KALMAN FILTERS

IRINA CAZAN

Abstract. The Kalman Filter is a statistical method that involves an algorithm which provides an efficient recursive approach to estimatating the states of a process by minimizing the mean of the squared error. The filter is a powerful tool in statistical signal processing that allows for acurate estimations of past, present and future states, even with an incomplete or imprecise system model.

1. Introduction and Motivation

Rudolf Kalman is an electrical engineer noted for his co-invention of the Kalman Filter, a mathematical technique widely used in the digital computing of control systems, navigation systems, avionics, and outer-space vehicles. The Kalman Filter extracts a signal from a long sequence of noisy or incomplete technical measurements, usually made by electronic systems.

Professor Kalman developed his theory in the late 1950s while working at the Research Institute for Advanced Studies in Baltimore, Maryland. His breakthrough is described in the paper "A New Approach to Linear Filtering and Prediction Problems" in 1960. [2]

The Kalman filter, and its later extensions to nonlinear problems, represents one of the most widely applied products of modern control theory. It has been used in space vehicle navigation and control (e.g. the Apollo vehicle), radar tracking algorithms for anti-ballistic missile applications, process control, and socio-economic systems. Its popularity in a variety of applications is due to the fact that the digital computer is effectively used in both the design phase as well as the implementation phase of the application. From a theoretical point of view, it brought under a common roof related concepts of filtering and control, and the duality between these two problems.

The Kalman filter uses elements of estimation theory to obtain the best unbiased estimator of a state of a dynamic system using previous measurement knowledge. Previous algorithms suffer from the computational limitation of using all previous information to estimate the state of the system at the next time step. In contrast, the Kalman filtering method makes use of the previous time step information and makes a-priori and a-posteriori predictions which are corrected using the new measurement. The prediction-correction iterations make use of Bayes' Rule and a collection of concepts which will be presented in section 2. Section 3 will cover in detail the system model. Section 4 contains a description of the Kalman filtering algorithm, which will be derived using two different approaches, followed by section 5 presenting one possible application of the filtering method.

2. Preliminaries

2.1. **Estimation Theory.** Estimation theory is a branch of statistics used in signal processing that deals with methods of extracting information from noisy observations. Its goal is to estimate the values of parameters based on measured data that has a random component. The parameters govern an underlying physical setting in such a way that the values of the parameters describe the distribution of the measured data. An estimator attempts to approximate the unknown parameters using measurements obtained from a population. A population can be defined as the set of elements with the characteristic one wishes to understand. Because there is rarely enough time or money to gather information from everyone or everything in a population, the goal becomes finding a representative sample (or subset) of that population. Estimation theory assumes that the observations contain an information-bearing quantity, therefore assuming that detection-based preprocessing has been performed (it is asking whether there is something in the observations worth estimating).

Definition 2.1. *(Random Variable) A random variable X is a function that assigns a real number, $X(\xi)$, to each outcome ξ in the sample space of a random experiment. $X : S \to \mathbb{R}$. The range of the random variable is a subset of the set of all real numbers.*

Remark. The function that assigns values to each outcome is fixed and deterministic. The randomness in the observed values is due to the underlying randomness of the arguments of the function X, namely the experiment outcomes ξ. In other words, the randomness in the observed values of X is induced by the underlying random experiment, and we should therefore be able to compute the probabilities of the observed values in terms of the probabilities of the underlying outcomes.

The behavior of a random variable is governed by chance, thus we can describe the behavior of a random variable in terms of probabilities. A random variable is completely described by telling the probability of each outcome.

The most fundamental property of an random variable X is its probability distribution function (PDF) $F_X(x)$, defined as

$$F_X(x) = P(X \leq x) \tag{2.1}$$

In the above equation, $F_X(x)$ is the PDF of the random variable X, and x is a nonrandom independent variable or constant. The probability density function (pdf) $f_x(x)$ is defined as the derivative of the PDF.

$$f_X(x) = \frac{dF_X(x)}{dx} \tag{2.2}$$

One of the most commonly used pdfs of a random variable and the one that we will be using later on in postulating the system model is the Gaussian or normal distribution. A random variable is called Gaussian or normal if its pdf is given by

the following equation

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\,[-\frac{(x-\mu_x)^2}{2\sigma^2}] \tag{2.3}$$

The quantities in the above equation will be explained in the following definitions. Furthermore, we will drop the index for the pdf $f_X(x)$, to simplify the notation.

Let X be a random variable and let $g : \mathbb{R} \to \mathbb{R}$ be a real-valued function defined on the sample space of X. Define $Y = g(X)$, that is Y is determined by evaluating the function $g(x)$ at the value assumed by the random variable X. Then $g(X)$ is also a random variable. The probabilities with which Y takes on various values depend on the function $g(x)$ as well as the distribution function of X.

Remark. The density function of a random variable completely describes the behavior of the variable. However, associated with any random variable are constants, or parameters, which are descriptive. Knowledge of the numerical values of these parameters gives the researcher quick insight into the nature of the variables. We consider three such parameters: the mean μ, the variance σ^2 and the covariance Σ.

To understand the reasoning behind many statistical methods, it is necessary to become familiar with one general concept, namely, the idea of mathematical expectation or expected value.

Definition 2.2. *(Expected Value) Let X be a real-valued random variable with density function $f(x)$. The expected value $\mu = E[X]$ is defined by*

$$\mu = E[X] = \int_{-\infty}^{\infty} x f(x) dx$$

provided the integral

$$\int_{-\infty}^{\infty} |x| f(x) dx$$

is finite. When used in a statistical context, the expected value of a random variable X is referred to as its mean and is denoted by μ or μ_x. The mean can be thought of as a measure of the "center of location" in the sense that it indicates where the "center" of the density lies.

Theorem 2.3. *(Properties of the expected value) Let X and Y be real-valued random variables and $c \in \mathbb{R}$ a scalar, then*

$$E[X+Y] = E[X] + E[Y]$$
$$E[cX] = cE[X]$$

Definition 2.4. *Let X and Y be independent real-valued continuous random variables with finite expected values. Then we have*

$$E[XY] = E[X]E[Y]$$

Definition 2.5. *(Variance) Let X be a real-valued random variable with density function $f(x)$. The variance $\sigma^2 = V[X]$ is defined by*

$$\sigma^2 = V[X] = \int_{-\infty}^{\infty} (x-\mu)^2 f(x)dx = E[(X-\mu)^2] = E[X^2] - (E[X])^2$$

Theorem 2.6. *(Properties of variance) If X and Y are real-valued random variables and $c \in \mathbb{R}$ a scalar, then we have*

(1) $V[c] = 0$
(2) $V[cX] = c^2 V[X]$
(3) $V[X + c] = V[X]$
(4) *If X and Y are independent,* $V[X + Y] = V[X] + V[Y]$

Definition 2.7. *(Covariance) We will consider a vector of random variables*

$$\boldsymbol{X} = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix}$$

where $X_1, , X_n$ are random variables, each with finite variance. Then the covariance matrix $\boldsymbol{\Sigma}$ is the matrix whose (i, j) entry is the covariance between X_i and X_j

$$\boldsymbol{\Sigma}_{ij} = Cov(X_i, X_j) = E[(X_i - \boldsymbol{\mu}_i)(X_j - \boldsymbol{\mu}_j)]$$

where $\mu_i = E[X_i]$, is the expected value of the ith entry in the vector $\boldsymbol{X}$. In other words, we have

$$\boldsymbol{\Sigma} = E[(\boldsymbol{X} - E[\boldsymbol{X}])(\boldsymbol{X} - E[\boldsymbol{X}])^T] = E[\boldsymbol{X}\boldsymbol{X}^T] - \boldsymbol{\mu}\boldsymbol{\mu}^T.$$

Definition 2.8. *The Moment Generating Function (MGF) of a random variable X, is $m_X(t) = E[e^{tX}]$ if the expectation is defined.*

2.2. **Gaussian Density Function.** We will assume from now on that all of the random vectors and processes used in the derivation of the filter are Gaussian. Let us begin by looking at the Gaussian probability density function in the multivariate setting. Thus, now we have

$$\boldsymbol{\mu} = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_n \end{bmatrix} \tag{2.4}$$

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1^2 \cdots \sigma_{1n} \\ \cdots \sigma_2^2 \cdots \\ \cdots\cdots\cdots \\ \cdots\cdots \sigma_n^2 \end{bmatrix} \tag{2.5}$$

Let $\mathbf{X}$ be a $\mathbb{R}^n$ -valued random variable, the Gaussian (normal) pdf is given by

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right]$$

where

$$\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f(x) dx_1 \cdots dx_n = 1,$$

$$E[\mathbf{X}] = \boldsymbol{\mu} = \text{mean value of vector},$$

$$E[(\mathbf{X} - \boldsymbol{\mu})(\mathbf{X} - \boldsymbol{\mu})^T] = \boldsymbol{\Sigma} = \text{covariance matrix of vector},$$

Also, $|\boldsymbol{\Sigma}|$ is the determinant of $\boldsymbol{\Sigma}$, $\boldsymbol{\Sigma}^{-1}$ is the matrix inverse of $\boldsymbol{\Sigma}$. We can observe that $f(X)$ is completely characterized via two parameters - mean and covariance matrix.

It is important to note that adding two (or more) Gaussian random vectors produces a Gaussian random vector with mean equal to the sum of means and variance equal to the sum of variances of the two random vectors. Expressed more formally,

Theorem 2.9. *Let $\boldsymbol{X}$ and $\boldsymbol{Y}$ be two normal (Gaussian) independent random vectors. Then $\boldsymbol{X} + \boldsymbol{Y}$ is also normally distributed. In particular, if $\boldsymbol{X} \sim N(\boldsymbol{\mu_x}, \boldsymbol{\Sigma_x})$ ($\boldsymbol{X}$ has a normal distribution) and $\boldsymbol{Y} \sim N(\boldsymbol{\mu}_y, \boldsymbol{\Sigma_y})$ and $\boldsymbol{X}$ and $\boldsymbol{Y}$ are independent, then $\boldsymbol{Z} = \boldsymbol{X} + \boldsymbol{Y} \sim N(\boldsymbol{\mu_x} + \boldsymbol{\mu_y}, \boldsymbol{\Sigma_x} + \boldsymbol{\Sigma_y})$.*

Proof. The moment generating function of the sum of two independent random variables X and Y is just the product of the two separate moment generating functions:

$$m_{X+Y}(t) = E[e^{t(X+Y)}] = E[e^{tX}]E[e^{tY}] = m_X(t)m_Y(t) \tag{2.6}$$

Remark. For a random variable $X \sim N(\mu, \sigma^2)$, we can determine the moment generating function to be

$$m_X(t) = \int_{-\infty}^{\infty} \frac{e^{tx}}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx = e^{\mu t} + \frac{\sigma^2 t^2}{2} \tag{2.7}$$

Therefore, using this in the above equation, we obtain

$$m_{X+Y}(t) = m_X(t)m_X(t) = e^{t\mu_x - \frac{\sigma_x^2 t^2}{2}} e^{t\mu_x - \frac{\sigma_y^2 t^2}{2}} = e^{t(\mu_x+\mu_y) - \frac{(\sigma_x^2+\sigma_y^2)t^2}{2}} \tag{2.8}$$

This is the moment generating function of the normal distribution with expected value $\mu_x + \mu_y$ and variance $\sigma_x^2 + \sigma_y^2$. Recall that two distinct distributions cannot both have the same moment generating function, hence the distribution of $X + Y$ must be this normal distribution.

Let

$$\boldsymbol{X} = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix}$$

and

$$\boldsymbol{Y} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix}$$

Then

$$\boldsymbol{X+Y} = \begin{bmatrix} X_1 + Y_1 \\ \vdots \\ X_n + Y_n \end{bmatrix}$$

Since by taking the expected value of a vector of random variables we take the expected value of each of its components, we obtain

$$E[\boldsymbol{X+Y}] = \begin{bmatrix} E[X_1 + Y_1] \\ \vdots \\ E[X_n + Y_n] \end{bmatrix} = \begin{bmatrix} E[X_1] + E[Y_1] \\ \vdots \\ E[X_n] + E[Y_n] \end{bmatrix} = \begin{bmatrix} \mu_{x,1} + \mu_{y,1} \\ \vdots \\ \mu_{x,n} + \mu_{y,n} \end{bmatrix} = \boldsymbol{\mu_x + \mu_y}$$

$$\begin{aligned}
\boldsymbol{\Sigma_{X+Y}} &= E[(\boldsymbol{X+Y})(\boldsymbol{X+Y})^T] - (\boldsymbol{\mu_X + \mu_Y})(\boldsymbol{\mu_X + \mu_Y})^T \\
&= E[(X_1 + Y_1)^2 + \cdots + (X_n + Y_n)^2] - (\boldsymbol{\mu_X + \mu_Y})(\boldsymbol{\mu_X + \mu_Y})^T \\
&= E[(X_1^2 + \cdots + X_n^2) + (Y_1^2 + \cdots + Y_n^2)] + 2E[X_1Y_1 + \cdots + X_nY_n] \\
&- ((\mu_{x,1} + \mu y, 1)^2 + \cdots + (\mu_{x,n} + \mu y, n)^2) \\
&= E[(X_1^2 + \cdots + X_n^2)] + E[(Y_1^2 + \cdots + Y_n^2)] + 2E[X_1Y_1 + \cdots + X_nY_n] \\
&- ((\mu_{x,1}^2 + \cdots + \mu_{x,n}^2) + (\mu_{y,1}^2 + \cdots + \mu_{y,n}^2) + 2(\mu_{x,1}\mu_{y,1} + \cdots + \mu_{x,n}\mu_{y,n})) \\
&= E[(X_1^2 + \cdots + X_n^2)] - (\mu_{x,1}^2 + \cdots + \mu_{x,n}^2) + E[(Y_1^2 + \cdots + Y_n^2)] - (\mu_{y,1}^2 + \cdots + \mu_{y,n}^2) \\
&= \boldsymbol{\Sigma_X + \Sigma_Y}
\end{aligned}$$

This completes our proof.

□

2.3. **Random Processes and the Markov Property.** A random process is a collection of continuous-valued random variables indexed by a continuous-valued parameter $x(t)$, $t_0 \leq t \leq t_f$. Since the state of continuous dynamic processes that occur in natural or man-made systems can never be known exactly, since they are subject to disturbances, these processes are random processes. To have a complete description of such a random process, we would need to know all possible joint density functions

$$f[x(t_1), x(t_2), \ldots, x(t_N)]$$

for all t's in the interval (t_0, t_f), where the index of t runs from 1 to ∞. It is not feasible to supply and use such a large amount of information for a given process. Most common processes have the property of being Markovian, with a Markov process being completely specified by giving the joint density function

$$f[x(t), x(\tau)] = f[x(t)|x(\tau)]f[x(\tau)]$$

for all t, τ in the interval (t_0, t_f). [4]

Definition 2.10. *(Purely Random Process or White Noise) In the case that* $p[x(t)|x(\tau)] = p[x(t)]$ *for all* t *and* τ *in* (t_0, t_f) *the process is called a purely random process or white noise. If the process involves an outside disturbance f(t), with* $p[f(t)|f(\tau)] \cong p[f(t)]$ *for* $|t - \tau| \geq T$*, with* T *much smaller that the average response time of the system analyzed, then* $f(t)$ *may be considered as white noise relative to the system.* [4]

Gauss-Markov Random Process. A Gauss-Markov random process is a Markov random process with the added restriction that $p[x(\tau)]$ and $p[x(t)|x(\tau)]$ are Gaussian density functions for all t, τ in the interval (t_0, t_f). Therefore, the density function $p[x(t)]$ of a Gauss-Markov process is completely described by giving the mean or expected value vector $\mu_{x(t)} = E[x(t)]$ and the covariance matrix $Cov_{x(t)} = E[x(t) - \mu_{x(t)}][x(t) - \mu_{x(t)}]^T$. The importance of this type of process flows from the fact that most natural and man-made dynamic processes may be approximated rather accurately using Gauss-Markov processes. [4] Furthermore, it is very common to approximate a non-Gaussian Markov process by a Gauss-Markov random process because there is usually limited statistical knowledge available about the actual process. [4]

2.4. **Least Squares Estimator and the Gauss - Markov Theorem.**

Least squares estimation (LSE) can be used whenever the probabilistic information about the data is not given. The approach here is to assume a system model (rather than probabilistic assumptions about the data) and achieve a design goal assuming this model. In order to do so, we will use a linear regression model to execute the estimation.

Definition 2.11. *Given a data set* $\{Y_i, X_{i1}, \ldots, X_{im}\}_{i=1}^n$*, a linear regression model assumes that the relationship between the dependent variable* Y_i *and the m-vector of input random variables is linear. This relationship is modeled through a disturbance term* ϵ_i*, an unobserved random variable that adds noise to the linear relationship between the dependent variable and regressors. Thus the model takes the form*

$$Y_i = \beta_1 X_{i1} + \cdots + \beta_m X_{im} + \epsilon_i = \boldsymbol{X}_i^T \beta + \epsilon_i \quad i=1, \ldots, n \tag{2.9}$$

so that $X_i^T \beta$ *is the inner product between vectors* X_i *and* β*. This can also be written in matrix form as*

$$\boldsymbol{Y} = \boldsymbol{X}^T \boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{2.10}$$

where

$$\boldsymbol{Y} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix}, \boldsymbol{X} = \begin{bmatrix} X_1^T \\ \vdots \\ X_n^T \end{bmatrix} = \begin{bmatrix} X_{11} \cdots X_{1m} \\ X_{21} \cdots X_{2m} \\ \vdots \ddots \vdots \\ X_{n1} \cdots X_{nm} \end{bmatrix}, \boldsymbol{\beta} = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_m \end{bmatrix}, \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

In concise form, the Gauss - Markov theorem states that in a linear regression model in which the errors have expectation zero, are uncorrelated and have equal variances, the best linear unbiased estimator (BLUE) of the coefficients is given by the ordinary least squares estimator. "Best" means giving the minimum mean squared error of the estimate. [8]

More precisely,

Theorem 2.12. *(Gauss-Markov) Assuming the following linear regression model*

$$\boldsymbol{Y} = \boldsymbol{X}^T\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{2.11}$$

where $\boldsymbol{\beta}$ is a vector of non-random unobservable parameters, $\boldsymbol{X}_{ij}$ are non-random and observable vectors, $\boldsymbol{\epsilon}$ is a vector of random variables, therefore the $\boldsymbol{Y}$'s are random vectors. If the following conditions are met:

1. *$E[\epsilon_i] = 0$, $V[\epsilon_i] = \sigma^2$ and $Cov(\epsilon_i, \epsilon_j) = 0$ for all $i \neq j$.*
2. *$\hat{\boldsymbol{\beta}}$ is an estimator of $\boldsymbol{\beta}$. We say that the estimator is unbiased if the following relationship holds: $E[\hat{\boldsymbol{\beta}}] = \boldsymbol{\beta}$.*
3. *Let $\sum_{j=1}^{m} \lambda_j \beta_j$ be any linear combination of the coefficients, then the mean squared error of the estimation is $E\left[\sum_{j=1}^{m} \lambda_j(\hat{\beta}_j - \beta_j)\right]$.*

Then the BLUE of $\boldsymbol{\beta}$ is the estimator with the minimum mean squared error for every linear combination parameters. The ordinary least squares estimator is the function:

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y} \quad \textit{of } \boldsymbol{X} \textit{ and } \boldsymbol{Y} \textit{ that minimizes the sum squares of residuals}:$$

$$\sum_{i=1}^{n}(\boldsymbol{Y}_i - \hat{\boldsymbol{Y}}_i)^2 = \sum_{i=1}^{n}(\boldsymbol{Y}_i - \sum_{j=1}^{K}\hat{\beta}_j\boldsymbol{Y}_{ij})^2$$

Hence, the ordinary least squares estimator is a BLUE.

We will skip the proof of the theorem here since it is not needed for our purposes.

Conditional expectations will play a crucial role in deriving the recursive LSE and in obtaining the Kalman filter. The calculus of conditional expectations is used in the context of a simple regression model in which a regression relationship is postulated in which the unknown parameters are regarded as fixed quantities. The next few derivations are meant to present some essential relationships within the underlying theoretical regression relationship. These will prove necessary in deriving the Kalman Filter algorithm in the next sections.

Let $\mathbf{X}$ and $\mathbf{Y}$ be random vectors with a joint distribution completely characterized by first and second order moments. We can define the second order moments of $\mathbf{X}$

and $\mathbf{Y}$:

$$V[\mathbf{X}] = E[\mathbf{X}\mathbf{X}^T] - E[\mathbf{X}]E[\mathbf{X}^T] \tag{2.12}$$

$$V[\mathbf{Y}] = E[\mathbf{Y}\mathbf{Y}^T] - E[\mathbf{Y}]E[\mathbf{Y}^T] \tag{2.13}$$

$$Cov(\mathbf{Y}, \mathbf{X}) = E[\mathbf{Y}\mathbf{X}^T] - E[\mathbf{Y}]E[\mathbf{X}^T] \tag{2.14}$$

Let the conditional expectation of $\mathbf{Y}$ given $\mathbf{X}$ be a linear functional of $\mathbf{X}$:

$$E[\mathbf{Y}|\mathbf{X}] = \boldsymbol{\alpha} + \mathbf{B}^T\mathbf{X} \tag{2.15}$$

The goal is to find vector $\boldsymbol{\alpha}$ and matrix $\mathbf{B}$ in terms of the second order moments described above. First, multiply the conditional expectation by the marginal density function of $\mathbf{X}$ and integrate with respect to $\mathbf{X}$ to get $E[E[\mathbf{Y}|\mathbf{X}]] = E[\mathbf{Y}]$

Using this on the above conditional expectation we obtain $E[\mathbf{Y}] = \boldsymbol{\alpha} + \mathbf{B}^T E[\mathbf{X}]$ and the following expression for $\boldsymbol{\alpha}$:

$$\boldsymbol{\alpha} = E[\mathbf{Y}] - B^T E[\mathbf{X}] \tag{2.16}$$

Second, we will multiply $E[\mathbf{Y}|\mathbf{X}]$ by $\mathbf{X}^T$ and by the marginal density function of $\mathbf{X}$ and take its expectation

$$E[\mathbf{Y}\mathbf{X}^T] = \boldsymbol{\alpha}E[\mathbf{X}] + \mathbf{B}^T E[\mathbf{X}\mathbf{X}^T] \tag{2.17}$$

Multiplying $E[\mathbf{Y}] = \boldsymbol{\alpha} + \mathbf{B}^T E[\mathbf{X}]$ by $E[\mathbf{X}^T]$ gives us

$$E[\mathbf{Y}]E[\mathbf{X}^T] = \boldsymbol{\alpha}E[\mathbf{X}] + \mathbf{B}^T E[\mathbf{X}]E[\mathbf{X}^T] \tag{2.18}$$

Subtracting equation (2.12) from (2.13) and using the definitions for the second order moments in (2.10), we have

$$Cov(\mathbf{Y}, \mathbf{X}) = \mathbf{B}^T Cov(\mathbf{X}) \quad \text{or} \quad \mathbf{B}^T = Cov(\mathbf{Y}, \mathbf{X})Cov^{-1}(\mathbf{X}) \tag{2.19}$$

Substituting the derived expressions for $\boldsymbol{\alpha}$, (2.12), and for $\mathbf{B}^T$, (2.14), into the definition for $E[\mathbf{Y}|\mathbf{X}]$, we obtain $E[\mathbf{Y}|\mathbf{X}] = \boldsymbol{\alpha} + \mathbf{B}^T\mathbf{X}$, or

$$E[\mathbf{Y}|\mathbf{X}] = E[\mathbf{Y}] + Cov(\mathbf{Y}, \mathbf{X})Cov^{-1}(\mathbf{X})(\mathbf{X} - E[\mathbf{X}]) \tag{2.20}$$

Below is a summary of our findings and other equations which can be derived similarly:

$$E[\mathbf{Y}|\mathbf{X}] = E[\mathbf{Y}] + Cov(\mathbf{Y}, \mathbf{X})Cov^{-1}(\mathbf{X})(\mathbf{X} - E[\mathbf{X}]) \tag{2.21}$$

$$Cov(\mathbf{Y}|\mathbf{X}) = Cov(\mathbf{Y}) - Cov(\mathbf{Y}, \mathbf{X})Cov^{-1}(\mathbf{X})Cov(\mathbf{X}, \mathbf{Y}) \tag{2.22}$$

$$E[E[\mathbf{Y}|\mathbf{X}]] = E[\mathbf{Y}] \tag{2.23}$$

$$Cov(E[\mathbf{Y}|\mathbf{X}]) = Cov(\mathbf{Y}, \mathbf{X})Cov^{-1}(\mathbf{X})Cov(\mathbf{X}, \mathbf{Y}) \tag{2.24}$$

$$Cov(\mathbf{Y}) = Cov(\mathbf{Y}, \mathbf{X}) + Cov(E[\mathbf{Y}|\mathbf{X}]) \tag{2.25}$$

$$Cov(\mathbf{Y} - E[\mathbf{Y}|\mathbf{X}], \mathbf{X}) = 0 \tag{2.26}$$

These will later be used in order to determine the recursive algorithm for Kalman filtering.

3. The Model

To begin deriving a model and solving the filtering problem, it is necessary to understand what is our goal and what is the information available from the instruments or sensors used to detect the desired signal. We want to estimate the state of the system, $\mathbf{X}$, which is a random vector containing n different system states, and we have available a set of measurements $\mathbf{X}$, which is a random vector composed of m measurements. State $\mathbf{X}$ is a random variable and belongs to the following time-controlled process governed by the linear stochastic difference equation:

$$\mathbf{X}_k = \mathbf{a}\mathbf{X}_{k-1} + \mathbf{b}\mathbf{u}_{k-1} + \mathbf{W}_{k-1} \tag{3.1}$$

and the sensors have the following sensing model:

$$\mathbf{Z}_k = \mathbf{h}\mathbf{X}_k + \mathbf{V}_k \tag{3.2}$$

where $\mathbf{W}_k$ and $\mathbf{V}_k$ are the process and measurement noise, respectively, which are zero-mean white-Gaussian random processes independent of each other: $\mathbf{W} \sim N(\mathbf{0}, \mathbf{Q})$ and $\mathbf{V} \sim N(\mathbf{0}, \mathbf{R})$, where $\mathbf{Q}$ and $\mathbf{R}$ covariance matrices defined as follows:

$$\mathbf{Q} = E[\mathbf{W}\mathbf{W}^T],$$
$$\mathbf{R} = E[\mathbf{V}\mathbf{V}^T].$$

The $n \times n$ matrix $\mathbf{a}$ relates state $\mathbf{X}$ at time step $k-1$ to the state at time step k, the $n \times l$ matrix $\mathbf{b}$ relates the optional control input $\mathbf{u} \in \mathbb{R}^l$ to state $\mathbf{X}$, and the $m \times n$ matrix $\mathbf{h}$ relates state $\mathbf{X}_k$ to measurement $\mathbf{Z}_k$. The above matrices are assumed to be constant for the simplicity of the derivation. In practice, they might change at each time step. [1]

In order to proceed with the derivation, it is necessary to introduce some notation that will be heavily used later on. The following definitions make use of Bayes' Rule. The essence of the Bayesian approach is to provide a mathematical rule explaining how one should change the existing beliefs in the light of new evidence. In other words, it allows scientists to combine new data with their existing knowledge. The process involves a prediction-correction algorithm: one makes an assumption (prediction) about the behavior of the system at the next time step (a priori estimate) and adjusts it (a posteriori estimate) using data from sensors or measuring instruments as soon as it becomes available (correction). Therefore, we define

$$\hat{\mathbf{X}}_k^- \in \mathbb{R}^n \tag{3.3}$$

$$\hat{\mathbf{X}}_k \in \mathbb{R}^n \tag{3.4}$$

$$\mathbf{e}_k^- = \mathbf{X}_k - \hat{\mathbf{X}}_k^- \tag{3.5}$$

$$\mathbf{e}_k = \mathbf{X}_k - \hat{\mathbf{X}}_k \tag{3.6}$$

$$\mathbf{\Sigma}_k^- = E[\mathbf{e}_k^-(\mathbf{e}_k^-)^T] \tag{3.7}$$

$$\mathbf{\Sigma}_k = E[\mathbf{e}_k\mathbf{e}_k^T] \tag{3.8}$$

where $\hat{\mathbf{X}}_k^-$ is the a - priori state estimate at step k (which means that we know the process prior to step k), $\hat{\mathbf{X}}_k$ is the a - posteriori state estimate at step k (which means that we know measurement $\mathbf{Z}_k$), $\mathbf{e}_k^-$ is the a - priori estimate error, $\mathbf{e}_k$ is the a - posteriori estimate error, $\mathbf{\Sigma}_k^-$ is the a - priori estimate error covariance matrix and $\mathbf{\Sigma}_k$ is the a - posteriori estimate error covariance matrix.

We want to find an equation that computes an a posteriori state estimate $\hat{\mathbf{X}}_k$ as a linear combination of an a priori estimate $\hat{\mathbf{X}}_k^-$ and a weighted difference between an actual measurement $\mathbf{Z}_k$ and a measurement prediction $\mathbf{h}\hat{\mathbf{X}}_k^-$.

$$\hat{\mathbf{X}}_k = \hat{\mathbf{X}}_k^- + \mathbf{K}_k(\mathbf{Z}_k - \mathbf{h}\hat{\mathbf{X}}_k^-) \tag{3.9}$$

In equation (3.9) there are two quantities playing a significant role: $\mathbf{Z}_k - \mathbf{h}\hat{\mathbf{X}}_k^-$ is called the measurement innovation or the residual and $\mathbf{K}_k$ is a $n \times m$ matrix called the Kalman gain.

We now need to formulate an estimation algorithm such that the following statistical conditions hold:

(1) The expected value of the state estimate is equal to the expected value of the true state. That is,on average,the estimate of the state will equal the true state.
(2) We want an estimation algorithm that minimizes the expected value of the square of the estimation error. That is,on average,the algorithm gives the smallest possible estimation error.

In order to do so, we need to identify the optimal value for the Kalman gain, $\mathbf{K}_k$. The above conditions translate into the following steps that need to be taken. First, we will take the expected value of the a posteriori error covariance and observe whether the estimator is biased. Then we will analyze the variances of the estimation errors in order to determine whether the Kalman gain computed by the error variance minimization process is optimal.

Theorem 3.1. *The optimal value of the Kalman gain matrix (the value which minimizes error covariance or achieves the best estimate) is*

$$\boldsymbol{K}_k = \mathbf{\Sigma}_k^- \boldsymbol{h}^T(\boldsymbol{h}\mathbf{\Sigma}_k^- \boldsymbol{h}^T + \boldsymbol{R})^{-1} \tag{3.10}$$

Proof. The main steps of the derivation are presented here, with the complete equations included in the appendix. Consider the dynamic process model and sensing model as stated in the beginning of the section with $e_k = \mathbf{X}_k - \hat{\mathbf{X}}_k$ the a posteriori estimate error. Let's begin by looking at the mean of this error by taking its expected value.

$$E[e_k] = E[\mathbf{X}_k - \hat{\mathbf{X}}_k] \qquad = E[(\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^- - \mathbf{K}_k\mathbf{V}_k].$$

Remark. Since $\mathbf{K}_k$ and $\mathbf{h}$ are constant at each time step and we are aiming to derive the expression for $\mathbf{K}_k$, we know that

$$\begin{aligned} E[\mathbf{K}_k] &= \mathbf{K}_k \\ E[\mathbf{h}] &= \mathbf{h} \\ E[\mathbf{K}_k^T] &= \mathbf{K}_k^T \\ E[\mathbf{h}^T] &= \mathbf{h}^T \end{aligned}$$

Given the above observation, we can now rewrite the mean of the a posteriori error covariance as follows

$$E[e_k] = (\mathbb{I} - \mathbf{K}_k\mathbf{h})E[e_k^-] - \mathbf{K}_k E[\mathbf{V}_k].$$

Therefore, if $E[\mathbf{V}_k] = 0$ and $E[e_k^-] = 0$, then $E[e_k] = 0$. This means that if the measurement noise is zero-mean for all k (which we know to be true because of the system model), and the initial estimate of the system state is set equal to its expected value, $\hat{\mathbf{X}}_0 = E[\mathbf{X}_0]$, then the expected value of $\hat{\mathbf{X}}_k$ will be equal to $\mathbf{X}_k$ for all k. Because of this, the estimator of the system state is unbiased, a property which holds regardless of the value of the gain matrix. Thus, the first requirement that we have outlined above is satisfied.

Keep in mind that the goal is to determine the optimal value of the Kalman gain matrix. The optimality criterion used is to minimize the aggregate variance of the estimation errors at time k. Let M_k be the aggregate variance of the estimation errors at time k. Then

$$\begin{aligned} M_k &= E[||\mathbf{X}_k - \hat{\mathbf{X}}_k||^2] = E[e_k^T e_k] \\ &= E[\mathrm{tr}(e_k e_k^T)] = \mathrm{tr}(E[e_k e_k^T]) = \mathrm{tr}(\mathbf{\Sigma}_k) \end{aligned}$$

where tr is the trace operator and $\mathbf{\Sigma}_k$ is the a posteriori estimate error. Let us now get an expression for $e_k e_k^T$, take its expected value and finally compute its trace. The derivations are below.

Step 1: First, we will derive an expression for $e_k e_k^T$.

$$\begin{aligned} e_k e_k^T &= ((\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^- - \mathbf{K}_k\mathbf{V}_k)((\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^- - \mathbf{K}_k\mathbf{V}_k)^T \\ &= (\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^- e_k^{-T}(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T - (\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^-\mathbf{V}_k^T\mathbf{K}_k^T \\ &- \mathbf{K}_k\mathbf{V}_k e_k^{-T}(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T + \mathbf{K}_k\mathbf{V}_k\mathbf{V}_k^T\mathbf{K}_k^T \end{aligned}$$

Step 2: Second, we will take the expected value of $e_k e_k^T$, or $E[e_k e_k^T]$.

$$E[e_k e_k^T] = E[\mathbb{I} - \mathbf{K}_k\mathbf{h}]\mathbf{\Sigma}_k^- E[(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T] - E[(\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^-\mathbf{V}_k^T\mathbf{K}_k^T] - E[\mathbf{K}_k\mathbf{V}_k e_k^{-T}(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T] + E[\mathbf{K}_k\mathbf{V}_k\mathbf{V}_k^T\mathbf{K}_k^T]$$

Remark. $E[\mathbf{V}_k] = 0$, $E[\mathbf{V}_k^T] = 0$ and $\mathbf{V}_k$ is independent of the other terms in the expectation it appears in (thus its expectation can be factored out), because it was defined as a measurement noise, which is white Gaussian noise. Thus, $E[(\mathbb{I} - \mathbf{Kh})e_k^-\mathbf{V}_k^T\mathbf{K}^T] = 0 = E[\mathbf{K}\mathbf{V}_k e_k^{-T}(\mathbb{I} - \mathbf{Kh})^T]$.

Therefore, $\mathbf{\Sigma}_k = E[e_k e_k^T]$ becomes:

$$E[e_k e_k^T] = E[\mathbb{I} - \mathbf{K}_k\mathbf{h}]\mathbf{\Sigma}_k^- E[(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T] + E[\mathbf{K}\mathbf{V}_k\mathbf{V}_k^T\mathbf{K}_k^T]$$

Step 3: In order to minimize matrix $\mathbf{\Sigma}_k$, we need to minimize its trace. Therefore, the goal now is to minimize $\mathrm{tr}(E[e_k e_k^T])$. We will do this by taking the derivative of the trace with respect to $\mathbf{K}_k$. To this effect, replace $\mathbf{K}_k$ with $\mathbf{K}_k + tU$, where t is a scalar and U is a direction matrix, and take the derivative with respect to t at $t = 0$.

Hence we get the following expression

$$M_k = \mathrm{tr}((\mathbb{I} - \mathbf{K}_k\mathbf{h})\mathbf{\Sigma}_k^-(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T - t((\mathbb{I}\mathbf{\Sigma}_k^-\mathbf{h}^T U^T + U\mathbf{h}\Sigma_k^-(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T) + t^2(\ldots) + \mathbf{K}_k\mathbf{R}\mathbf{K}_k^T + t\mathbf{K}_k\mathbf{R}U^T + tU\mathbf{R}\mathbf{K}_k^T + t^2(\ldots))$$

Now take the derivative with respect to t at $t = 0$ and set equal to 0. Only the coefficients of the linear terms will remain.

$$\frac{\partial M_k}{\partial \mathbf{K}_k} = -\mathrm{tr}((\mathbb{I} - \mathbf{Kh})\mathbf{\Sigma}_k^-(U\mathbf{h})^T) - \mathrm{tr}(U\mathbf{h}\mathbf{\Sigma}_k^-(\mathbb{I} - \mathbf{Kh})^T + \mathrm{tr}(\mathbf{K}\mathbf{R}U^T) + \mathrm{tr}(U\mathbf{R}\mathbf{K}^T) = 0$$

Since $\mathrm{tr}(A^T) = \mathrm{tr}(A)$ because diagonals are preserved, notice that $\mathrm{tr}(U^T\mathbf{K}) = \mathrm{tr}((\mathbf{K}^T U)^T) = \mathrm{tr}(\mathbf{K}^T U)$. Also, by the same reasoning, $\mathrm{tr}(U\mathbf{h}\mathbf{\Sigma}_k^-(\mathbb{I} - \mathbf{Kh})^T) = \mathrm{tr}((\mathbb{I}_{\mathbf{K}}\mathbf{h})\mathbf{\Sigma}_k^-(UH)^T)$. Thus, the above expression becomes:

$$\frac{\partial M_k}{\partial \mathbf{K}_k} = \mathrm{tr}(-U\mathbf{h}\mathbf{\Sigma}_k^-(\mathbb{I} - \mathbf{Kh})^T + \mathbf{K}^T U\mathbf{R}) = 0$$

Since $\mathbf{R} = E[\mathbf{V}\mathbf{V}^T]$ and $\mathbf{\Sigma}_k^- = E[e_k^- e_k^{-T}]$, they are both symmetric matrices. Hence, $\mathbf{R}^T = \mathbf{R}$ and $\mathbf{\Sigma}_k^{-T} = \mathbf{\Sigma}_k^-$. Also, for symmetric matrix $\mathbf{R}$ and matrices A, B, C, the following are true: $\mathrm{tr}(RA) = \mathrm{tr}((A^T R)^T) = \mathrm{tr}(A^T R) = \mathrm{tr}(RA^T)$ and $\mathrm{tr}(ABC) = \mathrm{tr}(CAB) = \mathrm{tr}(BCA)$.

Using these, we get

$$\frac{\partial M_k}{\partial \mathbf{K}_k} = \mathrm{tr}(-\mathbf{h}\Sigma_k^-(\mathbb{I} - \mathbf{Kh})^T + \mathbf{R}\mathbf{K}^T)U) = 0$$

There are n^2 choices for the direction matrix $U_{n\times n}$, depending where the 1 is placed (all other entries are zeros). Therefore:

$$\begin{aligned} &-\mathbf{h}\mathbf{\Sigma}_k^-(\mathbb{I}-\mathbf{K}\mathbf{h})^T+\mathbf{R}\mathbf{K}^T)=0\\ \Leftrightarrow&\mathbf{K}=\mathbf{\Sigma}_k^-\mathbf{h}^T(\mathbf{h}\mathbf{\Sigma}_k^-\mathbf{h}^T+\mathbf{R})^{-1}\end{aligned}$$

□

Let us now look at the limiting cases when the measurement covariance matrix $\mathbf{R}$ converges to $\mathbf{0}$ and when the a priori estimate error covariance $\mathbf{\Sigma}_k^-$ converges to $\mathbf{0}$. [1]

$\lim_{\mathbf{R}_k\to\mathbf{0}}\mathbf{K}_k=\mathbf{\Sigma}_k^-\mathbf{h}^T(\mathbf{h}\mathbf{\Sigma}_k^-\mathbf{h}^T)^{-1}=\mathbf{h}^{-1}$ The gain weights the residual more heavily

$\lim_{\mathbf{\Sigma}_k^-\to\mathbf{0}}\mathbf{K}_k=0$ The gain weights the residual less heavily

4. The Kalman Filter Algorithm

The Kalman filter is built as a feedback control mechanism. It estimates the process state at a point in time and combines this a priori estimate with the noisy measurements to get an a posteriori estimate. The prediction part of the method is the time update, or projection of the a posteriori estimate from time step k to time step $k+1$. The a posteriori estimate now becomes the new a priori estimate after the projection. In the correction phase the system receives data about the process that it postulated a belief about and adjusts the estimator in a measurement update at that time step.

Time update equations:

$$\hat{\mathbf{X}}_k^- = \mathbf{a}\hat{\mathbf{X}_{k-1}} + \mathbf{b}u_{k-1} \tag{4.1}$$

$$\mathbf{\Sigma}_k^- = \mathbf{a}\mathbf{\Sigma}_{k-1}\mathbf{a}^T + \mathbf{Q} \tag{4.2}$$

Measurement update equations:

$$\mathbf{K}_k = \mathbf{\Sigma}_k^-\mathbf{h}^T(\mathbf{h}\mathbf{\Sigma}_k^-\mathbf{h}^T+\mathbf{R})^{-1} \tag{4.3}$$

$$\hat{\mathbf{X}}_k = \hat{\mathbf{X}}_k^- + \mathbf{K}_k(\mathbf{Z}_k - \mathbf{h}\hat{\mathbf{X}}_k^-) \tag{4.4}$$

$$\mathbf{\Sigma}_k = (\mathbb{I}-\mathbf{K}_k\mathbf{h})\mathbf{\Sigma}_k^- \tag{4.5}$$

The previous section has shown how to obtain the above equations by using minimization techniques and had heavily relied on matrix algebra.

The issue that needs to be explored is: how do we know that the recursive algorithm will provide us with the optimal estimator of the process states? If we obtain a set a equations, is there a way to tell with certainty that for any application using it will result in an accurate estimator? To tackle this challenge, let us consider two methods of deriving the Kalman method and compare them. The first one was already included in the previous chapter and it involved minimizing the a posteriori error covariance, a rather intuitive way to go about solving the problem. The findings will be summarized below. The second method which will be explained in detail in this section makes use

of recursive least square estimation and will hinge on the Gauss - Markov Theorem presented in the Preliminaries section. According to the theorem, the recursive LSE method will allow us to find the best linear unbiased estimator (BLUE) every time if the underlying assumptions are satisfied. After the derivation is complete we will compare the methods and draw our conclusions.

We are given the following regression model, which is the sensing model for the dynamic process assumed in section 3:

$$\mathbf{Z}_k = \mathbf{h}\mathbf{X}_k + \mathbf{V}_k$$

where $E[\mathbf{V}_k] = 0$, $V[\mathbf{V}_k] = \mathbf{R}$, and $Cov(\mathbf{V}_k, \mathbf{V}_l) = 0$ for $k \neq l$. We also know that the system states, the $\mathbf{X}$s are related in the following way:

$$\mathbf{X}_k = \mathbf{a}\mathbf{X}_{k-1} + \mathbf{b}u_{k-1} + \mathbf{W}_{k-1}$$

with $\hat{\mathbf{X}}_k$ a linear estimator of $\mathbf{X}_k$. Since the assumptions of the Gauss - Markov Theorem are satisfied, we know that the estimators for $\mathbf{X}_k$ at each time step will be BLUEs.

Now, given a set of observations at time k, $\mathbf{Z}_k$, we want to derive the a posteriori estimates $\hat{\mathbf{X}}_k = E[\mathbf{X}_k|\mathbf{Z}_k]$ and $\Sigma_k = Cov(\mathbf{X}_k|\mathbf{Z}_k)$ from the a priori estimates $\hat{\mathbf{X}_k^-}$ and $\mathbf{\Sigma}_k^-$. To begin with, let $\mathbf{X}$ be $\mathbf{Z}_k$ and $\mathbf{Y}$ be $\mathbf{X}_k$ in (2.17) and get:

$$E[\mathbf{X}_k|\mathbf{Z}_k] = E[\mathbf{X}_k|\mathbf{Z}_{k-1}] + Cov(\mathbf{X}_k, \mathbf{Z}_k|\mathbf{Z}_{k-1})Cov^{-1}(\mathbf{Z}_k|\mathbf{Z}_{k-1})(\mathbf{Z}_k - E[\mathbf{Z}_k|\mathbf{Z}_{k-1}]) \tag{4.6}$$

Following $E[\mathbf{X}_k|\mathbf{Z}_{k-1}]$, there are three quantities that we are particularly interested in: the covariance, the variance-covariance, and the error from predicting $\mathbf{Z}_k$ using the available observations or measurements.

$$Cov(\mathbf{X}_k, \mathbf{Z}_k|\mathbf{Z}_{k-1}) = E[(\mathbf{X}_k - \hat{\mathbf{X}}_k^-)\mathbf{Z}_k^T] = E[(\mathbf{X}_k - \hat{\mathbf{X}}_k^-)(\mathbf{h}\mathbf{X}_k)^T] = \mathbf{\Sigma}_k^-\mathbf{h} \tag{4.7}$$

$$Cov(\mathbf{Z}_k|\mathbf{Z}_{k-1}) = Cov(\mathbf{h}(\mathbf{X}_k - \hat{\mathbf{X}}_k^-) + Cov(\mathbf{V}_k) = \mathbf{h}\ \boldsymbol{Sigma}_k^-\mathbf{h}^T + \mathbf{R} \tag{4.8}$$

$$\mathbf{Z}_k - E[\mathbf{Z}_k|\mathbf{Z}_{k-1}] = \mathbf{Z}_k - \mathbf{h}\hat{\mathbf{X}}_k^- := F_k \tag{4.9}$$

Using these derivations, we can return to equation (2.21) and get

$$\hat{\mathbf{X}}_k = \hat{\mathbf{X}}_k^- + \mathbf{\Sigma}_k^-\mathbf{h}^T(\mathbf{h}\ \boldsymbol{Sigma}_k^-\mathbf{h}^T + \mathbf{R})^{-1})(\mathbf{Z}_k - \mathbf{h}\mathbf{X}_k^-) \tag{4.10}$$

Similarly, using equation (2.16) we have that

$$Cov(\mathbf{X}_k|\mathbf{Z}_k) = Cov(\mathbf{X}_k|\mathbf{Z}_{k-1}) = Cov(\mathbf{X}_k, \mathbf{Z}_k|\mathbf{Z}_{k-1})Cov^{-1}(\mathbf{Z}_k, \mathbf{X}_k|\mathbf{Z}_{k-1}) \tag{4.11}$$

$$\mathbf{\Sigma}_k = \mathbf{\Sigma}_k^- - \mathbf{\Sigma}_k^-\mathbf{h}^T(\mathbf{h}\mathbf{\Sigma}_k^- + \mathbf{R})^{-1}\mathbf{h}\mathbf{\Sigma}_k^- \tag{4.12}$$

Summing up, we now have the measurement update equations to run the recursive least-squares algorithm.

$$\mathbf{F}_k = \mathbf{Z}_k - \mathbf{h}\hat{\mathbf{X}}_k^- \quad \text{Residual} \tag{4.13}$$

$$\mathbf{S}_k = \mathbf{h}\boldsymbol{\Sigma}_k^-\mathbf{h}^T + \mathbf{R} \quad \text{Error covariance} \tag{4.14}$$

$$\mathbf{K}_k = \boldsymbol{\Sigma}_k^-\mathbf{h}^T\mathbf{S}_k \quad \text{Filter gain} \tag{4.15}$$

$$\hat{\mathbf{X}}_k = \hat{\mathbf{X}}_k^- + \mathbf{K}_k\mathbf{F}_k \quad \text{Parameter estimate} \tag{4.16}$$

$$\boldsymbol{\Sigma}_k = (\mathbb{I} - \mathbf{K}_k\mathbf{h}^T)\boldsymbol{\Sigma}_k^- \quad \text{Estimate covariance} \tag{4.17}$$

The time update is performed as stated in the previous model, since it does not depend on the derivation method. The time update only involves the forward projection of the a priori estimate and error covariance using the system model. Therefore, we now have the full set of prediction-correction equations:

Time update/ Prediction phase

$$\hat{\mathbf{X}}_k^- = \mathbf{a}\hat{\mathbf{X}}_{k-1} + \mathbf{b}\mathbf{u}_{k-1} \tag{4.18}$$

$$\boldsymbol{\Sigma}_k^- = \mathbf{a}\boldsymbol{\Sigma}_{k-1}\mathbf{a}^T + \mathbf{Q} \tag{4.19}$$

Measurement update/ Correction phase

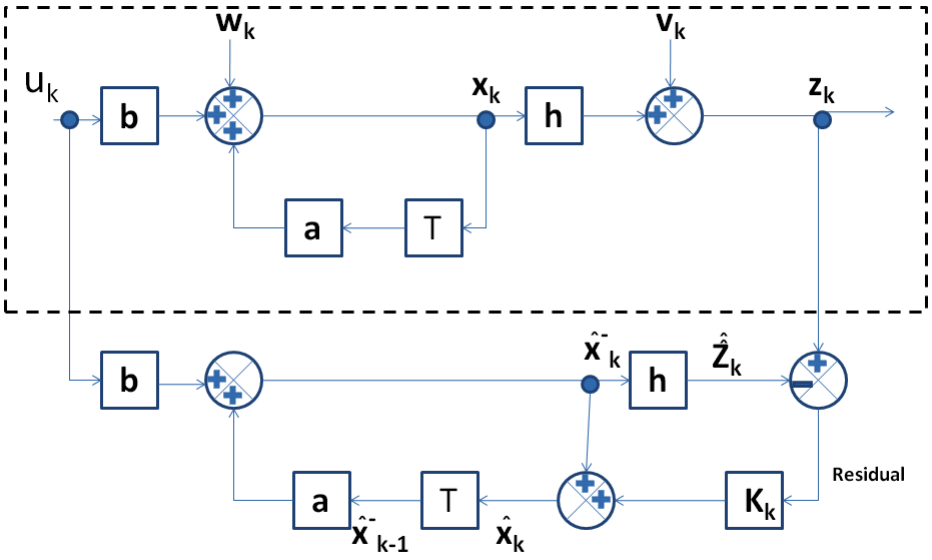
$$\mathbf{K}_k = \boldsymbol{\Sigma}_k^-\mathbf{h}^T\mathbf{S}_k \tag{4.20}$$

$$\hat{\mathbf{X}}_k = \hat{\mathbf{X}}_k^- + \mathbf{K}_k\mathbf{F}_k \tag{4.21}$$

$$\boldsymbol{\Sigma}_k = (\mathbb{I} - \mathbf{K}_k\mathbf{h}^T)\boldsymbol{\Sigma}_k^- \tag{4.22}$$

The results obtained by the two optimization methods used in the last two sections to obtain the Kalman gain for the best state estimator coincide. Therefore, the Kalman filter algorithm derived by the two methods is optimal under the model assumptions. In the first method I approached the problem by minimizing the a posteriori error covariance using the first order condition of setting the derivative equal to 0 and solving for the gain. The second method relies heavily on the Gauss - Markov Theorem assumptions to conclude that the estimator obtained using the computed Kalman gain is the best linear unbiased estimator. Because the same result has been reached using two very different methods points out that, given the assumed model, we have come up with the best estimator for the system states at each time step.

Below is a representation of the predictor-corrector mechanism that constitutes the Kalman filter. Each time an input passes through a box, it gets multiplied with the amount written inside. Inputs are added/subtracted at each node, according to indications. All notation appearing in the figure have been defined in the model section, and T represents the time delay from one time step to another.



The top part in the dashed box simply represents the system model characterized by equations (3.8) and (3.9). The whole picture shows the time and measurement updates that happen at each iteration.

5. Application: Estimate the Position and Velocity of a Moving Vehicle

The Kalman filter is a tool that can estimate the parameters of a wide range of processes. From the statistical point of view, a Kalman filter estimates the states of a linear system. The Kalman filter is not only an excellent practical tool, but it is theoretically attractive as well because, as we have shown in section 3, it minimizes the variance of the estimation error. This section will present the use of the filter to solve a vehicle navigation problem. In order to control the position of an automated vehicle, we must first have a reliable estimate of the vehicle's present position. Kalman filtering provides a tool for obtaining the best linear unbiased estimate.

Suppose we want to model a vehicle driving in a straight line. The state consists of the vehicle position p and velocity v. The input u is the commanded acceleration and the output z is the measured position. Suppose we are able to change the acceleration and measure the position every T seconds. In this case, the position p and the velocity v will be governed by the following equations:

$$p_{k+1} = p_k + Tv_k + \frac{1}{2}T^2u_k \tag{5.1}$$

$$v_{k+1} = v_k + Tu_k \tag{5.2}$$

However, the previous equations do not give a precise value for p_{k+1} or for v_{k+1}: the position and velocity will be perturbed by noise due to wind, road conditions,

and other realities of the environment the vehicle is moving in. The position and velocity noises are random variables that change with time. Therefore, more realistic expressions for p and v are:

$$p_{k+1} = p_k + Tv_k + \frac{1}{2}T^2u_k + \tilde{p}_k \tag{5.3}$$

$$v_{k+1} = v_k + Tu_k + \tilde{v}_k \tag{5.4}$$

where $\tilde{p}_k$ and $\tilde{v}_k$ are the position and velocity noise, respectively.

Let us define a state vector x that consists of position and velocity:

$$\mathbf{x}_k = \begin{bmatrix} p_k \\ v_k \end{bmatrix} \tag{5.5}$$

Since the measured output is equal to the position of the vehicle, our model is characterized by the following two equations

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} \frac{T^2}{2} \\ T \end{bmatrix} u_k + \mathbf{w}_k \tag{5.6}$$

$$\mathbf{z}_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_k + \mathbf{s}_k \tag{5.7}$$

where $\mathbf{s}_k$ is the measurement noise due to sensor inaccuracies. We want to estimate the vehicle position and velocity at each time step, therefore estimate the system state $\mathbf{x}$.

Assume the position is measured with an error of 10 feet, or one standard deviation, the commanded acceleration is a constant 1 foot/sec^2 The acceleration noise is 0.2 feet/sec^2, again, one standard deviation. Let the position be measured 10 times per second, or $T = 0.1$. Since $T = 0.1$, the linear model that represents the vehicle system can now be derived from the system model by a simple substitution:

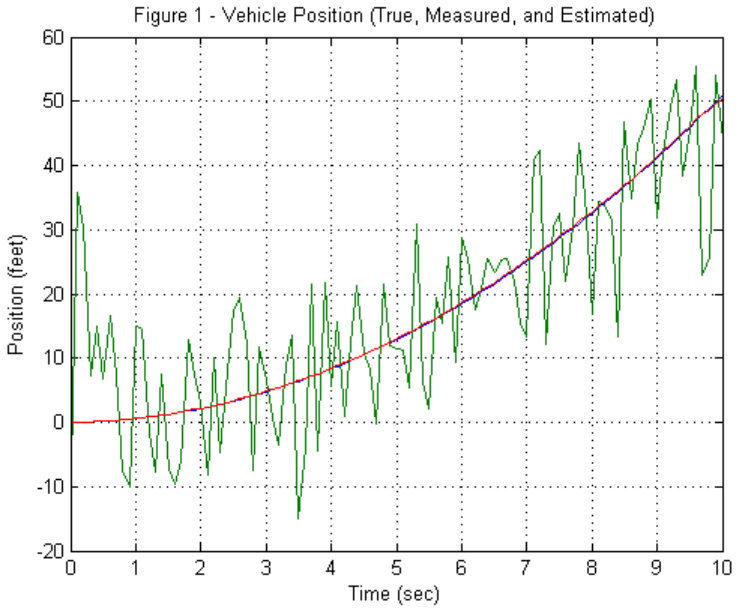
$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & 0.1 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0.005 \\ 0.1 \end{bmatrix} u_k + \mathbf{w}_k \tag{5.8}$$

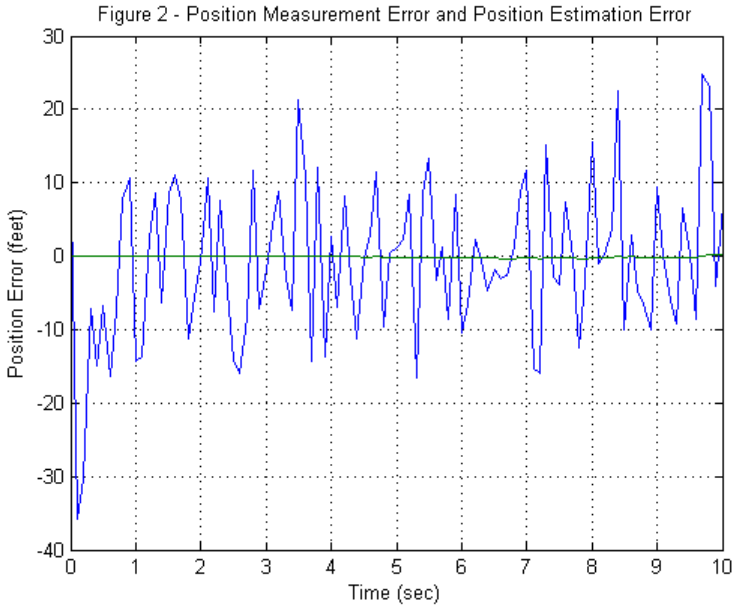
$$\mathbf{z}_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_k + \mathbf{s}_k \tag{5.9}$$

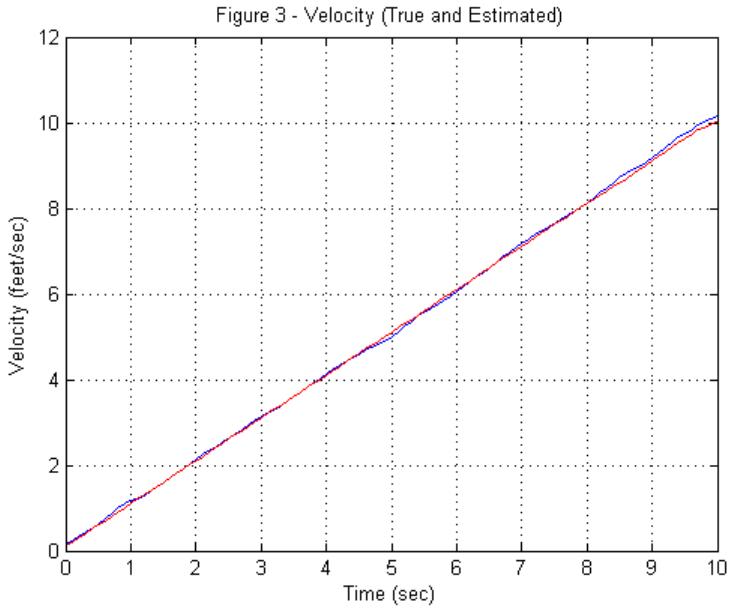
Because the standard deviation of the measurement noise is 10 feet, $R = 100$. Now, let us derive the Q matrix. Since the position is proportional to 0.005 times the acceleration, and the acceleration noise is 0.2 feet/sec^2 , the variance of the position noise is 10^{-6}. Similarly, since the velocity is proportional to 0.1 times the acceleration, the variance of the velocity noise is $4 * 10^{-4}$. The covariance of the position noise and velocity noise is equal to the standard deviation of the position noise times the standard deviation of the velocity noise, which is $2 * 10^{-5}$. We will now combine these to obtain the Q matrix:

$$Q = \begin{bmatrix} 10^{-6} & 2 * 10^{-5} \\ 2 * 10^{-5} & 4 * 10^{-4} \end{bmatrix} \tag{5.10}$$

We will work with the following initial conditions: $\hat{x}_0$ as initial estimate of position and velocity and $\Sigma_{0)}$ as the uncertainty in the initial estimate. Running the Kalman filter equations using a MATLAB routine, the following results were obtained:







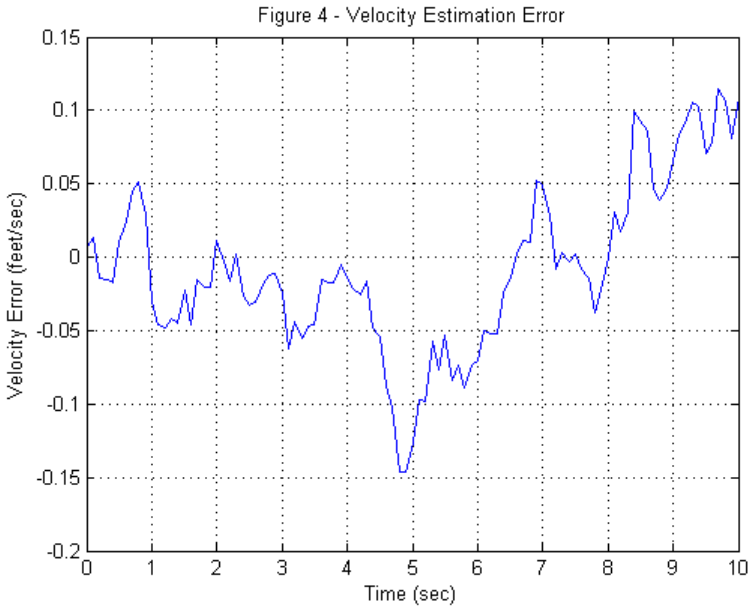


Figure 1 shows the true position of the vehicle, the measured position, and the estimated position. The two smooth curves are the true position and the estimated position, and they almost coincide. The noisy curve is the measured position. Figure 2 shows the error between the true position and the measured position, and the error between the true position and the Kalman filtered estimated position. Figure 3 shows the advantage that we get from the Kalman filter: since the vehicle velocity is part of the state $\mathbf{x}$, we get a velocity estimate along with the position estimate. Figure 4 shows the error between the true velocity and the Kalman filtered estimated velocity.

Acknowledgements

I would like to thank Professors Liam O'Brian and Andreas Malmendier for their supervision and support.

References

[1] *G. Welch, G. Bishop*, An Introduction to the Kalman Filter, Department of Computer Science, University of North Carolina at Chapel Hill, TR 95-041.
[2] *R.E. Kalman*, A New Approach to Linear Filtering and Prediction Problems, Transaction of the ASME - Journal of Basic Engineering, pp. 35-45 (March 1960).
[3] *H.W. Sorenson*, Least Squares Estimation: from Gauss to Kalman, IEEE Spectrum, vol. 7, pp. 63-68, July 1970.
[4] *A. Bryson, Y.-C Ho*, Applied Optimal Control: Optimization, Estimation, and Control
[5] *S. M. Kay*, Fundamentals of Statistical Signal Processing: Estimation Theory, Prentice-Hall, 1993
[6] *P. Zarchan, H. Musoff*, Fundamentals of Kalman Filtering: A Practical Approach, 3rd Edition, 2009
[7] *G. Casella, R. Berger*, Statistical Inference, Second Edition, Thomson Learning, 2001
[8] *D.S.G. Pollock*, The Kalman Filter etc., 2000

Appendix

Kalman Gain Derivation

$$\begin{aligned}
E[e_k] &= E[\mathbf{X}_k - \hat{\mathbf{X}}_k] = E[\mathbf{X}_k - \hat{\mathbf{X}}_k^- - \mathbf{K}(\mathbf{Z}_k - \mathbf{h}\hat{\mathbf{X}}_k^-)] \\
&= E[\mathbf{X}_k - (\mathbb{I} - \mathbf{K}_k\mathbf{h})\hat{\mathbf{X}}_k^- - \mathbf{K}_k\mathbf{Z}_k] \\
&= E[\mathbf{X}_k - (\mathbb{I} - \mathbf{K}_k\mathbf{h})\hat{\mathbf{X}}_k^- - \mathbf{K}_k(\mathbf{h}\mathbf{X}_k + \mathbf{V}_k)] \\
&= E[(\mathbb{I} - \mathbf{K}_k\mathbf{h})\mathbf{X}_k - (\mathbb{I} - \mathbf{K}_k\mathbf{h})\hat{\mathbf{X}}_k^- - \mathbf{K}_k\mathbf{V}_k] \\
&= E[(\mathbb{I} - \mathbf{K}_k\mathbf{h})(\mathbf{X}_k - \hat{\mathbf{X}}_k^-) - \mathbf{K}_k\mathbf{V}_k] \\
&= E[(\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^- - \mathbf{K}_k\mathbf{V}_k].
\end{aligned}$$

$$E[e_k] = (\mathbb{I} - \mathbf{K}_k\mathbf{h})E[e_k^-] - \mathbf{K}_kE[\mathbf{V}_k].$$

$$\begin{aligned}
M_k &= E[||\mathbf{X}_k - \hat{\mathbf{X}}_k||^2] = E[e_k^T e_k] \\
&= E[\mathrm{tr}(e_k e_k^T)] = \mathrm{tr}(E[e_k e_k^T]) = \mathrm{tr}(\mathbf{\Sigma}_k)
\end{aligned}$$

Step 1:

$$\begin{aligned}
e_k e_k^T &= ((\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^- - \mathbf{K}_k\mathbf{V}_k)((\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^- - \mathbf{K}_k\mathbf{V}_k)^T \\
&= ((\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^- - \mathbf{K}_k\mathbf{V}_k)(e_k^{-T}(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T - v_k^T\mathbf{K}_k^T) \\
&= (\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^-e_k^{-T}(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T - (\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^-\mathbf{V}_k^T\mathbf{K}_k^T \\
&- \mathbf{K}_k\mathbf{V}_ke_k^{-T}(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T + \mathbf{K}_k\mathbf{V}_k\mathbf{V}_k^T\mathbf{K}_k^T
\end{aligned}$$

Step 2:

$$\begin{aligned}
E[e_k e_k^T] &= E[\mathbb{I} - \mathbf{K}_k\mathbf{h}]\mathbf{\Sigma}_k^-E[(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T] - E[(\mathbb{I} - \mathbf{K}_k\mathbf{h})e_k^-\mathbf{V}_k^T\mathbf{K}_k^T] \\
&- E[\mathbf{K}_k\mathbf{V}_ke_k^{-T}(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T] + E[\mathbf{K}_k\mathbf{V}_k\mathbf{V}_k^T\mathbf{K}_k^T]
\end{aligned}$$

Therefore, $\mathbf{\Sigma}_k = E[e_k e_k^T]$ becomes:

$$E[e_k e_k^T] = E[\mathbb{I} - \mathbf{K}_k\mathbf{h}]\mathbf{\Sigma}_k^-E[(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T] + E[\mathbf{K}\mathbf{V}_k\mathbf{V}_k^T\mathbf{K}_k^T]$$

Step 3: Minimize $\mathrm{tr}(E[e_k e_k^T])$. We will do this by taking the derivative of the trace with respect to $\mathbf{K}_k$. To this effect, replace $\mathbf{K}_k$ with $\mathbf{K}_k + tU$, where t is a scalar and U is a direction matrix, and take the derivative with respect to t at $t = 0$.

$$\begin{aligned}
M_k &= tr((\mathbb{I} - (\mathbf{K}_k + tU)\mathbf{h})\Sigma_k^-(\mathbb{I} - (\mathbf{K}_k + tU)\mathbf{h})^T + (\mathbf{K}_k + tU)\mathbf{R}(\mathbf{K}_k + tU)^T) \\
&= tr((\mathbb{I} - \mathbf{K}_k\mathbf{h} - tU\mathbf{h})\Sigma_k^-(\mathbb{I} - \mathbf{K}_k\mathbf{h} - tU\mathbf{h})^T + (\mathbf{K}_k + tU)\mathbf{R}(\mathbf{K}_k^T + tU^T)) \\
&= tr((\mathbb{I} - \mathbf{K}_k\mathbf{h})\Sigma_k^- - tU\mathbf{h}\Sigma_k^-)((\mathbb{I} - \mathbf{K}_k\mathbf{h})^T - (t\mathbf{h}^TU^T)) + (\mathbf{K}_k\mathbf{R} + tU\mathbf{R})(\mathbf{K}_k^T + tU^T) \\
&= \mathrm{tr}((\mathbb{I} - \mathbf{K}_k\mathbf{h})\mathbf{\Sigma}_k^-(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T - t((\mathbb{I}\mathbf{\Sigma}_k^-\mathbf{h}^TU^T + U\mathbf{h}\Sigma_k^-(\mathbb{I} - \mathbf{K}_k\mathbf{h})^T) \\
&+ t^2(\ldots) + \mathbf{K}_k\mathbf{R}\mathbf{K}_k^T + t\mathbf{K}_k\mathbf{R}U^T + tU\mathbf{R}\mathbf{K}_k^T + t^2(\ldots))
\end{aligned}$$

Now take the derivative with respect to t at $t = 0$ and set equal to 0. Only the coefficients of the linear terms will remain.

$$\begin{aligned}
\frac{\partial M_k}{\partial \mathbf{K}_k} =& -\mathrm{tr}((\mathbb{I} - \mathbf{K}\mathbf{h})\mathbf{\Sigma}_k^-(U\mathbf{h})^T) - \mathrm{tr}(U\mathbf{h}\mathbf{\Sigma}_k^-(\mathbb{I} - \mathbf{K}\mathbf{h})^T \\
&+ \mathrm{tr}(\mathbf{K}\mathbf{R}U^T) + \mathrm{tr}(U\mathbf{R}\mathbf{K}^T) = 0
\end{aligned}$$

$$\begin{aligned}
\frac{\partial M_k}{\partial \mathbf{K}_k} &= -2tr(U\mathbf{h}\Sigma_k^-(\mathbb{I}_\mathbf{K}\mathbf{h})^T) + 2tr(\mathbf{K}^TU\mathbf{R}) = 0 \\
\Leftrightarrow \frac{\partial M_k}{\partial \mathbf{K}_k} &= \mathrm{tr}(-U\mathbf{h}\mathbf{\Sigma}_k^-(\mathbb{I} - \mathbf{K}\mathbf{h})^T + \mathbf{K}^TU\mathbf{R}) = 0
\end{aligned}$$

$$\begin{aligned}
\frac{\partial M_k}{\partial \mathbf{K}_k} &= tr(-\mathbf{h}\Sigma_k^-(\mathbb{I} - \mathbf{K}\mathbf{h})^TU + \mathbf{R}\mathbf{K}^TU) = 0 \\
\Leftrightarrow \frac{\partial M_k}{\partial \mathbf{K}_k} &= \mathrm{tr}(-\mathbf{h}\Sigma_k^-(\mathbb{I} - \mathbf{K}\mathbf{h})^T + \mathbf{R}\mathbf{K}^T)U) = 0
\end{aligned}$$

There are n^2 choices for the direction matrix $U_{n\times n}$, depending where the 1 is placed (all other entries are zeros). Therefore:

$$\begin{aligned}
&- \mathbf{h}\mathbf{\Sigma}_k^-(\mathbb{I} - \mathbf{K}\mathbf{h})^T + \mathbf{R}\mathbf{K}^T) = 0 \\
&(-\mathbf{h}\Sigma_k^-(\mathbb{I} - \mathbf{K}\mathbf{h})^T + \mathbf{R}\mathbf{K}^T)^T = 0 \\
&- (\mathbb{I} - \mathbf{K}\mathbf{h})\Sigma_k^-\mathbf{h}^T + \mathbf{K}\mathbf{R} = 0 \\
&- \Sigma_k^-\mathbf{h}^T + \mathbf{K}\mathbf{h}\Sigma_k^-\mathbf{h}^T + \mathbf{K}\mathbf{R} = 0 \\
&\mathbf{K}(\mathbf{h}\Sigma_k^-\mathbf{h}^T + \mathbf{R}) = \Sigma_k^-\mathbf{h}^T \\
\Leftrightarrow &\mathbf{K} = \mathbf{\Sigma}_k^-\mathbf{h}^T(\mathbf{h}\mathbf{\Sigma}_k^-\mathbf{h}^T + \mathbf{R})^{-1}
\end{aligned}$$