

# An Elementary Introduction to Kalman Filtering

Yan Pei

University of Texas at Austin  
ypei@cs.utexas.edu

Donald S. Fussell

University of Texas at Austin  
fussell@cs.utexas.edu

Swarnendu Biswas

Indian Institute of Technology Kanpur  
swarnendu@cse.iitk.ac.in

Keshav Pingali

University of Texas at Austin  
pingali@cs.utexas.edu

## ABSTRACT

Kalman filtering is a classic state estimation technique used in application areas such as signal processing and autonomous control of vehicles. It is now being used to solve problems in computer systems such as controlling the voltage and frequency of processors.

Although there are many presentations of Kalman filtering in the literature, they usually deal with particular systems like autonomous robots or linear systems with Gaussian noise, which makes it difficult to understand the general principles behind Kalman filtering. In this paper, we first present the abstract ideas behind Kalman filtering at a level accessible to anyone with a basic knowledge of probability theory and calculus, and then show how these concepts can be applied to the particular problem of state estimation in linear systems. This separation of concepts from applications should make it easier to understand Kalman filtering and to apply it to other problems in computer systems.

## KEYWORDS

Kalman filtering, data fusion, uncertainty, noise, state estimation, covariance, BLUE, linear systems

## 1 INTRODUCTION

Kalman filtering is a state estimation technique invented in 1960 by Rudolf E. Kálmán [16]. Because of its ability to extract useful information from noisy data and its small computational and memory requirements, it is used in many application areas including spacecraft navigation, motion planning in robotics, signal processing, and wireless sensor networks [12, 21, 28–30]. Recent work has used Kalman filtering in controllers for computer systems [5, 13, 14, 24].

Although many introductions to Kalman filtering are available in the literature [1–4, 6–11, 18, 22, 26, 30], they are usually focused on particular applications like robot motion or state estimation in linear systems. This can make it difficult to see how to apply Kalman filtering to other problems. Other presentations derive Kalman filtering as an application of Bayesian inference assuming that noise is Gaussian. This leads to the common misconception that Kalman filtering

can be applied only if noise is Gaussian [15]. The goal of this paper is to present the abstract concepts behind Kalman filtering in a way that is accessible to most computer scientists while clarifying the key assumptions, and then show how the problem of state estimation in linear systems can be solved as an application of these general concepts.

Abstractly, Kalman filtering can be seen as a particular approach to combining approximations of an unknown value to produce a better approximation. Suppose we use two devices of different designs to measure the temperature of a CPU core. Because devices are usually noisy, the measurements are likely to differ from the actual temperature of the core. Since the devices are of different designs, let us assume that noise affects the two devices in unrelated ways (this is formalized using the notion of correlation in Section 2). Therefore, the measurements  $x_1$  and  $x_2$  are likely to be different from each other and from the actual core temperature  $x_c$ . A natural question is the following: is there a way to combine the information in the noisy measurements  $x_1$  and  $x_2$  to obtain a good approximation of the actual temperature  $x_c$ ?

One *ad hoc* solution is to use the formula  $0.5 \cdot x_1 + 0.5 \cdot x_2$  to take the average of the two measurements, giving them equal weight. Formulas of this sort are called *linear estimators* because they use a weighted sum to fuse values; for our temperature problem, their general form is  $\beta \cdot x_1 + \alpha \cdot x_2$ . In this paper, we use the term *estimate* to refer to both a noisy measurement and to a value computed by an estimator, since both are approximations of unknown values of interest.

Suppose we have additional information about the two devices; say the second one uses more advanced temperature sensing. Since we would have more confidence in the second measurement, it seems reasonable that we should discard the first one, which is equivalent to using the linear estimator  $0.0 \cdot x_1 + 1.0 \cdot x_2$ . Kalman filtering tells us that in general, this intuitively reasonable linear estimator is not “optimal”; paradoxically, there is useful information even in the measurement from the lower-quality device, and the optimal estimator is one in which the weight given to each measurement is proportional to the confidence we have in the device producing that measurement. Only if we have no

confidence whatever in the first device should we discard its measurement.

Section 2 describes how these intuitive ideas can be quantified. Estimates are modeled as random samples from *distributions*, and confidence in estimates is quantified in terms of the *variances* and *covariances* of these distributions.<sup>1</sup> Sections 3-5 develop the two key ideas behind Kalman filtering.

- (1) How should estimates be fused optimally?

Section 3 shows how to fuse *scalar* estimates such as temperatures optimally. It is also shown that the problem of fusing more than two estimates can be reduced to the problem of fusing two estimates at a time without any loss in the quality of the final estimate. Section 4 extends these results to estimates that are *vectors*, such as state vectors representing the estimated position and velocity of a robot.

- (2) In some applications, estimates are vectors but only a part of the vector can be measured directly. For example, the state of a spacecraft may be represented by its position and velocity, but only its position may be observable. In such situations, how do we obtain a complete estimate from a partial estimate?

Section 5 shows how the *Best Linear Unbiased Estimator (BLUE)* can be used for this. Intuitively, it is assumed that there is a linear relationship between the observable and hidden parts of the state vector, and this relationship is used to compute an estimate for the hidden part of the state, given an estimate for the observable part.

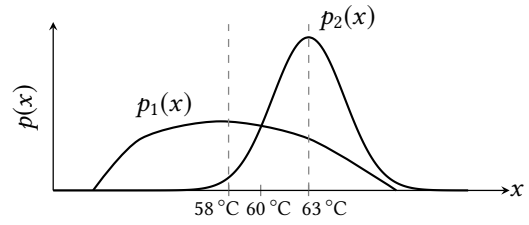
Section 6 uses these ideas to solve the state estimation problems for linear systems, which is the usual context for presenting Kalman filters. Section 7 briefly discusses extensions of Kalman filtering for nonlinear systems.

## 2 FORMALIZATION OF ESTIMATES

This section makes precise the notions of *estimates* and *confidence* in estimates.

### 2.1 Scalar estimates

To model the behavior of devices producing noisy measurements, we associate each device  $i$  with a *random variable* that has a *probability density function* (pdf)  $p_i(x)$  such as the ones shown in Figure 1 (the x-axis in this figure represents temperature). Random variables need not be Gaussian.<sup>2</sup> Obtaining a measurement from device  $i$  corresponds to drawing a random sample from the distribution for that device. We write  $x_i \sim p_i(\mu_i, \sigma_i^2)$  to denote that  $x_i$  is a random variable



**Figure 1: Using pdfs to model devices with systematic and random errors. Ground truth is 60 °C. Dashed lines are means of pdfs.**

with pdf  $p_i$  whose mean and variance are  $\mu_i$  and  $\sigma_i^2$  respectively; following convention, we use  $x_i$  to represent a random sample from this distribution as well.

Means and variances of distributions model different kinds of inaccuracies in measurements. Device  $i$  is said to have a *systematic error* or *bias* in its measurements if the mean  $\mu_i$  of its distribution is not equal to the actual temperature  $x_c$  (in general, to the value being estimated, which is known as *ground truth*); otherwise, the instrument is *unbiased*. Figure 1 shows pdfs for two devices that have different amounts of systematic error. The variance  $\sigma_i^2$  on the other hand is a measure of the *random error* in the measurements. The impact of random errors can be mitigated by taking many measurements with a given device and averaging their values, but this approach will not help to reduce systematic error.

In the usual formulation of Kalman filtering, it is assumed that measuring devices do not have systematic errors. However, we do not have the luxury of taking many measurements of a given state, so we must take into account the impact of random error on a single measurement. Therefore, confidence in a device is modeled formally by the variance of the distribution associated with that device; the smaller the variance, the higher our confidence in the measurements made by the device. In Figure 1, the fact that we have less confidence in the first device has been illustrated by making  $p_1$  more spread out than  $p_2$ , giving it a larger variance.

The informal notion that noise should affect the two devices in “unrelated ways” is formalized by requiring that the corresponding random variables be *uncorrelated*. This is a weaker condition than requiring them to be *independent*, as explained in the Appendix A. Suppose we are given the measurement made by one of the devices (say  $x_1$ ) and we have to guess what the other measurement ( $x_2$ ) might be. If knowing  $x_1$  does not give us any new information about what  $x_2$  might be, the random variables are independent. This is expressed formally by the equation  $p(x_2|x_1) = p(x_2)$ ; intuitively, knowing the value of  $x_1$  does not change the pdf for the possible values of  $x_2$ . If the random variables are

<sup>1</sup>Basic concepts including probability density function, mean, expectation, variance and covariance are introduced in Appendix A.

<sup>2</sup>The role of Gaussians in Kalman filtering is discussed in Section 6.5.

only uncorrelated, knowing  $x_1$  might give us new information about  $x_2$  such as restricting its possible values but the mean of  $x_2|x_1$  will still be  $\mu_2$ . Using expectations, this can be written as  $E[x_2|x_1] = E[x_2]$ , which is equivalent to requiring that  $E[(x_1 - \mu_1)(x_2 - \mu_2)]$ , the covariance between the two variables, be equal to zero. This is obviously a weaker condition than independence.

Although the discussion in this section has focused on measurements, the same formalization can be used for estimates produced by an estimator. Lemma 2.1(i) shows how the mean and variance of a linear combination of pairwise uncorrelated random variables can be computed from the means and variances of the random variables [19]. The mean and variance can be used to quantify bias and random errors for the estimator as in the case of measurements.

An *unbiased estimator* is one whose mean is equal to the unknown value being estimated and it is preferable to a biased estimator with the same variance. Only unbiased estimators are considered in this paper. Furthermore, an unbiased estimator with a smaller variance is preferable to one with a larger variance since we would have more confidence in the estimates it produces. As a step towards generalizing this discussion to estimators that produce vector estimates, we refer to the variance of an unbiased scalar estimator as the *Mean Square Error* of that estimator or *MSE* for short.

Lemma 2.1(ii) asserts that if a random variable is pairwise uncorrelated with a set of random variables, it is uncorrelated with any linear combination of those variables.

LEMMA 2.1. Let  $x_1 \sim p_1(\mu_1, \sigma_1^2), \dots, x_n \sim p_n(\mu_n, \sigma_n^2)$  be a set of pairwise uncorrelated random variables. Let  $y = \sum_{i=1}^n \alpha_i x_i$  be a random variable that is a linear combination of the  $x_i$ 's.

(i) The mean and variance of  $y$  are:

$$\mu_y = \sum_{i=1}^n \alpha_i \mu_i \quad (1)$$

$$\sigma_y^2 = \sum_{i=1}^n \alpha_i^2 \sigma_i^2 \quad (2)$$

(ii) If a random variable  $x_{n+1}$  is pairwise uncorrelated with  $x_1, \dots, x_n$ , it is uncorrelated with  $y$ .

## 2.2 Vector estimates

In some applications, estimates are vectors. For example, the state of a mobile robot might be represented by a vector containing its position and velocity. Similarly, the vital signs of a person might be represented by a vector containing his temperature, pulse rate and blood pressure. In this paper, we denote a vector by a boldfaced lowercase letter, and a matrix by an uppercase letter.

The covariance matrix  $\Sigma_{\mathbf{x}\mathbf{x}}$  of a random variable  $\mathbf{x}$  is the matrix  $E[(\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}})(\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}})^T]$ , where  $\boldsymbol{\mu}_{\mathbf{x}}$  is the mean of  $\mathbf{x}$ . Intuitively, entry  $(i,j)$  of this matrix is the covariance between the  $i$  and  $j$  components of vector  $\mathbf{x}$ ; in particular, entry  $(i,i)$  is the variance of the  $i^{th}$  component of  $\mathbf{x}$ . A random variable  $\mathbf{x}$  with a pdf  $p$  whose mean is  $\boldsymbol{\mu}_{\mathbf{x}}$  and covariance matrix is  $\Sigma_{\mathbf{x}\mathbf{x}}$  is written as  $\mathbf{x} \sim p(\boldsymbol{\mu}_{\mathbf{x}}, \Sigma_{\mathbf{x}\mathbf{x}})$ . The inverse of the covariance matrix ( $\Sigma_{\mathbf{x}\mathbf{x}}^{-1}$ ) is called the *precision* or *information* matrix.

*Uncorrelated random variables:* The cross-covariance matrix  $\Sigma_{\mathbf{v}\mathbf{w}}$  of two random variables  $\mathbf{v}$  and  $\mathbf{w}$  is the matrix  $E[(\mathbf{v} - \boldsymbol{\mu}_{\mathbf{v}})(\mathbf{w} - \boldsymbol{\mu}_{\mathbf{w}})^T]$ . Intuitively, element  $(i,j)$  of this matrix is the covariance between elements  $\mathbf{v}(i)$  and  $\mathbf{w}(j)$ . If the random variables are uncorrelated, all entries in this matrix are zero, which is equivalent to saying that every component of  $\mathbf{v}$  is uncorrelated with every component of  $\mathbf{w}$ . Lemma 2.2 generalizes Lemma 2.1.

LEMMA 2.2. Let  $\mathbf{x}_1 \sim p_1(\boldsymbol{\mu}_1, \Sigma_1), \dots, \mathbf{x}_n \sim p_n(\boldsymbol{\mu}_n, \Sigma_n)$  be a set of pairwise uncorrelated random variables of length  $m$ . Let  $\mathbf{y} = \sum_{i=1}^n A_i \mathbf{x}_i$ .

(i) The mean and covariance matrix of  $\mathbf{y}$  are the following:

$$\boldsymbol{\mu}_{\mathbf{y}} = \sum_{i=1}^n A_i \boldsymbol{\mu}_i \quad (3)$$

$$\Sigma_{\mathbf{y}\mathbf{y}} = \sum_{i=1}^n A_i \Sigma_i A_i^T \quad (4)$$

(ii) If a random variable  $\mathbf{x}_{n+1}$  is pairwise uncorrelated with  $\mathbf{x}_1, \dots, \mathbf{x}_n$ , it is uncorrelated with  $\mathbf{y}$ .

The MSE of an unbiased estimator  $\mathbf{y}$  is  $E[(\mathbf{y} - \boldsymbol{\mu}_{\mathbf{y}})^T(\mathbf{y} - \boldsymbol{\mu}_{\mathbf{y}})]$ , which is the sum of the variances of the components of  $\mathbf{y}$ ; if  $\mathbf{y}$  has length 1, this reduces to variance as expected. The MSE is also the sum of the diagonal elements of  $\Sigma_{\mathbf{y}\mathbf{y}}$  (this is called the *trace* of  $\Sigma_{\mathbf{y}\mathbf{y}}$ ).

## 3 FUSING SCALAR ESTIMATES

Section 3.1 discusses the problem of fusing two scalar estimates. Section 3.2 generalizes this to the problem of fusing  $n > 2$  scalar estimates. Section 3.3 shows that fusing  $n > 2$  estimates can be done iteratively by fusing two estimates at a time without any loss of quality in the final estimate.

### 3.1 Fusing two scalar estimates

We now consider the problem of choosing the optimal values of the parameters  $\alpha$  and  $\beta$  in the linear estimator  $\beta x_1 + \alpha x_2$  for fusing estimates  $x_1$  and  $x_2$  from uncorrelated random variables.

The first reasonable requirement is that if the two estimates  $x_1$  and  $x_2$  are equal, fusing them should produce the same value. This implies that  $\alpha + \beta = 1$ . Therefore the linear

estimators of interest are of the form

$$y_\alpha(x_1, x_2) = (1-\alpha)x_1 + \alpha x_2 \quad (5)$$

If  $x_1$  and  $x_2$  in Equation 5 are considered to be unbiased estimators of some quantity of interest, then  $y_\alpha$  is an unbiased estimator for any value of  $\alpha$ . How should optimality of such an estimator be defined? One reasonable definition is that the optimal value of  $\alpha$  *minimizes the variance of  $y_\alpha$*  since this will produce the highest-confidence fused estimates as discussed in Section 2. The variance (MSE) of  $y_\alpha$  can be determined from Lemma 2.1:

$$\sigma_y^2(\alpha) = (1-\alpha)^2 \sigma_1^2 + \alpha^2 \sigma_2^2 \quad (6)$$

**THEOREM 3.1.** *Let  $x_1 \sim p_1(\mu_1, \sigma_1^2)$  and  $x_2 \sim p_2(\mu_2, \sigma_2^2)$  be uncorrelated random variables. Consider the linear estimator  $y_\alpha(x_1, x_2) = (1-\alpha)x_1 + \alpha x_2$ . The variance of the estimator is minimized for  $\alpha = \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}$ .*

This result can be proved by setting the derivative of  $\sigma_y^2(\alpha)$  with respect to  $\alpha$  to zero and solving equation for  $\alpha$ .

PROOF.

$$\begin{aligned} \frac{d}{d\alpha} \sigma_y^2(\alpha) &= -2(1-\alpha) \sigma_1^2 + 2\alpha \sigma_2^2 \\ &= 2\alpha (\sigma_1^2 + \sigma_2^2) - 2 \sigma_1^2 = 0 \\ \alpha &= \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} \end{aligned} \quad (7)$$

The second order derivative of  $\sigma_y^2(\alpha)$ ,  $(\sigma_1^2 + \sigma_2^2)$ , is positive, showing that  $\sigma_y^2(\alpha)$  reaches a minimum at this point.  $\square$

In the literature, the optimal value of  $\alpha$  is called the *Kalman gain*  $K$ . Substituting  $K$  into the linear fusion model, we get the optimal linear estimator  $y(x_1, x_2)$ :

$$y(x_1, x_2) = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} x_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} x_2 \quad (8)$$

As a step towards fusion of  $n > 2$  estimates, it is useful to rewrite this as follows:

$$y(x_1, x_2) = \frac{\frac{1}{\sigma_1^2}}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} x_1 + \frac{\frac{1}{\sigma_2^2}}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} x_2 \quad (9)$$

Substituting the optimal value of  $\alpha$  into Equation 6, we get

$$\sigma_y^2 = \frac{1}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} \quad (10)$$

The expressions for  $y$  and  $\sigma_y^2$  are complicated because they contain the reciprocals of variances. If we let  $v_1$  and  $v_2$  denote

the precisions of the two distributions, the expressions for  $y$  and  $v_y$  can be written more simply as follows:

$$y(x_1, x_2) = \frac{v_1}{v_1 + v_2} x_1 + \frac{v_2}{v_1 + v_2} x_2 \quad (11)$$

$$v_y = v_1 + v_2 \quad (12)$$

These results say that the weight we should give to an estimate is proportional to the confidence we have in that estimate, and that we have more confidence in the fused estimate than in the individual estimates, which is intuitively reasonable. To use these results, we need only the variances of the distributions. In particular, the pdfs  $p_i$ , which are usually not available in applications, are not needed, and the proof of Theorem 3.1 does not require these pdf's to have the same mean.

### 3.2 Fusing multiple scalar estimates

The approach in Section 3.1 can be generalized to optimally fuse multiple pairwise uncorrelated estimates  $x_1, x_2, \dots, x_n$ . Let  $y_{n,\alpha}(x_1, \dots, x_n)$  denote the linear estimator for fusing the  $n$  estimates given parameters  $\alpha_1, \dots, \alpha_n$ , which we denote by  $\alpha$ . The notation  $y_\alpha(x_1, x_2)$  introduced in the previous section can be considered to be an abbreviation of  $y_{2,\alpha}(x_1, x_2)$ .

**THEOREM 3.2.** *Let  $x_i \sim p_i(\mu_i, \sigma_i^2)$  for  $(1 \leq i \leq n)$  be a set of pairwise uncorrelated random variables. Consider the linear estimator  $y_{n,\alpha}(x_1, \dots, x_n) = \sum_{i=1}^n \alpha_i x_i$  where  $\sum_{i=1}^n \alpha_i = 1$ . The variance of the estimator is minimized for*

$$\alpha_i = \frac{\frac{1}{\sigma_i^2}}{\sum_{j=1}^n \frac{1}{\sigma_j^2}}$$

PROOF. From Lemma 2.1,  $\sigma_y^2(\alpha) = \sum_{i=1}^n \alpha_i^2 \sigma_i^2$ . To find the values of  $\alpha_i$  that minimize the variance  $\sigma_y^2$  under the constraint that the  $\alpha_i$ 's sum to 1, we use the method of Lagrange multipliers. Define

$$f(\alpha_1, \dots, \alpha_n) = \sum_{i=1}^n \alpha_i^2 \sigma_i^2 + \lambda (\sum_{i=1}^n \alpha_i - 1)$$

where  $\lambda$  is the Lagrange multiplier. Taking the partial derivatives of  $f$  with respect to each  $\alpha_i$  and setting these derivatives to zero, we find  $\alpha_1 \sigma_1^2 = \alpha_2 \sigma_2^2 = \dots = \alpha_n \sigma_n^2 = -\lambda/2$ . From this, and the fact that sum of the  $\alpha_i$ 's is 1, the result follows.  $\square$

The minimal variance is given by the following expression:

$$\sigma_{y_n}^2 = \frac{1}{\sum_{j=1}^n \frac{1}{\sigma_j^2}} \quad (13)$$

As in Section 3.1, these expressions are more intuitive if the variance is replaced with precision: the contribution of  $x_i$



to the value of  $y_n(x_1, \dots, x_n)$  is proportional to  $x_i$ 's confidence.

$$y_n(x_1, \dots, x_n) = \sum_{i=1}^n \frac{v_i}{v_1 + \dots + v_n} * x_i \quad (14)$$

$$v_{y_n} = \sum_{i=1}^n v_i \quad (15)$$

Equations 14 and 15 generalize Equations 11 and 12.

### 3.3 Incremental fusing is optimal

In many applications, the estimates  $x_1, x_2, \dots, x_n$  become available successively over a period of time. While it is possible to store all the estimates and use Equations 14 and 15 to fuse all the estimates from scratch whenever a new estimate becomes available, it is possible to save both time and storage if one can do this fusion incrementally. We show that just as a sequence of numbers can be added by keeping a running sum and adding the numbers to this running sum one at a time, a sequence of  $n > 2$  estimates can be fused by keeping a "running estimate" and fusing estimates from the sequence one at a time into this running estimate without any loss in the quality of the final estimate. In short, we want to show that  $y_n(x_1, x_2, \dots, x_n) = y_2(y_2(y_2(x_1, x_2), x_3), \dots, x_n)$ .

A little bit of algebra shows that if  $n > 2$ , Equations 14 and 15 for the optimal linear estimator and its precision can be expressed as shown in Equations 16 and 17.

$$y_n(x_1, \dots, x_n) = \frac{v_{y_{n-1}}}{v_{y_{n-1}} + v_n} y_{n-1}(x_1, \dots, x_{n-1}) + \frac{v_n}{v_{y_{n-1}} + v_n} x_n \quad (16)$$

$$v_{y_n} = v_{y_{n-1}} + v_n \quad (17)$$

This shows that  $y_n(x_1, \dots, x_n) = y_2(y_{n-1}(x_1, \dots, x_{n-1}), x_n)$ . Using this argument recursively gives the required result.<sup>3</sup>

To make the connection to Kalman filtering, it is useful to derive the same result using a pictorial argument. Figure 2 shows the process of incrementally fusing the  $n$  estimates. In this picture, time progresses from left to right, the precision of each estimate is shown in parentheses next to it, and the weights on the edges are the weights from Equation 11. The contribution of each  $x_i$  to the final value  $y_2(y_2(y_2(x_1, x_2), x_3), \dots, x_n)$  is given by the product of the weights on the path from  $x_i$  to the final value and this product is obviously equal to the weight of  $x_i$  in Equation 14, showing that incremental fusion is optimal.

### 3.4 Summary

The results in this section can be summarized informally as follows. *When using a linear estimator to fuse uncertain scalar estimates, the weight given to each estimate should be*

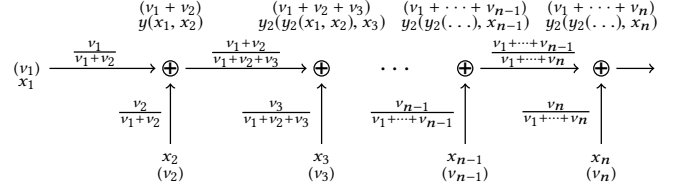


Figure 2: Dataflow graph for incremental fusion.

*inversely proportional to the variance of the random variable from which that estimate is obtained. Furthermore, when fusing  $n > 2$  estimates, estimates can be fused incrementally without any loss in the quality of the final result.* These results are often expressed formally in terms of the Kalman gain  $K$ , as shown below; the equations can be applied recursively to fuse multiple estimates. Note that if  $v_1 \gg v_2$ ,  $K \approx 0$  and  $y(x_1, x_2) \approx x_1$ ; conversely if  $v_1 \ll v_2$ ,  $K \approx 1$  and  $y(x_1, x_2) \approx x_2$ .

$$x_1 \sim p_1(\mu_1, \sigma_1^2), \quad x_2 \sim p_2(\mu_2, \sigma_2^2)$$

$$K = \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} = \frac{v_2}{v_1 + v_2} \quad (18)$$

$$y(x_1, x_2) = x_1 + K(x_2 - x_1) \quad (19)$$

$$\sigma_y^2 = (1 - K)\sigma_1^2 \quad \text{or} \quad v_y = v_1 + v_2 \quad (20)$$

## 4 FUSING VECTOR ESTIMATES

The results in Section 3 for fusing scalar estimates can be extended to vectors by replacing *variances* with *covariance matrices*.

### 4.1 Fusing multiple vector estimates

For vectors, the linear estimator is

$$\mathbf{y}_{n,A}(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n) = \sum_{i=1}^n A_i \mathbf{x}_i \quad \text{where} \quad \sum_{i=1}^n A_i = I \quad (21)$$

Here  $A$  stands for the matrix parameters  $(A_1, \dots, A_n)$ . All the vectors  $\mathbf{x}_i$  are assumed to be of the same length. To simplify notation, we omit the subscript  $n$  in  $\mathbf{y}_{n,A}$  in the discussion below since it is obvious from the context.

*Optimality:* The parameters  $A_1, \dots, A_n$  in the linear data fusion model are chosen to minimize  $MSE(\mathbf{y}_A)$  which is  $E[(\mathbf{y}_A - \boldsymbol{\mu}_{y_A})^T (\mathbf{y}_A - \boldsymbol{\mu}_{y_A})]$ , as explained in Section 2.

Theorem 4.1 generalizes Theorem 3.2 to the vector case. The proof of this theorem uses matrix derivatives [23] (see Appendix B) and is given in Appendix C since it is not needed for understanding the rest of this paper. Comparing Theorems 4.1 and 3.2, we see that the expressions are similar, the main difference being that the role of variance in the scalar case is played by the covariance matrix in the vector case.

<sup>3</sup>We thank Mani Chandy for showing us this approach to proving the result.

THEOREM 4.1. Let  $\mathbf{x}_i \sim p_i(\boldsymbol{\mu}_i, \Sigma_i)$  for  $(1 \leq i \leq n)$  be a set of pairwise uncorrelated random variables. Consider the linear estimator  $\mathbf{y}_A(\mathbf{x}_1, \dots, \mathbf{x}_n) = \sum_{i=1}^n A_i \mathbf{x}_i$ , where  $\sum_{i=1}^n A_i = I$ . The value of  $MSE(\mathbf{y}_A)$  is minimized for

$$A_i = \left( \sum_{j=1}^n \Sigma_j^{-1} \right)^{-1} \Sigma_i^{-1} \quad (22)$$

Therefore the optimal estimator is

$$\mathbf{y}(\mathbf{x}_1, \dots, \mathbf{x}_n) = \left( \sum_{j=1}^n \Sigma_j^{-1} \right)^{-1} \sum_{i=1}^n \Sigma_i^{-1} \mathbf{x}_i \quad (23)$$

The covariance matrix of  $\mathbf{y}$  can be computed by using Lemma 2.2.

$$\Sigma_{yy} = \left( \sum_{j=1}^n \Sigma_j^{-1} \right)^{-1} \quad (24)$$

In the vector case, precision is the inverse of a covariance matrix, denoted by  $N$ . Equations 25–26 use precision to express the optimal estimator and its variance and generalize Equations 14–15 to the vector case.

$$\mathbf{y}(\mathbf{x}_1, \dots, \mathbf{x}_n) = N_y^{-1} \sum_{i=1}^n N_i \mathbf{x}_i \quad (25)$$

$$N_y = \sum_{j=1}^n N_j \quad (26)$$

As in the scalar case, fusion of  $n > 2$  vector estimates can be done incrementally without loss of precision. The proof is similar to the scalar case, and is omitted.

There are several equivalent expressions for the Kalman gain for the fusion of two estimates. The following one, which is easily derived from Equation 22, is the vector analog of Equation 18:

$$K = \Sigma_1(\Sigma_1 + \Sigma_2)^{-1} \quad (27)$$

The covariance matrix of the optimal estimator  $\mathbf{y}(\mathbf{x}_1, \mathbf{x}_2)$  is the following.

$$\Sigma_{yy} = \Sigma_1(\Sigma_1 + \Sigma_2)^{-1} \Sigma_2 \quad (28)$$

$$= K \Sigma_2 = \Sigma_1 - K \Sigma_1 \quad (29)$$

## 4.2 Summary

The results in this section can be summarized in terms of the Kalman gain  $K$  as follows.

$$\mathbf{x}_1 \sim p_1(\boldsymbol{\mu}_1, \Sigma_1), \quad \mathbf{x}_2 \sim p_2(\boldsymbol{\mu}_2, \Sigma_2)$$

$$K = \Sigma_1(\Sigma_1 + \Sigma_2)^{-1} = (N_1 + N_2)^{-1} N_2 \quad (30)$$

$$\mathbf{y}(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1 + K(\mathbf{x}_2 - \mathbf{x}_1) \quad (31)$$

$$\Sigma_{yy} = (I - K)\Sigma_1 \quad \text{or} \quad N_y = N_1 + N_2 \quad (32)$$

## 5 BEST LINEAR UNBIASED ESTIMATOR (BLUE)

In some applications, the state of the system is represented by a vector but only part of the state can be measured directly, so it is necessary to estimate the hidden portion of the state corresponding to a measured value of the visible state. This section describes an estimator called the *Best Linear Unbiased Estimator* (BLUE) [17, 20, 27] for doing this.

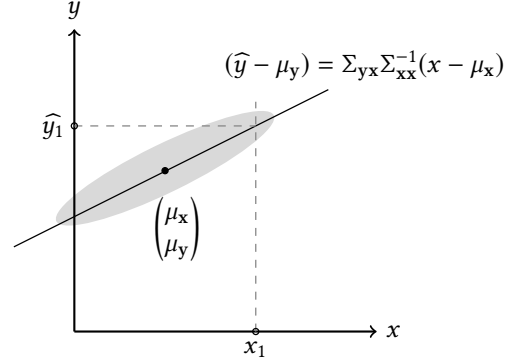


Figure 3: BLUE line corresponding to Equation 33.

Consider the general problem of determining a value for vector  $\mathbf{y}$  given a value for a vector  $\mathbf{x}$ . If there is a functional relationship between  $\mathbf{x}$  and  $\mathbf{y}$  (say  $\mathbf{y} = F(\mathbf{x})$  and  $F$  is given), it is easy to compute  $\mathbf{y}$  given a value for  $\mathbf{x}$  (say  $\mathbf{x}_1$ ).

In our context however,  $\mathbf{x}$  and  $\mathbf{y}$  are random variables, so such a precise functional relationship will not hold. Figure 3 shows an example in which  $x$  and  $y$  are scalar-valued random variables. The gray ellipse in this figure, called a *confidence ellipse*, is a projection of the joint distribution of  $x$  and  $y$  onto the  $(x, y)$  plane, that shows where some large proportion of the  $(x, y)$  values are likely to be. Suppose  $x$  takes the value  $x_1$ . Even within the confidence ellipse, there are many points  $(x_1, y)$ , so we cannot associate a single value of  $y$  with  $x_1$ . One possibility is to compute the mean of the  $y$  values associated with  $x_1$  (that is, the expectation  $E[y|x=x_1]$ ), and return this as the estimate for  $y$  if  $x=x_1$ . This requires knowing the joint distribution of  $x$  and  $y$ , which may not always be available.

In some problems, we can assume that there is an unknown linear relationship between  $\mathbf{x}$  and  $\mathbf{y}$  and that uncertainty comes from noise. Therefore, we can use a technique similar to the ordinary least squares (OLS) method to estimate this linear relationship, and use it to return the best estimate of  $y$  for any given value of  $x$ . In Figure 3, we see that although there are many points  $(x_1, y)$ , the  $y$  values are clustered around the line shown in the figure so the value  $\hat{y}_1$  is a reasonable estimate for the value of  $y$  corresponding to  $x_1$ . This line, called the *best linear unbiased estimator* (BLUE), is the analog of ordinary least squares (OLS) for distributions.

*Computing BLUE.* Consider the estimator  $\hat{\mathbf{y}}_{A,b}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$ . We choose  $\mathbf{A}$  and  $\mathbf{b}$  so that this is an unbiased estimator with minimal MSE. The “ $\hat{\cdot}$ ” over the  $\mathbf{y}$  is notation that indicates that we are computing an estimate for  $\mathbf{y}$ .

**THEOREM 5.1.** Let  $\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \sim p\left(\begin{pmatrix} \boldsymbol{\mu}_x \\ \boldsymbol{\mu}_y \end{pmatrix}, \begin{pmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{pmatrix}\right)$ . The estimator  $\hat{\mathbf{y}}_{A,b}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$  for estimating the value of  $\mathbf{y}$  for a given value of  $\mathbf{x}$  is an unbiased estimator with minimal MSE if

$$\begin{aligned} \mathbf{b} &= \boldsymbol{\mu}_y - \mathbf{A}(\boldsymbol{\mu}_x) \\ \mathbf{A} &= \Sigma_{yx}\Sigma_{xx}^{-1} \end{aligned}$$

The proof of Theorem 5.1 is straightforward. For an unbiased estimator,  $E[\hat{\mathbf{y}}] = E[\mathbf{y}]$ . This implies that  $\mathbf{b} = \boldsymbol{\mu}_y - \mathbf{A}(\boldsymbol{\mu}_x)$  so an unbiased estimator is of the form  $\hat{\mathbf{y}}_A(\mathbf{x}) = \boldsymbol{\mu}_y + \mathbf{A}(\mathbf{x} - \boldsymbol{\mu}_x)$ . Note that this is equivalent to asserting that the BLUE line must pass through the point  $(\boldsymbol{\mu}_x, \boldsymbol{\mu}_y)$ . Setting the derivative of  $MSE_A(\hat{\mathbf{y}}_A)$  with respect to  $\mathbf{A}$  to zero [23] and solving for  $\mathbf{A}$ , we find that the best linear unbiased estimator is

$$\hat{\mathbf{y}} = \boldsymbol{\mu}_y + \Sigma_{yx}\Sigma_{xx}^{-1}(\mathbf{x} - \boldsymbol{\mu}_x) \quad (33)$$

This equation can be understood intuitively as follows. If we have no information about  $\mathbf{x}$  and  $\mathbf{y}$ , the best we can do is the estimate  $(\boldsymbol{\mu}_x, \boldsymbol{\mu}_y)$ , which lies on the BLUE line. However, if we know that  $\mathbf{x}$  has a particular value  $\mathbf{x}_1$ , we can use the correlation between  $\mathbf{y}$  and  $\mathbf{x}$  to estimate a better value for  $\mathbf{y}$  from the difference  $(\mathbf{x}_1 - \boldsymbol{\mu}_x)$ . Note that if  $\Sigma_{yx} = 0$  (that is,  $\mathbf{x}$  and  $\mathbf{y}$  are uncorrelated), the best estimate of  $\mathbf{y}$  is just  $\boldsymbol{\mu}_y$ , so knowing the value of  $\mathbf{x}$  does not give us any additional information about  $\mathbf{y}$  as one would expect. In Figure 3, this corresponds to the case when the BLUE line is parallel to the  $x$ -axis. At the other extreme, suppose that  $\mathbf{y}$  and  $\mathbf{x}$  are functionally related so  $\mathbf{y} = \mathbf{C}\mathbf{x}$ . In that case, it is easy to see that  $\Sigma_{yx} = \mathbf{C}\Sigma_{xx}$ , so  $\hat{\mathbf{y}} = \mathbf{C}\mathbf{x}$  as expected. In Figure 3, this corresponds to the case when the confidence ellipse shrinks down to the BLUE line.

Equation 33 is a generalization of ordinary least squares in the sense that if we compute the relevant means and variances of a set of discrete data  $(x_i, y_i)$  and substitute into Equation 33, we get the same line that is obtained by using OLS.

## 6 KALMAN FILTERS FOR LINEAR SYSTEMS

In this section, we apply the algorithms developed in Sections 3-5 to the particular problem of state estimation in linear systems, which is the classical application of Kalman filtering.

Figure 4a shows how the evolution of the state of such a system over time can be computed if the initial state  $\mathbf{x}_0$  and the model of the system dynamics are known precisely. Time advances in discrete steps. The state of the system

at any time step is a function of the state of the system at the previous time step and the control inputs applied to the system during that interval. This is usually expressed by an equation of the form  $\mathbf{x}_t = f_t(\mathbf{x}_{t-1}, \mathbf{u}_t)$  where  $\mathbf{u}_t$  is the control input. The function  $f_t$  is nonlinear in the general case, and can be different for different steps. If the system is linear, the relation for state evolution over time can be written as  $\mathbf{x}_t = F_t\mathbf{x}_{t-1} + B_t\mathbf{u}_t$ , where  $F_t$  and  $B_t$  are time-dependent matrices that can be determined from the physics of the system. Therefore, if the initial state  $\mathbf{x}_0$  is known exactly and the system dynamics are modeled perfectly by the  $F_t$  and  $B_t$  matrices, the evolution of the state over time can be computed precisely as shown in Figure 4a.

In general however, we may not know the initial state exactly, and the system dynamics and control inputs may not be known precisely. These inaccuracies may cause the state computed by the model to diverge unacceptably from the actual state over time. To avoid this, we can make measurements of the state after each time step. If these measurements were exact, there would of course be no need to model the system dynamics. However, in general, the measurements themselves are imprecise.

Kalman filtering was invented to solve the problem of state estimation in such systems. Figure 4b shows the dataflow of the computation, and we use it to introduce standard terminology. An estimate of the initial state, denoted by  $\hat{\mathbf{x}}_{0|0}$ , is assumed to be available. At each time step  $t=1, 2, \dots$ , the system model is used to provide an estimate of the state at time  $t$  using information from time  $t-1$ . This step is called *prediction* and the estimate that it provides is called the *a priori* estimate and denoted by  $\hat{\mathbf{x}}_{t|t-1}$ . The *a priori* estimate is then fused with  $\mathbf{z}_t$ , the state estimate obtained from the measurement at time  $t$ , and the result is the *a posteriori* state estimate at time  $t$ , denoted by  $\hat{\mathbf{x}}_{t|t}$ . This *a posteriori* estimate is used by the model to produce the *a priori* estimate for the next time step and so on. As described below, the *a priori* and *a posteriori* estimates are the means of certain random variables; the covariance matrices of these random variables are shown within parentheses above each estimate in Figure 4b, and these are used to weight estimates when fusing them.

Section 6.1 presents the state evolution model and *a priori* state estimation. Section 6.2 discusses how state estimates are fused if an estimate of the entire state can be obtained by measurement; Section 6.3 addresses this problem when only a portion of the state can be measured directly.

### 6.1 State evolution model and prediction

The evolution of the state over time is described by a series of random variables  $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \dots$

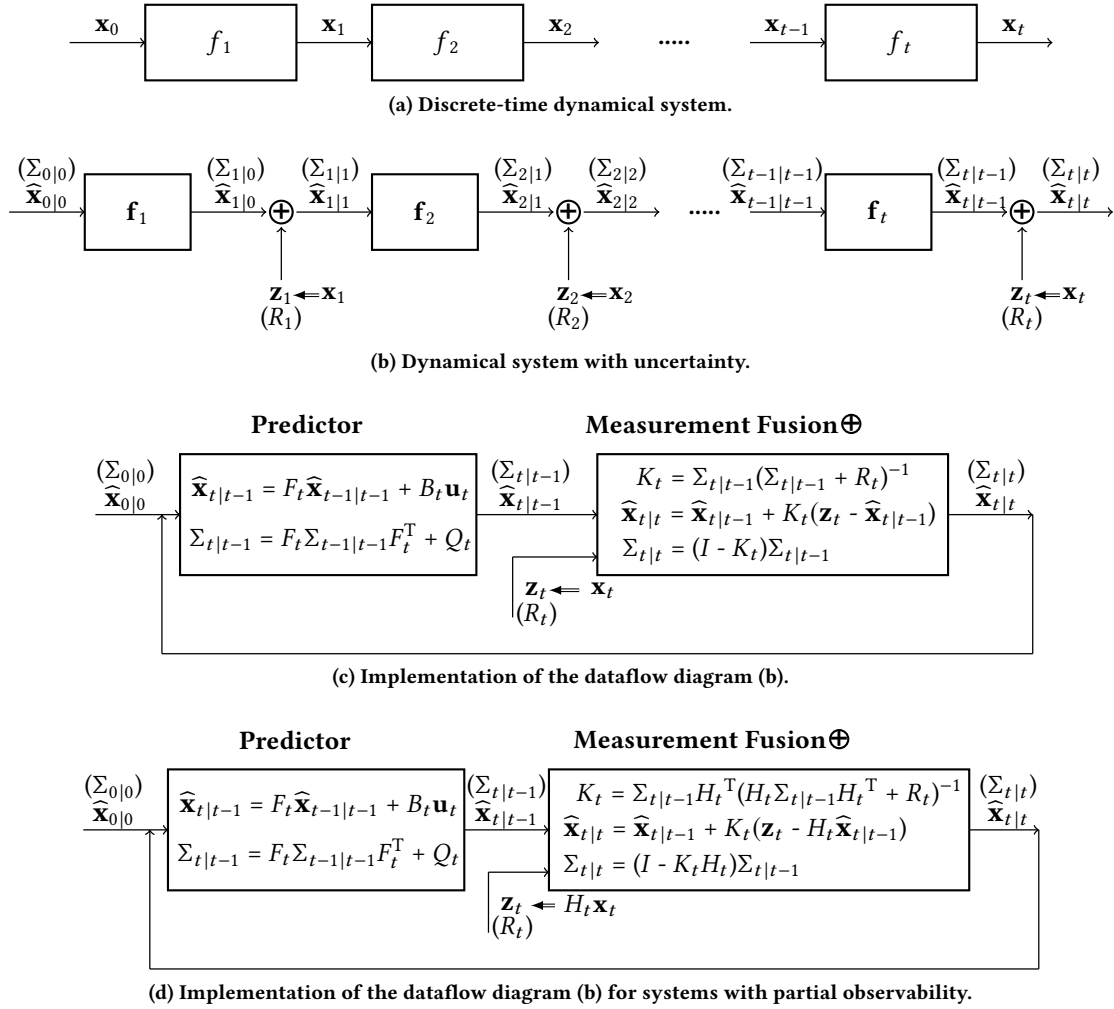


Figure 4: State estimation using Kalman filtering.

- The random variable  $\mathbf{x}_0$  captures the likelihood of different initial states.
- The random variables at successive time steps are related by the following linear model:

$$\mathbf{x}_t = F_t \mathbf{x}_{t-1} + B_t \mathbf{u}_t + \mathbf{w}_t \quad (34)$$

Here,  $\mathbf{u}_t$  is the control input, which is assumed to be deterministic, and  $\mathbf{w}_t$  is a zero-mean noise term that models all the uncertainty in the system. The covariance matrix of  $\mathbf{w}_t$  is denoted by  $Q_t$ , and the noise terms in different time steps are assumed to be uncorrelated to each other (that is,  $E[\mathbf{w}_i \mathbf{w}_j] = 0$  if  $i \neq j$ ) and to  $\mathbf{x}_0$ .

For estimation, we have a random variable  $\mathbf{x}_{0|0}$  that captures **our belief** about the likelihood of different states at time  $t=0$ , and two random variables  $\mathbf{x}_{t|t-1}$  and  $\mathbf{x}_{t|t}$  at each

time step  $t = 1, 2, \dots$  that capture our beliefs about the likelihood of different states at time  $t$  before and after fusion with the measurement respectively. The mean and covariance matrix of a random variable  $\mathbf{x}_{i|j}$  are denoted by  $\hat{\mathbf{x}}_{i|j}$  and  $\Sigma_{i|j}$  respectively. We assume  $E[\hat{\mathbf{x}}_{0|0}] = E[\mathbf{x}_0]$  (no bias).

Prediction essentially uses  $\mathbf{x}_{t-1|t-1}$  as a proxy for  $\mathbf{x}_{t-1}$  in Equation 34 to determine  $\mathbf{x}_{t|t-1}$  as shown in Equation 35.

$$\mathbf{x}_{t|t-1} = F_t \mathbf{x}_{t-1|t-1} + B_t \mathbf{u}_t + \mathbf{w}_t \quad (35)$$

For state estimation, we need only the mean and covariance matrix of  $\mathbf{x}_{t|t-1}$ . The Predictor box in Figure 4 computes these values; the covariance matrix is obtained from Lemma 2.2 under the assumption that  $\mathbf{w}_t$  is uncorrelated with  $\mathbf{x}_{t-1|t-1}$ , which is justified in Section 6.2.



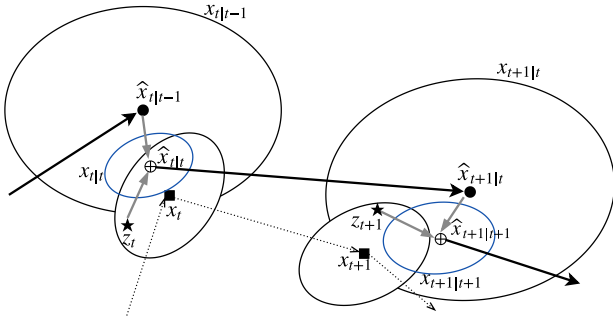


Figure 5: Pictorial representation of Kalman filtering.

## 6.2 Fusing complete observations of the state

If the entire state can be measured at each time step, the imprecise measurement at time  $t$  is modeled as follows:

$$\mathbf{z}_t = \mathbf{x}_t + \mathbf{v}_t \quad (36)$$

where  $\mathbf{v}_t$  is a zero-mean noise term with covariance matrix  $R_t$ . The noise terms in different time steps are assumed to be uncorrelated with each other (that is,  $E[\mathbf{v}_i \mathbf{v}_j^T]$  is zero if  $i \neq j$ ) as well as with  $\mathbf{x}_{0|0}$  and all  $\mathbf{w}_k$ . A subtle point here is that  $\mathbf{x}_t$  in this equation is the actual state of the system at time  $t$  (that is, a particular realization of the random variable  $\mathbf{x}_t$ ), so variability in  $\mathbf{z}_t$  comes only from  $\mathbf{v}_t$  and its covariance matrix  $R_t$ .

Therefore, we have two imprecise estimates for the state at each time step  $t = 1, 2, \dots$ , the *a priori* estimate from the predictor ( $\hat{\mathbf{x}}_{t|t-1}$ ) and the one from the measurement ( $\mathbf{z}_t$ ). If  $\mathbf{z}_t$  is uncorrelated to  $\mathbf{x}_{t|t-1}$ , we can use Equations 30-32 to fuse the estimates as shown in Figure 4c.

The assumptions that (i)  $\mathbf{x}_{t-1|t-1}$  is uncorrelated with  $\mathbf{w}_t$ , which is used in prediction, and (ii)  $\mathbf{x}_{t|t-1}$  is uncorrelated with  $\mathbf{z}_t$ , which is used in fusion, are easily proved to be correct by induction on  $t$ , using Lemma 2.2(ii). Figure 4b gives the intuition:  $\mathbf{x}_{t|t-1}$  for example is an affine function of the random variables  $\mathbf{x}_{0|0}, \mathbf{w}_1, \mathbf{v}_1, \mathbf{w}_2, \mathbf{v}_2, \dots, \mathbf{w}_t$ , and is therefore uncorrelated with  $\mathbf{v}_t$  (by assumption about  $\mathbf{v}_t$  and Lemma 2.2(ii)) and hence with  $\mathbf{z}_t$ .

Figure 5 shows the computation pictorially using confidence ellipses to illustrate uncertainty. The dotted arrows at the bottom of the figure show the evolution of the state, and the solid arrows show the computation of the *a priori* estimates and their fusion with measurements.

## 6.3 Fusing partial observations of the state

In some problems, only a portion of the state can be measured directly. The observable portion of the state is specified formally using a full row-rank matrix  $H_t$  called the *observation matrix*: if the state is  $\mathbf{x}$ , what is observable is  $H_t \mathbf{x}$ . For

example, if the state vector has two components and only the first component is observable,  $H_t$  can be  $\begin{bmatrix} 1 & 0 \end{bmatrix}$ . In general, the  $H_t$  matrix can specify a linear relationship between the state and the observation, and it can be time-dependent. The imprecise measurement model introduced in Equation 36 becomes:

$$\mathbf{z}_t = H_t \mathbf{x}_t + \mathbf{v}_t \quad (37)$$

The hidden portion of the state can be specified using a matrix  $C_t$ , which is an orthogonal complement of  $H_t$ . For example, if  $H_t = \begin{bmatrix} 1 & 0 \end{bmatrix}$ , one choice for  $C_t$  is  $\begin{bmatrix} 0 & 1 \end{bmatrix}$ .

Figure 4d shows the computation for this case. The fusion phase can be understood intuitively in terms of the following steps.

- (i) The observable part of the *a priori* estimate of the state  $H_t \hat{\mathbf{x}}_{t|t-1}$  is fused with the measurement  $\mathbf{z}_t$ , using the techniques developed in Sections 3-4. The quantity  $\mathbf{z}_t - H_t \hat{\mathbf{x}}_{t|t-1}$  is called the *innovation*. The result is the *a posteriori* estimate of the observable state  $H_t \hat{\mathbf{x}}_{t|t}$ .
- (ii) The BLUE in Section 5 is used to obtain the *a posteriori* estimate of the hidden state  $C_t \hat{\mathbf{x}}_{t|t}$  by adding to the *a priori* estimate of the hidden state  $C_t \hat{\mathbf{x}}_{t|t-1}$  a value obtained from the product of the covariance between  $H_t \mathbf{x}_{t|t-1}$  and  $C_t \mathbf{x}_{t|t-1}$  and the difference between  $H_t \hat{\mathbf{x}}_{t|t-1}$  and  $H_t \hat{\mathbf{x}}_{t|t}$ .
- (iii) The *a posteriori* estimates of the observable and hidden portions of the state are composed to produce the *a posteriori* estimate of the entire state  $\hat{\mathbf{x}}_{t|t}$ .

The actual implementation produces the final result directly without going through these steps as shown in Figure 4d, but these incremental steps are useful for understanding how all this works, and they are implemented as follows.

- (i) The *a priori* estimate of the observable part of the state is  $H_t \hat{\mathbf{x}}_{t|t-1}$  and the covariance is  $H_t \Sigma_{t|t-1} H_t^T$ . The *a posteriori* estimate is obtained directly from Equation 31:

$$H_t \hat{\mathbf{x}}_{t|t} = H_t \hat{\mathbf{x}}_{t|t-1} + H_t \Sigma_{t|t-1} H_t^T (H_t \Sigma_{t|t-1} H_t^T + R_t)^{-1} (\mathbf{z}_t - H_t \hat{\mathbf{x}}_{t|t-1})$$

Let  $K_t = \Sigma_{t|t-1} H_t^T (H_t \Sigma_{t|t-1} H_t^T + R_t)^{-1}$ . The *a posteriori* estimate of the observable state can be written in terms of  $K_t$  as follows:

$$H_t \hat{\mathbf{x}}_{t|t} = H_t \hat{\mathbf{x}}_{t|t-1} + H_t K_t (\mathbf{z}_t - H_t \hat{\mathbf{x}}_{t|t-1}) \quad (38)$$

- (ii) The *a priori* estimate of the hidden state is  $C_t \hat{\mathbf{x}}_{t|t-1}$ . The covariance between the hidden portion  $C_t \mathbf{x}_{t|t-1}$  and the observable portion  $H_t \mathbf{x}_{t|t-1}$  is  $C_t \Sigma_{t|t-1} H_t^T$ . The difference between the *a priori* estimate and *a posteriori* estimate of  $H_t \mathbf{x}$  is  $H_t K_t (\mathbf{z}_t - H_t \hat{\mathbf{x}}_{t|t-1})$ . Therefore the *a posteriori* estimate of the hidden portion of the state is

obtained directly from Equation 33:

$$\begin{aligned} C_t \widehat{\mathbf{x}}_{t|t} &= C_t \widehat{\mathbf{x}}_{t|t-1} \\ &+ (C_t \Sigma_{t|t-1} H_t^T) (H_t \Sigma_{t|t-1} H_t^T)^{-1} H_t K_t (\mathbf{z}_t - H_t \widehat{\mathbf{x}}_{t|t-1}) \\ &= C_t \widehat{\mathbf{x}}_{t|t-1} + C_t K_t (\mathbf{z}_t - H_t \widehat{\mathbf{x}}_{t|t-1}) \end{aligned} \quad (39)$$

(iii) Putting the *a posteriori* estimates (38) and (39) together,

$$\begin{pmatrix} H_t \\ C_t \end{pmatrix} \widehat{\mathbf{x}}_{t|t} = \begin{pmatrix} H_t \\ C_t \end{pmatrix} \widehat{\mathbf{x}}_{t|t-1} + \begin{pmatrix} H_t \\ C_t \end{pmatrix} K_t (\mathbf{z}_t - H_t \widehat{\mathbf{x}}_{t|t-1})$$

Since  $\begin{pmatrix} H_t \\ C_t \end{pmatrix}$  is invertible, it can be canceled from the left and right hand sides, giving the equation

$$\widehat{\mathbf{x}}_{t|t} = \widehat{\mathbf{x}}_{t|t-1} + K_t (\mathbf{z}_t - H_t \widehat{\mathbf{x}}_{t|t-1}) \quad (40)$$

To compute  $\Sigma_{t|t}$ , Equation 40 can be rewritten as  $\widehat{\mathbf{x}}_{t|t} = (I - K_t H_t) \widehat{\mathbf{x}}_{t|t-1} + K_t \mathbf{z}_t$ . Since  $\mathbf{x}_{t|t-1}$  and  $\mathbf{z}_t$  are uncorrelated, it follows from Lemma 2.2 that

$$\Sigma_{t|t} = (I - K_t H_t) \Sigma_{t|t-1} (I - K_t H_t)^T + K_t R_t K_t^T$$

Substituting the value of  $K_t$  and simplifying, we get

$$\Sigma_{t|t} = (I - K_t H_t) \Sigma_{t|t-1} \quad (41)$$

Figure 4d puts all this together. In the literature, this dataflow is referred to as Kalman filtering. Unlike in Sections 3 and 4, the Kalman gain is not a dimensionless value here. If  $H_t = I$ , the computations in Figure 4d reduce to those of Figure 4c as expected.

Equation 40 shows that the *a posteriori* state estimate is a linear combination of the *a priori* state estimate ( $\widehat{\mathbf{x}}_{t|t-1}$ ) and the measurement ( $\mathbf{z}_t$ ). The optimality of this linear unbiased estimator is shown in the Appendix D. In Section 3.3, it was shown that incremental fusion of scalar estimates is optimal. The dataflow of Figures 4(c,d) computes the *a posteriori* state estimate at time  $t$  by incrementally fusing measurements from the previous time steps, and this incremental fusion can be shown to be optimal using a similar argument.

## 6.4 Example: falling body

To demonstrate the effectiveness of the Kalman filter, we consider an example in which an object falls from the origin at time  $t=0$  with an initial speed of 0 m/s and an expected constant acceleration of 9.8 m/s<sup>2</sup> due to gravity. Note that acceleration in reality may not be constant due to factors such as wind, air friction, and so on.

The state vector of the object contains two components, one for the distance from the origin  $s(t)$  and one for the velocity  $v(t)$ . We assume that only the velocity state can be measured at each time step. If time is discretized in steps of 0.25 seconds, the difference equation for the dynamics of the

system is easily shown to be the following:

$$\begin{pmatrix} v_n \\ s_n \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0.25 & 1 \end{pmatrix} \begin{pmatrix} v_{n-1} \\ s_{n-1} \end{pmatrix} + \begin{pmatrix} 0 & 0.25 \\ 0 & 0.5 \times 0.25^2 \end{pmatrix} \begin{pmatrix} 0 \\ 9.8 \end{pmatrix} \quad (42)$$

where we assume  $\begin{pmatrix} v_0 \\ s_0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$  and  $\Sigma_0 = \begin{pmatrix} 80 & 0 \\ 0 & 10 \end{pmatrix}$ .

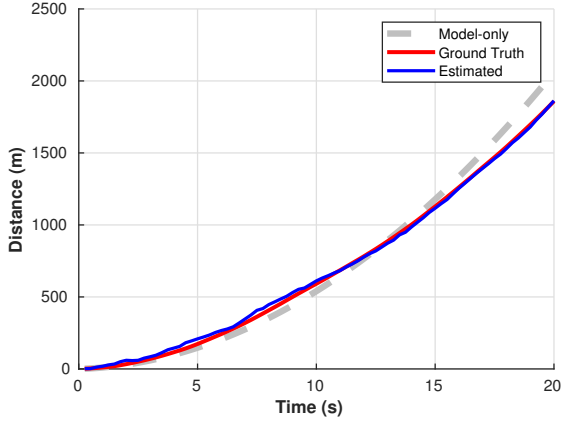
The gray lines in Figure 6 show the evolution of velocity and distance with time according to this model. Because of uncertainty in modeling the system dynamics, the actual evolution of the velocity and position will be different in practice. The red lines in Figure 6 show one trajectory for this evolution, corresponding to a Gaussian noise term with covariance  $\begin{pmatrix} 2 & 2.5 \\ 2.5 & 4 \end{pmatrix}$  in Equation 34 (because this noise term is random, there are many trajectories for the evolution, and we are just showing one of them). The red lines correspond to “ground truth” in our example.

The green points in Figure 6b show the noisy measurements of velocity at different time steps, assuming the noise is modeled by a Gaussian with variance 8. The blue lines show the *a posteriori* estimates of the velocity and position. It can be seen that the *a posteriori* estimates track the ground truth quite well even when the ideal system model (the gray lines) is inaccurate and the measurements are noisy. The cyan bars in the right figure show the variance of the velocity at different time steps. Although the initial variance is quite large, application of Kalman filtering is able to reduce it rapidly in few time steps.

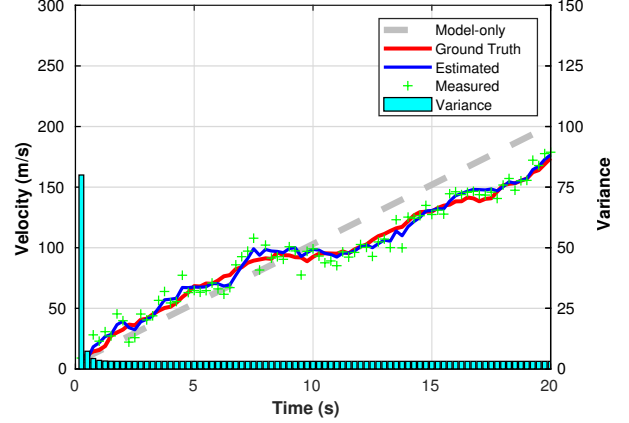
## 6.5 Discussion

This section shows that Kalman filtering for state estimation in linear systems can be derived from two elementary ideas: optimal linear estimators for fusing uncorrelated estimates and best linear unbiased estimators for correlated variables. This is a different approach to the subject than the standard presentations in the literature. One standard approach is to use Bayesian inference. The other approach is to assume that the *a posteriori* state estimator is a linear combination of the form  $A_t \widehat{\mathbf{x}}_{t|t-1} + B_t \mathbf{z}_t$ , and then find the values of  $A_t$  and  $B_t$  that produce an unbiased estimator with minimum *MSE*. We believe that the advantage of the presentation given here is that it exposes the concepts and assumptions that underlie Kalman filtering.

Most presentations in the literature also begin by assuming that the noise terms  $\mathbf{w}_t$  in the state evolution equation and  $\mathbf{v}_t$  in the measurement are Gaussian. Some presentations [1, 10] use properties of Gaussians to derive the results in Sections 3 although as we have seen, these results do not depend on distributions being Gaussians. Gaussians however enter the picture in a deeper way if one considers *nonlinear* estimators. It can be shown that if the noise terms are not Gaussian, there may be nonlinear estimators whose *MSE* is lower than that



(a) Evolution of state: Distance



(b) Evolution of state: Velocity

Figure 6: Estimates of the object's state over time.

of the linear estimator presented in Figure 4d. However if the noise is Gaussian, this linear estimator is as good as any unbiased nonlinear estimator (that is, the linear estimator is a *minimum variance unbiased estimator* (MVUE)). This result is proved using the Cramer-Rao lower bound [25].

## 7 EXTENSION TO NONLINEAR SYSTEMS

The *Extended Kalman Filter* (EKF) and *Unscented Kalman Filter* (UKF) are heuristic approaches to using Kalman filtering for nonlinear systems. The state evolution and measurement equations for nonlinear systems with additive noise can be written as follows; in these equations,  $f$  and  $h$  are nonlinear functions.

$$\mathbf{x}_t = f(\mathbf{x}_{t-1}, \mathbf{u}_t) + \mathbf{w}_t \quad (43)$$

$$\mathbf{z}_t = h(\mathbf{x}_t) + \mathbf{v}_t \quad (44)$$

Intuitively, the EKF constructs linear approximations to the nonlinear functions  $f$  and  $h$  and applies the Kalman filter equations, while the UKF constructs approximations to probability distributions and propagates these through the nonlinear functions to construct approximations to the posterior distributions.

*EKF.* Examining Figure 4d, we see that the *a priori* state estimate in the predictor can be computed using the system model:  $\hat{\mathbf{x}}_{t|t-1} = f(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t)$ . However, since the system dynamics and measurement equations are nonlinear, it is not clear how to compute the covariance matrices for the *a priori* estimate and the measurement. In the EKF, these matrices are computed by linearizing Equations 43 and 44 using the Taylor series expansions for the nonlinear functions  $f$  and

$h$ . This requires computing the following *Jacobians*<sup>4</sup>, which play the role of  $F_t$  and  $H_t$  in Figure 4d.

$$F_t = \left. \frac{\partial f}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t} \quad H_t = \left. \frac{\partial h}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_{t|t-1}}$$

The EKF performs well in some applications such as navigation systems and GPS [29].

*UKF.* When the system dynamics and observation models are highly nonlinear, the Unscented Kalman Filter (UKF) [15] can be an improvement over the EKF. The UKF is based on the *unscented transformation*, which is a method for computing the statistics of a random variable  $\mathbf{x}$  that undergoes a nonlinear transformation ( $\mathbf{y} = g(\mathbf{x})$ ). The random variable  $\mathbf{x}$  is sampled using a carefully chosen set of *sigma points* and these sample points are propagated through the nonlinear function  $g$ . The statistics of  $\mathbf{y}$  are estimated using a weighted sample mean and covariance of the posterior sigma points. The UKF tends to be more robust and accurate than the EKF but has higher computation overhead due to the sampling process.

## 8 CONCLUSION

In this paper, we have shown that two concepts - optimal linear estimators for fusing uncorrelated estimates and best linear unbiased estimators for correlated variables - provide the underpinnings for Kalman filtering. By combining these ideas, standard results on Kalman filtering for linear systems can be derived in an intuitive and straightforward way that is simpler than other presentations of this material in the

<sup>4</sup>The Jacobian matrix is the matrix of all first order partial derivatives of a vector-valued function.

literature. This approach makes clear the assumptions that underlie the optimality results associated with Kalman filtering, and should make it easier to apply Kalman filtering to problems in computer systems.

## ACKNOWLEDGMENTS

This research was supported by NSF grants 1337281, 1406355, and 1618425, and by DARPA contracts FA8750-16-2-0004 and FA8650-15-C-7563. The authors would like to thank K. Mani Chandy (Caltech), Ivo Babuska (UT Austin) and Augusto Ferrante (Padova) for their feedback on this paper.

## REFERENCES

- [1] Tim Babb. 2018. How a Kalman filter works, in pictures | Bzarg. <https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/>. (2018). Accessed: 2018-11-30.
- [2] A. V. Balakrishnan. 1987. *Kalman Filtering Theory*. Optimization Software, Inc., Los Angeles, CA, USA.
- [3] Allen L. Barker, Donald E. Brown, and Worthy N. Martin. 1994. *Bayesian Estimation and the Kalman Filter*. Technical Report. Charlottesville, VA, USA.
- [4] Alex Becker. 2018. Kalman Filter Overview. <https://www.kalmanfilter.net/default.aspx>. (2018). Accessed: 2018-11-08.
- [5] K. Bergman. 2009. Nanophotonic Interconnection Networks in Multi-core Embedded Computing. In *2009 IEEE/LEOS Winter Topicals Meeting Series*. 6–7. <https://doi.org/10.1109/LEOSWT.2009.4771628>
- [6] Liyu Cao and Howard M. Schwartz. 2004. Analysis of the Kalman Filter Based Estimation Algorithm: An Orthogonal Decomposition Approach. *Automatica* 40, 1 (Jan. 2004), 5–19. <https://doi.org/10.1016/j.automatica.2003.07.011>
- [7] Charles K. Chui and Guanrong Chen. 2017. *Kalman Filtering: With Real-Time Applications* (5th ed.). Springer Publishing Company, Incorporated.
- [8] R.L. Eubank. 2005. *A Kalman Filter Primer (Statistics: Textbooks and Monographs)*. Chapman & Hall/CRC.
- [9] Geir Evensen. 2006. *Data Assimilation: The Ensemble Kalman Filter*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.
- [10] Rodney Faragher. 2012. Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation. *IEEE Signal Processing Magazine* 29, 5 (September 2012), 128–132.
- [11] Mohinder S. Grewal and Angus P. Andrews. 2014. *Kalman Filtering: Theory and Practice with MATLAB* (4th ed.). Wiley-IEEE Press.
- [12] Anne-Kathrin Hess and Anders Rantzer. 2010. Distributed Kalman Filter Algorithms for Self-localization of Mobile Devices. In *Proceedings of the 13th ACM International Conference on Hybrid Systems: Computation and Control (HSCC '10)*. ACM, New York, NY, USA, 191–200. <https://doi.org/10.1145/1755952.1755980>
- [13] Connor Imes and Henry Hoffmann. 2016. Bard: A Unified Framework for Managing Soft Timing and Power Constraints. In *International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation (SAMOS)*.
- [14] C. Imes, D. H. K. Kim, M. Maggio, and H. Hoffmann. 2015. POET: A Portable Approach to Minimizing Energy Under Soft Real-time Constraints. In *21st IEEE Real-Time and Embedded Technology and Applications Symposium*. 75–86. <https://doi.org/10.1109/RTAS.2015.7108419>
- [15] Simon J. Julier and Jeffrey K. Uhlmann. 2004. Unscented Filtering and Nonlinear Estimation. *Proc. IEEE* 92, 3 (2004), 401–422. <https://doi.org/10.1109/JPROC.2003.823141>
- [16] Rudolph Emil Kalman. 1960. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME—Journal of Basic Engineering* 82, Series D (1960), 35–45.
- [17] Peter K Kitanidis. 1987. Unbiased Minimum-variance Linear State Estimation. *Automatica* 23, 6 (Nov. 1987), 775–778. [https://doi.org/10.1016/0005-1098\(87\)90037-9](https://doi.org/10.1016/0005-1098(87)90037-9)
- [18] Anders Lindquist and Giorgio Picci. 2017. *Linear Stochastic Systems*. Springer-Verlag.
- [19] Peter S Maybeck. 1982. *Stochastic models, estimation, and control*. Vol. 3. Academic press.
- [20] Jerry M Mendel. 1995. *Lessons in Estimation Theory for Signal Processing, Communications, and Control*. Pearson Education.
- [21] Kaushik Nagarajan, Nicholas Gans, and Roozbeh Jafari. 2011. Modeling Human Gait Using a Kalman Filter to Measure Walking Distance. In *Proceedings of the 2nd Conference on Wireless Health (WH '11)*. ACM, New York, NY, USA, Article 34, 2 pages. <https://doi.org/10.1145/2077546.2077584>
- [22] Eduardo F. Nakamura, Antonio A. F. Loureiro, and Alejandro C. Frery. 2007. Information Fusion for Wireless Sensor Networks: Methods, Models, and Classifications. *ACM Comput. Surv.* 39, 3, Article 9 (Sept. 2007). <https://doi.org/10.1145/1267070.1267073>
- [23] Kaare Brandt Petersen and Michael Syskind Pedersen. 2012. The Matrix Cookbook. [http://www2.imm.dtu.dk/pubdb/views/publication\\_details.php?id=3274](http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=3274). (Nov. 2012). Version 20121115.
- [24] Raghavendra Pradyumna Pothukuchi, Amin Ansari, Petros Voulgaris, and Josep Torrellas. 2016. Using Multiple Input, Multiple Output Formal Control to Maximize Resource Efficiency in Architectures. In *Computer Architecture (ISCA), 2016 ACM/IEEE 43rd Annual International Symposium on*. IEEE, 658–670.
- [25] C.R. Rao. 1945. Information and the Accuracy Attainable in the Estimation of Statistical Parameters. *Bulletin of the Calcutta Mathematical Society* 37 (1945), 81–89.
- [26] Matthew B. Rhudy, Roger A. Salguero, and Keaton Holappa. 2017. A Kalman Filtering Tutorial for Undergraduate Students. *International Journal of Computer Science & Engineering Survey* 8, 1 (Feb. 2017).
- [27] Sailes K. Sengupta. 1995. Fundamentals of Statistical Signal Processing: Estimation Theory. *Technometrics* 37, 4 (1995), 465–466. <https://doi.org/10.1080/00401706.1995.10484391>
- [28] Éfren L. Souza, Eduardo F. Nakamura, and Richard W. Pazzi. 2016. Target Tracking for Sensor Networks: A Survey. *ACM Comput. Surv.* 49, 2, Article 30 (June 2016), 31 pages. <https://doi.org/10.1145/2938639>
- [29] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. 2005. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press.
- [30] Greg Welch and Gary Bishop. 1995. *An Introduction to the Kalman Filter*. Technical Report. Chapel Hill, NC, USA.
- [31] Yan Pei, Swarnendu Biswas, Donald S. Fussell, and Keshav Pingali. 2017. An Elementary Introduction to Kalman Filtering. *ArXiv e-prints* (Oct. 2017). arXiv:1710.04055



# Appendices

## A BASIC PROBABILITY THEORY AND STATISTICS TERMINOLOGY

*Probability density function.* For a continuous random variable  $x$ , a *probability density function* (pdf) is a function  $p(x)$  whose value provides a relative likelihood that the value of the random variable will equal  $x$ . The integral of the pdf within a range of values is the probability that the random variable will take a value within that range.

If  $g(x)$  is a function of  $x$  with pdf  $p(x)$ , the *expected value* or *expectation* of  $g(x)$  is  $E[g(x)]$ , defined as the following integral:

$$E[g(x)] = \int_{-\infty}^{\infty} g(x)p(x)dx$$

By definition, the *mean*  $\mu_x$  of a random variable  $x$  is  $E[x]$ . The *variance* of a random variable  $x$  measures the variability of the distribution. For the set of possible values of  $x$ , variance (denoted by  $\sigma_x^2$ ) is defined by  $\sigma_x^2 = E[(x - \mu_x)^2]$ . The variance of a continuous random variable  $x$  can be written as the following integral:

$$\sigma_x^2 = \int_{-\infty}^{\infty} (x - \mu)^2 p(x)dx$$

If  $x$  is discrete and all outcomes are equally likely, then  $\sigma_x^2 = \frac{\sum (x_i - \mu_x)^2}{n}$ . The standard deviation  $\sigma_x$  is the square root of the variance.

*Covariance.* The *covariance* of two random variables is a measure of their joint variability. The covariance between random variables  $x_1 : p_1 \sim (\mu_1, \sigma_1^2)$  and  $x_2 : p_2 \sim (\mu_2, \sigma_2^2)$  is the expectation  $E[(x_1 - \mu_1)(x_2 - \mu_2)]$ . Two random variables are *uncorrelated* or *not correlated* if their covariance is zero. This is not the same concept as *independence* of random variables.

Two random variables are independent if knowing the value of one of the variables does not give us any information about the possible values of the other one. This is written formally as  $p(x_1|x_2) = p(x_1)$ ; intuitively, knowing the value of  $x_2$  does not change the probability that  $p_1$  takes a particular value.

Independent random variables are uncorrelated but random variables can be uncorrelated even if they are not independent. It can be shown that if  $x_1$  and  $x_2$  are not correlated,  $E[x_1|x_2] = E[x_1]$ ; intuitively, knowing the value of  $x_2$  may change the probability that  $x_1$  takes a particular value, but the mean of the resulting distribution remains the same as the mean of  $x_1$ . A special case of this that is easy to understand are examples in which knowing  $x_2$  restricts the possible values of  $x_1$  without changing the mean. Consider a random variable  $u : U$  that is uniformly distributed over the unit circle, and consider random variables  $x_1 : [-1, 1]$  and

$x_2 : [-1, 1]$  that are the projections of  $u$  on the  $x$  and  $y$  axes respectively. Given a value for  $x_2$ , there are only two possible values for  $x_1$ , so  $x_1$  and  $x_2$  are not independent. However, the mean of these values is 0, which is the mean of  $x_1$ , so  $x_1$  and  $x_2$  are not correlated.

## B MATRIX DERIVATIVES

If  $f(X)$  is a scalar function of a matrix  $X$ , the matrix derivative  $\frac{\partial f(X)}{\partial X}$  is defined as the matrix

$$\begin{pmatrix} \frac{\partial f(X)}{\partial X(1,1)} & \cdots & \frac{\partial f(X)}{\partial X(1,n)} \\ \vdots & \ddots & \vdots \\ \frac{\partial f(X)}{\partial X(n,1)} & \cdots & \frac{\partial f(X)}{\partial X(n,n)} \end{pmatrix}$$

LEMMA B.1. Let  $X$  be a  $m \times n$  matrix,  $\mathbf{a}$  be a  $m \times 1$  vector,  $\mathbf{b}$  be a  $n \times 1$  vector.

$$\frac{\partial \mathbf{a}^T X \mathbf{b}}{\partial X} = \mathbf{a} \mathbf{b}^T \quad (45)$$

$$\frac{\partial (\mathbf{a}^T X^T X \mathbf{b})}{\partial X} = X(\mathbf{a} \mathbf{b}^T + \mathbf{b} \mathbf{a}^T) \quad (46)$$

$$\frac{\partial (\text{trace}(X B X^T))}{\partial X} = X B^T + X B \quad (47)$$

See Petersen and Pedersen for a proof [23].

## C PROOF OF THEOREM 4.1

Theorem 4.1, which is reproduced below for convenience, can be proved using matrix derivatives.

THEOREM. Let pairwise uncorrelated estimates  $\mathbf{x}_i (1 \leq i \leq n)$  drawn from distributions  $p_i(\mathbf{x}) = (\boldsymbol{\mu}_i, \Sigma_i)$  be fused using the linear model  $\mathbf{y}_A(\mathbf{x}_1, \dots, \mathbf{x}_n) = \sum_{i=1}^n A_i \mathbf{x}_i$ , where  $\sum_{i=1}^n A_i = I$ . The MSE( $\mathbf{y}_A$ ) is minimized for

$$A_i = \left( \sum_{j=1}^n \Sigma_j^{-1} \right)^{-1} \Sigma_i^{-1}.$$

PROOF. To use the Lagrange multiplier approach, we can convert the constraint  $\sum_{i=1}^n A_i = I$  into a set of  $m^2$  scalar equations (for example, the first equation would be  $A_1(1, 1) + A_2(1, 1) + \dots + A_n(1, 1) = 1$ ), and then introduce  $m^2$  Lagrange multipliers, which can denoted by  $\lambda(1, 1), \dots, \lambda(m, m)$ .

This obscures the matrix structure of the problem so it is better to implement this idea implicitly. Let  $\Lambda$  be an  $m \times m$  matrix in which each entry is one of the scalar Lagrange multipliers we would have introduced in the approach described above. Analogous to the inner product of vectors, we can define the inner product of two matrices as  $\langle A, B \rangle = \text{trace}(A^T B)$  (it is easy to see that  $\langle A, B \rangle$  is  $\sum_{i=1}^m \sum_{j=1}^m A(i, j) B(i, j)$ ).



Using this notation, we can formulate the optimization problem using Lagrange multipliers as follows:

$$f(A_1, \dots, A_n) = E\left\{\sum_{i=1}^n (\mathbf{x}_i - \boldsymbol{\mu}_i)^T A_i^T A_i (\mathbf{x}_i - \boldsymbol{\mu}_i)\right\} + \left\langle \Lambda, \left(\sum_{i=1}^n A_i - I\right) \right\rangle$$

Taking the matrix derivative of  $f$  with respect to each  $A_i$  and setting each derivative to zero to find the optimal values of  $A_i$  gives us the equation  $E\{2A_i(\mathbf{x}_i - \boldsymbol{\mu}_i)(\mathbf{x}_i - \boldsymbol{\mu}_i)^T + \Lambda\} = 0$ .

This equation can be written as  $2A_i\Sigma_i + \Lambda = 0$ , which implies

$$A_1\Sigma_1 = A_2\Sigma_2 = \dots = A_n\Sigma_n = -\frac{\Lambda}{2}$$

Using the constraint that the sum of all  $A_i$  equals to the identity matrix  $I$  gives us the desired expression for  $A_i$ :

$$A_i = \left(\sum_{j=1}^n \Sigma_j^{-1}\right)^{-1} \Sigma_i^{-1}$$

□

## D PROOF OF THE OPTIMALITY OF EQUATION 40

We show that  $(\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + K_t(\mathbf{z}_t - H_t\hat{\mathbf{x}}_{t|t-1}))$  (Equation 40) is an optimal unbiased linear estimator for fusing the *a priori* state estimate with the measurement at each step. The proof has two steps: we show that this estimator is unbiased, and then show it is optimal.

*Unbiased condition:* We prove a more general result that characterizes unbiased linear estimators for this problem, assuming that the prediction stage (Figure 4(d)) is unchanged. The general form of the linear estimator for computing the *a posteriori* state estimate is

$$\hat{\mathbf{y}}_{t|t} = A_t * \hat{\mathbf{y}}_{t|t-1} + B_t * \mathbf{z}_t \quad (48)$$

It is unbiased if  $E[\hat{\mathbf{y}}_{t|t}] = E[\mathbf{z}_t]$ , and we show that this is true if  $A_t = (I - B_t) * H_t$ .

The proof is by induction on  $t$ . By assumption,  $E[\hat{\mathbf{y}}_{0|0}] = E[\mathbf{x}_0]$ .

Assume inductively that  $E[\hat{\mathbf{y}}_{t-1|t-1}] = E[\mathbf{x}_{t-1}]$ .

(a) We first prove that the predictor is unbiased.

$$\hat{\mathbf{y}}_{t|t-1} = F_t * \hat{\mathbf{y}}_{t-1|t-1} + B_t * \mathbf{u}_t \quad (\text{Predictor in Figure 4})$$

$$\begin{aligned} E[\hat{\mathbf{y}}_{t|t-1}] &= F_t * E[\hat{\mathbf{y}}_{t-1|t-1}] + B_t * \mathbf{u}_t \\ &= F_t * E[\mathbf{x}_{t-1}] + B_t * \mathbf{u}_t \quad (\text{By inductive assumption}) \\ &= E[F_t * \mathbf{x}_{t-1} + B_t * \mathbf{u}_t] \\ &= E[F_t * \mathbf{x}_{t-1} + B_t * \mathbf{u}_t + \mathbf{w}_t] \quad (\mathbf{w}_t \text{ is zero mean}) \\ &= E[\mathbf{x}_t] \quad (\text{From state evolution equation 34}) \end{aligned}$$

(b) We prove that the estimator in Equation 48 is unbiased if  $A_t = (I - B_t) * H_t$ .

$$\begin{aligned} E[\hat{\mathbf{y}}_{t|t}] &= E[A_t * \hat{\mathbf{y}}_{t|t-1} + B_t * \mathbf{z}_t] \quad (\text{From Equation 48}) \\ &= A_t * E[\hat{\mathbf{y}}_{t|t-1}] + B_t * E[\mathbf{z}_t] \\ &= A_t * E[\mathbf{x}_t] + B_t * E[H_t \mathbf{x}_t + \mathbf{v}_t] \quad (\text{Equation 37 for } \mathbf{z}_t) \\ &= A_t * E[\mathbf{x}_t] + B_t * H_t * E[\mathbf{x}_t] \quad (\text{Because } \mathbf{v}_t \text{ is zero mean}) \\ &= (A_t + B_t * H_t) * E[\mathbf{x}_t] \end{aligned}$$

The estimator is unbiased if  $(A_t + B_t * H_t) = I$ , which is equivalent to requiring that  $A_t = (I - B_t * H_t)$ . Therefore the general unbiased linear estimator is of the form

$$\begin{aligned} \hat{\mathbf{y}}_{t|t} &= (I - B_t * H_t) * \hat{\mathbf{y}}_{t|t-1} + B_t * \mathbf{z}_t \\ &= \hat{\mathbf{y}}_{t|t-1} + B_t * (\mathbf{z}_t - H_t * \hat{\mathbf{y}}_{t|t-1}) \end{aligned} \quad (49)$$

Since Equation 40 is of this form, it is an unbiased linear estimator.

*Optimality:* We now show that using  $B_t = K_t$  at each step is optimal, assuming that this is done at all time steps before  $t$ . Since  $\mathbf{z}_t$  and  $\mathbf{y}_{t|t-1}$  are uncorrelated, we can use Lemma 2.2 to compute the covariance matrix of  $\hat{\mathbf{y}}_{t|t}$ , denoted by  $\Xi_{t|t}$ . This gives  $\Xi_{t|t} = (I - B_t H_t) \Sigma_{t|t-1} (I - B_t H_t)^T + B_t R_t B_t^T$ . The MSE is the trace of this matrix, and we need to find  $B_t$  that minimizes this trace. Using matrix derivatives (Equation 47), we see that

$$\frac{\partial(\text{trace}(\Xi_{t|t}))}{\partial B_t} = -2(I - B_t H_t) \Sigma_{t|t-1} H_t^T + 2 * B_t R_t$$

Setting this zero and solving for  $B_t$  gives  $B_t = \Sigma_{t|t-1} H_t^T [H_t \Sigma_{t|t-1} H_t^T + R_t]^{-1}$ . This is exactly  $K_t$ , proving that Equation 40 is an optimal unbiased linear estimator.

*Comment:* This proof of optimality provides another way of deriving Equation 40. We believe the constructive approach described in Section 6 provides more insight into how and why Kalman filtering works. The ideas in that construction can be used to provide a different proof of optimality.