

CHAPTER 10

Additional topics in Kalman filtering

The use of wrong *a priori* statistics in the design of a Kalman filter can lead to large estimation errors or even to a divergence of errors.

—Raman Mehra [Meh72]

The previous chapters covered the essentials of Kalman filtering and should provide a firm foundation for further studies. This chapter discusses some additional important topics related to Kalman filtering. Section 10.1 talks about how to verify that a Kalman filter is operating reliably. When we run computer-based simulations of a Kalman filter, we can tell if the filter is working because we are in control of the simulation model and so we can compare the true state with the estimated state. However, in the real world we do not know what the true state is – after all, that is why we need a Kalman filter. In those situations, it is more difficult to verify that the Kalman filter's estimates are reliable.

Section 10.2 discusses multiple-model estimation, which is a way of estimating system states when we are not sure of which model is governing the dynamics of the system. This can be useful when the system model changes due to events of which the engineer may not be aware. Section 10.3 discusses reduced-order filtering. Many system models are of high order, which means that the corresponding Kalman filter will also be of high order. The high order of the filter may prevent the real-time implementation of the Kalman filter due to computational constraints. In these

cases a smaller, suboptimal filter (called a reduced-order filter) can be designed to give acceptable estimation performance at a lower computational cost. Section 10.4 discusses robust Kalman filtering, which is a way of making the filter less sensitive to variations in the assumed system model. Section 10.5 discusses the topic of delayed measurements. Sometimes the measurements do not arrive at the filter in chronological order because of processing delays. In these cases, we can modify the filter to optimally incorporate measurements that arrive at the filter in the wrong sequence.

10.1 VERIFYING KALMAN FILTER PERFORMANCE

We can verify Kalman filter performance, or adjust the gain of the Kalman filter, using our knowledge of the statistics of the innovations. The innovations is defined as $(y_k - H_k \hat{x}_k^-)$, and in this section we will show that it is a zero-mean white stochastic process with a covariance of $(H_k P_k^- H_k^T + R_k)$.

Recall our original system model, along with the one-step *a priori* update equation for the state estimate:

$$\begin{aligned} x_k &= F_{k-1} x_{k-1} + w_{k-1} \\ y_k &= H_k x_k + v_k \\ \hat{x}_{k+1}^- &= F_k \hat{x}_k^- + F_k K_k (y_k - H_k \hat{x}_k^-) \end{aligned} \quad (10.1)$$

The innovations is defined as the quantity in parentheses in the update equation. The innovations can be thought of as the part of the measurement that contains new information and that is therefore used to update the state estimate (apart from our knowledge of the state transition matrix). If the innovations was zero then the state estimate would simply be updated according to the state transition matrix. A nonzero innovations allows the measurement to affect the state estimate. The innovations r_k can be written as

$$\begin{aligned} r_k &= y_k - H_k \hat{x}_k^- \\ &= (H_k x_k + v_k) - H_k \hat{x}_k^- \\ &= H_k (x_k - \hat{x}_k^-) + v_k \\ &= H_k \epsilon_k + v_k \end{aligned} \quad (10.2)$$

where ϵ_k , the *a priori* estimation error, is defined by the above equation. The covariance of the innovations is given as

$$E[r_k r_i^T] = E[(H_k \epsilon_k + v_k)(H_i \epsilon_i + v_i)^T] \quad (10.3)$$

Let us see what the covariance is when $k \neq i$. We can assume without loss of generality that $k > i$. We then obtain

$$E[r_k r_i^T] = H_k E(\epsilon_k \epsilon_i^T) H_i^T + H_k E(\epsilon_k v_i^T) \quad (10.4)$$

Note that two of the cross terms reduced to zero because of the whiteness of v_k , and the fact that the estimation error ϵ_i is independent of v_k for $k > i$. In order to evaluate this covariance, we need to evaluate $E(\epsilon_k \epsilon_i^T)$ and $E(\epsilon_k v_i^T)$. First we

will evaluate $E(\epsilon_k \epsilon_i^T)$. In order to evaluate this term, notice that the *a priori* state estimate can be written as follows:

$$\begin{aligned}\hat{x}_{k+1}^- &= F_k \hat{x}_k^- + F_k K_k (y_k - H_k \hat{x}_k^-) \\ &= F_k \hat{x}_k^- + F_k K_k (H_k x_k + v_k - H_k \hat{x}_k^-) \\ &= F_k \hat{x}_k^- + F_k K_k H_k (x_k - \hat{x}_k^-) + F_k K_k v_k\end{aligned}\quad (10.5)$$

The *a priori* estimation error can be written as

$$\begin{aligned}\epsilon_{k+1} &= x_{k+1} - \hat{x}_{k+1}^- \\ &= F_k (x_k - \hat{x}_k^-) - F_k K_k H_k (x_k - \hat{x}_k^-) + w_k - F_k K_k v_k \\ &= F_k (I - K_k H_k) \epsilon_k + (w_k - F_k K_k v_k) \\ &= \tilde{\phi}_k \epsilon_k + v'_k\end{aligned}\quad (10.6)$$

where $\tilde{\phi}_k$ and v'_k are defined by the above equation. This is a linear discrete-time system for ϵ_k with the state transition matrix

$$\tilde{\phi}_{k,i} = \begin{cases} \tilde{\phi}_{k-1} \tilde{\phi}_{k-2} \cdots \tilde{\phi}_i & k > i \\ I & k = i \end{cases}\quad (10.7)$$

ϵ_k can be solved from the initial condition ϵ_i as follows:

$$\epsilon_k = \tilde{\phi}_{k,i} \epsilon_i + \sum_{j=i}^{k-1} \tilde{\phi}_{k,j+1} v'_j\quad (10.8)$$

The covariance of $\epsilon_k \epsilon_i^T$ can be written as

$$E(\epsilon_k \epsilon_i^T) = E \left[\left(\tilde{\phi}_{k,i} \epsilon_i + \sum_{j=i}^{k-1} \tilde{\phi}_{k,j+1} v'_j \right) \epsilon_i^T \right]\quad (10.9)$$

We see that all of the $v'_j \epsilon_i^T$ terms in the above expression are zero-mean. This is because all of the v'_j noise terms occur at time i or later and so do not affect ϵ_i . [Note from Equation (10.6) that ϵ_i is affected only by the noise terms at time $(i-1)$ or earlier.] Therefore,

$$E(v'_j \epsilon_i^T) = 0 \quad (j \geq i)\quad (10.10)$$

We therefore see that Equation (10.9) can be written as

$$\begin{aligned}E(\epsilon_k \epsilon_i^T) &= \tilde{\phi}_{k,i} E(\epsilon_i \epsilon_i^T) \\ &= \tilde{\phi}_{k,i} P_i^-\end{aligned}\quad (10.11)$$

Now that we have computed $E(\epsilon_k \epsilon_i^T)$, we need to solve for $E(\epsilon_k v_i^T)$ in order to arrive at our goal, which is the evaluation of Equation (10.4). $E(\epsilon_k v_i^T)$ can be written as

$$E(\epsilon_k v_i^T) = E \left[\left(\tilde{\phi}_{k,i} \epsilon_i + \sum_{j=i}^{k-1} \tilde{\phi}_{k,j+1} v'_j \right) v_i^T \right]\quad (10.12)$$

The $\epsilon_i v_i^T$ term in the above expression is zero-mean, and the $v_j' v_i^T$ terms are zero-mean for $j > i$. The above covariance can therefore be written as

$$\begin{aligned} E(\epsilon_k v_i^T) &= E(\tilde{\phi}_{k,i+1} v_i^T) \\ &= E(\tilde{\phi}_{k,i+1} (w_i - F_i K_i v_i)^T) \\ &= -\tilde{\phi}_{k,i+1} F_i K_i R_i \end{aligned} \quad (10.13)$$

Substituting this equation, along with Equation (10.11), into Equation (10.4) gives

$$\begin{aligned} E(r_k r_i^T) &= H_k E(\epsilon_k \epsilon_i^T) H_i^T + H_k E(\epsilon_k v_i^T) \\ &= H_k \tilde{\phi}_{k,i} P_i^- H_i^T - H_k \tilde{\phi}_{k,i+1} F_i K_i R_i \\ &= H_k \tilde{\phi}_{k,i+1} (\tilde{\phi}_i P_i^- H_i^T - F_i K_i R_i) \end{aligned} \quad (10.14)$$

Now use the fact from Equation (10.6) that $\tilde{\phi}_i = F_i(I - K_i H_i)$ to obtain

$$\begin{aligned} E(r_k r_i^T) &= H_k \tilde{\phi}_{k,i+1} (F_i P_i^- H_i^T - F_i K_i H_i P_i^- H_i^T - F_i K_i R_i) \\ &= H_k \tilde{\phi}_{k,i+1} [F_i P_i^- H_i^T - F_i K_i (H_i P_i^- H_i^T + R_i)] \end{aligned} \quad (10.15)$$

Now use the fact that $K_i = P_i^- H_i^T (H_i P_i^- H_i^T + R_i)^{-1}$ (the standard Kalman filter gain equation) to obtain

$$\begin{aligned} E(r_k r_i^T) &= H_k \tilde{\phi}_{k,i+1} (F_i P_i^- H_i^T - F_i P_i^- H_i^T) \\ &= 0 \quad \text{for } k > i \end{aligned} \quad (10.16)$$

So we see that the innovations r_k is white noise. Our next task is to determine its covariance. In order to do this we write the covariance as

$$\begin{aligned} E(r_k r_k^T) &= E[(y_k - H_k \hat{x}_k^-)(y_k - H_k \hat{x}_k^-)^T] \\ &= E\{[H_k(x_k - \hat{x}_k^-) + v_k][H_k(x_k - \hat{x}_k^-) + v_k]^T\} \\ &= H_k E(\epsilon_k \epsilon_k^T) H_k^T + E(v_k v_k^T) \\ &= H_k P_k^- H_k^T + R_k \end{aligned} \quad (10.17)$$

We therefore see that the innovations is a white noise process with zero mean and a covariance of $(H_k P_k^- H_k^T + R_k)$. While the Kalman filter is operating, we can process the innovations, compute its mean and covariance, and verify that it is white with the expected mean and covariance. If it is colored, nonzero-mean, or has the wrong covariance, then there is something wrong with the filter. The most likely reason for such a discrepancy is a modeling error. In particular, an incorrect value of F , H , Q , or R could cause the innovations to statistically deviate from its theoretically expected behavior. Statistical methods can then be used to tune F , H , Q , and R in order to force the innovations to be white zero-mean noise with a covariance of $(H_k P_k^- H_k^T + R_k)$ [Meh70, Meh72]. This concept is illustrated in Figure 10.1. A scalar example is presented in Problem 10.1.

Alternatively, if the engineer is uncertain of the correct values of F , H , Q , and R , then a bank of Kalman filters can be run in parallel, each Kalman filter with a value of F , H , Q , and R that the engineer thinks may be likely. Then the innovations can be inspected in each filter, and the one that matches theory is assumed to have

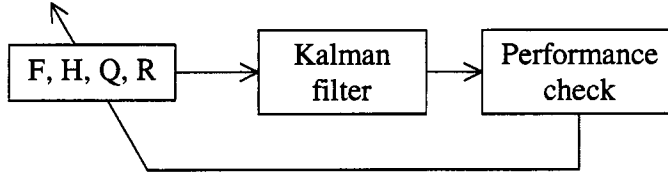


Figure 10.1 This figure illustrates how the performance of a Kalman filter can be used to tune the values of F , H , Q , and R in order to obtain residual statistics that agree with theory. Alternatively, the Kalman gain K could be tuned directly.

the correct F , H , Q , and R , so the state estimate that comes out of that filter is probably the most correct. See [Kob03] for an application of this idea.

The analysis of this section can also be conducted for the continuous-time Kalman filter. The continuous-time innovations, $y(t) - H(t)\hat{x}(t)$, is a zero-mean white stochastic process with a covariance $R(t)$ (see Problem 10.2).

10.2 MULTIPLE-MODEL ESTIMATION

Suppose our system model is not known, or the system model changes depending on unknown factors. We can use multiple Kalman filters (one for each possible system model) and combine the state estimates to obtain a refined state estimate. Remember Bayes' rule from Section 2.1:

$$\Pr(x|y) = \frac{\Pr(y|x)\Pr(x)}{\Pr(y)} \quad (10.18)$$

Suppose that a random variable x can take one of N mutually exclusive values x_1, \dots, x_N . Then we can use Bayes' rule to write

$$\begin{aligned} \Pr(y) &= \Pr(y|x_1)\Pr(x_1) + \dots + \Pr(y|x_N)\Pr(x_N) \\ \Pr(x|y) &= \frac{\text{pdf}(y|x)\Pr(x)}{\sum_{i=1}^N \text{pdf}(y|x_i)\Pr(x_i)} \end{aligned} \quad (10.19)$$

where we have used the fact that the probability of an event occurring is directly proportional to the value of its pdf. Now suppose that we have the time-invariant system

$$\begin{aligned} x_k &= Fx_{k-1} + Gu_{k-1} + w_{k-1} \\ y_k &= Hx_k + v_k \\ w_k &\sim N(0, Q) \\ v_k &\sim N(0, R) \end{aligned} \quad (10.20)$$

The parameter set p is defined as the set (F, G, H, Q, R) . Suppose that p can take one of N possible values p_1, \dots, p_N . The question that we want to answer in this section is as follows: Given the measurements y_k , what is the probability that $p = p_j$? From Equation (10.19) this probability can be written as

$$\Pr(p_j|y_k) = \frac{\text{pdf}(y_k|p_j)\Pr(p_j)}{\sum_{i=1}^N \text{pdf}(y_k|p_i)\Pr(p_i)} \quad (10.21)$$

Now think about the probability that measurement y_k is observed given the fact that $p = p_j$. If $p = p_j$ then the state will take on some value x_k that is determined by the parameter set p_j . We therefore see that

$$\begin{aligned}\Pr(y_k|p_j) &= \Pr(y_k|x_k) \\ \text{pdf}(y_k|p_j) &= \text{pdf}(y_k|x_k)\end{aligned}\quad (10.22)$$

However, if our state estimate is accurate, then we know that $x_k \approx \hat{x}_k^-$. Therefore, the above equation can be written as

$$\text{pdf}(y_k|p_j) \approx \text{pdf}(y_k|\hat{x}_k^-) \quad (10.23)$$

The right side of the equation is the pdf of the measurement y_k given the fact that the state is \hat{x}_k^- . But since $y_k \approx H\hat{x}_k^- + v_k$, this pdf is approximately equal to the pdf of $(y_k - H\hat{x}_k^-)$. We therefore have

$$\begin{aligned}\text{pdf}(y_k|p_j) &\approx \text{pdf}(y_k - H\hat{x}_k^-) \\ &= \text{pdf}(r_k)\end{aligned}\quad (10.24)$$

where r_k is the residual defined in Section 10.1. From Section 10.1 we see that if w_k , v_k , and x_0 are Gaussian, then the residual r_k is a linear combination of Gaussian random variables. Recall from Section 2.4.2 that a linear combination of Gaussian random variables is itself Gaussian. In Section 10.1 we found the mean and variance of r_k . The pdf of r_k , which is approximated by the pdf of y_k given p_j , can therefore be approximated as

$$\text{pdf}(y_k|p_j) \approx \frac{\exp(-r_k^T S_k^{-1} r_k / 2)}{(2\pi)^{q/2} |S_k|^{1/2}} \quad (10.25)$$

where $r_k = y_k - H_k \hat{x}_k^-$, $S_k = H_k P_k^- H_k^T + R_k$, and q is the number of measurements.

Now from Bayes' rule we can write the following equation for the probability that $p = p_j$ given the fact that the measurement y_{k-1} is observed.

$$\Pr(p_j|y_{k-1}) = \frac{\Pr(y_{k-1}|p_j)\Pr(p_j)}{\Pr(y_{k-1})} \quad (10.26)$$

If we are presently at time k , then the measurement at time $(k-1)$ is a given. The value of the measurement at time $(k-1)$ is a certain event with a probability equal to one. Therefore, $\Pr(y_{k-1}|p_j) = \Pr(y_{k-1}) = 1$ and the above equation becomes

$$\Pr(p_j|y_{k-1}) = \Pr(p_j) \quad (10.27)$$

Now in Equation (10.21) we can substitute this equation for $\Pr(p_j)$, and we substitute Equation (10.25) for $\text{pdf}(y_k|p_j)$. This gives a time-recursive equation for evaluating the probability that $p = p_j$ given the fact that the measurement was equal to y_k . The multiple-model estimator can be summarized as follows.

The multiple-model estimator

1. For $j = 1, \dots, N$, initialize the probabilities of each parameter set before any measurements are obtained. These probabilities are denoted as $\Pr(p_j|y_0)$ ($j = 1, \dots, N$).

2. At each time step k we perform the following steps.

- (a) Run N Kalman filters, one for each parameter set p_j ($j = 1, \dots, N$). The *a priori* state estimate and covariance of the j th filter are denoted as \hat{x}_{kj}^- and P_{kj}^- .
- (b) After the measurement at time k is received, for each parameter set approximate the pdf of y_k given p_j as follows:

$$\text{pdf}(y_k|p_j) \approx \frac{\exp(-r_k^T S_k^{-1} r_k/2)}{(2\pi)^{q/2} |S_k|^{1/2}} \quad (10.28)$$

where $r_k = y_k - H_k \hat{x}_{kj}^-$, $S_k = H P_{kj}^- H^T + R_k$, and q is the number of measurements.

- (c) Estimate the probability that $p = p_j$ as follows.

$$\Pr(p_j|y_k) = \frac{\text{pdf}(y_k|p_j) \Pr(p_j|y_{k-1})}{\sum_{i=1}^N \text{pdf}(y_k|p_i) \Pr(p_i|y_{k-1})} \quad (10.29)$$

- (d) Now that each parameter set p_j has an associated probability, we can weight each \hat{x}_{kj}^- and P_{kj}^- accordingly to obtain

$$\begin{aligned} \hat{x}_k^- &= \sum_{j=1}^N \Pr(p_j|y_k) \hat{x}_{kj}^- \\ P_k^- &= \sum_{j=1}^N \Pr(p_j|y_k) P_{kj}^- \end{aligned} \quad (10.30)$$

- (e) We can estimate the true parameter set in one of several ways, depending on our application. For example, we can use the parameter set with the highest conditional probability as our parameter estimate, or we can estimate the parameter set as a weighted average of the parameter sets:

$$\hat{p} = \begin{cases} \operatorname{argmax}_{p_j} \Pr(p_j|y_k) & \text{max-probability method} \\ \sum_{j=1}^N \Pr(p_j|y_k) p_j & \text{weighted-average method} \end{cases} \quad (10.31)$$

As time progresses, some of the $\Pr(p_j|y_k)$ terms will approach zero. Those p_j possibilities can then be eliminated and the number N can be reduced.

In Equation (10.31), the function $\operatorname{argmax}_x f(x)$ returns the value of x at which the maximum of $f(x)$ occurs. For example, $\max(1-x)^2 = 0$ because the maximum of $(1-x)^2$ is 0, but $\operatorname{argmax}_x (1-x)^2 = 1$ because $(1-x)^2$ attains its maximum value when $x = 1$. A similar definition holds for the function argmin .

■ EXAMPLE 10.1

In this example, we consider a second-order system identification problem [Ste94]. Suppose that we have a continuous-time system with discrete-time measurements described as follows:

$$\begin{aligned}
\dot{x} &= Ax + Bw_1 \\
&= \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} x + \begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix} w_1 \\
y_k &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_k + v_k \\
w_1 &\sim N(0, Q_c) \\
v_k &\sim N(0, R)
\end{aligned} \tag{10.32}$$

The damping ratio $\zeta = 0.1$, and the process and measurement noise covariances Q_c and R are respectively equal to 1000 and $10I$. The natural frequency $\omega_n = 2$, but this is not known to the engineer. The engineer knows that ω_n^2 is either 4, 4.4, or 4.8 with the following *a priori* probabilities:

$$\begin{aligned}
\Pr(\omega_n^2 = 4) &= 0.1 \\
\Pr(\omega_n^2 = 4.4) &= 0.6 \\
\Pr(\omega_n^2 = 4.8) &= 0.3
\end{aligned} \tag{10.33}$$

The state equation can be written as

$$\begin{aligned}
\dot{x} &= Ax + w \\
w &\sim N(0, BQ_cB^T)
\end{aligned} \tag{10.34}$$

We can discretize the system using the technique given in Section 1.4. If the measurements are obtained every 0.1 seconds, then we discretize the state equation with a sample time of $T = 0.1$ to obtain

$$\begin{aligned}
x_k &= Fx_{k-1} + \Lambda w'_{k-1} \\
y_k &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_k + v_k \\
F &= \exp(AT) \\
\Lambda &= (F - I)F^{-1}
\end{aligned} \tag{10.35}$$

From Section 8.1 we know that the covariance Q' of the discrete-time noise w'_k is given as

$$Q' \approx BQ_cB^TT \tag{10.36}$$

This means that the discrete-time process dynamics can be written as

$$\begin{aligned}
x_k &= Fx_{k-1} + w_{k-1} \\
w_k &\sim N(0, Q) \\
Q &= (F - I)F^{-1}(BQ_cB^TT)F^{-T}(F^T - I)
\end{aligned} \tag{10.37}$$

The multiple-model estimator described in this section was run on this example. Three Kalman filters running in parallel each generate an estimate of the state. As the filters run, the probability of each parameter is updated by the multiple-model Kalman filter. Figure 10.2 shows the parameter probabilities for a typical simulation run. It is seen that even though the correct parameter has the lowest initial probability, the multiple-model filter estimate converges to the correct parameter after a few seconds.

▽▽▽

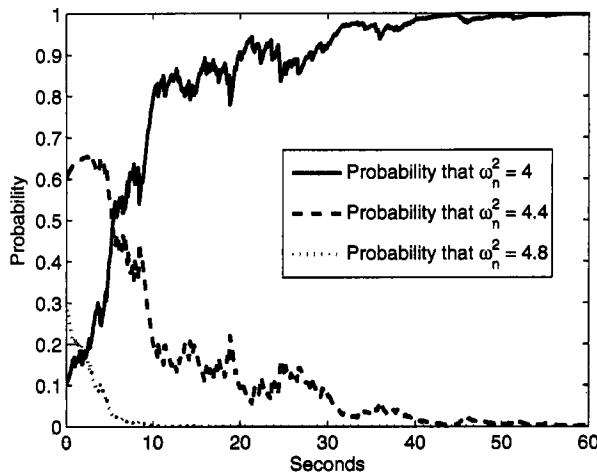


Figure 10.2 Parameter probabilities for the multiple-model Kalman filter for Example 10.1. The true parameter value is 4, and the filter converges to the correct parameter after a few seconds.

10.3 REDUCED-ORDER KALMAN FILTERING

If a user wants to estimate only a subset of the state vector, then a reduced-order filter can be designed. This can be the case, for example, in a real-time application where computational effort is a main consideration. Even in off-line applications, some types of problems (e.g., weather forecasting) can involve tens of thousands of states, which naturally motivates reduced-order filtering as a means to reduce computational effort [Pha98, Bal01].

Various approaches to reduced-order filtering have been proposed over the years. For example, if the dynamic model of the underlying system can be reduced to a lower-order model that approximates the full-order model, then the reduced-order model can form the basis of a normally designed Kalman filter [Kel99]. This is the approach taken in [Gli94, Sot99] for motor state estimation, in [Sim69, Ara94] for navigation system alignment, in [Bur93, Pat98] for image processing, and in [Cha96] for audio processing. If some of the states are not observable then the Kalman filter Riccati equation reduces to a lower-order equation [Yon80]. Reduced-order filtering can be implemented by approximating the covariance with a lower-rank SVD-like decomposition [Pha98, Bal01]. If some of the measurements are noise free, or if there are known equality constraints between some of the states, then the Kalman filter is a filter with an order that is lower than the underlying system [Bry65, Hae98] as discussed in Section 7.5.1 of this book. Optimal reduced-order filters are obtained from first principles in [Ber85, Nag87]. A more heuristic approach to reduced-order filtering is to decouple portions of the matrix multiplications in the Kalman filter equations [Chu87]. In this section we will present two different approaches to reduced-order filtering.

10.3.1 Anderson's approach to reduced-order filtering

Anderson and Moore [And79] suggest a framework for reduced-order filtering that is fully developed in [Sim06a] and in this section. This approach is based on the idea that we do not always need to estimate all of the states of a system. Sometimes, with a system that has n states, we are interested only in estimating m linear combinations of the states, where $m < n$. In this case, it stands to reason that we could devise a filter with an order less than n that estimates the m linear combinations that we are interested in. Suppose our state space system is given as

$$\begin{aligned}\bar{x}_{k+1} &= \bar{F}\bar{x}_k + \bar{G}w_k \\ y_k &= \bar{H}\bar{x}_k + v_k\end{aligned}\tag{10.38}$$

We desire to estimate the following m linear combinations of the state: $T_1^T \bar{x}$, $T_2^T \bar{x}$, \dots , $T_m^T \bar{x}$, where each T_i^T is a row vector. Define the $n \times n$ matrix T as

$$T = \begin{bmatrix} T_1^T \\ \vdots \\ T_m^T \\ S \end{bmatrix}\tag{10.39}$$

where S is arbitrary as long as it makes T a nonsingular $n \times n$ matrix. Now perform the state transformation

$$x = T\bar{x}\tag{10.40}$$

This means that $\bar{x} = T^{-1}x$. From these relationships we can obtain a state space description of the system in terms of the new state as follows:

$$\begin{aligned}T^{-1}x_{k+1} &= \bar{F}T^{-1}x_k + \bar{G}w_k \\ x_{k+1} &= T\bar{F}T^{-1}x_k + T\bar{G}w_k \\ &= Fx_k + Gw_k \\ y_k &= \bar{H}T^{-1}x_k + v_k \\ &= Hx_k + v_k\end{aligned}\tag{10.41}$$

where F , G , and H are defined by the above equations. Remember that our goal is to estimate the first m elements of x , which we will denote as \tilde{x} . We therefore partition x as follows:

$$x = \begin{bmatrix} \tilde{x} \\ \tilde{\tilde{x}} \end{bmatrix}\tag{10.42}$$

We can then write equations for \tilde{x}_{k+1} , $\tilde{\tilde{x}}_{k+1}$, and y_k as follows:

$$\begin{aligned}\tilde{x}_{k+1} &= F_{11}\tilde{x}_k + F_{12}\tilde{\tilde{x}}_k + G_1w_k \\ \tilde{\tilde{x}}_{k+1} &= F_{21}\tilde{x}_k + F_{22}\tilde{\tilde{x}}_k + G_2w_k \\ y_k &= H_1\tilde{x}_k + H_2\tilde{\tilde{x}}_k + v_k\end{aligned}\tag{10.43}$$

where the F_{ij} , G_i , and H_i matrices are appropriately dimensioned partitions of F , G , and H . Now we propose the following form for the one-step *a posteriori* estimator of \tilde{x} :

$$\hat{\tilde{x}}_{k+1}^+ = F_{11}\hat{\tilde{x}}_k^+ + K_k(y_{k+1} - H_1F_{11}\hat{\tilde{x}}_k^+)\tag{10.44}$$

This predictor/corrector form for the estimate of \tilde{x} is very similar to the predictor/corrector form of the standard Kalman filter. The estimation error is given as follows:

$$\begin{aligned}
 e_{k+1} &= \tilde{x}_{k+1} - \hat{\tilde{x}}_{k+1}^+ \\
 &= F_{11}(\tilde{x}_k - \hat{\tilde{x}}_k^+) + F_{12}\tilde{x}_k + G_1w_k - K_k(y_{k+1} - H_1F_{11}\hat{\tilde{x}}_k^+) \\
 &= (I - K_kH_1)F_{11}e_k + [F_{12} - K_k(H_1F_{12} - H_2F_{22})]\tilde{x}_k - \\
 &\quad K_kH_2F_{21}\tilde{x}_k - K_kv_{k+1} + [G_1 - K_k(H_1G_1 + H_2G_2)]w_k
 \end{aligned} \tag{10.45}$$

Now we will introduce the following notation for various covariance matrices:

$$\begin{aligned}
 P_k &= E(e_k e_k^T) \\
 \tilde{P}_k &= E(\tilde{x}_k \tilde{x}_k^T) \\
 \tilde{\tilde{P}}_k &= E(\tilde{\tilde{x}}_k \tilde{\tilde{x}}_k^T) \\
 \Sigma_k &= E(\tilde{x}_k \tilde{\tilde{x}}_k^T) \\
 \tilde{\Pi}_k &= E(\hat{\tilde{x}}_k \tilde{x}_k^T) \\
 \tilde{\tilde{\Pi}}_k &= E(\hat{\tilde{x}}_k \tilde{\tilde{x}}_k^T)
 \end{aligned} \tag{10.46}$$

With this notation and the equations given earlier in this section, we can obtain the following expressions for these covariances:

$$\begin{aligned}
 \tilde{P}_{k+1} &= F_{11}\tilde{P}_kF_{11}^T + (F_{11}\Sigma_kF_{12}^T) + (\cdots)^T + F_{12}\tilde{\tilde{P}}_kF_{12}^T + G_1Q_kG_1^T \\
 \tilde{\tilde{P}}_{k+1} &= F_{21}\tilde{P}_kF_{21}^T + (F_{21}\Sigma_kF_{22}^T) + (\cdots)^T + F_{22}\tilde{\tilde{P}}_kF_{22}^T + G_2Q_kG_2^T \\
 \Sigma_{k+1} &= F_{11}\tilde{P}_kF_{21}^T + F_{11}\Sigma_kF_{22}^T + F_{12}\tilde{\tilde{P}}_kF_{22}^T + G_1Q_kG_2^T \\
 \tilde{\Pi}_{k+1} &= (I - K_kH_1)F_{11}(\tilde{\Pi}_kF_{11}^T + \tilde{\tilde{\Pi}}_kF_{12}^T) + \\
 &\quad K_k(H_1F_{11} + H_2F_{21})(\tilde{P}_kF_{11}^T + \Sigma_kF_{12}^T) + \\
 &\quad K_k(H_1F_{12} + H_2F_{22})(\Sigma_k^TF_{11}^T + \tilde{\tilde{P}}_kF_{12}^T) + \\
 &\quad K_k(H_1G_1 + H_2G_2)Q_kG_1^T \\
 \tilde{\tilde{\Pi}}_{k+1} &= (I - K_kH_1)F_{11}(\tilde{\Pi}_kF_{21}^T + \tilde{\tilde{\Pi}}_kF_{22}^T) + \\
 &\quad K_k(H_1F_{11} + H_2F_{21})(\tilde{P}_kF_{21}^T + \Sigma_kF_{22}^T) + \\
 &\quad K_k(H_1F_{12} + H_2F_{22})(\Sigma_k^TF_{21}^T + \tilde{\tilde{P}}_kF_{22}^T) + \\
 &\quad K_k(H_1G_1 + H_2G_2)Q_kG_2^T \\
 P_{k+1} &= (I - K_kH_1)F_{11}P_kF_{11}^T(I - K_kH_1)^T + \\
 &\quad \left[(I - K_kH_1)F_{11}(\Sigma_k - \tilde{\tilde{\Pi}}_k)Y_{k+1} \right] + \left[\cdots \right]^T + \\
 &\quad \left[(I - K_kH_1)F_{11}(\tilde{\Pi}_k - \tilde{P}_k)F_{21}^TH_2^TK_k^T \right] + \left[\cdots \right]^T + \\
 &\quad Y_{k+1}^T\tilde{\tilde{P}}_kY_{k+1} - (Y_{k+1}^T\Sigma_k^TF_{21}^TH_2^TK_k^T) + (\cdots)^T + \\
 &\quad K_kH_2F_{21}\tilde{P}_kF_{21}^TH_2^TK_k^T + K_kR_{k+1}K_k^T + \\
 &\quad [G_1 - K_k(H_1G_1 + H_2G_2)]Q_k[G_1 - K_k(H_1G_1 + H_2G_2)]^T
 \end{aligned} \tag{10.47}$$

where Y_k is defined as

$$Y_k = [F_{12} - K_k(H_1F_{12} + H_2F_{22})]^T \tag{10.48}$$

Now we can find the optimal reduced-order gain K_k at each time step as follows:

$$\begin{aligned} K_k &= \operatorname{argmin} \operatorname{Tr} P_{k+1} \\ \frac{\partial \operatorname{Tr} P_{k+1}}{\partial K_k} &= 0 \end{aligned} \quad (10.49)$$

In order to compute the partial derivative we have to remember from Section 1.1.3 that

$$\begin{aligned} \frac{\partial \operatorname{Tr}(ABA^T)}{\partial A} &= AB + AB^T \\ \frac{\partial \operatorname{Tr}(AB)}{\partial A} &= B^T \\ \frac{\partial \operatorname{Tr}(BA^T)}{\partial A} &= B \end{aligned} \quad (10.50)$$

Armed with these tools we can compute the partial derivative of Equation 10.49 and set it equal to zero to obtain

$$K_k = A_k^{-1} B_k \quad (10.51)$$

where A_k and B_k are given as follows:

$$\begin{aligned} A_k &= H_1 F_{11} P_k F_{11}^T H_1^T + \left[H_1 F_{11} (\Sigma_k - \tilde{\Pi}_k) (H_1 F_{12} + H_2 F_{22})^T \right] + \\ &\quad \left[\cdots \right]^T + \left[H_{11} F_{11} (\tilde{P}_k - \tilde{\Pi}_k) F_{21}^T H_2^T \right] + \left[\cdots \right]^T + \\ &\quad (H_1 F_{12} + H_2 F_{22}) \tilde{P}_k (H_1 F_{12} + H_2 F_{22})^T + \\ &\quad \left[(H_1 F_{12} + H_2 F_{22}) \Sigma_k^T F_{21}^T H_2^T \right] + \left[\cdots \right]^T + H_2 F_{21} \tilde{P}_k F_{21}^T H_2^T + \\ &\quad R_{k+1} + (H_1 G_1 + H_2 G_2) Q_k (H_2 G_1 + H_2 G_2)^T \\ B_k &= \left(F_{11} P_k + F_{12} \Sigma_k^T - F_{12} \tilde{\Pi}_k^T \right) F_{11}^T H_1^T + \\ &\quad \left(F_{11} \Sigma_k - F_{11} \tilde{\Pi}_k + F_{12} \tilde{P}_k \right) (H_1 F_{12} + H_2 F_{22})^T + \\ &\quad \left(F_{11} \tilde{P}_k - F_{11} \tilde{\Pi}_k + F_{12} \Sigma_k^T \right) F_{21}^T H_2^T + G_1 Q_k (H_1 G_1 + H_2 G_2)^T \end{aligned} \quad (10.52)$$

Equation (10.51) ends up being a long and complicated expression for the reduced-order gain. In fact, this reduced-order filter is probably more computationally expensive than the full-order filter (depending on the values of m and n). However, if the gain of the reduced-order filter converges to steady state, then it can be computed off-line to obtain savings in real-time computational cost and memory usage. However, note that the reduced-order filter may not be stable, even if the full-order Kalman filter is stable.

■ EXAMPLE 10.2

Suppose we are given the following system:

$$x_{k+1} = \begin{bmatrix} 0.9 & 0.1 \\ 0.2 & 0.7 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} w_k$$

$$\begin{aligned}
y_k &= \begin{bmatrix} 0 & 1 \end{bmatrix} x_k + v_k \\
w_k &\sim (0, 0.1) \\
v_k &\sim (0, 1)
\end{aligned} \tag{10.53}$$

We want to find a reduced-order estimator of the first element of x . In this example the reduced-order gain of Equation (10.51) converges to a steady-state value after about 80 time steps. The estimation-error variance of the reduced-order filter converges to a value that is about 10% higher than the estimation-error variance of the full-order filter for the first state, as shown in Figure 10.3. The estimation error for the reduced-order filter and the full-order filter is shown in Figure 10.3 for a typical simulation. In this example, the standard deviation of the estimation error was 0.46 for the full-order filter and 0.50 for the reduced-order filter. The steady-state full-order estimator is given as follows:

$$\begin{aligned}
\hat{x}_{k+1}^- &= \begin{bmatrix} 0.9 & 0.1 \\ 0.2 & 0.7 \end{bmatrix} \hat{x}_k^+ \\
\hat{x}_k^+ &= \hat{x}_k^- + K(y_k - \begin{bmatrix} 0 & 1 \end{bmatrix} \hat{x}_k^-) \\
K &= \begin{bmatrix} 0.1983 \\ 0.1168 \end{bmatrix}
\end{aligned} \tag{10.54}$$

The steady-state reduced-order estimator is given as follows:

$$\begin{aligned}
\hat{\tilde{x}}_{k+1}^+ &= 0.9\hat{\tilde{x}}_k^+ + K_r[y_{k+1} - (0)(0.9)\hat{\tilde{x}}_k^+] \\
&= 0.9\hat{\tilde{x}}_k^+ + K_r y_{k+1} \\
K_r &= 0.1420
\end{aligned} \tag{10.55}$$

▽▽▽

10.3.2 The reduced-order Schmidt–Kalman filter

Stanley Schmidt's approach to reduced-order filtering can be used if the states are decoupled from each other in the dynamic equation [Sch66, Bro96, Gre01]. This happens, for instance, if colored measurement noise is accounted for by augmenting the state vector (see Section 7.2.2). In fact, satellite navigation with colored measurement noise was the original motivation for this approach.

Suppose we have a system in the form

$$\begin{aligned}
\begin{bmatrix} \tilde{x}_{k+1} \\ \tilde{\tilde{x}}_{k+1} \end{bmatrix} &= \begin{bmatrix} F_1 & 0 \\ 0 & F_2 \end{bmatrix} \begin{bmatrix} \tilde{x}_k \\ \tilde{\tilde{x}}_k \end{bmatrix} + \begin{bmatrix} \tilde{w}_k \\ \tilde{\tilde{w}}_k \end{bmatrix} \\
\tilde{w}_k &\sim (0, Q_1) \\
\tilde{\tilde{w}}_k &\sim (0, Q_2) \\
y_k &= \begin{bmatrix} H_1 & H_2 \end{bmatrix} \begin{bmatrix} \tilde{x}_k \\ \tilde{\tilde{x}}_k \end{bmatrix} + v_k \\
v_k &\sim (0, R)
\end{aligned} \tag{10.56}$$

We want to estimate \tilde{x}_k but we do not care about estimating $\tilde{\tilde{x}}_k$. Suppose we use a Kalman filter to estimate the entire state vector. The estimation-error covariance

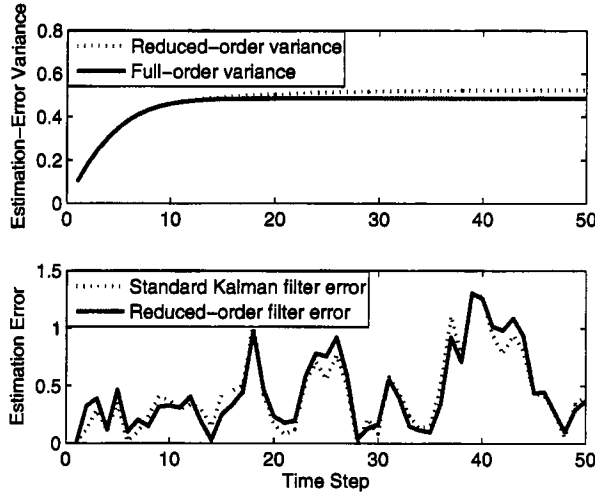


Figure 10.3 Results for Example 10.2. The top figure shows the analytical estimation-error variances for the first state for the full-order filter and the reduced-order filter. As expected, the reduced-order filter has a higher estimation-error variance, but the small degradation in performance may be worth the computational savings, depending on the application. The bottom figure shows typical error magnitudes for the estimate of the first state for the full-order filter and the reduced-order filter. The reduced-order filter has slightly larger estimation errors.

can be partitioned as follows:

$$P = \begin{bmatrix} \tilde{P} & \Sigma \\ \Sigma^T & \tilde{\tilde{P}} \end{bmatrix} \quad (10.57)$$

We are omitting the time subscripts for ease of notation. The Kalman gain is usually written as $K = P^{-1}H^T(HP^{-1}H^T + R)^{-1}$. With our new notation it can be written as follows:

$$\begin{aligned} K &= \begin{bmatrix} \tilde{K} \\ \tilde{\tilde{K}} \end{bmatrix} \\ &= \begin{bmatrix} \tilde{P}^{-1} & \Sigma^{-1} \\ (\Sigma^{-1})^T & \tilde{\tilde{P}}^{-1} \end{bmatrix} \times \\ &\quad \begin{bmatrix} H_1^T \\ H_2^T \end{bmatrix} \left[\begin{pmatrix} H_1 & H_2 \end{pmatrix} \begin{pmatrix} \tilde{P}^{-1} & \Sigma^{-1} \\ (\Sigma^{-1})^T & \tilde{\tilde{P}}^{-1} \end{pmatrix} \begin{pmatrix} H_1^T \\ H_2^T \end{pmatrix} + R \right]^{-1} \end{aligned} \quad (10.58)$$

By multiplying out this equation we can write the formula for \tilde{K} as follows.

$$\tilde{K} = (\tilde{P}^{-1}H_1^T + \Sigma^{-1}H_2^T)\alpha^{-1} \quad (10.59)$$

where α is defined as

$$\alpha = H_1\tilde{P}^{-1}H_1^T + H_1\Sigma^{-1}H_2^T + H_2(\Sigma^{-1})^TH_1^T + H_2\tilde{\tilde{P}}^{-1}H_2^T + R \quad (10.60)$$

The measurement-update equation for \hat{x} is normally written as $\hat{x}_k^+ = \hat{x}_k^- + K(y_k - H\hat{x}_k^-)$. With our new notation it is written as

$$\begin{bmatrix} \hat{x}_k^+ \\ \hat{\tilde{x}}_k^+ \end{bmatrix} = \begin{bmatrix} \tilde{K} \\ \tilde{\tilde{K}} \end{bmatrix} \left(y_k - H_1 \hat{x}_k^- - H_2 \hat{\tilde{x}}_k^- \right) \quad (10.61)$$

Since we are not going to estimate \tilde{x} with the reduced-order filter, we set $\hat{\tilde{x}}_k^- = 0$ in the above equation to obtain the following measurement-update equation for \hat{x}_k^+ :

$$\hat{x}_k^+ = \hat{x}_k^- + \tilde{K} \left(y_k - H_1 \hat{x}_k^- \right) \quad (10.62)$$

The measurement-update equation for P is usually written as $P^+ = (I - KH)P^- (I - KH)^T + KRK^T$. With our new notation it is written as

$$\begin{bmatrix} \tilde{P}^+ & \Sigma^+ \\ (\Sigma^+)^T & \tilde{\tilde{P}}^+ \end{bmatrix} = \left[\begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} - \begin{pmatrix} \tilde{K} \\ \tilde{\tilde{K}} \end{pmatrix} \begin{pmatrix} H_1 & H_2 \end{pmatrix} \right] \begin{bmatrix} \tilde{P}^- & \Sigma^- \\ (\Sigma^-)^T & \tilde{\tilde{P}}^- \end{bmatrix} \times \\ \left[\begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} - \begin{pmatrix} \tilde{K} \\ \tilde{\tilde{K}} \end{pmatrix} \begin{pmatrix} H_1 & H_2 \end{pmatrix} \right]^T + \begin{pmatrix} \tilde{K} \\ \tilde{\tilde{K}} \end{pmatrix} R \begin{pmatrix} \tilde{K}^T & \tilde{\tilde{K}}^T \end{pmatrix} \quad (10.63)$$

At this point, we assume that $\tilde{\tilde{K}} = 0$. This can be justified if the measurement noise associated with the \tilde{x} states is large, or if H_2 is small, or if the elements of \tilde{x} are small. The \tilde{x} elements are then referred to as consider states, nuisance states, or nuisance variables, because they are only partially used in the reduced-order state estimator, and because we are not interested in estimating them. Based on Equation (10.63), the update equation for \tilde{P}^+ can then be written as

$$\begin{aligned} \tilde{P}^+ &= (I - \tilde{K}H_1)\tilde{P}^-(I - \tilde{K}H_1)^T - \tilde{K}H_2(\Sigma^-)^T(I - \tilde{K}H_1)^T - \\ &\quad (I - \tilde{K}H_1)\Sigma^-H_2^T\tilde{K}^T + \tilde{K}H_2\tilde{\tilde{P}}^-H_2^T\tilde{K}^T + \tilde{K}R\tilde{K}^T \end{aligned} \quad (10.64)$$

Multiplying out the above equation and then using the definition of α from Equation (10.60) results in

$$\begin{aligned} \tilde{P}^+ &= \tilde{P}^- - \tilde{K}H_1\tilde{P}^- - \tilde{P}^-H_1^T\tilde{K}^T + \tilde{K}\alpha\tilde{K}^T - \tilde{K}H_2(\Sigma^-)^T - \Sigma^-H_2^T\tilde{K}^T \\ &= \tilde{P}^- - \tilde{K}H_1\tilde{P}^- - \tilde{P}^-H_1^T\tilde{K}^T + (\tilde{P}^-H_1^T + \Sigma^-H_2^T)\tilde{K}^T - \tilde{K}H_2(\Sigma^-)^T - \\ &\quad \Sigma^-H_2^T\tilde{K}^T \\ &= (I - \tilde{K}H_1)\tilde{P}^- - \tilde{K}H_2(\Sigma^-)^T \end{aligned} \quad (10.65)$$

This gives the measurement-update equation for \tilde{P}^+ . We can go through similar manipulations with Equation (10.63) to obtain

$$\begin{aligned} \Sigma^+ &= (I - \tilde{K}H_1)\Sigma^- - \tilde{K}H_2\tilde{\tilde{P}}^- \\ \tilde{\tilde{P}}^+ &= \tilde{\tilde{P}}^- \end{aligned} \quad (10.66)$$

Putting it all together results in the reduced-order Schmidt-Kalman filter. We can summarize the reduced-order filter as follows.

The reduced-order Schmidt–Kalman filter

1. The system and measurement equations are given in Equation (10.56).
2. At each time step execute the following equations to obtain \hat{x}_k , the estimate of the desired part of the state:

$$\begin{aligned}
 \tilde{K}_k &= (\tilde{P}_k^- H_1^T + \Sigma_k^- H_2^T) \alpha_k^{-1} \\
 \alpha_k &= H_1 \tilde{P}_k^- H_1^T + H_1 \Sigma_k^- H_2^T + H_2 (\Sigma_k^-)^T H_1^T + H_2 \tilde{P}_k^- H_2^T + R \\
 \hat{x}_k^+ &= \hat{x}_k^- + \tilde{K}_k (y_k - H_1 \hat{x}_k^-) \\
 \tilde{P}_k^+ &= (I - \tilde{K}_k H_1) \tilde{P}_k^- - \tilde{K}_k H_2 (\Sigma_k^-)^T \\
 \Sigma_k^+ &= (I - \tilde{K}_k H_1) \Sigma_k^- - \tilde{K}_k H_2 \tilde{P}_k^- \\
 \tilde{P}_k^+ &= \tilde{P}_k^- \\
 \hat{x}_{k+1}^- &= F_1 \hat{x}_k^+ \\
 \tilde{P}_{k+1}^- &= F_1 \tilde{P}_k^+ F_1^T + Q_1 \\
 \Sigma_{k+1}^- &= F_1 \Sigma_k^+ F_2^T \\
 \tilde{P}_{k+1}^- &= F_1 \tilde{P}_k^+ F_2^T + Q_2
 \end{aligned} \tag{10.67}$$

■ EXAMPLE 10.3

Consider the following system:

$$\begin{aligned}
 x_{k+1} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_k + w_k \\
 y_k &= \begin{bmatrix} 1 & 1 \end{bmatrix} x_k + v_k \\
 w_k &\sim (0, Q) \quad Q = \text{diag}(1, 0) \\
 v_k &\sim (0, R) \quad R = 1
 \end{aligned} \tag{10.68}$$

Figure 10.4 shows a typical example of the estimation error of the first element of the state vector for the full-order filter and the reduced-order filter. It is seen that the performances of the two estimators are virtually identical. In other words, we can save a lot of computational effort with only a marginal degradation of estimation performance by using the reduced-order filter.

▽▽▽

10.4 ROBUST KALMAN FILTERING

The Kalman filter works well, but it assumes that the system model and noise statistics are known. If any of these assumptions are violated then the filter estimates can degrade. This was noted early in the history of Kalman filtering [Soo65, Hef66, Nis66].

Daniel Pena and Irwin Guttman give an overview of several methods of robustifying the Kalman filter [Spa88, Chapter 9]. For example, although the Kalman filter

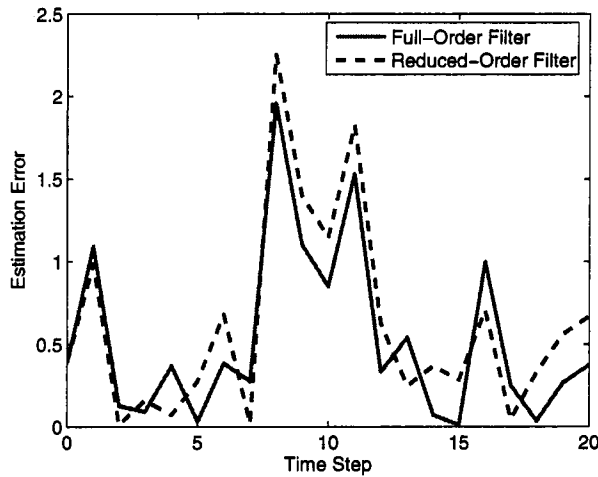


Figure 10.4 Results for Example 10.3. Typical error magnitudes for the estimate of the first state for the full-order filter and the reduced-order filter. The reduced-order filter has only slightly larger estimation errors.

is the optimal linear filter, it is not the optimal filter in general for non-Gaussian noise. Noise in nature is often approximately Gaussian but with heavier tails, and the Kalman filter can be modified to accommodate these types of density functions [Mas75, Mas77, Tsa83]. Sometimes, measurements do not contain any useful information but consist entirely of noise (probabilistically), and the Kalman filter can be modified to deal with this possibility also [Nah69, Sin73, Ath77, Bar78].

The problem of Kalman filtering with uncertainties in the system matrix F_k , the measurement matrix H_k , and the noise covariances Q_k and R_k , has been considered by several authors [Xie94, Zha95, Hsi96, The96, Xie04]. This can be called adaptive filtering or robust filtering. Comparisons of adaptive filtering methods for navigation are presented in [Hid03]. Continuous-time adaptive filtering is discussed in [Bar05, Mar05]. Methods for identifying the noise covariances Q and R are presented in [Meh70, Meh72, Als74, Mye76]. Additional material on robust Kalman filtering can be found in [Che93].

In this section we present a conceptually straightforward way of making the Kalman filter more robust to uncertainties in Q and R [Kos04]. Suppose we have the linear time-invariant system

$$\begin{aligned}
 x_{k+1} &= Fx_k + w_k \\
 y_k &= Hx_k + v_k \\
 w_k &\sim (0, Q) \\
 v_k &\sim (0, R)
 \end{aligned} \tag{10.69}$$

Now suppose that a general steady-state gain K (not necessarily the Kalman gain) is used in a predictor/corrector type of state estimator. The state estimate update equations are then given as follows:

$$\begin{aligned}
\hat{x}_{k+1}^- &= F\hat{x}_k^+ \\
\hat{x}_{k+1}^+ &= \hat{x}_{k+1}^- + K(y_{k+1} - H\hat{x}_{k+1}^-) \\
&= F\hat{x}_k^+ + K(Hx_{k+1} + v_{k+1} - HF\hat{x}_k^+) \\
&= KHx_{k+1} + (I - KH)F\hat{x}_k^+ + Kv_{k+1} \\
&= (KHFx_k + KHw_k) + (I - KH)F\hat{x}_k^+ + Kv_{k+1} \quad (10.70)
\end{aligned}$$

The error in the *a posteriori* state estimate can be written as

$$\begin{aligned}
e_{k+1} &= x_{k+1} - \hat{x}_{k+1}^+ \\
&= (Fx_k + w_k) - [(KHFx_k + KHw_k) + (I - KH)F\hat{x}_k^+ + Kv_{k+1}] \\
&= (I - KH)Fx_k + (I - KH)w_k - (I - KH)F\hat{x}_k^+ - Kv_{k+1} \\
&= (I - KH)Fe_k + (I - KH)w_k - Kv_{k+1} \quad (10.71)
\end{aligned}$$

So the covariance of the estimation error can be written as

$$\begin{aligned}
P_{k+1} &= E(e_{k+1}e_{k+1}^T) \\
&= (I - KH)FP_kF^T(I - KH)^T + (I - KH)Q(I - KH)^T + \\
&\quad KRK^T \quad (10.72)
\end{aligned}$$

The steady-state covariance P satisfies the following Riccati equation:

$$P = (I - KH)FPF^T(I - KH)^T + (I - KH)Q(I - KH)^T + KRK^T \quad (10.73)$$

Note that we derived this without making any assumption on the optimality of the filter gain K . That is, this equation holds regardless of what filter gain K we use. Now we can consider what happens when there is no measurement noise, and what happens when there is no process noise. Define P_1 as the steady-state estimation-error covariance when $R = 0$, and P_2 as the steady-state estimation-error covariance when $Q = 0$. The above equation for P shows that

$$\begin{aligned}
P_1 &= (I - KH)FP_1F^T(I - KH)^T + (I - KH)Q(I - KH)^T \\
P_2 &= (I - KH)FP_2F^T(I - KH)^T + KRK^T \quad (10.74)
\end{aligned}$$

Adding these two covariances together results in

$$\begin{aligned}
P_1 + P_2 &= (I - KH)FP_1F^T(I - KH)^T + (I - KH)Q(I - KH)^T + \\
&\quad (I - KH)FP_2F^T(I - KH)^T + KRK^T \\
&= (I - KH)F(P_1 + P_2)F^T(I - KH)^T + \\
&\quad (I - KH)Q(I - KH)^T + KRK^T \quad (10.75)
\end{aligned}$$

Comparing this equation with Equation (10.73) shows that P and the sum $(P_1 + P_2)$ both satisfy the same Riccati equation. This shows that

$$P = P_1 + P_2 \quad (10.76)$$

This shows an interesting linearity property of a general predictor/corrector type of state estimator. The estimation covariance is equal to the sum of the covariance

due to process noise only and the covariance due to measurement noise only. Recall from Chapter 5 that the Kalman filter was designed to minimize the trace of P . So the Kalman filter minimizes the trace of $(P_1 + P_2)$.

Now suppose that the true process noise and measurement noise covariances are different from those assumed by the Kalman filter. The filter is designed under the assumption that the noise covariances are Q and R , but the true noise covariances are \tilde{Q} and \tilde{R} :

$$\begin{aligned}\tilde{Q} &= (1 + \alpha)Q \\ \tilde{R} &= (1 + \beta)R\end{aligned}\tag{10.77}$$

where α and β are unknown scalars. These differences between the assumed and true covariances will result in a change in the estimation-error covariance of the filter. The true estimation-error covariance \tilde{P} will be equal to the assumed covariance P plus some difference ΔP . This can be written as

$$\begin{aligned}\tilde{P} &= (I - KH)F\tilde{P}F^T(I - KH)^T + (I - KH)\tilde{Q}(I - KH)^T + K\tilde{R}K^T \\ P + \Delta P &= (I - KH)F(P + \Delta P)F^T(I - KH)^T + \\ &\quad (1 + \alpha)(I - KH)Q(I - KH)^T + (1 + \beta)KRK^T\end{aligned}\tag{10.78}$$

Comparing this equation with Equation (10.73) shows that

$$\Delta P = (I - KH)F\Delta P F^T(I - KH)^T + \alpha(I - KH)Q(I - KH)^T + \beta KRK^T\tag{10.79}$$

Now we repeat this same line of reasoning for the computation of the true estimation-error covariance when the process noise is zero ($\tilde{P}_1 = P_1 + \Delta P_1$) and the true estimation-error covariance when the measurement noise is zero ($\tilde{P}_2 = P_2 + \Delta P_2$). Equation (10.74) shows that

$$\begin{aligned}\tilde{P}_1 &= (I - KH)F\tilde{P}_1F^T(I - KH)^T + (I - KH)\tilde{Q}(I - KH)^T \\ P_1 + \Delta P_1 &= (I - KH)F(P_1 + \Delta P_1)F^T(I - KH)^T + \\ &\quad (1 + \alpha)(I - KH)Q(I - KH)^T \\ \tilde{P}_2 &= (I - KH)F\tilde{P}_2F^T(I - KH)^T + K\tilde{R}K^T \\ P_2 + \Delta P_2 &= (I - KH)F(P_2 + \Delta P_2)F^T(I - KH)^T + (1 + \beta)KRK^T\end{aligned}\tag{10.80}$$

Comparing these equations with Equation (10.74) shows that

$$\begin{aligned}\Delta P_1 &= (I - KH)F\Delta P_1F^T(I - KH)^T + \alpha(I - KH)Q(I - KH)^T \\ \Delta P_2 &= (I - KH)F\Delta P_2F^T(I - KH)^T + \beta KRK^T\end{aligned}\tag{10.81}$$

Adding these two equations and comparing with Equation (10.79) shows that

$$\Delta P = \Delta P_1 + \Delta P_2\tag{10.82}$$

Comparing Equations (10.74) and (10.81) shows that

$$\begin{aligned}\Delta P_1 &= \alpha P_1 \\ \Delta P_2 &= \beta P_2\end{aligned}\tag{10.83}$$

Combining Equations (10.82) and (10.83) shows that

$$\Delta P = \alpha P_1 + \beta P_2\tag{10.84}$$

Now suppose that α and β are independent zero-mean random variables with variances σ_1^2 and σ_2^2 , respectively. The previous equation shows that

$$\begin{aligned} E[\text{Tr}(\Delta P)] &= E(\alpha)\text{Tr}(P_1) + E(\beta)\text{Tr}(P_2) \\ &= 0 \\ E\{[\text{Tr}(\Delta P)]^2\} &= E\{[\alpha\text{Tr}(X_1) + \beta\text{Tr}(X_2)]^2\} \\ &= \sigma_1^2\text{Tr}^2(P_1) + \sigma_2^2\text{Tr}^2(P_2) \end{aligned} \quad (10.85)$$

This gives the variance of the change in the estimation-error covariance due to changes in the process and measurement-noise covariances. A robust filter should try to minimize this variance. In other words, a robust filter should have an estimation-error covariance that is insensitive to changes in the process and measurement-noise covariances. So the performance index of a robust filter can be written as follows:

$$\begin{aligned} J &= \rho\text{Tr}(P) + (1 - \rho)E\{[\text{Tr}(\Delta P)]^2\} \\ &= \rho[\text{Tr}(P_1) + \text{Tr}(P_2)] + (1 - \rho)[\sigma_1^2\text{Tr}^2(P_1) + \sigma_2^2\text{Tr}^2(P_2)] \end{aligned} \quad (10.86)$$

where ρ is the relative importance given to filter performance under nominal conditions (i.e., when Q and R are as expected), and $(1 - \rho)$ is the relative importance given to robustness. In other words, $(1 - \rho)$ is the relative weight given to minimizing the variation of the estimation-error covariance due to changes in Q and R . If $\rho = 1$ then we have the standard Kalman filter. If $\rho = 0$ then we will minimize changes in the estimation-error covariance, but the nominal estimation-error covariance may be poor. So ρ should be chosen to balance nominal performance and robustness. Unfortunately, the performance index J cannot be minimized analytically, so numerical methods must be used. P_1 and P_2 are functions of the gain K and can be computed using a DARE function in control system software. The partial derivative of J with respect to K must be computed numerically, and then the value of K can be changed using a gradient-descent method to decrease J .

■ EXAMPLE 10.4

Suppose we have a discretized second-order Newtonian system that is driven by an acceleration input. $x(1)$ represents position, $x(2)$ represents velocity, u_k represents the known acceleration input, and w_k represents the noisy acceleration input. This is the same as the system described in Example 9.1. The system is described as follows:

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} u_k + w_k \\ y_k &= \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + v_k \\ w_k &\sim (0, Q) \\ v_k &\sim (0, R) \\ Q &= q^2 \begin{bmatrix} T^4/4 & T^3/2 \\ T^3/2 & T/2 \end{bmatrix} \end{aligned} \quad (10.87)$$

The sample time $T = 0.1$. The variance q^2 of the acceleration noise is equal to 0.2^2 , and the variance R of the measurement noise is equal to 10^2 . Now

suppose that Q and R have relative uncertainties of one (one standard deviation). That is, $\sigma_1^2 = \sigma_2^2 = 1$. Suppose we find the robust filter gain using equal weighting for both nominal and robust performance (i.e., $\rho = 0.5$). Table 10.1 shows the average performance of the robust filter and the standard Kalman filter when Q and R change by factors of -0.8 and 3 , respectively. One question that remains is, How does the robust filter perform under nominal conditions? That is, since the Kalman filter is optimal, the robust filter will not perform as well as the Kalman filter when Q and R are equal to their nominal values. However, Table 10.2 shows that the performance degradation is marginal. In fact, the robust filter performs identically to the optimal filter (to two decimal places) under nominal conditions. During the gradient-descent optimization of Equation (10.86), the nominal part of the cost function increases from 2.02 to 2.04 , the robust part of the cost function decreases from 2.54 to 2.38 , and the total cost function decreases from 2.28 to 2.21 .

Table 10.1 RMS estimation errors for Example 10.4 over 100 seconds when the noise covariances are not nominal ($\rho = 0.5$, $\sigma_1 = \sigma_2 = 1$, $\alpha = -0.8$, $\beta = 3$)

	Position	Velocity
Standard Filter	4.62	0.38
Robust Filter	4.47	0.32

Table 10.2 RMS estimation errors for Example 10.4 over 100 seconds when the noise covariances are nominal ($\rho = 0.5$, $\sigma_1 = \sigma_2 = 1$, $\alpha = 0$, $\beta = 0$)

	Position	Velocity
Standard Filter	1.38	0.19
Robust Filter	1.38	0.19

▽▽▽

The robust filtering approach presented here opens several possible research topics. For example, under what conditions is the robust filter stable? Is the gain of the robust filter equal to the gain of a standard Kalman filter for some other related system? What is the true estimation-error covariance of the robust filter?

10.5 DELAYED MEASUREMENTS AND SYNCHRONIZATION ERRORS

In decentralized filtering systems, observations are often collected at various physical locations, and then transmitted in bulk to a central processing computer. In this type of setup, the measurements may not arrive at the processing computer synchronously. That is, the computer may receive measurements out of sequence. This is typically the case in target-tracking systems. Various approaches have been

taken to deal with this problem [Ale91, Bar95, Kas96, Lar98, Mal01]. The case of delayed measurements with uncertainty in the measurement sampling time is discussed in [Tho94a, Tho94b]. The approach to filtering delayed measurements that is presented here is based on [Bar02].

First we will present yet another form of the Kalman filter that will provide the basis for the delayed-measurement filter. Then we will derive the optimal way to incorporate delayed measurements into the Kalman filter estimate and covariance. In this section, we will have to change our notation slightly in order to carry out the derivation of the delayed measurement Kalman filter. We will use the following notation to represent a discrete-time system:

$$\begin{aligned}x(k) &= F(k-1)x(k-1) + w(k-1) \\ y(k) &= H(k)x(k) + v(k)\end{aligned}\tag{10.88}$$

where $w(k)$ and $v(k)$ are independent zero-mean white noise process with covariances $Q(k)$ and $R(k)$, respectively.

10.5.1 A statistical derivation of the Kalman filter

Suppose that we have an *a priori* estimate $\hat{x}^-(k)$ at time k , and we want to find an optimal way to update the state estimate based on the measurement at time k . We want our update equation to be linear (for reasons of mathematical tractability) so we decide to update the state estimate at time k with the equation

$$\hat{x}^+(k) = K(k)y(k) + b(k)\tag{10.89}$$

where $K(k)$ and $b(k)$ are a matrix and vector to be determined. Our first state estimation criterion is unbiasedness. We can see by taking the mean of Equation (10.89) that

$$\overline{\hat{x}^+}(k) = K(k)\bar{y}(k) + b(k)\tag{10.90}$$

This gives us the constraint that

$$b(k) = \bar{x}(k) - K(k)\bar{y}(k)\tag{10.91}$$

This will ensure that $\hat{x}^+(k)$ is unbiased regardless of the value of the gain matrix $K(k)$. Next we find the gain matrix $K(k)$ that minimizes the trace of the estimation error. First recall that

$$\begin{aligned}P_z &= E[(z - \bar{z})(z - \bar{z})^T] \\ &= E(zz^T) - \bar{z}\bar{z}^T\end{aligned}\tag{10.92}$$

for any general random vector z . Now set $z = x(k) - \hat{x}^+(k)$. With this definition of z we see that $\bar{z} = 0$. The quantity we want to minimize is given by the trace of the following matrix:

$$\begin{aligned}P^+(k) &= E[(x(k) - \hat{x}^+(k))(x(k) - \hat{x}^+(k))^T] \\ &= P_z + \bar{z}\bar{z}^T\end{aligned}\tag{10.93}$$

P_z can be computed as follows:

$$\begin{aligned}
 P_z &= E \{ [x(k) - \hat{x}^+(k) - E(x(k) - \hat{x}^+(k))] [\cdot \cdot]^T \} \\
 &= E \{ [x(k) - (K(k)y(k) + b(k)) - \bar{x}(k) - (K(k)\bar{y}(k) + b(k))] [\cdot \cdot]^T \} \\
 &= E \{ [(x(k) - \bar{x}(k)) - K(k)(y(k) - \bar{y}(k))] [\cdot \cdot]^T \} \\
 &= P^-(k) - K(k)P_{yx} - P_{xy}K^T(k) + K(k)P_yK^T(k)
 \end{aligned} \tag{10.94}$$

We are using the symbol P_{yx} to denote the cross covariance between y_k and x_k , P_{xy} to denote the cross covariance between x_k and y_k , and P_y to denote the covariance of y_k . Recall that $P_{xy} = P_{yx}^T$. We have omitted the subscript k on P_{yx} , P_{xy} , and P_y for notational convenience. We combine the above equation with (10.93) to obtain

$$\begin{aligned}
 \text{Tr } P^+(k) &= \text{Tr} (P^-(k) - K(k)P_{yx} - P_{xy}K(k)^T + K(k)P_yK(k)^T) + \text{Tr}(\bar{z}\bar{z}^T) \\
 &= \text{Tr} (P^-(k) - K(k)P_{yx} - P_{xy}K(k)^T + K(k)P_yK(k)^T) + \\
 &\quad ||\bar{x}(k) - K(k)\bar{y}(k) - b(k)||^2 \\
 &= \text{Tr} [(K(k) - P_{xy}P_y^{-1})P_y(K(k) - P_{xy}P_y^{-1})^T] + \\
 &\quad \text{Tr} (P^-(k) - P_{xy}P_y^{-1}P_{xy}^T) + ||\bar{x}(k) - K(k)\bar{y}(k) - b(k)||^2
 \end{aligned} \tag{10.95}$$

where we have used the fact that $\text{Tr}(AB) = \text{Tr}(BA)$ for compatibly dimensioned matrices [see Equation (1.26)]. We want to choose $K(k)$ and $b(k)$ in order to minimize the above expression. The second term is independent of $K(k)$ and $b(k)$, and the first and third terms are always nonnegative. The first and third terms can be minimized to zero when

$$\begin{aligned}
 K(k) &= P_{xy}P_y^{-1} \\
 b(k) &= \bar{x}(k) - K(k)\bar{y}(k)
 \end{aligned} \tag{10.96}$$

Note that this is the same value for $b(k)$ that we obtained in Equation (10.91) when we enforced unbiasedness in the state estimate. With these values of $K(k)$ and $b(k)$, we see that the first and third terms in (10.95) are equal to zero, so the estimation-error covariance $P^+(k)$ can be seen to be equal to the second term. Substituting these values into Equation (10.89) we obtain

$$\begin{aligned}
 \hat{x}^+(k) &= K(k)y(k) + \bar{x}(k) - K(k)\bar{y}(k) \\
 &= K(k)y(k) + \hat{x}^-(k) - K(k)H(k)\hat{x}^-(k) \\
 &= \hat{x}^-(k) + K(k)(y(k) - H(k)\hat{x}^-(k)) \\
 P^+(k) &= P^-(k) - P_{xy}P_y^{-1}P_{xy}^T \\
 &= P^-(k) - K(k)P_yK^T(k)
 \end{aligned} \tag{10.97}$$

Straightforward calculations (see Problem 10.8) show that P_{xy} and P_y can be computed as

$$\begin{aligned}
 P_{xy} &= P^-(k)H(k)^T \\
 P_y &= H(k)P^-(k)H(k)^T + R(k)
 \end{aligned} \tag{10.98}$$

Now consider our linear discrete-time system:

$$\begin{aligned}
 x(k) &= F(k-1)x(k-1) + w(k-1) \\
 y(k) &= H(k)x(k) + v(k)
 \end{aligned} \tag{10.99}$$

The noise processes $w(k)$ and $v(k)$ are white, zero-mean, and uncorrelated, with covariances $Q(k)$ and $R(k)$, respectively. We saw in Chapter 4 how the mean and covariance of the state propagates between measurement times. Those equations, along with the measurement-update equations derived above, provide the following Kalman filter equations:

$$\begin{aligned}
 \hat{x}^-(k) &= F(k-1)\hat{x}^+(k-1) \\
 P^-(k) &= F(k-1)P^+(k-1)F^T(k-1) + Q(k) \\
 P_{xy} &= P^-(k)H^T(k) \\
 P_y &= H(k)P^-(k)H^T(k) + R(k) \\
 K(k) &= P_{xy}P_y^{-1} \\
 \hat{x}^+(k) &= \hat{x}^-(k) + K(k)(y(k) - H(k)\hat{x}^-(k)) \\
 P^+(k) &= P^-(k) - K(k)P_yK^T(k) \\
 &= P^-(k) - P_{xy}P_y^{-1}P_{xy}^T
 \end{aligned} \tag{10.100}$$

These equations appear much different than the Kalman filter equations derived earlier in this book, but actually they are mathematically identical for linear systems.

10.5.2 Kalman filtering with delayed measurements

Now we need to complicate the notation a little bit more in order to derive the Kalman filter with delayed measurements. We will write our system equations as

$$\begin{aligned}
 x(k) &= F(k, k-1)x(k-1) + w(k, k-1) \\
 y(k) &= H(k)x(k) + v(k)
 \end{aligned} \tag{10.101}$$

$F(k, k-1)$ is the matrix that quantifies the state transition from time $(k-1)$ to time k . Similarly, $w(k, k-1)$ is the effect of the process noise on the state from time $(k-1)$ to time k . We can then generalize the state-space equation to the following:

$$x(k) = F(k, k_0)x(k_0) + w(k, k_0) \tag{10.102}$$

where k_0 is any time index less than k . The above equation can be solved for $x(k_0)$ as

$$x(k_0) = F(k_0, k)[x(k) - w(k, k_0)] \tag{10.103}$$

where $F(k_0, k) = F^{-1}(k, k_0)$. Note that $F(k, k_0)$ should always be invertible if it comes from a real system, because $F(k, k_0)$ comes from a matrix exponential that is always invertible (see Sections 1.2 and 1.4). The noise $w(k, k_0)$ is the cumulative effect of all of the process noise on the state from time k_0 to time k . Its covariance is defined as $Q(k, k_0)$:

$$w(k, k_0) \sim [0, Q(k, k_0)] \tag{10.104}$$

At time k we have the standard *a posteriori* Kalman filter estimate, which is the expected value of the state $x(k)$ conditioned on all of the measurements up to and including time k . We also have the *a posteriori* covariance of the estimate:

$$\begin{aligned}
 \hat{x}(k) &= E[x(k)|y(1), \dots, y(k)] \\
 &= E[x(k)|Y(k)] \\
 P(k) &= E\{[x(k) - \hat{x}(k)][x(k) - \hat{x}(k)]^T | Y(k)\}
 \end{aligned} \tag{10.105}$$

where $Y(k)$ is defined by the above equation; that is, $Y(k)$ is all of the measurements up to and including time k that have been processed by the Kalman filter. (There may be some measurements before time k that have not yet been processed by the filter. These measurements are not part of $Y(k)$.)

Now suppose an out-of-sequence measurement arrives. That is, we obtain a measurement from time $k_0 < k$ that we want to incorporate into the estimate and covariance at time k . The problem is how to modify the state estimate and covariance on the basis of this new measurement. The modified state estimate and covariance are given as follows:

$$\begin{aligned}\hat{x}(k|k_0) &= E[x(k)|Y(k), y(k_0)] \\ P(k|k_0) &= E\{[x(k) - \hat{x}(k, k_0)][x(k) - \hat{x}(k, k_0)]^T | Y(k), y(k_0)\} \quad (10.106)\end{aligned}$$

The approach here is to use the new measurement at time k_0 to obtain an updated state estimate and covariance at time k_0 , and then use those quantities to update the estimate and covariance at time k . We can use Equation (10.103) to obtain

$$\begin{aligned}E[x(k_0)|Y(k)] &= F(k_0, k)E[x(k) - w(k, k_0)|Y(k)] \\ &= F(k_0, k)[\hat{x}^-(k) - \hat{w}(k, k_0)] \quad (10.107)\end{aligned}$$

where $\hat{w}(k, k_0)$ is defined by the above equation; it is the expected value of the cumulative effect of the process noise from time k_0 to time k , conditioned on all of the measurements up to and including time k [but not including measurement $y(k_0)$]. Now define the vector

$$z(k) = \begin{bmatrix} x(k) \\ w(k, k_0) \end{bmatrix} \quad (10.108)$$

In general, we define the covariance of vector a conditioned on vector c , and the cross covariance of vectors a and b conditioned on vector c , as follows:

$$\begin{aligned}\text{Cov}(a|c) &= E[(a - \bar{a})(a - \bar{a})^T | c] \\ \text{Cov}(a, b|c) &= E[(a - \bar{a})(b - \bar{b})^T | c] \quad (10.109)\end{aligned}$$

We can then generalize Equation (10.100) to obtain

$$\begin{aligned}\hat{z}(k) &= \hat{z}^-(k) + \\ &\text{Cov}[z(k), y(k)|Y(k-1)]\text{Cov}^{-1}[y(k)|Y(k-1)](y(k) - H(k)\hat{x}^-(k)) \\ \text{Cov}[z(k)|Y(k)] &= \text{Cov}[z(k)|Y(k-1)] - \\ &\text{Cov}[z(k), y(k)|Y(k-1)]\text{Cov}^{-1}[y(k)|Y(k-1)]\text{Cov}[y(k), z(k)|Y(k-1)] \quad (10.110)\end{aligned}$$

The first covariance on the right side of the above $\hat{z}(k)$ equation can be written as

$$\text{Cov}[z(k), y(k)|Y(k-1)] = \begin{bmatrix} \text{Cov}[x(k), y(k)|Y(k-1)] \\ \text{Cov}[w(k, k_0), y(k)|Y(k-1)] \end{bmatrix} \quad (10.111)$$

Now consider the first covariance in the above equation. This can be written as

$$\begin{aligned}\text{Cov}[x(k), y(k)|Y(k-1)] &= \text{Cov}\{x(k)(H(k)x(k) + v(k))^T | Y(k-1)\} \\ &= \text{Cov}\{x(k)[H(k)x(k)]^T | Y(k-1)\} \\ &= \text{Cov}\{x(k)\} H^T(k) \\ &= P^-(k)H^T(k) \quad (10.112)\end{aligned}$$

where the covariance of $x(k)$ and $v(k)$ is zero since they are independent. Now consider the second covariance on the right side of Equation (10.111). This can be written as

$$\begin{aligned}
 \text{Cov}[w(k, k_0), y(k)|Y(k-1)] &= E \{ w(k, k_0)[y(k) - \hat{y}^-(k)]^T | Y(k-1) \} \\
 &= E \{ w(k, k_0)[H(k)(F(k, k_0)x(k_0) + w(k, k_0)) + v(k) - \hat{y}^-(k)]^T | Y(k-1) \} \\
 &= E \{ w(k, k_0)w^T(k, k_0)H^T(k) \} \\
 &= Q(k, k_0)H^T(k)
 \end{aligned} \tag{10.113}$$

where the cross covariances of $w(k, k_0)$ with $x(k_0)$, $v(k)$, and $\hat{y}^-(k)$ are zero since they are independent. We are using the notation $\hat{y}^-(k)$ to denote the expected value of $y(k)$ based on measurements up to (but not including) time k . Now consider the conditional covariance of $y(k)$ in Equation (10.110). This was derived in Equation (10.17) in Section 10.1 as

$$\text{Cov}[y(k)|Y(k-1)] = H(k)P^-(k)H^T(k) + R(k) \tag{10.114}$$

We will write this expression more compactly as

$$\text{Cov}[r(k)] = S(k) \tag{10.115}$$

where the residual $r(k) = y(k) - H(k)\hat{x}^-(k)$ and its covariance $S(k)$ are defined by the two above equations. Substituting Equations (10.112) and (10.113) into Equation (10.111), and then substituting into Equation (10.110), gives

$$\begin{aligned}
 \hat{z}(k) &= \begin{bmatrix} \hat{x}(k) \\ \hat{w}(k, k_0) \end{bmatrix} \\
 &= \begin{bmatrix} \hat{x}^-(k) \\ \hat{w}^-(k, k_0) \end{bmatrix} + \begin{bmatrix} P^-(k)H^T(k) \\ Q(k, k_0)H^T(k) \end{bmatrix} S^{-1}(k)r(k)
 \end{aligned} \tag{10.116}$$

This shows that

$$\begin{aligned}
 \hat{w}(k, k_0) &= \hat{w}^-(k, k_0) + Q(k, k_0)H^T(k)S^{-1}(k)r(k) \\
 &= Q(k, k_0)H^T(k)S^{-1}(k)r(k)
 \end{aligned} \tag{10.117}$$

because $E[\hat{w}(k, k_0)|Y(k-1)] = 0$ [since $w(k, k_0)$ is independent of the measurements]. Substituting this expression into Equation (10.107) gives

$$E[x(k_0)|Y(k)] = F(k_0, k) [\hat{x}(k) - Q(k, k_0)H^T(k)S^{-1}(k)r(k)] \tag{10.118}$$

This is called the retrodiction of the state estimate from time k back to time k_0 . Whereas a prediction equation is used to predict the state at some future time, a retrodiction equation is used to predict the state at some past time. In this case, the state estimate at time k [i.e., $\hat{x}(k)$] is retrodicted back to time k_0 to obtain the state estimate at time k_0 , which is denoted above as $E[x(k_0)|Y(k)]$. Note that $E[x(k_0)|Y(k)]$ is computed on the basis of all the measurements up to and including time k , but does not consider the measurement at time k_0 .

Now we can write Equation (10.110) as follows:

$$\begin{aligned}
 \text{Cov}[z(k)|Y(k)] &= \text{Cov} \left\{ \begin{bmatrix} x(k) \\ w(k, k_0) \end{bmatrix} \middle| Y(k) \right\} \\
 &= \text{Cov} \left\{ \begin{bmatrix} x(k) \\ w(k, k_0) \end{bmatrix} \middle| Y(k-1) \right\} - \\
 &\quad \text{Cov} \left\{ \begin{bmatrix} x(k) \\ w(k, k_0) \end{bmatrix}, y(k)|Y(k-1) \right\} \text{Cov}^{-1}[y(k)|Y(k-1)] \times \\
 &\quad \text{Cov} \left\{ y(k), \begin{bmatrix} x(k) \\ w(k, k_0) \end{bmatrix}^T \middle| Y(k-1) \right\} \\
 &= \begin{bmatrix} \text{Cov}[x(k)|Y(k-1)] & \text{Cov}[x(k), w(k, k_0)|Y(k-1)] \\ \text{Cov}[w(k, k_0), x(k)|Y(k-1)] & \text{Cov}[w(k, k_0)|Y(k-1)] \end{bmatrix} - \\
 &\quad \begin{bmatrix} \text{Cov}[x(k), y(k)|Y(k-1)] \\ \text{Cov}[w(k, k_0), y(k)|Y(k-1)] \end{bmatrix} \text{Cov}^{-1}[y(k)|Y(k-1)] \times \\
 &\quad \begin{bmatrix} \text{Cov}[x(k), y(k)|Y(k-1)] \\ \text{Cov}[w(k, k_0), y(k)|Y(k-1)] \end{bmatrix}^T \quad (10.119)
 \end{aligned}$$

From Equation (10.102) we can write

$$\begin{aligned}
 \text{Cov}[x(k), w(k, k_0)|Y(k-1)] &= E[x(k)w^T(k, k_0)|Y(k-1)] \\
 &= E \{ [F(k, k_0)x(k_0) + w(k, k_0)]w^T(k, k_0)|Y(k-1) \} \\
 &= E \{ w(k, k_0)w^T(k, k_0)|Y(k-1) \} \\
 &= Q(k, k_0) \quad (10.120)
 \end{aligned}$$

where we have used the independence of $x(k_0)$ and $w(k, k_0)$. Now substitute this equation along with Equations (10.112), (10.113), and (10.114) into Equation (10.119) to obtain

$$\begin{aligned}
 \text{Cov} \left\{ \begin{bmatrix} x(k) \\ w(k, k_0) \end{bmatrix} \middle| Y(k) \right\} &= \begin{bmatrix} P^-(k) & Q(k, k_0) \\ Q(k, k_0) & Q(k, k_0) \end{bmatrix} - \\
 &\quad \begin{bmatrix} P^-(k)H^T(k) \\ Q(k, k_0)H^T(k) \end{bmatrix} S^{-1}(k) \begin{bmatrix} P^-(k)H^T(k) \\ Q(k, k_0)H^T(k) \end{bmatrix}^T \quad (10.121)
 \end{aligned}$$

From this equation we can write the conditional covariance of $w(k, k_0)$, and cross covariance of $x(k)$ and $w(k, k_0)$, as follows:

$$\begin{aligned}
 P_w(k, k_0) &= \text{Cov}[w(k, k_0)|Y(k)] \\
 &= Q(k, k_0) - Q(k, k_0)H^T(k)S^{-1}(k)H(k)Q(k, k_0) \\
 P_{xw}(k, k_0) &= \text{Cov}[x(k), w(k, k_0)|Y(k)] \\
 &= Q(k, k_0) - P^-(k)H^T(k)S^{-1}(k)H(k)Q(k, k_0) \quad (10.122)
 \end{aligned}$$

Using this in Equation (10.103) gives the conditional covariance of the state retrodiction as follows:

$$\begin{aligned}
P(k_0, k) &= \text{Cov}[x(k_0)|Y(k)] \\
&= F(k_0, k) \text{Cov}[x(k) - w(k, k_0)|Y(k)] F^T(k_0, k) \\
&= F(k_0, k) \{ \text{Cov}[x(k)|Y(k)] - \text{Cov}[x(k), w(k, k_0)|Y(k)] - \\
&\quad \text{Cov}^T[x(k), w(k, k_0)|Y(k)] + \text{Cov}[w(k, k_0)|Y(k)] \} F^T(k_0, k) \\
&= F(k_0, k) \{ P^+(k) - P_{xw}(k, k_0) - P_{xw}^T(k, k_0) + \\
&\quad P_w(k, k_0) \} F^T(k_0, k)
\end{aligned} \tag{10.123}$$

Using the above along with Equation (10.101) we obtain the conditional covariance of $y(k_0)$ as

$$\begin{aligned}
S(k_0) &= \text{Cov}[y(k_0)|Y(k)] \\
&= E \{ [H(k_0)x(k_0) + v(k_0)][H(k_0)x(k_0) + v(k_0)]^T | Y(k) \} \\
&= H(k_0)P(k_0, k)H^T(k_0) + R(k_0)
\end{aligned} \tag{10.124}$$

We can use Equations (10.101) and (10.103) to obtain the conditional covariance between $x(k)$ and $y(k_0)$ as

$$\begin{aligned}
P_{xy}(k, k_0) &= \text{Cov}[x(k), y(k_0)|Y(k)] \\
&= \text{Cov} \{ x(k), H(k_0)F(k_0, k)[x(k) - w(k, k_0)] + v(k_0) | Y(k) \} \\
&= [P^+(k) - P_{xw}(k, k_0)] F^T(k_0, k) H^T(k_0)
\end{aligned} \tag{10.125}$$

We can substitute this into the top partition of the $\hat{z}(k)$ expression in Equation (10.110) to obtain the estimate of $x(k)$ which is updated on the basis of the measurement $y(k_0)$:

$$\hat{x}(k, k_0) = \hat{x}(k) + P_{xy}(k, k_0) S^{-1}(k_0) [y(k_0) - H(k_0)\hat{x}(k_0, k)] \tag{10.126}$$

where $\hat{x}(k_0, k)$ is the retrodiction of the state estimate given in Equation (10.118). From the top partition of the $\text{Cov}[z(k)|Y(k)]$ expression in Equation (10.110) we obtain

$$\begin{aligned}
\text{Cov}[x(k)|Y(k), y(k_0)] &= P(k, k_0) \\
&= P(k) - P_{xy}(k, k_0) S^{-1}(k_0) P_{xy}^T(k, k_0)
\end{aligned} \tag{10.127}$$

These equations show how the state estimate and its covariance can be updated on the basis of an out-of-sequence measurement. The delayed-measurement Kalman filter can be summarized as follows.

The delayed-measurement Kalman filter

1. The Kalman filter is run normally on the basis of measurements that arrive sequentially. If we are presently at time k in the Kalman filter, then we have $\hat{x}^-(k)$ and $P^-(k)$, the *a priori* state estimate and covariance that are based on measurements up to and including time $(k-1)$. We also have $\hat{x}(k)$ and $P(k)$, the *a posteriori* state estimate and covariance that are based on measurements up to and including time k .

2. If we receive a measurement $y(k_0)$, where $k_0 < k$, then we can update the state estimate and its covariance to $\hat{x}(k, k_0)$ and $P(k, k_0)$ as follows.

- (a) Retrodict the state estimate from k back to k_0 as shown in Equation (10.118):

$$\begin{aligned} S(k) &= H(k)P^-(k)H^T(k) + R(k) \\ \hat{x}(k_0, k) &= F(k_0, k) [\hat{x}(k) - Q(k, k_0)H^T(k)S^{-1}(k)r(k)] \end{aligned} \quad (10.128)$$

- (b) Compute the covariance of the retrodicted state using Equations (10.122) and (10.123):

$$\begin{aligned} P_w(k, k_0) &= Q(k, k_0) - Q(k, k_0)H^T(k)S^{-1}(k)H(k)Q(k, k_0) \\ P_{xw}(k, k_0) &= Q(k, k_0) - P^-(k)H^T(k)S^{-1}(k)H(k)Q(k, k_0) \\ P(k_0, k) &= F(k_0, k) \{ P(k) - P_{xw}(k, k_0) - P_{xw}^T(k, k_0) + \\ &\quad P_w(k, k_0) \} F^T(k_0, k) \end{aligned} \quad (10.129)$$

- (c) Compute the covariance of the retrodicted measurement at time k_0 using Equation (10.124):

$$S(k_0) = H(k_0)P(k_0, k)H^T(k_0) + R(k_0) \quad (10.130)$$

- (d) Compute the covariance of the state at time k and the retrodicted measurement at time k_0 using Equation (10.125):

$$P_{xy}(k, k_0) = [P(k) - P_{xw}(k, k_0)]F^T(k_0, k)H^T(k_0) \quad (10.131)$$

- (e) Use the delayed measurement $y(k_0)$ to update the state estimate and its covariance:

$$\begin{aligned} \hat{x}(k, k_0) &= \hat{x}(k) + P_{xy}(k, k_0)S^{-1}(k_0)[y(k_0) - H(k_0)\hat{x}(k_0, k)] \\ P(k, k_0) &= P(k) - P_{xy}(k, k_0)S^{-1}(k_0)P_{xy}^T(k, k_0) \end{aligned} \quad (10.132)$$

It is possible to make some simplifying approximations to this delayed measurement filter in order to decrease computational cost with only a slight degradation in performance [Bar02].

10.6 SUMMARY

In this chapter we discussed some important topics related to Kalman filtering that extend beyond standard results. We have seen how to verify if a Kalman filter is operating reliably. This gives us a quantifiable confidence in the accuracy of our filter estimates. We also discussed multiple-model estimation, which is a way of estimating system states when we are not sure of which model is governing the dynamics of the system. This can be useful when the system model changes in unpredictable ways. We discussed reduced-order filtering, which can be used to estimate a subset of the system states while saving computational effort. We derived a robust Kalman filter, which makes the filter less sensitive to variations

in the assumed system model. Robust filtering naturally leads into the topic of H_∞ filtering, which we will discuss in Chapter 11. Finally, we derived a way to update the state estimate when a measurement arrives at the filter in the wrong chronological order because of processing delays.

There are several other important extensions to Kalman filtering that we have not had time to discuss in this chapter. One is the variable structure filter, which is a combination of the Kalman filter with variable structure control. This guarantees stability under certain conditions and often provides performance better than the Kalman filter, especially when applied to nonlinear systems [Hab03]. Another recent proposal is the proportional integral Kalman filter, which adds an integral term to the measurement state update and thereby improves stability and reduces steady-state tracking errors [Bas99]. Another interesting topic is the use of a perturbation estimator to estimate the process noise. This allows model uncertainties to be lumped with process noise so that the process-noise estimate increases the robustness of the filter [Kwo03].

PROBLEMS

Written exercises

10.1 In this problem we consider the scalar system

$$\begin{aligned}x_{k+1} &= x_k + w_k \\ y_k &= x_k + v_k\end{aligned}$$

where w_k and v_k are white and uncorrelated with respective variances Q and R , which are unknown. A suboptimal steady-state value of K is used in the state estimator since Q and R are unknown.

- Use the expression for P_k^- along with the first expression for P_k^+ in Equation (5.19) to find the steady-state value of P_k^- as a function of the suboptimal value of K and the true values of Q and R . [Note that the first expression for P_k^+ in Equation (5.19) does not depend on the value for K_k being optimal.]
- Now suppose that $E(r_k^2)$ and $E(r_{k+1}r_k)$ are found numerically as the filter runs. Find the true value of R and the steady-state value of P_k^- as a function of $E(r_k^2)$ and $E(r_{k+1}r_k)$.
- Use your results from parts (a) and (b) to find the true value of Q .

10.2 Show that the innovations $r = y - C\hat{x}$ of the continuous-time Kalman filter is white with covariance R .

10.3 Consider the system described in Problem 5.1. Find the steady-state variance of the Kalman filter innovations when $Q = R$ and when $Q = 2R$.

10.4 Consider the system of Problem 10.3 with $Q = R = 1$. Suppose the Kalman filter for the system has reached steady state. At time k the innovations $r_k = y_k - \hat{x}_k^-$.

- Find an approximate value for $\text{pdf}(y_k|p)$ (where p is the model used in the Kalman filter) if $r_k = 0$, if $r_k = 1$, and if $r_k = 2$.

- b) Suppose that the use of model p_1 gives $r_k = 0$, model p_2 gives $r_k = 1$, and model p_3 gives $r_k = 2$. Further suppose that $\Pr(p_1|y_{k-1}) = 1/4$, $\Pr(p_2|y_{k-1}) = 1/4$, and $\Pr(p_3|y_{k-1}) = 1/2$. Find $\Pr(p_j|y_k)$ for $j = 1, 2, 3$.

10.5 Consider the system described in Example 4.1 where the measurement consists of the predator population. Suppose that we want to estimate $x(1) + x(2)$, the sum of the predator and prey populations. Create an equivalent system with transformed states such that our goal is to estimate the first element of the transformed state vector.

10.6 Consider the system

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_k + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w_k \\ y_k &= \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + v_k \end{aligned}$$

where w_k and v_k are uncorrelated zero-mean white noise processes with variances q and R , respectively.

- Use Anderson's approach to reduced-order filtering to estimate the first element of the state vector. Find steady-state values for \tilde{P} , $\tilde{\Sigma}$, $\tilde{\Pi}$, and P . Find the steady-state gain K of the reduced-order filter.
- Use the full-order filter to estimate the entire state vector. Find steady-state values for P and K .
- Comment on the comparison between your answer for P in part (a) and part (b).

10.7 Consider the reduced-order filter of Example 10.3 with the initial condition $\tilde{P}_0^+ = 1$.

- Find analytical expressions for the steady-state values of \tilde{K} , α , \tilde{P}^+ , Σ^+ , \tilde{P}^- , \tilde{P}^- , Σ^- , and \tilde{P}^- .
- What does the reduced-order filter indicate for the steady-state *a posteriori* estimation-error variance of the first state? Find an analytical expression for the true steady-state *a posteriori* estimation-error variance of the first state when the reduced-order filter is used. Your answer should be a function of $x(2)$. Solve for the true steady-state *a posteriori* estimation-error variance of the first state when $x(2) = 0$, when $x(2) = 1$, and when $x(2) = 2$.
- What is the steady-state *a posteriori* estimation-error variance of the first state when the full-order filter is used?

10.8 Verify that the two expressions in Equation (10.98) are respectively equal to the cross-covariance of x and y , and the covariance of y .

10.9 Suppose you have the linear system $x_{k+1} = Fx_k + w_k$, where $w_k \sim (0, Q_k)$ is zero-mean white noise. Define $w(k+2, k)$ as the cumulative effect of all of the process noise on the state from time k to time $(k+2)$. What are the mean and covariance of $w(k+2, k)$?

10.10 Suppose that a Kalman filter is running with

$$F = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

$$\begin{aligned}
 H &= \begin{bmatrix} 1 & 0 \end{bmatrix} \\
 Q &= \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \\
 R &= 1 \\
 P^+(k) &= \begin{bmatrix} 1/2 & 0 \\ 0 & 1 \end{bmatrix}
 \end{aligned}$$

An out-of-sequence measurement from time $(k - 1)$ is received at the filter.

- What was the value of $P^-(k)$?
- Use the delayed-measurement filter to find the quantities $P_w(k, k - 1)$, $P_{xw}(k, k - 1)$, $P(k - 1, k)$, $P_{xy}(k, k - 1)$, and $P(k, k - 1)$.
- Realizing that the measurement at time $(k - 1)$ was not received at time $(k - 1)$, derive the value of $P^-(k - 1)$. Now suppose that the measurement was received in the correct sequence at time $(k - 1)$. Use the standard Kalman filter equations to compute $P^+(k - 1)$, $P^-(k)$, and $P^+(k)$. How does your computed value of $P^+(k)$ compare with the value of $P(k, k - 1)$ that you computed in part (b) of this problem?

10.11 Under what conditions will P_y in Equation (10.100) be invertible for all k ?

Computer exercises

10.12 Consider the equations

$$\begin{aligned}
 300x + 400y &= 700 \\
 100x + 133y &= 233
 \end{aligned}$$

- What is the solution of these equations?
- What is the solution of these equations if each constant in the second equation increases by 1?
- What is the condition number of the original set of equations?

10.13 Repeat Problem 10.12 for the equations

$$\begin{aligned}
 300x + 400y &= 700 \\
 100x + 200y &= 200
 \end{aligned}$$

Comment on the difference between this set of equations and the set given in Problem 10.12.

10.14 Tire tread is measured every τ weeks. After τ weeks, 20% of the tread has worn off, so we can model the dynamics of the tread height as $x_{k+1} = fx_k + w_k$, where $f = 0.8$, and w_k is zero-mean white noise with a variance of 0.01. We measure the tread height every τ weeks with zero-mean white measurement noise that has a variance of 0.01. The initial tread height is known to be exactly 1 cm. Write a program to simulate the system and a Kalman filter to estimate the tread height.

- Run the program for 10 time steps per tire, and for 1000 tires. What is the mean of the 10,000 measurement residuals?

- b) Suppose the Kalman filter designer incorrectly believes that 30% of the tread wears off every τ weeks. What is the mean of the 10,000 measurement residuals in this case?
- c) Suppose the Kalman filter designer incorrectly believes that 10% of the tread wears off every τ weeks. What is the mean of the 10,000 measurement residuals in this case?

10.15 Consider the system described in Problem 10.14. Suppose the engineer does not know the true value of f but knows the initial probabilities $\Pr(f = 0.8) = \Pr(f = 0.85) = \Pr(f = 0.9) = 1/3$. Run the multiple-model estimator for 10 time steps on 100 tires to estimate f . The f probabilities at each time step can be taken as the mean of the 100 f probabilities that are obtained from the 100 tire simulations, and similarly for the f estimate at each time step. Plot the f probabilities and the f estimate as a function of time.

10.16 Consider a scalar system with $F = H = 1$ and nominal noise variances $Q = R = 5$. The true but unknown noise variances \tilde{Q} and \tilde{R} are given as

$$\begin{aligned}\tilde{Q} &= (1 + \alpha)Q \\ \tilde{R} &= (1 + \beta)R \\ E(\alpha^2) &= \sigma_1^2 = 1/2 \\ E(\beta^2) &= \sigma_2^2 = 1\end{aligned}$$

where α and β are independent zero-mean random variables. The variance of the *a posteriori* estimation error is P if $\alpha = \beta = 0$. In general, α and β are nonzero and the variance of the estimation error is $P + \Delta P$. Plot P , $E(\Delta P^2)$, and $(P + E(\Delta P^2))$ as a function of K for $K \in [0.3, 0.7]$. What are the minimizing values of K for the three plots?