, k) \qquad (5.278\text{a}) \\ \tilde{\mathbf{y}}_k &= \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k, k) \qquad (5.278\text{b}) \end{aligned}}$$

Note that a continuous-time model can always be written using eqn. (5.278a) through an appropriate numerical integration scheme. It is again assumed that $\mathbf{w}_k$ and $\mathbf{v}_k$ are zero-mean Gaussian noise processes with covariances given by Q_k and R_k, respectively. We first rewrite the Kalman filter update equations in Table 5.9 as[41]

$$\boxed{\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k \boldsymbol{\upsilon}_k} \tag{5.279a}$$

$$\boxed{P_k^+ = P_k^- - K_k P_k^{\upsilon\upsilon} K_k^T} \tag{5.279b}$$

where $\boldsymbol{\upsilon}_k$ is the *innovations process*, given by

$$\begin{aligned} \boldsymbol{\upsilon}_k &\equiv \tilde{\mathbf{y}}_k - \hat{\mathbf{y}}_k^- \\ &= \tilde{\mathbf{y}}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-, \mathbf{u}_k, k) \end{aligned} \tag{5.280}$$

The covariance of $\boldsymbol{\upsilon}_k$ is defined by $P_k^{\upsilon\upsilon}$. The gain K_k is computed by

$$\boxed{K_k = P_k^{xy} (P_k^{\upsilon\upsilon})^{-1}} \tag{5.281}$$

where P_k^{xy} is the cross-correlation matrix between $\hat{\mathbf{x}}_k^-$ and $\hat{\mathbf{y}}_k^-$.

The Unscented filter uses a different propagation than the form given by the standard extended Kalman filter. Given an $n \times n$ covariance matrix P, a set of order n points can be generated from the columns (or rows) of the matrices $\pm\sqrt{nP}$. The set of points is zero-mean, but if the distribution has mean $\boldsymbol{\mu}$, then simply adding $\boldsymbol{\mu}$ to each of the points yields a symmetric set of $2n$ points having the desired mean and covariance.[39] Due to the symmetric nature of this set, its odd central moments are zero, so its first three moments are the same as the original Gaussian distribution. This is the foundation for the Unscented filter. A complete derivation of this filter is beyond the scope of the present text, so only the final results are presented here. Various methods can be used to handle the process noise and measurement noise in the Unscented filter. One approach involves augmenting the covariance matrix with

$$P_k^a = \begin{bmatrix} P_k^+ & P_k^{xw} & P_k^{xv} \\ (P_k^{xw})^T & Q_k & P_k^{wv} \\ (P_k^{xv})^T & (P_k^{wv})^T & R_k \end{bmatrix} \tag{5.282}$$

where P_k^{xw} is the correlation between the state error and process noise, P_k^{xv} is the correlation between the state error and measurement noise, and P_k^{wv} is the correlation between the process noise and measurement noise, which are all zero for most systems. Augmenting the covariance requires the computation of $2(q+l)$ additional sigma points (where q is the dimension of $\mathbf{w}_k$ and l is the dimension of $\mathbf{v}_k$, which does not necessarily have to be the same dimension, m, as the output in this case), but the effects of the process and measurement noise in terms of the impact on the mean and covariance are introduced with the same order of accuracy as the uncertainty in the state.

The general formulation for the propagation equations are given as follows. First, the following set of *sigma points* are computed:

$$\boldsymbol{\sigma}_k \leftarrow 2L \text{ columns from } \pm\gamma\sqrt{P_k^a} \tag{5.283a}$$

$$\boldsymbol{\chi}_k^a(0) = \hat{\mathbf{x}}_k^a \tag{5.283b}$$

$$\boldsymbol{\chi}_k^a(i) = \boldsymbol{\sigma}_k(i) + \hat{\mathbf{x}}_k^a \tag{5.283c}$$

where $\hat{\mathbf{x}}_k^a$ is an augmented state defined by

$$\mathbf{x}_k^a = \begin{bmatrix} \mathbf{x}_k \\ \mathbf{w}_k \\ \mathbf{v}_k \end{bmatrix}, \quad \hat{\mathbf{x}}_k^a = \begin{bmatrix} \hat{\mathbf{x}}_k \\ \mathbf{0}_{q\times 1} \\ \mathbf{0}_{m\times 1} \end{bmatrix} \tag{5.284}$$

and L is the size of the vector $\hat{\mathbf{x}}_k^a$. The parameter γ is given by

$$\gamma = \sqrt{L+\lambda} \tag{5.285}$$

where the composite scaling parameter, λ, is given by

$$\lambda = \alpha^2(L+\kappa) - L \tag{5.286}$$

The constant α determines the spread of the sigma points and is usually set to a small positive value (e.g., $1\times 10^{-4} \le \alpha \le 1$).[42] Also, the significance of the parameter κ will be discussed shortly. Efficient methods to compute the matrix square root can be found by using the Cholesky decomposition (see Appendix A) or using eqn. (5.209). If an orthogonal matrix square root is used, then the sigma points lie along the eigenvectors of the covariance matrix. Note that there are a total of $2L$ values for $\boldsymbol{\sigma}_k$ (the positive and negative square roots). The transformed set of sigma points are evaluated for each of the points by

$$\boldsymbol{\chi}_{k+1}(i) = \mathbf{f}(\boldsymbol{\chi}_k^x(i), \boldsymbol{\chi}_k^w(i), \mathbf{u}_k, k) \tag{5.287}$$

where $\boldsymbol{\chi}_k^x(i)$ is a vector of the first n elements of $\boldsymbol{\chi}_k^a(i)$, and $\boldsymbol{\chi}_k^w(i)$ is a vector of the next q elements of $\boldsymbol{\chi}_k^a(i)$, with

$$\boldsymbol{\chi}_k^a(i) = \begin{bmatrix} \boldsymbol{\chi}_k^x(i) \\ \boldsymbol{\chi}_k^w(i) \\ \boldsymbol{\chi}_k^v(i) \end{bmatrix} \tag{5.288}$$

where $\chi_k^v(i)$ is a vector of the last l elements of $\chi_k^a(i)$, which will be used to compute the output covariance. We now define the following weights:

$$W_0^{\text{mean}} = \frac{\lambda}{L+\lambda} \tag{5.289a}$$

$$W_0^{\text{cov}} = \frac{\lambda}{L+\lambda} + (1-\alpha^2+\beta) \tag{5.289b}$$

$$W_i^{\text{mean}} = W_i^{\text{cov}} = \frac{1}{2(L+\lambda)}, \quad i = 1, 2, \ldots, 2L \tag{5.289c}$$

where β is used to incorporate prior knowledge of the distribution (a good starting guess is $\beta = 2$).

The predicted mean for the state estimate is calculated using a weighted sum of the points $\chi_{k+1}^x(i)$, which is given by

$$\boxed{\hat{\mathbf{x}}_{k+1}^- = \sum_{i=0}^{2L} W_i^{\text{mean}} \chi_{k+1}^x(i)} \tag{5.290}$$

The predicted covariance is given by

$$\boxed{P_{k+1}^- = \sum_{i=0}^{2L} W_i^{\text{cov}} [\chi_{k+1}^x(i) - \hat{\mathbf{x}}_{k+1}^-][\chi_{k+1}^x(i) - \hat{\mathbf{x}}_{k+1}^-]^T} \tag{5.291}$$

The mean observation is given by

$$\boxed{\hat{\mathbf{y}}_{k+1}^- = \sum_{i=0}^{2L} W_i^{\text{mean}} \gamma_{k+1}(i)} \tag{5.292}$$

where

$$\gamma_{k+1}(i) = \mathbf{h}(\chi_{k+1}^x(i), \mathbf{u}_{k+1}, \chi_{k+1}^v(i), k+1) \tag{5.293}$$

The output covariance is given by

$$\boxed{P_{k+1}^{yy} = \sum_{i=0}^{2L} W_i^{\text{cov}} [\gamma_{k+1}(i) - \hat{\mathbf{y}}_{k+1}^-][\gamma_{k+1}(i) - \hat{\mathbf{y}}_{k+1}^-]^T} \tag{5.294}$$

Then the innovations covariance is simply given by

$$\boxed{P_{k+1}^{\upsilon\upsilon} = P_{k+1}^{yy}} \tag{5.295}$$

Finally the cross correlation matrix is determined using

$$\boxed{P_{k+1}^{xy} = \sum_{i=0}^{2L} W_i^{\text{cov}} [\chi_{k+1}^x(i) - \hat{\mathbf{x}}_{k+1}^-][\gamma_{k+1}(i) - \hat{\mathbf{y}}_{k+1}^-]^T} \tag{5.296}$$

The filter gain is then computed using eqn. (5.281), and the state vector can now be updated using eqn. (5.279). Even though propagations on the order of $2n$ are required for the Unscented filter, the computations may be comparable to the extended Kalman filter (especially if the continuous-time covariance equation needs to be integrated and a numerical Jacobian matrix is evaluated). Also, if the measurement noise, $\mathbf{v}_k$, appears linearly in the output (with $l = m$), then the augmented state can be reduced because the system state does not need to augmented with the measurement noise. In this case the covariance of the measurement error is simply added to the innovations covariance, with $P^{\upsilon\upsilon}_{k+1} = P^{yy}_{k+1} + R_{k+1}$. This can greatly reduce the computational requirements in the Unscented filter.

The scalar κ in the previous set of equations is a convenient parameter for exploiting knowledge (if available) about the higher moments of the given distribution.[41] In scalar systems (i.e., for $L = 1$), a value of $\kappa = 2$ leads to errors in the mean and variance that are sixth order. For higher-dimensional systems choosing $\kappa = 3 - L$ minimizes the mean-squared-error up to the fourth order.[39] However, caution should be exercised when κ is negative since a possibility exists that the predicted covariance can become non-positive semi-definite. A modified form has been suggested for this case (see Ref. [39]). Also, a square root version of the Unscented filter is presented in Ref. [42] that avoids the need to re-factorize at each step. Furthermore, Ref. [42] presents an Unscented Particle filter, which makes no assumptions on the form of the probability densities, i.e., full nonlinear, non-Gaussian estimation.

Example 5.9: In this example a comparison is made between the extended Kalman filter and the Unscented filter to estimate the altitude, velocity, and ballistic coefficient of a vertically falling body.[43] The geometry of the problem is shown in Figure 5.10, where $x_1(t)$ is the altitude, $x_2(t)$ is the downward velocity, $r(t)$ is the range (measured by a radar), M is the horizontal distance, and Z is the radar altitude. The truth model is given by

$$\begin{aligned}
\dot{x}_1(t) &= -x_2(t) \\
\dot{x}_2(t) &= -e^{-\alpha\, x_1(t)} x_2^2(t)\, x_3(t) \\
\dot{x}_3(t) &= 0
\end{aligned}$$

where $x_3(t)$ is the (constant) ballistic coefficient and α is a constant (5×10^{-5}) that relates the air density with altitude. The discrete-time range measurement at time t_k is given by

$$\tilde{y}_k = \sqrt{M^2 + (x_{1_k} - Z)^2} + \upsilon_k$$

where the variance of υ_k is given by 1×10^4, and $M = Z = 1 \times 10^5$. Note that the dynamic model contains no process noise so that $Q_k = 0$.

The extended Kalman filter requires various partials to be computed. The matrix F from Table 5.9 is given by

$$F = e^{-\alpha\,\hat{x}_1} \begin{bmatrix} 0 & -e^{\alpha\,\hat{x}_1} & 0 \\ \alpha\,\hat{x}_2^2\hat{x}_3 & -2\hat{x}_2\hat{x}_3 & -\hat{x}_2^2 \\ 0 & 0 & 0 \end{bmatrix}$$

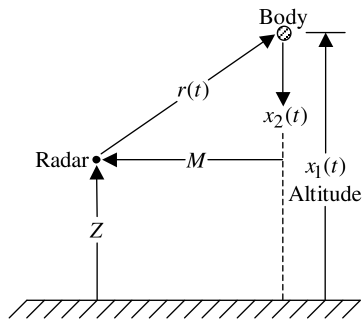

Figure 5.10: Vertically Falling Body Example

The matrix H is given by

$$H = \left[\frac{\hat{x}_1 - Z}{\sqrt{M^2 + (\hat{x}_1 - Z)^2}} \; 0 \; 0 \right]$$

The Kalman filter covariance propagation is carried out by converting F into discrete-time form with the known sampling interval, using eqn. (5.35) to propagate to P_{k+1}^-. For the Unscented filter, since $n = 3$ then $\kappa = 0$, which minimizes the maximum error up to fourth order. The true state and initial estimates are given by

$$\begin{aligned} x_1(0) &= 3 \times 10^5 & \qquad \hat{x}_1(0) &= 3 \times 10^5 \\ x_2(0) &= 2 \times 10^4 & \qquad \hat{x}_2(0) &= 2 \times 10^4 \\ x_3(0) &= 1 \times 10^{-3} & \qquad \hat{x}_3(0) &= 3 \times 10^{-5} \end{aligned}$$

Clearly, an error is present in the ballistic coefficient value. Physically this corresponds to assuming that the body is "heavy" whereas in reality the body is "light." The initial covariance for both filters is given by

$$P(0) = \begin{bmatrix} 1 \times 10^6 & 0 & 0 \\ 0 & 4 \times 10^6 & 0 \\ 0 & 0 & 1 \times 10^{-4} \end{bmatrix}$$

Measurements are sampled at 1-second intervals. In the original test[43] all differential equations were integrated using a fourth-order Runge-Kutta method with a step size of 1/64 second. In our simulations only the truth trajectory has been generated in this manner. The integration step size in both filters has been set to the measurement sample interval (1 second), which further stresses both filters.

Figure 5.11 depicts the average magnitude of the position error by each filter using a Monte Carlo simulation consisting of 100 runs. At the beginning stage where the altitude is high there is little difference between both filters. We should note that correct estimation of x_3 cannot take place at high altitudes due to the small air density.[43] The most severe nonlinearities start taking effect at about 9 seconds,

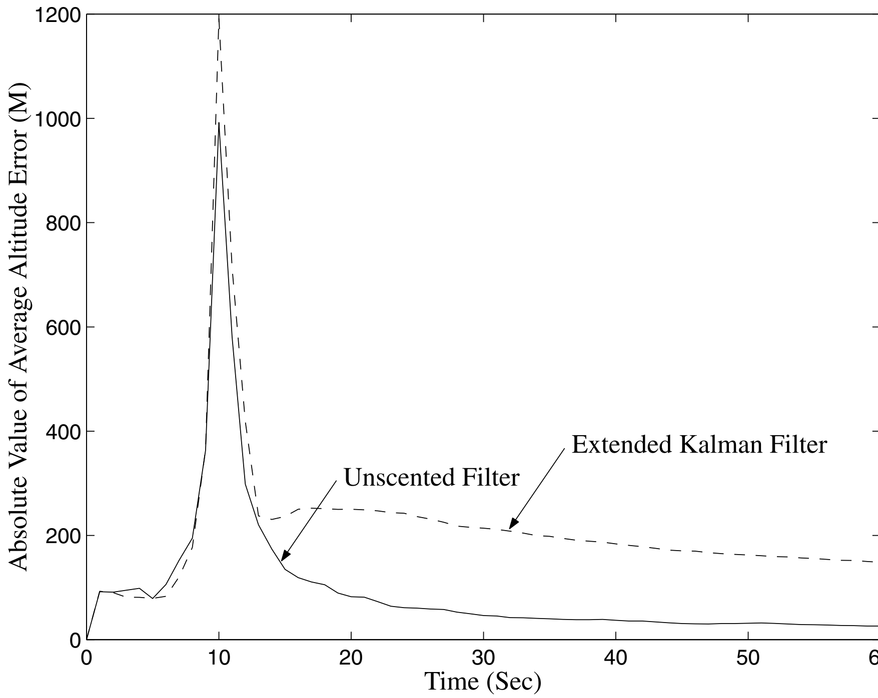

Figure 5.11: Absolute Mean Position Error

where the effects of drag become significant. Large errors are present in both filters, which corresponds to the time when the altitude of the body is the same as the radar (this occurs at 10 seconds where the system is nearly unobservable). However, the Unscented filter has a smaller error-spike than the extended Kalman filter. Finally, the extended Kalman filter converges much slower than the Unscented filter, which is due to the highly nonlinear nature of the model. Similar results are also obtained for the other states. For the x_3 state the extended Kalman filter converges to an order of magnitude larger than the Unscented filter, which attests to the power of using the Unscented filter for highly nonlinear systems.

5.7.7 Robust Filtering

The design of robust filters attempts to maintain filter responses and error signals to within some tolerances despite the effects of uncertainty on the system. Uncertainty may take many forms, but among the most common involve noise (structural) uncertainty and system model uncertainties. The basic idea of one of these designs, called H_∞ filtering, minimizes a "worst-case" loss function, which can be shown to be a *minimax problem* where the maximum "energy" in the error is minimized over all noise trajectories that lead to the same problem.[44] Unfortunately the mathematics behind this theory is intense, involving Hilbert spaces, and well beyond the present text. Therefore, only a brief introduction is presented here.

A good introduction to the H_∞ theory is provided in Refs. [45] and [46]. Before we present the main results of robust filtering we first give an introduction to the operator norms $||G(s)||_2$ and $||G(s)||_\infty$, where $G(s)$ is a proper rational transfer function. The 2-norm of $G(s)$ is defined by

$$||G(s)||_2 = \left\{ \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{Tr}[G(j\omega)\, G^T(-j\omega)]\, d\omega \right\}^{1/2} \tag{5.297}$$

The ∞-norm of $G(s)$ is defined by

$$||G(s)||_\infty = \sup_\omega \bar{\sigma}[G(j\omega)] \tag{5.298}$$

where sup denotes the supremum and $\bar{\sigma}$ is the largest singular value. One way to compute $||G(s)||_\infty$ is to take the supremum of the largest singular value of $[G(j\omega)]$ over all frequencies ω. Also, from $y(s) = G(s)\, u(s)$, if $||u(s)||_2 < \infty$ and $G(s)$ is proper with no poles on the imaginary axis, then[45]

$$||G(s)||_\infty = \sup_u \frac{||y||_2}{||u||_2} \tag{5.299}$$

A closed-form solution for computing $||G(s)||_2$ is possible, derived using either the controllability or observability Gramians, but a closed-form solution of $||G(s)||_\infty$ is not possible in general. Consider the state-space representation of $G(s)$, given by eqn. (3.11). The procedure to compute $||G(s)||_\infty$ involves searching for the scalar $\gamma > 0$ that yields $||G(s)||_\infty < \gamma$, if and only if $\bar{\sigma}(D) < \gamma$ and the following matrix has no eigenvalues on the imaginary axis:

$$\mathcal{H} \equiv \begin{bmatrix} F + B\,W^{-1} D^T H & B\,W^{-1} B^T \\ -H^T (I + D\,W^{-1} D^T)\, H & -(A + B\,W^{-1} D^T H)^T \end{bmatrix} \tag{5.300}$$

where $W = \gamma^2 I - D^T D$. A proof of this result can be found in Ref. [46]. An iterative solution for γ can be found using a bisection algorithm.[46]

The filtering results presented in this section involve continuous-time models and measurements. Discrete-time systems are discussed in Ref. [8]. We first rewrite the system in eqn. (5.117) as

$$\dot{\mathbf{x}}(t) = F(t)\,\mathbf{x}(t) + B(t)\,\mathbf{u}(t) + G(t)\,\mathbf{w}(t) \tag{5.301a}$$

$$\tilde{\mathbf{y}}(t) = H(t)\,\mathbf{x}(t) + D(t)\,\mathbf{w}(t) \tag{5.301b}$$

Note that the same noise term $\mathbf{w}(t)$ is added in the dynamic model and measurement equations. But the covariance of the measurement noise can be derived directly using $D(t)$, i.e., $R(t) = D(t)\, Q(t)\, D^T(t)$. Also, without loss in generality we can assume that measurement noise can be normalized so that $D(t)\,D^T(t) = I$. Finally it is assumed that the process and measurement noise are uncorrelated so that $D(t)\,G^T(t) = 0$.

The following worst-case loss function is now defined with known initial conditions:

$$J = \sup_{\mathbf{0}\neq\mathbf{w}} \frac{||\mathbf{x}-\hat{\mathbf{x}}||_2^2}{||\mathbf{w}||_2^2} \tag{5.302}$$

with $\mathbf{x}(t_0) = \mathbf{0}$. Note that if the initial condition is not zero, since the system is linear by subtracting the contribution from the nonzero initial condition, then the assumption is valid without loss in generality. Our goal is to determine a filter, given $\gamma > 0$, such that $J < \gamma^2$. Reference [44] has shown that the following filter achieves this condition:

$$\boxed{\begin{aligned}\dot{\hat{\mathbf{x}}}(t) &= F(t)\,\hat{\mathbf{x}}(t) + B(t)\,\mathbf{u}(t)\\ &\quad + P(t)\,H^T(t)[\tilde{\mathbf{y}}(t) - H(t)\,\hat{\mathbf{x}}(t)], \quad \hat{\mathbf{x}}(t_0) = \mathbf{0}\end{aligned}} \tag{5.303}$$

where

$$\boxed{\begin{aligned}\dot{P}(t) &= F(t)\,P(t) + P(t)\,F^T(t) - P(t)\,[H^T(t)\,H(t) - \gamma^{-2}I]\,P(t)\\ &\quad + G(t)\,G^T(t), \quad P(t_0) = 0\end{aligned}} \tag{5.304}$$

Notice that the H_∞ filter bears a striking resemblance to the classical Kalman filter in §5.4. As $\gamma \to \infty$ eqn. (5.304) becomes the corresponding Kalman filter Riccati equation with known initial conditions. For time-invariant systems a steady-state approach can be used. In this case γ can be chosen to be as small as possible such that the Hamiltonian matrix corresponding to the algebraic version of eqn. (5.304), with $\dot{P}(t) = 0$, does not have any eigenvalues on the imaginary axis. Therefore a bisection approach discussed previously can be used to determine γ. If the initial condition is not known then the following loss function is used:

$$J = \sup_{\mathbf{0}\neq\mathbf{w}} \frac{||\mathbf{x}-\hat{\mathbf{x}}||_2^2}{||\mathbf{w}||_2^2 + \mathbf{x}_0^T\,S\,\mathbf{x}_0} \tag{5.305}$$

with $\mathbf{x}(t_0) = \mathbf{x}_0$ and where S is a positive definite symmetric matrix. The solution to this problem is equivalent to eqn. (5.304) but with $P(t_0) = S^{-1}$. Also, the correlated case can be constructed by using the following modifications:

$$F(t) \leftarrow F(t) - G(t)\,D^T(t)\,H(t) \tag{5.306a}$$

$$G(t) \leftarrow G(t)\,[I - D^T(t)\,D(t)] \tag{5.306b}$$

and the filters are obtained simply by superposition, treating $G(t)\,D^T(t)\tilde{\mathbf{y}}(t)$ as a known quantity.[44]

Example 5.10: In this simple example the performance characteristics of the H_∞ filter approach are investigated for a simple scalar and autonomous system, given by

$$\dot{x}(t) = f\,x(t) + g\,w(t)$$
$$\tilde{y}(t) = h\,x(t) + v(t)$$

Note that $w(t)$ and $v(t)$ are not correlated. The steady-state value for $p \equiv P(t)$ in eqn. (5.304) can be found by solving the following algebraic Riccati equation:

$$2fp - (h^2 - \gamma^{-2})\,p^2 + g^2 = 0$$

which gives

$$p = \frac{f \pm \sqrt{f^2 + g^2(h^2 - \gamma^{-2})}}{h^2 - \gamma^{-2}}$$

Consider the case where p has non-complex values, given by the following condition:

$$f^2 + g^2(h^2 - \gamma^{-2}) \geq 0$$

which yields

$$\gamma^2 \geq \frac{g^2}{f^2 + g^2 h^2} \tag{5.307}$$

If we choose the limiting case where γ^2 is equal to the previous expression, then $p = -g^2/f$. Note that p is positive only when f is negative. Therefore, the original system must be stable, which is an undesired consequence of the H_∞ filter approach.

In order to maintain non-complex values for p we can choose γ^2 to be given by

$$\gamma^2 = \frac{g^2}{f^2 + g^2 h^2 - \alpha^2}$$

where α^2 is a scalar that must satisfy $0 \leq \alpha^2 < (f^2 + g^2 h^2)$. When α^2 approaches its upper bound then γ^{-2} approaches 0, which yields the standard Kalman filter Riccati equation. When $\alpha = 0$ then $p = -g^2/f$. Substituting γ^2 into the Riccati equation yields

$$p = \frac{g^2(f \pm \alpha)}{(\alpha + f)(\alpha - f)}$$

Since f is required to be negative then

$$p = \frac{g^2}{\alpha - f}$$

For the range of valid α the following inequality is true (which is left as an exercise for the reader):

$$\frac{g^2}{\alpha - f} > \frac{f + \sqrt{f^2 + g^2 h^2}}{h^2}$$

Note that the right-hand side of the previous equation is the solution of p for the standard Kalman filter. This inequality shows that the gain in the H_∞ filter will always be larger than the gain in the Kalman filter, which means that the bandwidth of the H_∞ filter is larger than the Kalman filter. Therefore, the H_∞ filter relies more on the measurements than the *a priori* state to obtain the state estimate, which is more robust to modelling errors, but allows more high-frequency noise in the estimate.

This section has introduced the basic concepts of robust filtering. This subject area (as well as robust control) is currently an evolving theory for which the benefits are yet unknown. Still the relationship between the H_∞ filter and Kalman filter is interesting, and in some multi-dimensional cases the H_∞ filter may provide some significant advantages over the Kalman filter. Other areas such as H_∞ adaptive filtering and nonlinear H_∞ filtering may be found in the references provided in this section, as well as the open literature. The reader is encouraged to pursue these references in order to evaluate the performance of robust filtering approaches for the reader's particular dynamical system studies.

5.8 Summary

The results of §5.2 provide the basis for all state estimation algorithms. One of the most fascinating aspects of the estimators developed in §5.2 is the similarity to the sequential estimation results in §1.3. This is truly remarkable since the results of Chapter 1 are applied to constant parameter estimation, while the results of this chapter are applied to parameters that are allowed to change during the estimation process. Another important aspect of state estimation is the similarity to feedback control, where the measurement is the quantity to be "tracked" by the feedback system. This similarity between control and estimation will be further expanded upon in Chapter 6.

The discrete-time Kalman filter developments of §5.3 are based upon the discrete-time sequential estimator of §5.2.1. The only difference between them is in how the gain matrix is derived. The driving force of any estimator is the location of the estimator poles. If these poles are well-known then Ackermann's formula should be employed to determine the gain matrix. However, in practice this is hardly ever the case. The Kalman filter also is a "pole-placement" method, but these poles are selected through rigorous use of known statistical properties of the process noise and measurement noise.

Several theoretical aspects of the Kalman filter are given in this chapter. One of the most important is the stability of the closed-loop Kalman filter state matrix, which is rigorously proved using Lyapunov's theorem. This stability is especially appeasing, since even if the model state matrix is unstable the Kalman filter will always be stable. Several other important aspects of the Kalman filter are shown in this chapter, including: the information filter form, sequential processing, the steady-state Kalman filter, correlated measurement and process noise cases, and the orthogonality principle. The derivation of the continuous-time Kalman filter is shown from two different approaches. The first approach is based upon a continuous-time covariance derivation, and the second approach is shown by applying a limiting argument to the discrete-time formulas. We believe that both approaches are important in understanding the intricacies of the linear Kalman filter.

The Kalman filter is probably one of the most studied algorithms to date. This fact is attested to by the plethora of publications in journals and books. Its popularity will continue for many years to come. An excellent overview of the history behind general filtering theory is given in Ref. [47]. A few of the several notable research results are presented in §5.7, which includes factorization methods, colored-noise Kalman filtering, adaptive filtering, error analysis, Unscented filtering, and robust filtering. This section has merely "scratched the surface" of the flood of research results obtained by studying the Kalman filter. Our own experiences have shown that every time we implement the Kalman filter or study its theoretical foundation, new insights are brought to the surface. The reader is also encouraged to further study the Kalman filter in the open literature (decentralized Kalman filtering,[14] approximate Kalman filtering,[48] and Particle filtering[42, 49, 50] are also interesting topics).

A summary of the key formulas presented in this chapter is given below.

- Ackermann's formula (Continuous-Time)

$$\dot{\hat{\mathbf{x}}} = F\,\hat{\mathbf{x}} + B\,\mathbf{u} + K[\tilde{\mathbf{y}} - H\,\hat{\mathbf{x}}]$$
$$\hat{\mathbf{y}} = H\,\hat{\mathbf{x}}$$
$$K = d(F)\begin{bmatrix} H \\ HF \\ HF^2 \\ \vdots \\ HF^{n-1} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \equiv d(F)\mathcal{O}^{-1}\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

- Ackermann's formula (Discrete-Time)

$$\hat{\mathbf{x}}_{k+1}^{-} = \Phi\,\hat{\mathbf{x}}_k^{+} + \Gamma\,\mathbf{u}_k$$
$$\hat{\mathbf{x}}_k^{+} = \hat{\mathbf{x}}_k^{-} + K[\tilde{\mathbf{y}}_k - H\,\hat{\mathbf{x}}_k^{-}]$$
$$K = d(\Phi)\begin{bmatrix} H\Phi \\ H\Phi^2 \\ H\Phi^3 \\ \vdots \\ H\Phi^{n} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \equiv d(\Phi)\Phi^{-1}\mathcal{O}_d^{-1}\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

- Kalman Filter (Discrete-Time)

$$\hat{\mathbf{x}}_{k+1}^- = \Phi_k \hat{\mathbf{x}}_k^+ + \Gamma_k \mathbf{u}_k$$
$$P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T$$

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k[\tilde{\mathbf{y}}_k - H_k \hat{\mathbf{x}}_k^-]$$
$$P_k^+ = [I - K_k H_k] P_k^-$$
$$K_k = P_k^- H_k^T [H_k P_k^- H_k^T + R_k]^{-1}$$

- Alternative Gain and Update Forms

$$K_k = P_k^+ H_k^T R_k^{-1}$$
$$\hat{\mathbf{x}}_k^+ = P_k^+ \left[\left(P_k^-\right)^{-1} \hat{\mathbf{x}}_k^- + H_k^T R_k^{-1} \tilde{\mathbf{y}}_k \right]$$

- Joseph's Form

$$P_k^+ = [I - K_k H_k] P_k^- [I - K_k H_k]^T + K_k R_k K_k^T$$

- Information Filter

$$\mathcal{P}_k^+ = \mathcal{P}_k^- + H_k^T R_k^{-1} H_k$$
$$\mathcal{P}_{k+1}^- = \left[I - \Psi_k \Upsilon_k \left(\Upsilon_k^T \Psi_k \Upsilon_k + Q_k^{-1} \right)^{-1} \Upsilon_k^T \right] \Psi_k$$
$$\Psi_k \equiv \Phi_k^{-T} \mathcal{P}_k^+ \Phi_k^{-1}$$
$$K_k = (\mathcal{P}_k^+)^{-1} H_k^T R_k^{-1}$$

- Sequential Processing

$$\begin{aligned} \tilde{\mathbf{z}}_k \equiv T_k \tilde{\mathbf{y}}_k &= T_k H_k \mathbf{x}_k + T_k \mathbf{v}_k \\ &\equiv \mathcal{H}_k \mathbf{x}_k + \boldsymbol{\upsilon}_k \end{aligned}$$
$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + P_k^+ \mathcal{H}_k^T \mathcal{R}_k^{-1} [\tilde{\mathbf{z}}_k - \mathcal{H}_k \hat{\mathbf{x}}_k^-]$$

$$K_{i_k} = \frac{P_{i-1_k}^- \mathcal{H}_{i_k}^T}{\mathcal{H}_{i_k} P_{i-1_k}^- \mathcal{H}_{i_k}^T + \mathcal{R}_{i_k}}$$
$$P_{i_k}^+ = [I - K_{i_k} \mathcal{H}_{i_k}] P_{i-1_k}^T, \quad P_{0_k}^+ = P_k^-$$

- Autonomous Kalman Filter (Discrete-Time)

$$\hat{\mathbf{x}}_{k+1} = \Phi\, \hat{\mathbf{x}}_k + \Gamma\, \mathbf{u}_k + \Phi\, K\, [\tilde{\mathbf{y}}_k - H\, \hat{\mathbf{x}}_k]$$
$$P = \Phi\, P\, \Phi^T - \Phi\, P\, H^T [H\, P\, H^T + R]^{-1} H\, P\, \Phi^T + \Upsilon\, Q\, \Upsilon^T$$
$$K = P\, H^T [H\, P\, H^T + R]^{-1}$$

- Correlated Kalman Filter (Discrete-Time)

$$\hat{\mathbf{x}}_{k+1}^{-} = \Phi_k \hat{\mathbf{x}}_k^{+} + \Gamma_k \mathbf{u}_k$$
$$P_{k+1}^{-} = \Phi_k P_k^{+} \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T$$

$$\hat{\mathbf{x}}_k^{+} = \hat{\mathbf{x}}_k^{-} + K_k[\tilde{\mathbf{y}}_k - H_k \hat{\mathbf{x}}_k^{-}]$$
$$P_k^{+} = [I - K_k H_k] P_k^{-} - K_k S_k^T \Upsilon_{k-1}^T$$

$$\begin{aligned} K_k = {} & [P_k^{-} H_k^T + \Upsilon_{k-1} S_k] \\ & \times [H_k P_k^{-} H_k^T + R_k + H_k \Upsilon_{k-1} S_k + S_k^T \Upsilon_{k-1}^T H_k^T]^{-1} \end{aligned}$$

- Continuous-Time to Discrete-Time Covariance Calculation

$$\mathcal{A} = \begin{bmatrix} -F & G\,Q\,G^T \\ 0 & F^T \end{bmatrix} \Delta t$$
$$\mathcal{B} = e^{\mathcal{A}} \equiv \begin{bmatrix} \mathcal{B}_{11} & \mathcal{B}_{12} \\ 0 & \mathcal{B}_{22} \end{bmatrix} = \begin{bmatrix} \mathcal{B}_{11} & \Phi^{-1}\mathcal{Q} \\ 0 & \Phi^T \end{bmatrix}$$

$$\Phi = \mathcal{B}_{22}^T$$
$$\mathcal{Q} = \Phi\,\mathcal{B}_{12}$$

- Kalman Filter (Continuous-Time)

$$\dot{\hat{\mathbf{x}}}(t) = F(t)\,\hat{\mathbf{x}}(t) + B(t)\,\mathbf{u}(t) + K(t)[\tilde{\mathbf{y}}(t) - H(t)\,\hat{\mathbf{x}}(t)]$$
$$\begin{aligned} \dot{P}(t) = {} & F(t)\,P(t) + P(t)\,F^T(t) \\ & - P(t)\,H^T(t)\,R^{-1}(t) H(t)\,P(t) + G(t)\,Q(t)\,G^T(t) \end{aligned}$$
$$K(t) = P(t)\,H^T(t)\,R^{-1}(t)$$

- Autonomous Kalman Filter (Continuous-Time)

$$\dot{\hat{\mathbf{x}}}(t) = F\,\hat{\mathbf{x}}(t) + B\,\mathbf{u}(t) + K[\tilde{\mathbf{y}}(t) - H\,\hat{\mathbf{x}}(t)]$$
$$F\,P + P\,F^T - P\,H^T\,R^{-1} H\,P + G\,Q\,G^T = 0$$
$$K = P\,H^T R^{-1}$$

- Correlated Kalman Filter (Continuous-Time)

$$\dot{\hat{\mathbf{x}}}(t) = F(t)\,\hat{\mathbf{x}}(t) + B(t)\,\mathbf{u}(t) + K(t)[\tilde{\mathbf{y}}(t) - H(t)\,\hat{\mathbf{x}}(t)]$$
$$\begin{aligned} \dot{P}(t) = {} & F(t)\,P(t) + P(t)\,F^T(t) \\ & - K(t)\,R(t)\,K^T(t) + G(t)\,Q(t)\,G^T(t) \end{aligned}$$
$$K(t) = \left[P(t)\,H^T(t) + G(t)\,S^T(t)\right] R^{-1}(t)$$

- Continuous-Discrete Kalman Filter

$$\dot{\hat{\mathbf{x}}}(t) = F(t)\,\hat{\mathbf{x}}(t) + B(t)\,\mathbf{u}(t)$$
$$\dot{P}(t) = F(t)\,P(t) + P(t)\,F^T(t) + G(t)\,Q(t)\,G^T(t)$$

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k[\tilde{\mathbf{y}}_k - H_k\hat{\mathbf{x}}_k^-]$$
$$P_k^+ = [I - K_k H_k]P_k^-$$
$$K_k = P_k^- H_k^T[H_k P_k^- H_k^T + R_k]^{-1}$$

- Extended Kalman Filter (Continuous-Time)

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{f}(\hat{\mathbf{x}}(t),\,\mathbf{u}(t),\,t) + K(t)[\tilde{\mathbf{y}}(t) - \mathbf{h}(\hat{\mathbf{x}}(t),\,t)]$$

$$\begin{aligned}\dot{P}(t) = {} & F(\hat{\mathbf{x}}(t),\,t)\,P(t) + P(t)\,F^T(\hat{\mathbf{x}}(t),\,t) \\ & - P(t)\,H^T(\hat{\mathbf{x}}(t),\,t)\,R^{-1}(t)H(\hat{\mathbf{x}}(t),\,t)\,P(t) + G(t)\,Q(t)\,G^T(t)\end{aligned}$$
$$F(\hat{\mathbf{x}}(t),\,t) \equiv \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}(t)}, \quad H(\hat{\mathbf{x}}(t),\,t) \equiv \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}(t)}$$
$$K(t) = P(t)\,H^T(\hat{\mathbf{x}}(t),\,t)\,R^{-1}(t)$$

- Continuous-Discrete Extended Kalman Filter

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{f}(\hat{\mathbf{x}}(t),\,\mathbf{u}(t),\,t)$$
$$\dot{P}(t) = F(\hat{\mathbf{x}}(t),\,t)\,P(t) + P(t)\,F^T(\hat{\mathbf{x}}(t),\,t) + G(t)\,Q(t)\,G^T(t)$$
$$F(\hat{\mathbf{x}}(t),\,t) \equiv \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}(t)}$$

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k[\tilde{\mathbf{y}}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-)]$$
$$P_k^+ = [I - K_k H_k(\hat{\mathbf{x}}_k^-)]P_k^-$$
$$H_k(\hat{\mathbf{x}}_k^-) \equiv \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_k^-}$$
$$K_k = P_k^- H_k^T(\hat{\mathbf{x}}_k^-)[H_k(\hat{\mathbf{x}}_k^-)P_k^- H_k^T(\hat{\mathbf{x}}_k^-) + R_k]^{-1}$$

- Iterated Extended Kalman Filter

$$\hat{\mathbf{x}}_{k_i}^+ = \hat{\mathbf{x}}_k^- + K_{k_i}\left[\tilde{\mathbf{y}}_k - \mathbf{h}(\hat{\mathbf{x}}_{k_i}^+) - H_k(\hat{\mathbf{x}}_{k_i}^+)\left(\hat{\mathbf{x}}_k^- - \hat{\mathbf{x}}_{k_i}^+\right)\right]$$
$$K_{k_i} = P_k^- H_k^T(\hat{\mathbf{x}}_{k_i}^+)\left[H_k(\hat{\mathbf{x}}_{k_i}^+)P_k^- H_k^T(\hat{\mathbf{x}}_{k_i}^+) + R_k\right]^{-1}$$
$$P_{k_i}^+ = \left[I - K_{k_i} H_k(\hat{\mathbf{x}}_{k_i}^+)\right]P_k^-$$
$$\hat{\mathbf{x}}_{k_0}^+ = \hat{\mathbf{x}}_k^-$$

- Square Root Information Filter

$$\mathcal{P}_k^+ \equiv (P_k^+)^{-1} = \mathcal{S}_k^{+T}\mathcal{S}_k^+$$
$$\mathcal{P}_k^- \equiv (P_k^-)^{-1} = \mathcal{S}_k^{-T}\mathcal{S}_k^-$$
$$R_k^{-1} = \mathcal{V}_k^T\mathcal{V}_k$$
$$Q_k = Z_k E_k Z_k^T$$
$$\Xi_k \equiv \Upsilon_k Z_k$$
$$\hat{\boldsymbol{\alpha}}_k^+ \equiv \mathcal{S}_k^+\hat{\mathbf{x}}_k^+$$
$$\hat{\boldsymbol{\alpha}}_k^- \equiv \mathcal{S}_k^-\hat{\mathbf{x}}_k^-$$
$$\mathcal{Q}_k^T\begin{bmatrix}\mathcal{S}_k^-\\ \mathcal{V}_k H_k\end{bmatrix} = \begin{bmatrix}\mathcal{S}_k^+\\ 0_{m\times n}\end{bmatrix}$$

for $i = 1$

$$\mathbf{a} = \mathcal{S}_k^+\Phi_k^{-1}\Xi_k(1)$$
$$b = \left[\mathbf{a}^T\mathbf{a} + 1/E_k(1,1)\right]^{-1}$$
$$c = \left[1 + \sqrt{b/E_k(1,1)}\right]^{-1}$$
$$\mathbf{d}^T = b\,\mathbf{a}^T\mathcal{S}_k^+\Phi_k^{-1}$$
$$\hat{\boldsymbol{\alpha}}_{k+1}^- = \hat{\boldsymbol{\alpha}}_k^+ - b\,c\,\mathbf{a}\mathbf{a}^T\,\hat{\boldsymbol{\alpha}}_k^+$$
$$\mathcal{S}_{k+1}^- = \mathcal{S}_k^+\Phi_k^{-1} - c\,\mathbf{a}\mathbf{d}^T$$

for $i > 1$

$$\mathbf{a} = \mathcal{S}_{k+1}^-\Xi_k(i)$$
$$b = \left[\mathbf{a}^T\mathbf{a} + 1/E_k(i,i)\right]^{-1}$$
$$c = \left[1 + \sqrt{b/E_k(i,i)}\right]^{-1}$$
$$\mathbf{d}^T = b\,\mathbf{a}^T\mathcal{S}_{k+1}^-$$
$$\hat{\boldsymbol{\alpha}}_{k+1}^- \leftarrow \hat{\boldsymbol{\alpha}}_{k+1}^- - b\,c\,\mathbf{a}\mathbf{a}^T\,\hat{\boldsymbol{\alpha}}_{k+1}^-$$
$$\mathcal{S}_{k+1}^- \leftarrow \mathcal{S}_{k+1}^- - c\,\mathbf{a}\mathbf{d}^T$$

- *U-D* Filter

$$P_{i_k}^- = U_{i_k}^- D_{i_k}^- U_{i_k}^{-T}$$
$$P_{i_k}^+ = U_{i_k}^+ D_{i_k}^+ U_{i_k}^{+T} = U_{i_k}^-\left[D_{i_k}^- - \frac{1}{\alpha_{i_k}}\mathbf{e}_{i_k}\mathbf{e}_{i_k}^T\right]U_{i_k}^{-T}$$
$$\alpha_{i_k} \equiv \mathcal{H}_{i_k}P_{i_k}^-\mathcal{H}_{i_k}^T + \mathcal{R}_{i_k}$$
$$\mathbf{e}_{i_k} \equiv D_{i_k}^- U_{i_k}^{-T}\mathcal{H}_{i_k}^T$$

$$\left[D_{i_k}^- - \frac{1}{\alpha_{i_k}}\mathbf{e}_{i_k}\mathbf{e}_{i_k}^T\right] = L_{i_k}^- \mathcal{E}_{i_k}^- L_{i_k}^{-T}$$
$$U_{i_k}^+ = U_{i-1_k}^+ L_{i_k}^-, \quad U_{0_k}^+ = U_k^-$$
$$D_{i_k}^+ = \mathcal{E}_{i_k}^-, \quad D_{0_k}^+ = D_k^-$$

$$K_{i_k} = \frac{1}{\alpha_{i_k}} U_{i_k}^- \mathbf{e}_{i_k}$$

$$W_{k+1}^- \equiv \left[\Phi_k U_k^+ \ \ \Xi_k\right]$$
$$\tilde{D}_{k+1}^- \equiv \begin{bmatrix} D_k^+ & 0_{n\times s} \\ 0_{s\times n} & E_k \end{bmatrix}$$
$$\left[\mathbf{w}(1)\ \mathbf{w}(2)\ \cdots\ \mathbf{w}(n)\right] = W_{k+1}^{-T}$$

$$\mathbf{c}(i) = \tilde{D}_{k+1}^- \mathbf{w}(i)$$
$$D_{k+1}^-(i,i) = \mathbf{w}^T(i)\,\mathbf{c}(i)$$
$$\mathbf{d}(i) = \mathbf{c}(i)/D_{k+1}^-(i,i)$$
$$U_{k+1}^-(j,i) = \mathbf{w}^T(j)\,\mathbf{d}(i), \quad j = 1, 2, \ldots, i-1$$
$$\mathbf{w}(j) \leftarrow \mathbf{w}(j) - U_{k+1}^-(j,i)\,\mathbf{w}(i), \quad j = 1, 2, \ldots, i-1$$

- Process-Noise Colored-Filter

$$\begin{bmatrix} \mathbf{x}_{k+1} \\ \chi_{k+1} \end{bmatrix} = \begin{bmatrix} \Phi & \Upsilon\mathcal{H} \\ 0 & \Psi \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ \chi_k \end{bmatrix} + \begin{bmatrix} \Gamma \\ 0 \end{bmatrix} \mathbf{u}_k + \begin{bmatrix} \Upsilon\mathcal{D} \\ \mathcal{V} \end{bmatrix} \boldsymbol{\omega}_k$$
$$\tilde{\mathbf{y}}_k = \left[H\ \ 0\right] \begin{bmatrix} \mathbf{x}_k \\ \chi_k \end{bmatrix} + \mathbf{v}_k$$

- Measurement-Noise Colored-Filter

$$\begin{bmatrix} \mathbf{x}_{k+1} \\ \chi_{k+1} \end{bmatrix} = \begin{bmatrix} \Phi & 0 \\ 0 & \Psi \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ \chi_k \end{bmatrix} + \begin{bmatrix} \Gamma \\ 0 \end{bmatrix} \mathbf{u}_k + \begin{bmatrix} \Upsilon & 0 \\ 0 & \mathcal{V} \end{bmatrix} \begin{bmatrix} \mathbf{w}_k \\ \boldsymbol{\omega}_k \end{bmatrix}$$
$$\tilde{\mathbf{y}}_k = \left[H\ \ \mathcal{H}\right] \begin{bmatrix} \mathbf{x}_k \\ \chi_k \end{bmatrix} + \mathcal{D}\boldsymbol{\omega}_k + \boldsymbol{\nu}_k$$

$$E\left\{\begin{bmatrix} \mathbf{w}_k \\ \boldsymbol{\omega}_k \end{bmatrix} \left[\mathbf{w}_k^T\ \ \boldsymbol{\omega}_k^T\right]\right\} = \begin{bmatrix} Q & 0 \\ 0 & \mathcal{Q} \end{bmatrix}$$
$$R = \mathcal{D}\mathcal{Q}\mathcal{D}^T + \mathcal{R}$$
$$S = \left[0\ \ \mathcal{D}\mathcal{Q}\right]$$

- Measurement-Noise Colored-Filter (Restricted Case)

$$\begin{aligned}\chi_{k+1} &= \Psi\,\chi_k + \mathcal{V}\,\omega_k\\ \mathbf{v}_k &= \chi_k\end{aligned}$$

$$\begin{aligned}\tilde{\gamma}_{k+1} &\equiv \tilde{\mathbf{y}}_{k+1} - \Psi\,\tilde{\mathbf{y}}_k - H\,\Gamma\,\mathbf{u}_k\\ &= \mathcal{H}\,\mathbf{x}_k + \mathcal{V}\,\omega_k + H\,\Upsilon\,\mathbf{w}_k\\ \mathcal{H} &\equiv H\,\Phi - \Psi\,H\end{aligned}$$

$$\begin{aligned}R &= \mathcal{V}\,\mathcal{Q}\,\mathcal{V}^T + H\,\Upsilon\,Q\,\Upsilon^T\,H^T\\ S &= H\,\Upsilon\,Q\end{aligned}$$

- Consistency of the Kalman Filter

$$\bar{\epsilon}_k = \frac{1}{M}\sum_{i=1}^{M}\epsilon_k(i) = \frac{1}{M}\sum_{i=1}^{M}\mathbf{e}_k^T(i)\,E_k^{-1}(i)\,\mathbf{e}_k(i)$$

$$\bar{\rho}_{k,j} = \frac{1}{\sqrt{m}}\sum_{i=1}^{M}\mathbf{e}_k^T(i)\left[\sum_{i=1}^{M}\mathbf{e}_k(i)\mathbf{e}_k^T(i)\sum_{i=1}^{M}\mathbf{e}_k(j)\mathbf{e}_k^T(j)\right]^{-1/2}\mathbf{e}_k(j)$$

$$[\bar{\boldsymbol{\mu}}_k]_j = \frac{1}{M}\sum_{i=1}^{M}\frac{[\mathbf{e}_k]_j}{\sqrt{[E_k]_{jj}}}, \quad j = 1, 2, \ldots, m$$

$$\bar{\epsilon} = \frac{1}{N}\sum_{k=1}^{N}\mathbf{e}_k^T\,E_k^{-1}\,\mathbf{e}_k$$

$$\bar{\rho}_j = \frac{1}{\sqrt{n}}\sum_{k=1}^{N}\mathbf{e}_k^T\mathbf{e}_{k+j}\left[\sum_{k=1}^{N}\mathbf{e}_k^T\mathbf{e}_k\sum_{k=1}^{N}\mathbf{e}_{k+j}^T\mathbf{e}_{k+j}\right]^{-1/2}$$

- Adaptive Filtering

$$\hat{C}_i = \frac{1}{N}\sum_{j=i}^{N}\mathbf{e}_j\mathbf{e}_{j-i}^T$$

$$\mathbf{e}_k \equiv \tilde{\mathbf{y}}_k - H\,\hat{\mathbf{x}}_k^-$$

$$\hat{R} = \hat{C}_0 - H\,\hat{Z}$$

$$\hat{Z} = (M^T M)^{-1}M^T\begin{bmatrix}\hat{C}_1 + H\,\Phi\,K\,\hat{C}_0\\ \hat{C}_2 + H\,\Phi\,K\,\hat{C}_1 + H\,\Phi^2 K\,\hat{C}_0\\ \vdots\\ \hat{C}_n + H\,\Phi\,K\,\hat{C}_{n-1} + \cdots + H\,\Phi^n K\,\hat{C}_0\end{bmatrix}$$

$$\delta\hat{P} = \Phi\Big[\delta P - (\hat{Z} + \delta P\, H^T)(\hat{C}_0 + H\,\delta P\, H^T)^{-1}(\hat{Z}^T + H\,\delta P)$$
$$+K\,\hat{Z}^T + \hat{Z}\,K^T - K\,\hat{C}_0\,K^T\Big]\Phi^T$$
$$\hat{K}^* = \Big[\hat{Z} + \delta\hat{P}\,H^T\Big]\Big[\hat{C}_0 + H\,\delta\hat{P}\,H^T\Big]^{-1}$$

- Error Analysis

$$\dot{\hat{\bar{\mathbf{x}}}} = \bar{F}\,\hat{\bar{\mathbf{x}}} + B\,\mathbf{u} + \bar{K}[\tilde{\mathbf{y}} - \bar{H}\,\hat{\bar{\mathbf{x}}}]$$
$$\dot{\bar{P}} = \bar{F}\,\bar{P} + \bar{P}\,\bar{F}^T - \bar{P}\,\bar{H}^T\bar{R}^{-1}\bar{H}\,\bar{P} + \bar{G}\,\bar{Q}\,\bar{G}^T$$
$$\bar{K} = \bar{P}\,\bar{H}^T\bar{R}^{-1}$$

$$P_{\tilde{\mathbf{x}}} = \bar{P} + \Delta V_{\tilde{\mathbf{x}}} + \boldsymbol{\mu}_{\tilde{\mathbf{x}}}\boldsymbol{\mu}_{\tilde{\mathbf{x}}}^T$$

$$\dot{\boldsymbol{\mu}}_{\tilde{\mathbf{x}}} = (\bar{F} - \bar{K}\,\bar{H})\,\boldsymbol{\mu}_{\tilde{\mathbf{x}}} + (\Delta F - \bar{K}\,\Delta H)\,\boldsymbol{\mu}_{\mathbf{x}}$$
$$\dot{\boldsymbol{\mu}}_{\mathbf{x}} = F\,\boldsymbol{\mu}_{\mathbf{x}}$$
$$\Delta F \equiv F - \bar{F}, \quad \Delta H \equiv H - \bar{H}$$

$$\Delta V_{\tilde{\mathbf{x}}} = (\bar{F} - \bar{K}\,\bar{H})\,\Delta V_{\tilde{\mathbf{x}}} + V_{\tilde{\mathbf{x}}}(\bar{F} - \bar{K}\,\bar{H})^T + V^T(\Delta F - \bar{K}\,\Delta H)^T$$
$$+ (\Delta F - \bar{K}\,\Delta H)\,V + (G\,Q\,G^T - \bar{G}\,\bar{Q}\,\bar{G}^T) + \bar{K}\,(R - \bar{R})\,\bar{K}^T$$

- Unscented Filtering

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k,\, \mathbf{w}_k,\, \mathbf{u}_k,\, k)$$
$$\tilde{\mathbf{y}}_k = \mathbf{h}(\mathbf{x}_k,\, \mathbf{u}_k,\, \mathbf{v}_k,\, k)$$

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k\boldsymbol{\upsilon}_k$$
$$P_k^+ = P_k^- - K_k\,P_k^{\upsilon\upsilon}\,K_k^T$$
$$\boldsymbol{\upsilon}_k \equiv \tilde{\mathbf{y}}_k - \hat{\mathbf{y}}_k^-$$
$$= \tilde{\mathbf{y}}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-,\, \mathbf{u}_k, k)$$

$$K_k = P_k^{xy}(P_k^{\upsilon\upsilon})^{-1}$$

$$P_k^a = \begin{bmatrix} P_k^+ & P_k^{xw} & P_k^{xv} \\ (P_k^{xw})^T & Q_k & P_k^{wv} \\ (P_k^{xv})^T & (P_k^{wv})^T & R_k \end{bmatrix}$$
$$\boldsymbol{\sigma}_k \leftarrow 2L \text{ columns from } \pm\gamma\sqrt{P_k^a}$$
$$\boldsymbol{\chi}_k^a(0) = \hat{\mathbf{x}}_k^a$$
$$\boldsymbol{\chi}_k^a(i) = \boldsymbol{\sigma}_k(i) + \hat{\mathbf{x}}_k^a$$

$$\mathbf{x}_k^a = \begin{bmatrix} \mathbf{x}_k \\ \mathbf{w}_k \\ \mathbf{v}_k \end{bmatrix}, \quad \hat{\mathbf{x}}_k^a = \begin{bmatrix} \hat{\mathbf{x}}_k \\ \mathbf{0}_{q\times 1} \\ \mathbf{0}_{m\times 1} \end{bmatrix}$$

$$W_0^{\text{mean}} = \frac{\lambda}{L+\lambda}$$

$$W_0^{\text{cov}} = \frac{\lambda}{L+\lambda} + (1-\alpha^2+\beta)$$

$$W_i^{\text{mean}} = W_i^{\text{cov}} = \frac{1}{2(L+\lambda)}, \quad i = 1, 2, \ldots, 2L$$

$$\chi_{k+1}(i) = \mathbf{f}(\chi_k^x(i), \chi_k^w(i), \mathbf{u}_k, k)$$

$$\hat{\mathbf{x}}_{k+1}^- = \sum_{i=0}^{2L} W_i^{\text{mean}} \chi_{k+1}^x(i)$$

$$P_{k+1}^- = \sum_{i=0}^{2L} W_i^{\text{cov}} [\chi_{k+1}^x(i) - \hat{\mathbf{x}}_{k+1}^-][\chi_{k+1}^x(i) - \hat{\mathbf{x}}_{k+1}^-]^T$$

$$\gamma_{k+1}(i) = \mathbf{h}(\chi_{k+1}^x(i), \mathbf{u}_{k+1}, \chi_{k+1}^v(i), k+1)$$

$$\hat{\mathbf{y}}_{k+1}^- = \sum_{i=0}^{2L} W_i^{\text{mean}} \gamma_{k+1}(i)$$

$$P_{k+1}^{yy} = \sum_{i=0}^{2L} W_i^{\text{cov}} [\gamma_{k+1}(i) - \hat{\mathbf{y}}_{k+1}^-][\gamma_{k+1}(i) - \hat{\mathbf{y}}_{k+1}^-]^T$$

$$P_{k+1}^{\upsilon\upsilon} = P_{k+1}^{yy}$$

$$P_{k+1}^{xy} = \sum_{i=0}^{2L} W_i^{\text{cov}} [\chi_{k+1}^x(i) - \hat{\mathbf{x}}_{k+1}^-][\gamma_{k+1}(i) - \hat{\mathbf{y}}_{k+1}^-]^T$$

- Robust Filtering

$$\begin{aligned} \dot{\hat{\mathbf{x}}}(t) &= F(t)\,\hat{\mathbf{x}}(t) + B(t)\,\mathbf{u}(t) \\ &\quad + P(t)\,H^T(t)[\tilde{\mathbf{y}}(t) - H(t)\,\hat{\mathbf{x}}(t)], \quad \hat{\mathbf{x}}(t_0) = \mathbf{0} \end{aligned}$$

$$\begin{aligned} \dot{P}(t) &= F(t)\,P(t) + P(t)\,F^T(t) - P(t)\,[H^T(t)\,H(t) - \gamma^{-2}I]\,P(t) \\ &\quad + G(t)\,G^T(t), \quad P(t_0) = S^{-1} \end{aligned}$$

Exercises

5.1 Write a general computer routine for Ackermann's formula in eqn. (5.19).

5.2 Design an estimator for a simple pendulum model, given by

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & 0 \end{bmatrix} \mathbf{x}(t)$$
$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}(t)$$

where both estimator eigenvalues are at $-10\omega_n$. Convert your estimator into discrete-time. Pick any initial conditions and simulate the performance of the estimator using synthetic measurements ($\tilde{y}_k = y_k + v_k$), with various values for the measurement-error variance. How do your estimates change as more noise is introduced into the measurement? Also, try changing the pole locations of the estimator for various noise levels.

5.3 The stick-fixed lateral equations of motion for a general aviation aircraft are given by[31]

$$\begin{bmatrix} \Delta\dot{\beta}(t) \\ \Delta\dot{p}(t) \\ \Delta\dot{r}(t) \\ \Delta\dot{\phi}(t) \end{bmatrix} = \begin{bmatrix} -0.254 & 0 & -1.0 & 0.182 \\ -16.02 & -8.40 & -2.19 & 0 \\ 4.488 & -0.350 & -0.760 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\beta(t) \\ \Delta p(t) \\ \Delta r(t) \\ \Delta\phi(t) \end{bmatrix}$$
$$y(t) = \Delta\phi(t)$$

where $\Delta\beta(t)$, $\Delta p(t)$, $\Delta r(t)$, and $\Delta\phi(t)$ are perturbations in sideslip, lateral angular velocities quantities, and roll angle, respectively. Determine the open-loop eigenvalues and the observability of the system. Design an estimator that places the poles at $s_1 = -10$, $s_2 = -20$, and $s_{3,4} = -10 \pm 2j$. Check the performance of this estimator through simulated runs for various initial condition errors.

5.4 In example 5.1 prove that the solutions for k_1 and k_2 solve the desired characteristic equation.

5.5 Consider the following system to be controlled:

$$\dot{\mathbf{x}}(t) = F\,\mathbf{x}(t) + B\,u(t)$$

Let $u(t) = -K\,\mathbf{x}(t)$, where K is a $1 \times n$ matrix. The closed-loop system matrix is given by $F - B\,K$ {compare this to eqn. (5.7)}. Suppose that a desired closed-loop characteristic equation is sought, with $d(s) = 0$. Following the steps in §5.2 derive Ackermann's formula for this control system. Also, derive an equivalent formula for a discrete-time system. What condition is required for K to exist (note: this control problem is the dual of the estimator design)?

5.6 Equation (5.22) represents an estimator for the predicted state. Derive a similar equation for the updated state using eqn. (5.20). Compare your result to eqn. (5.22).

5.7 ♣ Prove that $\Phi[I-KH]$ and $[I-KH]\Phi$ have the same eigenvalues.

5.8 In order to design a discrete-time estimator in eqn. (5.26), the system must be observable and the inverse of Φ must exist. Discuss the physical connotations for the inverse of Φ to exist.

5.9 Consider the following second-order continuous-time system:

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w \equiv F\,\mathbf{x} + G\,w$$

where $\mathbf{x} \equiv \begin{bmatrix} \theta & \omega \end{bmatrix}^T$ and the variance of w is given by q. Suppose we have measurements of θ only, so that $H = \begin{bmatrix} 1 & 0 \end{bmatrix}$. A simple method to study the behavior of discrete-time measurements is to assume continuous-time measurements with variance given by $R(t) = \sigma^2_{\text{sensor}}\Delta t$, where Δt is the sampling interval. Note the relation to eqn. (5.148) for this substitution. This will be a reasonable approximation if the sampling interval is much shorter than the time constants of interest. Using this approximation, solve for all the elements of the 2×2 continuous-time steady-state covariance matrix, P, shown in Table 5.5 in terms of q, σ^2_{sensor} and Δt.

5.10 Consider the following first-order discrete-time system:

$$x_{k+1} = \phi\, x_k + w_k$$

where w_k is a zero-mean Gaussian noise process with variance q. Derive a closed-form expression for the variance of x_k, where $p_k \equiv E\left\{x_k^2\right\}$. What is the steady-state variance? Also, discuss the properties of the steady-state value in terms of the stability of the system (i.e., in terms of ϕ).

5.11 Consider the following discrete-time model:

$$\begin{aligned} x_{k+1} &= x_k \\ \tilde{y}_k &= x_k + v_k \end{aligned}$$

where v_k is a zero-mean Gaussian noise process with variance r. Note that this system has no process noise, so $Q = 0$. Using the discrete-time Kalman filter equations in Table 5.1 derive a closed-form recursive solution for the gain K in terms of r, P_0 (the initial error-variance) and k (the time index). Discuss the properties of this simple Kalman filter as k increases.

5.12 Consider the following truth model for a simple second-order system:

$$\mathbf{x}_{k+1} = \begin{bmatrix} 9.9985\times 10^{-1} & 9.8510\times 10^{-3} \\ -2.9553\times 10^{-2} & 9.7030\times 10^{-1} \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} 4.9502\times 10^{-5} \\ 9.8510\times 10^{-3} \end{bmatrix} w_k$$

$$\tilde{y}_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_k + v_k$$

where the sampling interval is given by 0.01 seconds. Using initial conditions of $\mathbf{x}_0 = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$, create a set of 1001 synthetic measurements with the following variances for the process noise and measurement noise: $Q = 1$ and $R = 0.01$. Run the Kalman filter in Table 5.1 with the given model and assumed values for Q and R. Test the convergence of the filter for various state and covariance initial condition errors. Also, compare the computed state errors with their respective 3σ bounds computed from the covariance matrix P_k.

5.13 Repeat the simulation in exercise 5.12 using the same state model but with the following measurement model:

$$\tilde{\mathbf{y}}_k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} \mathbf{x}_k + \mathbf{v}_k$$

where $R = \text{diag}\begin{bmatrix} 0.01 & 0.01 & 0.01 \end{bmatrix}$. Do the added measurements yield better estimates (compare the values of P_k with the previous simulation)?

5.14 Repeat the simulation in exercise 5.13 using the information filter and sequential processing algorithm shown in §5.3.3. Compare the computational loads (in terms of Floating Point Operations) of the conventional Kalman filter with both the information filter and sequential processing algorithm.

5.15 Using the truth model in exercise 5.12, with initial conditions of $\mathbf{x}_0 = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$, create a set of 1001 synthetic measurements with the following variances for the process noise and measurement noise: $Q = 0$ and $R = 0.01$. Run the Kalman filter in Table 5.1 with the following assumed model:

$$\Phi = \begin{bmatrix} 9.9990 \times 10^{-1} & 9.8512 \times 10^{-3} \\ -1.9702 \times 10^{-2} & 9.7035 \times 10^{-1} \end{bmatrix}, \quad \Upsilon = \begin{bmatrix} 4.9503 \times 10^{-5} \\ 9.8512 \times 10^{-3} \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

Can you pick a value for Q that yields accurate estimates with this incorrect model (try various values to "tune" Q)? Compare your estimate errors with the theoretical 3σ bounds.

5.16 In example 5.3 the discrete-time process-noise covariance is shown without derivation. Fully derive this expression. Also, reproduce the results of this example using your own simulation.

5.17 Write a general program that solves the discrete-time algebraic Riccati equation using the eigenvalue/eigenvector decomposition algorithm of the Hamiltonian matrix derived in §5.3.4. Compare the steady-state values computed from your program to the values computed by the Kalman filter covariance propagation and update in problems 5.12 and 5.13.

5.18 Consider the following delayed-state measurement problem:

$$\mathbf{x}_k = \Phi_{k-1}\mathbf{x}_{k-1} + \Gamma_{k-1}\mathbf{u}_{k-1} + \Upsilon_{k-1}\mathbf{w}_{k-1}$$
$$\tilde{\mathbf{y}}_k = H_k\mathbf{x}_k + J_k\mathbf{x}_{k-1} + \mathbf{v}_k$$

where $\mathbf{w}_{k-1}$ and $\mathbf{v}_k$ are uncorrelated. Show that the measurement model can be rewritten as

$$\tilde{\mathbf{y}}_k = (H_k + J_k\Phi_{k-1}^{-1})\mathbf{x}_k + (\mathbf{v}_k - J_k\Phi_{k-1}^{-1}\mathbf{w}_{k-1})$$

What is the covariance of the new measurement error? What is the correlation between the new measurement error and process noise? Derive a correlated Kalman filter for the delayed-state measurement problem that is independent of Φ_{k-1}^{-1} (hint: use the following equation: $\Upsilon_{k-1}Q_{k-1}\Upsilon_{k-1}^T = P_k^- - \Phi_{k-1}P_{k-1}^+\Phi_{k-1}^T$).

5.19 ♣ Prove that the covariance for the correlated discrete-time Kalman filter in §5.3.5 is lower when $S_k \neq 0$ than with $S_k = 0$. Why is this true?

5.20 Fully show that the first-order approximation of eqn. (5.135) is given by eqn. (5.136).

5.21 Use the numerical solution in eqn. (5.140) to prove the analytical solution of the discrete-time process noise covariance in example 5.3.

5.22 Prove that the continuous-time Kalman filter estimation error is orthogonal to the state estimate, i.e., $E\left\{\hat{\mathbf{x}}(t)\,\tilde{\mathbf{x}}^T(t)\right\} = 0$, where $\tilde{\mathbf{x}}(t) \equiv \hat{\mathbf{x}}(t) - \mathbf{x}(t)$.

5.23 Using the methods of §5.4.2 find the relationship between the discrete-time correlation matrix S_k in eqn. (5.104) and the continuous-time correlation matrix $S(t)$ in eqn. (5.181).

5.24 Consider the steady-state continuous-time Kalman filter in Table 5.5 for a second-order system with $Q \equiv \operatorname{diag}\left[q_1\ q_2\right]$ and $R = I$. Using the dynamical model in exercise 5.2, find closed-form values for q_1 and q_2 in terms of ω_n that yield estimator eigenvalues at $-10\omega_n$. Discuss the aspects of using the Kalman filter over Ackermann's formula for pole-placement (which method do you think is easier)?

5.25 Prove that the eigenvalues of the Hamiltonian matrix in eqn. (5.168) are symmetric about the imaginary axis (i.e., if λ is an eigenvalue of $\mathcal{H}$, then $-\lambda$ is also an eigenvalue of $\mathcal{H}$).

5.26 Write a general program that solves the continuous-time algebraic Riccati equation using the eigenvalue/eigenvector decomposition algorithm of the Hamiltonian matrix derived in §5.4.4. Check your program for the solution you found in exercise 5.24 (use any value for ω_n).

5.27 The solution for the steady-state variance in example 5.4 is given by $p = r(a+f)$, where $a = \sqrt{f^2 + r^{-1}q}$. Show that another solution is given by $p = q/(a-f)$.

5.28 ♣ Prove that the covariance for the correlated continuous-time Kalman filter in §5.4.5 is lower when $S(t) \neq 0$ than with $S(t) = 0$.

5.29 Consider the following continuous-time model with discrete-time measurements (where the state quantities are explained in exercise 5.3):

$$\begin{bmatrix} \Delta\dot{\beta}(t) \\ \Delta\dot{p}(t) \\ \Delta\dot{r}(t) \\ \Delta\dot{\phi}(t) \end{bmatrix} = \begin{bmatrix} -0.254 & 0 & -1.0 & 0.182 \\ -16.02 & -8.40 & -2.19 & 0 \\ 4.488 & -0.350 & -0.760 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\beta(t) \\ \Delta p(t) \\ \Delta r(t) \\ \Delta\phi(t) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} w(t)$$

$$\tilde{y}_k = \Delta\phi_k + v_k$$

Assume that the measurements are sampled every 0.01 seconds. Using initial conditions of $\left[\pi/180\ \pi/180\ \pi/180\ \pi/180\right]^T$ radians, create a set of 1001 synthetic measurements with the following variances for the process noise and measurement noise: $Q = 0.001$ and $R = (0.1\pi/180)^2$ (note: Q is the continuous-time variance and R is the discrete-time covariance). Run the Kalman filter in Table 5.7 with the given model and assumed values for Q and R. Test the convergence of the filter for various state and covariance initial condition errors. Also, compare the computed state errors with their respective 3σ bounds computed from the covariance matrix $P(t)$.

5.30 Consider a linear Kalman filter with no measurements. Discuss the stability of the propagated covariance matrix with no state updates for stable, unstable, and marginally stable system-state matrices.

5.31 ♣ Using the approximations shown in §5.4.2 derive an algebraic Riccati equation for the continuous-discrete Kalman filter in Table 5.7, assuming that the system matrices F, G, and H are constants and that the noise processes are stationary. Compare your result to the algebraic Riccati equation in Table 5.5. Write a program that solves the algebraic Riccati equation you derived. Compare the steady-state values computed from your program to the values computed by the Kalman filter covariance propagation and update in exercise 5.29.

5.32 Consider the following first-order system:

$$\dot{x}(t) = x^2(t) + w(t)$$

$$\tilde{y}_k = x_k^{-1} + v_k$$

where $w(t)$ and v_k are zero-mean Gaussian noise processes with variances q and r, respectively. Derive the continuous-discrete extended Kalman filter equations in Table 5.9 for this system. Create synthetic measurements of this system for various values of x_0, P_0, q, and r. Test the performance of the extended Kalman filter using simulated computer runs. Compare the computed state errors with their respective 3σ bounds computed from the covariance matrix $P(t)$. Also, try changing the sampling interval in your simulations. Discuss the effects of the sampling interval on the overall covariance $P(t)$.

5.33 Consider the following model that is used to simulate the demodulation of angle-modulated signals:[30]

$$\begin{bmatrix} \dot{\lambda}(t) \\ \dot{\theta}(t) \end{bmatrix} \begin{bmatrix} -1/\beta & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \lambda(t) \\ \theta(t) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} w(t)$$
$$\tilde{y}_k = \sqrt{2}\sin(\omega_c t_k + \theta_k) + v_k$$

where the message $\lambda(t)$ has a first-order Butterworth spectrum, being modulated as the output of a first-order, time-invariant linear system with one real pole driven by a continuous zero-mean Gaussian noise process, $w(t)$, with variance q. This message is then passed through an integrator to give $\theta(t)$, which is then employed to phase modulate a carrier signal with frequency ω_c. The measurement noise process v_k is also zero-mean Gaussian noise with variance r.

Create 1001 synthetic measurements, sampled every 0.01 seconds, of the aforementioned system using the following parameters: $\omega_c = 5$ (rad/sec), $\beta = 1$, $q = 0.5$, $r = 1$, and initial conditions of $\lambda_0 = \pi$ (rad/sec) and $\theta_0 = \pi/6$ (rad). Run the extended Kalman filter in Table 5.9 with the given model and assumed values for Q and R. Test the convergence of the filter for various initial condition errors and values for P_0. Also, compare the computed state errors with their respective 3σ bounds computed from the covariance matrix $P(t)$. Finally, is it possible to use a fully discrete-time version of the extended Kalman filter on this system?

5.34 Consider the following second-order system:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 \\ -a & -b \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$
$$\tilde{y}_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_k + v_k$$

Create 1001 synthetic measurements, sampled every 0.01 seconds, of the aforementioned system using the following parameters: $a = b = 3$, $R = 0.0001$, $u(t) = 0$, and $\mathbf{x}_0 = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$. Append the model to include states to estimate the parameters a and b, so that the Kalman filter propagation model is given by

$$\dot{\hat{\mathbf{x}}}(t) = \begin{bmatrix} \hat{x}_2(t) \\ -\hat{x}_1(t)\,\hat{x}_3(t) - \hat{x}_2(t)\,\hat{x}_4(t) \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} u(t)$$
$$\hat{y}_k = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \hat{\mathbf{x}}_k$$

where $\hat{x}_3$ and $\hat{x}_4$ are estimates of a and b, respectively. Run the extended Kalman filter given in Table 5.9 with the given model to estimate a and b.

Use the following matrices for G and Q:

$$G = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad Q = \begin{bmatrix} q & 0 \\ 0 & q \end{bmatrix}$$

Try various values for q to test the performance of the extended Kalman filter. Also, compare the computed state errors with their respective 3σ bounds computed from the covariance matrix $P(t)$. Try adding a nonzero control input into the system, e.g., let $u(t) = 10\sin(t) - 8\cos(t) + 5\sin(2t) + 3\cos(2t)$. Does this help the observability of the system? Finally, try increasing R by an order of magnitude (as well as other values) and repeat the entire procedure.

5.35 Reproduce the results using the extended Kalman filter with Van Der Pol's model in examples 5.5 and 5.6 using your own simulation. Check the sensitivity of the extended Kalman filter for various initial condition errors. Can you find initial conditions that cause the filter to become unstable? For the parameter identification simulation, pick various values of q and discuss the performance of the identification results.

5.36 Consider the following first-order nonlinear system:

$$\dot{x}(t) = 0$$
$$\tilde{y}_k = \sin(x_k t_k) + v_k$$

Create 201 synthetic measurements, sampled every 0.1 seconds, of the aforementioned system using the following parameters: $t_0 = 0$, $x_k = 1$ for all time and $R = 0.1$. Develop an extended Kalman filter to estimate the frequency x_k with the following starting conditions: $\hat{x}_0 = 10$ and $P_0 = 1$ (note: $\hat{x}_{k+1}^- = \hat{x}_k^+$ and $P_{k+1}^- = P_k^+$ for this system). How does your EKF perform for this problem? Next, try an iterated Kalman filter using eqns. (5.198). Compare the performance of the iterated Kalman filter to the standard extended Kalman filter.

5.37 Consider the following formulas to simulate the effects of roundoff errors in a Kalman filter:

$$1 + \epsilon \overset{r}{\neq} 1$$
$$1 + \epsilon^2 \overset{r}{=} 1$$

where $\overset{r}{=}$ means equal to rounding and $\epsilon << 1$. Consider a scalar measurement update of a two-state problem with the following characteristics:

$$P_k^- = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad R = \epsilon^2$$

The exact covariance update is given by

$$P_k^+ = \begin{bmatrix} \epsilon^2/(1+\epsilon^2) & 0 \\ 0 & 1 \end{bmatrix}$$

Using the roundoff errors introduced previously compute the update covariance using: 1) the conventional Kalman filter form in eqn. (5.44), 2) Joseph's form in eqn. (5.39), 3) the SRIF factorization using eqn. (5.203), and 4) the U-D factorization using eqn. (5.214). Discuss the performance characteristics of each approach. Also, redo the problem with $H = \begin{bmatrix} 1 & 1 \end{bmatrix}$.

5.38 An SRIF approach can also be implemented for the extended Kalman filter. Derive this filter by using the inverse linearized dynamics to eliminate the state in the extended SRIF. Note: in the linear discrete-time SRIF Φ_k^{-1} is used to eliminate the state (use the inverse linearized dynamics in the extended SRIF).

5.39 Derive continuous-time versions of the colored-noise filters shown in §5.7.2.

5.40 Create synthetic measurements using the dynamical model and shaping filter discussed in example 5.7. Pick various values for the gust noise parameter a and process noise q, and run the Kalman filter given by Table 5.7 for the full model (including the shaping filter). With the same synthetic measurements run the standard Kalman filter using only the dynamical model without the shaping filter, tuning q until reasonable estimates are achieved. Under what cases does the "reduced-order" Kalman filter provide good estimates (i.e., when colored-noise process noise exists, but when only the standard Kalman filter is used)?

5.41 In example 5.8 the covariance of the measurement residual is stated to be given by

$$E_k = H_k^T P_k H_k + R_k$$

Prove that this covariance is correct.

5.42 Reproduce the results of example 5.8 using a single run of measurements. Also, try multiple (Monte Carlo) runs and use eqns. (5.235), (5.237), and (5.239) to check the consistency of the Kalman filter for various values of q. Since the truth is known for the example, consistency tests can also be applied to the state error with $\mathbf{e}_k = \hat{\mathbf{x}}_k - \mathbf{x}_k$ and $E_k = P_k$. Check the consistency of the Kalman filter using the state error with a single run and with multiple runs.

5.43 Write a computer program that computes the autocorrelation coefficients using eqns. (5.251) and (5.252). Create Gaussian noise values for $\mathbf{e}_k$ using a random noise generator and numerically check the confidence limit given by eqn. (5.253). Also, try non-Gaussian values for $\mathbf{e}_k$.

5.44 Using the adaptive methods of §5.7.4 estimate the measurement and process noise variances from exercise 5.12. How well do your estimates compare with their respective true values? Also, use the adaptive approach to find an "optimal" value for q using the synthetic measurements created with the model in exercise 5.12, but with the model in exercise 5.15 in the Kalman filter.

5.45 ♣ Derive the error analysis results of §5.7.5.

5.46 Consider the following nominal model:

$$\bar{F} = \begin{bmatrix} 0 & 1 \\ -3 & -3 \end{bmatrix}, \quad \bar{G} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \bar{H} = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

with $\bar{R} = \bar{Q} = 1$. Compute the steady-state continuous-time covariance using eqn. (5.269b). Next consider the following actual system model:

$$F = \begin{bmatrix} 0 & 1 \\ -a & -3 \end{bmatrix}$$

where $a > 0$. Compute the covariance of the error introduced by this modelling error using the methods shown in §5.7.5. Also, for this system evaluate the performance of the Kalman filter for the following error cases: 1) errors in Q alone, 2) errors in Q and a together, and 3) errors in R and a together. Which case seems to be the most sensitive in the Kalman filter design?

5.47 ♣ Consider the following one-dimensional random variable y that is related to x by the following nonlinear transformation:

$$y = x^2$$

where x is a Gaussian noise process with mean μ and variance σ_x^2. Prove that the true variance of y is given by

$$\sigma_y^2 = 2\sigma_x^4 + 4\mu\sigma_x^2$$

Compute an approximation of the true σ_y^2 by linearizing the nonlinear transformation. Next, compute an approximation of the true σ_y^2 by using the methods described in §5.7.6. Which approach yields better results?

5.48 Reproduce the results using the extended Kalman filter and Unscented filter of the vertically falling-body problem in example 5.9. Check the performance of both algorithms for various sampling intervals.

5.49 Implement the Unscented filter to estimate the damping coefficient c for Van der Pol's equation in examples 5.5 and 5.6. How does the performance of the Unscented filter compare to the extended Kalman filter for various initial condition errors?

5.50 Implement the Unscented filter to estimate the frequency of the model shown in exercise 5.36. Try various values of α in your Unscented filter (even outside the recommended upper bound of 1). Compare the performance of the Unscented filter to the iterated Kalman filter and standard extended Kalman filter.

5.51 ♣ Derive the H_∞ filtering results of §5.7.7.

5.52 Derive the last inequality shown in example 5.10.

5.53 Create synthetic measurements using the model described in example 5.10. Using known errors in f compare the performance of the standard Kalman filter to the performance of the H_∞ filter. Is the H_∞ filter more robust?

5.54 Using the synthetic measurements created in exercise 5.29, run the standard Kalman filter and H_∞ filter with various errors in the assumed model. Can you find a parameter change in the assumed model that yields better performance characteristics using the using H_∞ filter over the standard Kalman filter? Discuss the effect of the parameter γ on the performance of the H_∞ filter for this system.

References

[1] Gelb, A., editor, *Applied Optimal Estimation*, The MIT Press, Cambridge, MA, 1974.

[2] Franklin, G.F., Powell, J.D., and Workman, M., *Digital Control of Dynamic Systems*, Addison Wesley Longman, Menlo Park, CA, 3rd ed., 1998.

[3] Kalman, R.E. and Bucy, R.S., "New Results in Linear Filtering and Prediction Theory," *Journal of Basic Engineering*, March 1961, pp. 95–108.

[4] Stengle, R.F., *Optimal Control and Estimation*, Dover Publications, New York, NY, 1994.

[5] Lewis, F.L., *Optimal Estimation with an Introduction to Stochastic Control Theory*, John Wiley & Sons, New York, NY, 1986.

[6] Kalman, R.E. and Joseph, P.D., *Filtering for Stochastic Processes with Applications to Guidance*, Interscience Publishers, New York, NY, 1968.

[7] Golub, G.H. and Van Loan, C.F., *Matrix Computations*, The Johns Hopkins University Press, Baltimore, MD, 3rd ed., 1996.

[8] Kailath, T., Sayed, A.H., and Hassibi, B., *Linear Estimation*, Prentice Hall, Upper Saddle River, NJ, 2000.

[9] Vaughan, D.R., "A Nonrecursive Algebraic Solution for the Discrete Riccati Equation," *IEEE Transactions on Automatic Control*, Vol. AC-15, No. 5, Oct. 1970, pp. 597–599.

[10] Fallon, L., "Gyroscopes," *Spacecraft Attitude Determination and Control*, edited by J.R. Wertz, chap. 6.5, Kluwer Academic Publishers, The Netherlands, 1978.

[11] Farrenkopf, R.L., "Analytic Steady-State Accuracy Solutions for Two Common Spacecraft Attitude Estimators," *Journal of Guidance and Control*, Vol. 1, No. 4, July-Aug. 1978, pp. 282–284.

[12] Bendat, J.S. and Piersol, A.G., *Engineering Applications of Correlation and Spectral Analysis*, John Wiley & Sons, New York, NY, 1980.

[13] van Loan, C.F., "Computing Integrals Involving the Matrix Exponential," *IEEE Transactions on Automatic Control*, Vol. AC-23, No. 3, June 1978, pp. 396–404.

[14] Brown, R.G. and Hwang, P.Y.C., *Introduction to Random Signals and Applied Kalman Filtering*, John Wiley & Sons, New York, NY, 3rd ed., 1997.

[15] Schweppe, F.C., *Uncertain Dynamic Systems*, Prentice Hall, Englewood Cliffs, NJ, 1973.

[16] Reid, W.T., *Riccati Differential Equations*, Academic Press, New York, NY, 1972.

[17] Vaughan, D.R., "A Negative Exponential Solution for the Matrix Riccati Equation," *IEEE Transactions on Automatic Control*, Vol. AC-14, No. 1, Feb. 1969, pp. 72–75.

[18] MacFarlane, A.G.J., "An Eigenvector Solution of the Optimal Linear Regulator," *Journal of Electronics and Control*, Vol. 14, No. 6, June 1963, pp. 643–654.

[19] Potter, J.E., "Matrix Quadratic Solutions," *SIAM Journal of Applied Mathematics*, Vol. 14, No. 3, May 1966, pp. 496–501.

[20] Laub, A.J., "A Schur Method for Solving Algebraic Riccati Equations," *IEEE Transactions on Automatic Control*, Vol. AC-24, No. 6, Dec. 1979, pp. 913–921.

[21] Bittanti, S., Laub, A., and Willems, J., editors, *The Riccati Equation*, Communications and Control Engineering Series, Springer-Verlag, Berlin, 1991.

[22] Wiener, N., *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*, John Wiley, New York, NY, 1949.

[23] Maybeck, P.S., *Stochastic Models, Estimation, and Control*, Vol. 1, Academic Press, New York, NY, 1979.

[24] Jazwinski, A.H., *Stochastic Processes and Filtering Theory*, Academic Press, San Diego, CA, 1970.

[25] Slotine, J.J.E. and Li, W., *Applied Nonlinear Control*, Prentice Hall, Englewood Cliffs, NJ, 1991.

[26] Battin, R.H., *Astronautical Guidance*, McGraw Hill, New York, NY, 1964.

[27] Kaminski, P.G., Bryson, A.E., and Schmidt, S.F., "Discrete Square Root Filtering: A Survey of Current Techniques," *IEEE Transactions on Automatic Control*, Vol. AC-16, No. 5, Dec. 1971, pp. 727–735.

[28] Bierman, G.J., *Factorization Methods for Discrete Sequential Estimation*, Academic Press, Orlando, FL, 1977.

[29] Crassidis, J.L., Andrews, S.F., Markley, F.L., and Ha, K., "Contingency Designs for Attitude Determination of TRMM," *Proceedings of the Flight Mechanics/Estimation Theory Symposium*, NASA-Goddard Space Flight Center, Greenbelt, MD, May 1995, pp. 419–433.

[30] Anderson, B.D.O. and Moore, J.B., *Optimal Filtering*, Prentice Hall, Englewood Cliffs, NJ, 1979.

[31] Nelson, R.C., *Flight Stability and Automatic Control*, McGraw-Hill, New York, NY, 1989.

[32] Bar-Shalom, Y., Li, X.R., and Kirubarajan, T., *Estimation with Applications to Tracking and Navigation*, John Wiley & Sons, New York, NY, 2001.

[33] Devore, J.L., *Probability and Statistics for Engineering and Sciences*, Duxbury Press, Pacific Grove, CA, 1995.

[34] Mehra, R.K., "On the Identification of Variances and Adaptive Kalman Filtering," *IEEE Transactions on Automatic Control*, Vol. AC-15, No. 2, April 1970, pp. 175–184.

[35] Maybeck, P.S., *Stochastic Models, Estimation, and Control*, Vol. 2, Academic Press, New York, NY, 1982.

[36] Brown, R.J. and Sage, A.P., "Error Analysis of Modeling and Bias Errors in Continuous Time State Estimation," *Automatica*, Vol. 7, No. 5, Sept. 1971, pp. 577–590.

[37] Sage, A.P. and White, C.C., *Optimum Systems Control*, Prentice Hall, Englewood Cliffs, NJ, 2nd ed., 1977.

[38] Daum, F.E., "Exact Finite-Dimensional Nonlinear Filters," *IEEE Transactions on Automatic Control*, Vol. AC-31, No. 7, July 1986, pp. 616–622.

[39] Julier, S.J., Uhlmann, J.K., and Durrant-Whyte, H.F., "A New Approach for Filtering Nonlinear Systems," *American Control Conference*, Seattle, WA, June 1995, pp. 1628–1632.

[40] Julier, S.J., Uhlmann, J.K., and Durrant-Whyte, H.F., "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators," *IEEE Transactions on Automatic Control*, Vol. AC-45, No. 3, March 2000, pp. 477–482.

[41] Bar-Shalom, Y. and Fortmann, T.E., *Tracking and Data Association*, Academic Press, Boston, MA, 1988.

[42] Wan, E. and van der Merwe, R., "The Unscented Kalman Filter," *Kalman Filtering and Neural Networks*, edited by S. Haykin, chap. 7, Wiley, 2001.

[43] Athans, M., Wishner, R.P., and Bertolini, A., "Suboptimal State Estimation for Continuous-Time Nonlinear Systems from Discrete Noisy Measurements," *IEEE Transactions on Automatic Control*, Vol. AC-13, No. 5, Oct. 1968, pp. 504–514.

[44] Nagpal, K.M. and Khargonekar, P.P., "Filtering and Smoothing in an H_∞ Setting," *IEEE Transactions on Automatic Control*, Vol. AC-36, No. 2, Feb. 1991, pp. 152–166.

[45] Francis, B.A., *A Course in H_∞ Control Theory*, Springer-Verlag, Berlin, 1987.

[46] Zhou, K., Doyle, J.C., and Glover, K., *Robust and Optimal Control*, Prentice Hall, Upper Saddle River, NJ, 1996.

[47] Kailath, T., "A View of Three Decades of Linear Filtering Theory," *IEEE Transactions on Information Theory*, Vol. IT-20, No. 2, March 1974, pp. 146–181.

[48] Chen, G., editor, *Approximate Kalman Filtering*, World Scientific, Singpore, China, 1993.

[49] Maskell, S. and Gordon, N., "A Tutorial on Particle Filters for On-line Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, Feb. 2002, pp. 174–189.

[50] Doucet, A., de Freitas, N., and Gordan, N., editors, *Sequential Monte Carlo Methods in Practice*, Springer-Verlag, New York, NY, 2001.