State Estimation and Filtering
Course notes

Andrea Garulli

Version 1.3

July 15, 2025

Contents

1 Bayesian Estimation **1**

1.1 Notation 1

1.2 Problem formulation 2

1.3 Minimum Mean Square Error Estimator 3

1.4 Linear Mean Square Error Estimator 5

1.5 Bayesian estimation in the Gaussian framework 10

1.6 Exercises 12

2 State estimation for linear systems **15**

2.1 Notation 15

2.2 State space representation of stochastic systems 16

2.3 The state estimation problem 20

2.4 The Kalman Filter 22

2.4.1 The prediction step 23

2.4.2 The correction step 24

2.4.3 Initialization 28

2.5 Properties of the Kalman Filter 28

2.5.1 The information matrix 29

2.5.2 The Kalman Filter as a dynamic system 30

2.5.3 The Kalman Filter as a state observer 31

2.5.4 The innovation process 32

2.5.5 Extension to non white disturbance and noise processes 33

2.6 Asymptotic behavior of the KF 35

2.6.1 The asymptotic Kalman Filter 37
2.7 Applications of the Kalman Filter 44

3 State estimation for nonlinear systems 51
3.1 Nonlinear State Estimation 51
3.1.1 The Extended Kalman Filter 52
3.1.2 The EKF prediction step 52
3.1.3 The EKF correction step 54
3.1.4 Properties of the EKF 56
3.2 The Continuous-Discrete Kalman Filter 57
3.3 The Unscented Kalman Filter 61
3.3.1 The Unscented Transform 61
3.3.2 The UKF Algorithm 65
3.4 The Particle Filter 67
3.4.1 Recursive computation of the a posteriori state distribution 68
3.4.2 The Particle Filter Algorithm 70

Chapter 1

Bayesian Estimation

In this chapter, we introduce the Bayesian estimation framework, which allows one to compute estimators of quantities modeled as random variables, based on the observations of other quantities, also described by random variables. First, the minimum mean square error estimator will be derived. The special class of linear estimators will be also addressed.

1.1 Notation

We start by recalling the main notation adopted in these notes. A random variable is denoted by a bold letter, while its realization by the corresponding standard letter (for example, x is a realization of the random variable $\boldsymbol{x}$). We will write $\boldsymbol{x} \in \mathbb{R}^m$ to define the vector of m random variables $[\boldsymbol{x}_1\ \boldsymbol{x}_2\ \dots \boldsymbol{x}_m]^T$ (where the superscript ^{T} denotes the vector transpose). The probability density function (pdf) of $\boldsymbol{x}$ is written as $f_{\boldsymbol{x}}(x)$, while $f_{\boldsymbol{x},\boldsymbol{y}}(x,y)$ is the joint pdf of the random variables $\boldsymbol{x}$ and $\boldsymbol{y}$ (notice that the same notation is adopted when $\boldsymbol{x}$ and $\boldsymbol{y}$ are vectors). The conditional pdf of $\boldsymbol{x}$, given the observation of $\boldsymbol{y}$, is defined as

$$f_{\boldsymbol{x}|\boldsymbol{y}}(x|y) = \frac{f_{\boldsymbol{x},\boldsymbol{y}}(x,y)}{f_{\boldsymbol{y}}(y)}.$$

The operator $\mathbf{E}[\cdot]$ is the expected value, while $\mathbf{E}[\cdot|\boldsymbol{y}]$ denotes the conditional expectation given the observation of $\boldsymbol{y}$. The notation $\boldsymbol{x} \sim (m_{\boldsymbol{x}}, P_{\boldsymbol{x}})$ means

that $\boldsymbol{x} \in \mathbb{R}^m$ is a vector random variable with mean $m_{\boldsymbol{x}} = \mathbf{E}[\boldsymbol{x}] \in \mathbb{R}^m$ and covariance matrix $P_{\boldsymbol{x}} = \mathbf{E}\left[(\boldsymbol{x} - m_{\boldsymbol{x}})(\boldsymbol{x} - m_{\boldsymbol{x}})^T\right] \in \mathbb{R}^{m \times m}$. When we write $\boldsymbol{x} \sim \mathcal{N}(m_{\boldsymbol{x}}, P_{\boldsymbol{x}})$, we mean that $\boldsymbol{x}$ is a vector of normally distributed (Gaussian) random variables, with mean $m_{\boldsymbol{x}}$ and covariance $P_{\boldsymbol{x}}$.

1.2 Problem formulation

In the Bayesian estimation setting, the objective is to estimate a random variable $\boldsymbol{x} \in \mathbb{R}^m$, by using observations of a random variable $\boldsymbol{y} \in \mathbb{R}^n$. Clearly, the complete knowledge on the stochastic relationship between $\boldsymbol{x}$ and $\boldsymbol{y}$ is given by the joint probability density function (pdf) $f_{\boldsymbol{x},\boldsymbol{y}}(x, y)$.

In particular, the aim is to find an estimator $\hat{\boldsymbol{x}} = T(\boldsymbol{y})$, where

$$T(\cdot) : \mathbb{R}^n \to \mathbb{R}^m$$

Definition 1.1. In the Bayesian setting, an estimator $T(\cdot)$ is *unbiased* if

$$\mathbf{E}[T(\boldsymbol{y})] = \mathbf{E}[\boldsymbol{x}].$$

In order to assess the performance of an estimator, it is necessary to introduce a criterion to evaluate the quality of the estimate.

Definition 1.2. We define *Bayes risk function* the quantity

$$J_r = \mathbf{E}[d(\boldsymbol{x}, T(\boldsymbol{y}))] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} d(x, T(y)) f_{\boldsymbol{x},\boldsymbol{y}}(x, y)\, \mathrm{d}x\mathrm{d}y$$

where $d(\boldsymbol{x}, T(\boldsymbol{y}))$ denotes the distance between $\boldsymbol{x}$ and its estimate $T(\boldsymbol{y})$, according to a suitable metric.

Since the distance $d(\boldsymbol{x}, T(\boldsymbol{y}))$ is a random variable, the estimation problem can be formulated in terms of the minimization of its expected value. This boils down to finding the estimator $T^*(\cdot)$ such that

$$T^*(\cdot) = \arg\min_{T(\cdot)} J_r.$$

1.3 Minimum Mean Square Error Estimator

A standard choice for the distance $d(\cdot)$ is the *quadratic error*

$$d(\boldsymbol{x}, T(\boldsymbol{y})) = \|\boldsymbol{x} - T(\boldsymbol{y})\|^2 .$$

Definition 1.3. The *minimum Mean Square Error (MSE)* estimator is defined as $\hat{\boldsymbol{x}}_{MSE} = T^*(\cdot)$, where

$$T^*(\cdot) = \arg\min_{T(\cdot)} \mathbf{E}\left[\|\boldsymbol{x} - T(\boldsymbol{y})\|^2\right]. \tag{1.1}$$

Notice that in (1.1), the expected value is computed with respect to both random variables $\boldsymbol{x}$ and $\boldsymbol{y}$, and hence it is necessary to know the joint pdf $f_{\boldsymbol{x},\boldsymbol{y}}(x, y)$.

The following fundamental result provides the solution to the minimum MSE estimation problem.

Theorem 1.1. *Let $\boldsymbol{x}$ be a random variable and $\boldsymbol{y}$ a vector of observations. The minimum MSE estimator $\hat{\boldsymbol{x}}_{MSE}$ of $\boldsymbol{x}$ based on $\boldsymbol{y}$ is equal to the conditional expected value of $\boldsymbol{x}$ given $\boldsymbol{y}$:*

$$\hat{\boldsymbol{x}}_{MSE} = \mathbf{E}\left[\boldsymbol{x}|\boldsymbol{y}\right] = \int_{-\infty}^{+\infty} x f_{\boldsymbol{x}|\boldsymbol{y}}(x|y)\mathrm{d}x. \tag{1.2}$$

Proof

In order to solve problem (1.1), we need to find the estimator $T(\cdot)$ minimizing

$$\begin{aligned}
\mathbf{E}\left[\|\boldsymbol{x} - T(\boldsymbol{y})\|^2\right] &= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \|\boldsymbol{x} - T(\boldsymbol{y})\|^2 f_{\boldsymbol{x},\boldsymbol{y}}(x, y)\,\mathrm{d}x\mathrm{d}y \\
&= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \|\boldsymbol{x} - T(\boldsymbol{y})\|^2 f_{\boldsymbol{x}|\boldsymbol{y}}(x|y)\mathrm{d}x f_{\boldsymbol{y}}(y)\mathrm{d}y \\
&= \int_{-\infty}^{+\infty} \mathbf{E}\left[\|\boldsymbol{x} - T(\boldsymbol{y})\|^2 |\boldsymbol{y}\right] f_{\boldsymbol{y}}(y)\mathrm{d}y.
\end{aligned}$$

Since both $\mathbf{E}\left[\|\boldsymbol{x} - T(\boldsymbol{y})\|^2 |\boldsymbol{y}\right]$ and $f_{\boldsymbol{y}}(y)$ are nonnegative for every $y \in \mathbb{R}^n$, and the latter does not depend on $T(\cdot)$, problem (1.1) boils down to

$$T^*(\cdot) = \arg\min_{T(\cdot)} \mathbf{E}\left[\|\boldsymbol{x} - T(\boldsymbol{y})\|^2 |\boldsymbol{y}\right]. \tag{1.3}$$

Now, by exploiting the fact that $\mathbf{E}[T(\boldsymbol{y})|\boldsymbol{y}] = T(\boldsymbol{y})$, one has

$$\begin{aligned}
&\mathbf{E}\left[\|\boldsymbol{x} - T(\boldsymbol{y})\|^2 |\boldsymbol{y}\right] = \mathbf{E}\left[(\boldsymbol{x} - T(\boldsymbol{y}))^T(\boldsymbol{x} - T(\boldsymbol{y}))|\boldsymbol{y}\right] = \\
&= \mathbf{E}\left[\boldsymbol{x}^T\boldsymbol{x}|\boldsymbol{y}\right] - T(\boldsymbol{y})^T\mathbf{E}\left[\boldsymbol{x}|\boldsymbol{y}\right] - \mathbf{E}\left[\boldsymbol{x}|\boldsymbol{y}\right]^T T(\boldsymbol{y}) + T(\boldsymbol{y})^T T(\boldsymbol{y}) = \\
&= \mathbf{E}\left[\|\boldsymbol{x}\|^2|\boldsymbol{y}\right] - \|\mathbf{E}\left[\boldsymbol{x}|\boldsymbol{y}\right]\|^2 + \|\mathbf{E}\left[\boldsymbol{x}|\boldsymbol{y}\right] - T(\boldsymbol{y})\|^2
\end{aligned}$$

whose minimum is clearly achieved by choosing $T(\boldsymbol{y}) = \mathbf{E}[\boldsymbol{x}|\boldsymbol{y}]$. □

The previous result states that the estimator minimizing the MSE is the *a posteriori* expected value of $\boldsymbol{x}$, given the observation of $\boldsymbol{y}$, which is indeed a function of y. Since it is easy to prove that

$$\mathbf{E}\left[\mathbf{E}\left[\boldsymbol{x}|\boldsymbol{y}\right]\right] = \mathbf{E}\left[\boldsymbol{x}\right],$$

one can conclude that the minimum MSE estimator is always unbiased.

The minimum MSE estimator has other attractive properties. In particulare, if we consider the matrix

$$Q(\boldsymbol{x}, T(\boldsymbol{y})) = \mathbf{E}\left[(\boldsymbol{x} - T(\boldsymbol{y}))(\boldsymbol{x} - T(\boldsymbol{y}))^T\right],$$

it can be shown that:

- $\hat{\boldsymbol{x}}_{MSE}$ is the estimator minimizing (in matricial sense) $Q(\boldsymbol{x}, T(\boldsymbol{y}))$, i.e.

$$Q(\boldsymbol{x}, \hat{\boldsymbol{x}}_{MSE}) \leq Q(\boldsymbol{x}, T(\boldsymbol{y})), \quad \forall T(\boldsymbol{y})$$

- $\hat{\boldsymbol{x}}_{MSE}$ minimizes every monotonically increasing scalar function of matrix $Q(\boldsymbol{x}, T(\boldsymbol{y}))$, like for example the trace of Q, corresponding to the MSE, $\mathbf{E}\left[\|\boldsymbol{x} - T(\boldsymbol{y})\|^2\right]$.

The computation of the minimum MSE estimator may be difficult, or even intractable, in practical problems, because it requires the knowledge of the joint pdf $f_{\boldsymbol{x},\boldsymbol{y}}(x, y)$ and the computation of the integral (1.2).

Example 1.1. Consider two random variables $\boldsymbol{x}$ and $\boldsymbol{y}$, whose joint pdf is given by

$$f_{\boldsymbol{x},\boldsymbol{y}}(x, y) = \begin{cases} -\frac{3}{2}x^2 + 2xy & \text{if } 0 \leq x \leq 1, \quad 1 \leq y \leq 2 \\ 0 & \text{elsewhere} \end{cases}$$

Let us find the minimum MSE estimator of $\boldsymbol{x}$ based on one observation of $\boldsymbol{y}$. From Theorem 1.1, we know that

$$\hat{\boldsymbol{x}}_{MSE} = \int_{-\infty}^{+\infty} x f_{\boldsymbol{x}|\boldsymbol{y}}(x|y)\mathrm{d}x.$$

First, we need to compute

$$f_{\boldsymbol{x}|\boldsymbol{y}}(x|y) = \frac{f_{\boldsymbol{x},\boldsymbol{y}}(x,y)}{f_{\boldsymbol{y}}(y)}.$$

The marginal pdf of $\boldsymbol{y}$ can be calculated from the joint pdf as

$$\begin{aligned} f_{\boldsymbol{y}}(y) &= \int_0^1 -\frac{3}{2}x^2 + 2xy\mathrm{d}x \\ &= -\frac{x^3}{2} + yx^2\Big|_{x=0}^{x=1} = y - \frac{1}{2}. \end{aligned}$$

Hence, the conditional pdf is given by

$$f_{\boldsymbol{x}|\boldsymbol{y}}(x|y) = \begin{cases} \frac{-\frac{3}{2}x^2+2xy}{y-\frac{1}{2}} & \text{if } 0 \le x \le 1, \quad 1 \le y \le 2 \\ 0 & \text{elsewhere} \end{cases}$$

Now, it is possible to compute the minimum MSE estimator

$$\begin{aligned} \hat{\boldsymbol{x}}_{MSE} &= \int_0^1 x\frac{-\frac{3}{2}x^2 + 2xy}{y-\frac{1}{2}}\mathrm{d}x \\ &= \frac{1}{y-\frac{1}{2}}\left(-\frac{3}{8}x^4 + \frac{2}{3}x^3y\right)\Big|_{x=0}^{x=1} = \frac{\frac{2}{3}y - \frac{3}{8}}{y-\frac{1}{2}}. \end{aligned}$$

△

1.4 Linear Mean Square Error Estimator

We now restrict our attention to the class of affine linear estimators

$$T(\boldsymbol{y}) = A\boldsymbol{y} + b \tag{1.4}$$

in which the matrix $A \in \mathbb{R}^{m\times n}$ and the vector $b \in \mathbb{R}^m$ are the coefficients of the estimator to be determined. Among all estimators of the form (1.4), we aim at finding the one minimizing the MSE.

Definition 1.4. The *Linear Mean Square Error (LMSE)* estimator is defined as $\hat{\boldsymbol{x}}_{LMSE} = A^*\boldsymbol{y} + b^*$, where

$$A^*, b^* = \arg\min_{A,b} \mathbf{E}\left[\|\boldsymbol{x} - A\boldsymbol{y} - b\|^2\right]. \tag{1.5}$$

Theorem 1.2. *Let $\boldsymbol{x}$ be a random variable and $\boldsymbol{y}$ a vector of observations, such that*

$$\mathbf{E}[\boldsymbol{x}] = m_{\boldsymbol{x}}, \qquad \mathbf{E}[\boldsymbol{y}] = m_{\boldsymbol{y}}$$

$$\mathbf{E}\left[\begin{pmatrix}\boldsymbol{x} - m_{\boldsymbol{x}}\\ \boldsymbol{y} - m_{\boldsymbol{y}}\end{pmatrix}\begin{pmatrix}\boldsymbol{x} - m_{\boldsymbol{x}}\\ \boldsymbol{y} - m_{\boldsymbol{y}}\end{pmatrix}^T\right] = \begin{pmatrix}P_{\boldsymbol{x}} & P_{\boldsymbol{xy}}\\ P_{\boldsymbol{xy}}^T & P_{\boldsymbol{y}}\end{pmatrix}.$$

Then, the solution of problem (1.5) *is given by*

$$\begin{aligned} A^* &= P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1},\\ b^* &= m_{\boldsymbol{x}} - P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}m_{\boldsymbol{y}}. \end{aligned}$$

and hence, the LMSE estimator $\hat{\boldsymbol{x}}_{LMSE}$ of $\boldsymbol{x}$ is given by

$$\hat{\boldsymbol{x}}_{LMSE} = m_{\boldsymbol{x}} + P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}(\boldsymbol{y} - m_{\boldsymbol{y}}).$$

Proof

First, observe that the cost to be minimized is

$$\mathbf{E}\left[\|\boldsymbol{x} - A\boldsymbol{y} - b\|^2\right] = \operatorname{tr}\left(\mathbf{E}\left[(\boldsymbol{x} - A\boldsymbol{y} - b)(\boldsymbol{x} - A\boldsymbol{y} - b)^T\right]\right).$$

Since the trace is a monotonically increasing function, solving problem (1.5) is equivalent to find A^*, b^* such that

$$\mathbf{E}\left[(\boldsymbol{x} - A^*\boldsymbol{y} - b^*)(\boldsymbol{x} - A^*\boldsymbol{y} - b^*)^T\right] \leq \mathbf{E}\left[(\boldsymbol{x} - A\boldsymbol{y} - b)(\boldsymbol{x} - A\boldsymbol{y} - b)^T\right] \ \forall A, b \tag{1.6}$$

Therefore, by denoting the estimation error as $\tilde{\boldsymbol{x}} = \boldsymbol{x} - A\boldsymbol{y} - b$, one gets

$$\begin{aligned}
\mathbf{E}\left[\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^T\right] &= \mathbf{E}\left[(\boldsymbol{x} - m_{\boldsymbol{x}} - A(\boldsymbol{y} - m_{\boldsymbol{y}}) + m_{\boldsymbol{x}} - Am_{\boldsymbol{y}} - b)\right. \\
&\qquad \left. \times (\boldsymbol{x} - m_{\boldsymbol{x}} - A(\boldsymbol{y} - m_{\boldsymbol{y}}) + m_{\boldsymbol{x}} - Am_{\boldsymbol{y}} - b)^T\right] \\
&= P_{\boldsymbol{x}} + AP_{\boldsymbol{y}}A^T - P_{\boldsymbol{xy}}A^T - AP_{\boldsymbol{yx}} \\
&\qquad + (m_{\boldsymbol{x}} - Am_{\boldsymbol{y}} - b)(m_{\boldsymbol{x}} - Am_{\boldsymbol{y}} - b)^T \\
&= P_{\boldsymbol{x}} + AP_{\boldsymbol{y}}A^T - P_{\boldsymbol{xy}}A^T - AP_{\boldsymbol{xy}}^T + P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}P_{\boldsymbol{xy}}^T - P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}P_{\boldsymbol{xy}}^T \\
&\qquad + (m_{\boldsymbol{x}} - Am_{\boldsymbol{y}} - b)(m_{\boldsymbol{x}} - Am_{\boldsymbol{y}} - b)^T \\
&= P_{\boldsymbol{x}} - P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}P_{\boldsymbol{xy}}^T + \left(P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1} - A\right)P_{\boldsymbol{y}}\left(P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1} - A\right)^T \\
&\qquad + (m_{\boldsymbol{x}} - Am_{\boldsymbol{y}} - b)(m_{\boldsymbol{x}} - Am_{\boldsymbol{y}} - b)^T. \qquad (1.7)
\end{aligned}$$

Observe that the last two terms of the previous expression are positive semidefinite matrices. Hence, the solution of problem (1.6) is obtained by choosing A^*, b^* such that the last two terms are equal to zero, i.e.

$$\begin{aligned}
A^* &= P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}; \\
b^* &= m_{\boldsymbol{x}} - Am_{\boldsymbol{y}} = m_{\boldsymbol{x}} - P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}m_{\boldsymbol{y}}.
\end{aligned}$$

This concludes the proof. □

The LMSE estimator is unbiased because the expected value of the estimation error is equal to zero. In fact,

$$\begin{aligned}
\mathbf{E}\left[\tilde{\boldsymbol{x}}\right] &= \mathbf{E}\left[\boldsymbol{x} - \hat{\boldsymbol{x}}_{LMSE}\right] = m_{\boldsymbol{x}} - \mathbf{E}\left[m_{\boldsymbol{x}} + P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}(\boldsymbol{y} - m_{\boldsymbol{y}})\right] \\
&= m_{\boldsymbol{x}} - m_{\boldsymbol{x}} + P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}\mathbf{E}\left[\boldsymbol{y} - m_{\boldsymbol{y}}\right] = 0.
\end{aligned}$$

By setting $A = A^*$ and $b = b^*$ in the last expression in (1.7), we can compute the variance of the estimation error of the LMSE estimator, which is equal to

$$\mathbf{E}\left[\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^T\right] = P_{\boldsymbol{x}} - P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}P_{\boldsymbol{xy}}^T.$$

It is worth noting that, by interpreting $P_{\boldsymbol{x}}$ as the *a priori* uncertainty on $\boldsymbol{x}$, $P_{\boldsymbol{x}} - P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}P_{\boldsymbol{xy}}^T$ represents the new uncertainty on $\boldsymbol{x}$ after having observed

the measurement $\boldsymbol{y}$. Since the matrix $P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}P_{\boldsymbol{xy}}^{T}$ is always positive semidefinite, the effect of the observations is that to reduce the uncertainty on $\boldsymbol{x}$. Moreover, such a reduction depends on the size of $P_{\boldsymbol{xy}}$, i.e., on the correlation between the measurement $\boldsymbol{y}$ and the unknown $\boldsymbol{x}$ (notice that there is no uncertainty reduction when $P_{\boldsymbol{xy}} = 0$, as expected).

It is worth stressing that in order to compute the LMSE estimator it is not necessary to know the joint pdf $f_{\boldsymbol{x},\boldsymbol{y}}(x, y)$, but only the first and second order statistics $m_{\boldsymbol{x}}, m_{\boldsymbol{y}}, P_{\boldsymbol{x}}, P_{\boldsymbol{y}}, P_{\boldsymbol{xy}}$.

An interesting property of the LMSE estimator is that the estimation error $\tilde{\boldsymbol{x}}$ is uncorrelated to the observations $\boldsymbol{y}$. In fact, one has

$$\begin{aligned}\mathbf{E}\left[\tilde{\boldsymbol{x}}\boldsymbol{y}^{T}\right] &= \mathbf{E}\left[\left(\boldsymbol{x} - m_{\boldsymbol{x}} - P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}(\boldsymbol{y} - m_{\boldsymbol{y}})\right)\boldsymbol{y}^{T}\right] \\ &= P_{\boldsymbol{xy}} - P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}P_{\boldsymbol{y}} = 0.\end{aligned} \tag{1.8}$$

This result is often known as *orthogonality principle*. Conversely, it is possible to show that if a linear estimator satisfies the orthogonality condition $\mathbf{E}\left[\tilde{\boldsymbol{x}}\boldsymbol{y}^{T}\right] = 0$, then it is the LMSE estimator.

Example 1.2. Let $\boldsymbol{y}_1, \boldsymbol{y}_2$ be two noisy observations of the scalar random variable $\boldsymbol{x}$, having mean $m_{\boldsymbol{x}}$ and variance $\sigma_{\boldsymbol{x}}^2$:

$$\begin{aligned}\boldsymbol{y}_1 &= \boldsymbol{x} + \boldsymbol{\varepsilon}_1 \ , \\ \boldsymbol{y}_2 &= \boldsymbol{x} + \boldsymbol{\varepsilon}_2 \ .\end{aligned}$$

Let $\boldsymbol{\varepsilon}_1, \boldsymbol{\varepsilon}_2$ be two independent random variables, with zero mean and variance $\sigma_{\boldsymbol{1}}^2, \sigma_{\boldsymbol{2}}^2$, respectively. Under the assumption that $\boldsymbol{x}$ and $\boldsymbol{\varepsilon}_i$, $i = 1, 2$, are independent, we aim at computing the LMSE estimator of $\boldsymbol{x}$.

Define the vectors $\boldsymbol{y} = (\boldsymbol{y}_1 \ \boldsymbol{y}_2)^T$ and $\boldsymbol{\varepsilon} = (\boldsymbol{\varepsilon}_1 \ \boldsymbol{\varepsilon}_2)^T$, and rewrite the measurement equations in the form

$$\boldsymbol{y} = \mathbb{1}\,\boldsymbol{x} + \boldsymbol{\varepsilon},$$

where $\mathbb{1} = (1 \ 1)^T$.

First, let us compute the mean of $\boldsymbol{y}$

$$\mathbf{E}[\boldsymbol{y}] = \mathbf{E}[\mathbb{1}\,\boldsymbol{x} + \boldsymbol{\varepsilon}] = \mathbb{1}\,m_{\boldsymbol{x}}$$

In order to find the estimate $\hat{\boldsymbol{x}}_{LMSE}$, we have to compute the covariance matrices $P_{\boldsymbol{xy}}$ and $P_{\boldsymbol{y}}$. We get

$$P_{\boldsymbol{xy}} = \mathbf{E}\left[(\boldsymbol{x} - m_{\boldsymbol{x}})\left(\mathbb{1}\left(x - m_{\boldsymbol{x}}\right) + \boldsymbol{\varepsilon}\right)^T\right] = \sigma_{\boldsymbol{x}}^2 \mathbb{1}^T,$$

because $\boldsymbol{x}$ and $\boldsymbol{\varepsilon}$ are uncorrelated. Moreover,

$$\begin{aligned} P_{\boldsymbol{y}} &= \mathbf{E}\left[\left(\mathbb{1}\left(x - m_{\boldsymbol{x}}\right) + \boldsymbol{\varepsilon}\right)\left(\mathbb{1}\left(x - m_{\boldsymbol{x}}\right) + \boldsymbol{\varepsilon}\right)^T\right] \\ &= \mathbb{1}\,\sigma_{\boldsymbol{x}}^2\,\mathbb{1}^T + P_{\boldsymbol{\varepsilon}}, \end{aligned}$$

where

$$P_{\boldsymbol{\varepsilon}} = \begin{pmatrix} \sigma_{\mathbf{1}}^2 & 0 \\ 0 & \sigma_{\mathbf{2}}^2 \end{pmatrix}$$

is the covariance matrix of $\boldsymbol{\varepsilon}$. Finally, let us compute the inverse of the measurement covariance matrix

$$\begin{aligned} P_{\boldsymbol{y}}^{-1} &= \left[\sigma_{\boldsymbol{x}}^2 \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} + \begin{pmatrix} \sigma_{\mathbf{1}}^2 & 0 \\ 0 & \sigma_{\mathbf{2}}^2 \end{pmatrix}\right]^{-1} \\ &= \begin{pmatrix} \sigma_{\boldsymbol{x}}^2 + \sigma_{\mathbf{1}}^2 & \sigma_{\boldsymbol{x}}^2 \\ \sigma_{\boldsymbol{x}}^2 & \sigma_{\boldsymbol{x}}^2 + \sigma_{\mathbf{2}}^2 \end{pmatrix}^{-1} \\ &= \frac{1}{\sigma_{\boldsymbol{x}}^2(\sigma_{\mathbf{1}}^2 + \sigma_{\mathbf{2}}^2) + \sigma_{\mathbf{1}}^2\sigma_{\mathbf{2}}^2} \begin{pmatrix} \sigma_{\boldsymbol{x}}^2 + \sigma_{\mathbf{2}}^2 & -\sigma_{\boldsymbol{x}}^2 \\ -\sigma_{\boldsymbol{x}}^2 & \sigma_{\boldsymbol{x}}^2 + \sigma_{\mathbf{1}}^2 \end{pmatrix}. \end{aligned}$$

Hence, the LMSE estimator is given by

$$\begin{aligned} \hat{\boldsymbol{x}}_{LMSE} &= m_{\boldsymbol{x}} + P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1}(\boldsymbol{y} - \mathbb{1}\, m_{\boldsymbol{x}}) \\ &= m_{\boldsymbol{x}} + \sigma_{\boldsymbol{x}}^2 \mathbb{1}^T P_{\boldsymbol{y}}^{-1}(\boldsymbol{y} - \mathbb{1}\, m_{\boldsymbol{x}}) \\ &= m_{\boldsymbol{x}} + \frac{\sigma_{\boldsymbol{x}}^2}{\sigma_{\boldsymbol{x}}^2(\sigma_{\mathbf{1}}^2 + \sigma_{\mathbf{2}}^2) + \sigma_{\mathbf{1}}^2\sigma_{\mathbf{2}}^2}(1\ 1) \begin{pmatrix} \sigma_{\boldsymbol{x}}^2 + \sigma_{\mathbf{2}}^2 & -\sigma_{\boldsymbol{x}}^2 \\ -\sigma_{\boldsymbol{x}}^2 & \sigma_{\boldsymbol{x}}^2 + \sigma_{\mathbf{1}}^2 \end{pmatrix} \begin{pmatrix} \boldsymbol{y}_1 - m_{\boldsymbol{x}} \\ \boldsymbol{y}_2 - m_{\boldsymbol{x}} \end{pmatrix} \\ &= m_{\boldsymbol{x}} + \frac{1}{\sigma_{\mathbf{1}}^2 + \sigma_{\mathbf{2}}^2 + \frac{\sigma_{\mathbf{1}}^2\sigma_{\mathbf{2}}^2}{\sigma_{\boldsymbol{x}}^2}}(\sigma_{\mathbf{2}}^2\ \sigma_{\mathbf{1}}^2) \begin{pmatrix} \boldsymbol{y}_1 - m_{\boldsymbol{x}} \\ \boldsymbol{y}_2 - m_{\boldsymbol{x}} \end{pmatrix} \\ &= m_{\boldsymbol{x}} + \frac{\sigma_{\mathbf{2}}^2\boldsymbol{y}_1 + \sigma_{\mathbf{1}}^2\boldsymbol{y}_2 - m_{\boldsymbol{x}}(\sigma_{\mathbf{1}}^2 + \sigma_{\mathbf{2}}^2)}{\sigma_{\mathbf{1}}^2 + \sigma_{\mathbf{2}}^2 + \frac{\sigma_{\mathbf{1}}^2\sigma_{\mathbf{2}}^2}{\sigma_{\boldsymbol{x}}^2}} \end{aligned}$$

$$= \frac{\frac{m_{\boldsymbol{x}}\sigma_{\mathbf{1}}^2\sigma_{\mathbf{2}}^2}{\sigma_{\boldsymbol{x}}^2} + \sigma_{\mathbf{2}}^2\boldsymbol{y}_1 + \sigma_{\mathbf{1}}^2\boldsymbol{y}_2}{\sigma_{\mathbf{1}}^2 + \sigma_{\mathbf{2}}^2 + \frac{\sigma_{\mathbf{1}}^2\sigma_{\mathbf{2}}^2}{\sigma_{\boldsymbol{x}}^2}} = \frac{\frac{m_{\boldsymbol{x}}}{\sigma_{\boldsymbol{x}}^2} + \frac{1}{\sigma_{\mathbf{1}}^2}\boldsymbol{y}_1 + \frac{1}{\sigma_{\mathbf{2}}^2}\boldsymbol{y}_2}{\frac{1}{\sigma_{\boldsymbol{x}}^2} + \frac{\sigma_{\mathbf{1}}^2+\sigma_{\mathbf{2}}^2}{\sigma_{\mathbf{1}}^2\sigma_{\mathbf{2}}^2}}$$
$$= \frac{\frac{m_{\boldsymbol{x}}}{\sigma_{\boldsymbol{x}}^2} + \frac{1}{\sigma_{\mathbf{1}}^2}\boldsymbol{y}_1 + \frac{1}{\sigma_{\mathbf{2}}^2}\boldsymbol{y}_2}{\frac{1}{\sigma_{\boldsymbol{x}}^2} + \frac{1}{\sigma_{\mathbf{1}}^2} + \frac{1}{\sigma_{\mathbf{2}}^2}}.$$

Notice that each measurement is weighted with a weight that is inversely proportional to the variance of the noise affecting the measurement. Moreover, the *a priori* information on $\boldsymbol{x}$ (i.e., its mean $m_{\boldsymbol{x}}$ and variance $\sigma_{\boldsymbol{x}}^2$), is treated as an additional observation of $\boldsymbol{x}$. In particular, it is interesting to observe that if $\sigma_x^2 \to +\infty$ (i.e., the *a priori* information on $\boldsymbol{x}$ is completely unreliable), the estimate $\hat{\boldsymbol{x}}_{LMSE}$ takes on the same form of the Gauss-Markov parametric estimate of the mean m_x. This highlights the relationship between Bayesian and parametric estimation. △

1.5 Bayesian estimation in the Gaussian framework

Now let us consider the case in which the random variables $\boldsymbol{x}, \boldsymbol{y}$ are jointly Gaussian, with mean and covariance matrix defined as in Theorem 1.2. This means that the joint pdf of $\boldsymbol{x}$ and $\boldsymbol{y}$ is given by

$$f_{\boldsymbol{\xi}}(\xi) = \frac{1}{(2\pi)^{(n+m)/2}(\det P_{\boldsymbol{\xi}})^{1/2}} e^{-\frac{1}{2}(\xi - m_{\boldsymbol{\xi}})^T P_{\boldsymbol{\xi}}^{-1}(\xi - m_{\boldsymbol{\xi}})}, \tag{1.9}$$

where

$$\boldsymbol{\xi} = \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix}, \quad m_{\boldsymbol{\xi}} = \begin{pmatrix} m_{\boldsymbol{x}} \\ m_{\boldsymbol{y}} \end{pmatrix}, \quad P_{\boldsymbol{\xi}} = \begin{pmatrix} P_{\boldsymbol{x}} & P_{\boldsymbol{xy}} \\ P_{\boldsymbol{xy}}^T & P_{\boldsymbol{y}} \end{pmatrix}. \tag{1.10}$$

The following result holds.

Theorem 1.3. *Let $\boldsymbol{x}$, $\boldsymbol{y}$ be distributed according to* (1.9)-(1.10)*. Then, the conditional expected value of $\boldsymbol{x}$ given the observation of $\boldsymbol{y}$ is given by*

$$\mathbf{E}[\boldsymbol{x}|\boldsymbol{y}] = m_{\boldsymbol{x}} + P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1}(\boldsymbol{y} - m_{\boldsymbol{y}}).$$

Moreover, the minimum covariance matrix of the estimation error achievable by any Bayesian estimator is equal to $P_{\boldsymbol{x}} - P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1} P_{\boldsymbol{xy}}^T$.

Proof

Let us introduce the new variable

$$\hat{\boldsymbol{x}} = \boldsymbol{x} - P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1} (\boldsymbol{y} - m_{\boldsymbol{y}}).$$

One has that $\mathbf{E}[\hat{\boldsymbol{x}}] = \mathbf{E}[\boldsymbol{x}] = m_{\boldsymbol{x}}$, and

$$\begin{aligned} P_{\hat{\boldsymbol{x}}} &= \mathbf{E}\left[(\hat{\boldsymbol{x}} - m_{\boldsymbol{x}})(\hat{\boldsymbol{x}} - m_{\boldsymbol{x}})^T\right] \\ &= \mathbf{E}\left[(\boldsymbol{x} - m_{\boldsymbol{x}} - P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1}(\boldsymbol{y} - m_{\boldsymbol{y}}))(\boldsymbol{x} - m_{\boldsymbol{x}} - P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1}(\boldsymbol{y} - m_{\boldsymbol{y}}))^T\right] \\ &= P_{\boldsymbol{x}} - P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1} P_{\boldsymbol{xy}}^T - P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1} P_{\boldsymbol{xy}}^T + P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1} P_{\boldsymbol{y}} P_{\boldsymbol{y}}^{-1} P_{\boldsymbol{xy}}^T \\ &= P_{\boldsymbol{x}} - P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1} P_{\boldsymbol{xy}}^T. \end{aligned}$$

Moreover, if we compute the cross-covariance between $\hat{\boldsymbol{x}}$ and $\boldsymbol{y}$, we obtain

$$P_{\hat{\boldsymbol{x}}\boldsymbol{y}} = P_{\boldsymbol{xy}} - P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1} P_{\boldsymbol{y}} = 0,$$

meaning that $\hat{\boldsymbol{x}}$ and $\boldsymbol{y}$ are uncorrelated. Since $\hat{\boldsymbol{x}}$ is by definition a linear combination of Gaussian random variables, it is itself a Gaussian random variable. Therefore, $\hat{\boldsymbol{x}}$ and $\boldsymbol{y}$ are also independent (being both jointly Gaussian and uncorrelated), and one can write their joint pdf as

$$f_{\hat{\boldsymbol{x}},\boldsymbol{y}}(\hat{x}, y) = f_{\hat{\boldsymbol{x}}}(\hat{x})\, f_{\boldsymbol{y}}(y). \tag{1.11}$$

Now, observe that the pairs of random variables $(\boldsymbol{x}, \boldsymbol{y})$ and $(\hat{\boldsymbol{x}}, \boldsymbol{y})$ satisfy the linear relationship

$$\begin{pmatrix} \boldsymbol{x} - m_{\boldsymbol{x}} \\ \boldsymbol{y} - m_{\boldsymbol{y}} \end{pmatrix} = \begin{pmatrix} I & P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{x}} - m_{\boldsymbol{x}} \\ \boldsymbol{y} - m_{\boldsymbol{y}} \end{pmatrix}$$

in which the transformation matrix has determinant equal to 1. Hence, one gets

$$\begin{aligned} f_{\boldsymbol{x},\boldsymbol{y}}(x, y) &= f_{\hat{\boldsymbol{x}},\boldsymbol{y}}\left(x - P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1}(y - m_{\boldsymbol{y}}), y\right) \\ &= f_{\hat{\boldsymbol{x}}}\left(x - P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1}(y - m_{\boldsymbol{y}})\right) f_{\boldsymbol{y}}(y) \end{aligned}$$

where the last equality comes from (1.11). Then, we can write the conditional pdf of $\boldsymbol{x}$ given $\boldsymbol{y}$ as

$$\begin{aligned} f_{\boldsymbol{x}|\boldsymbol{y}}(x|y) &= \frac{f_{\boldsymbol{x},\boldsymbol{y}}(x,y)}{f_{\boldsymbol{y}}(y)} = \frac{f_{\hat{\boldsymbol{x}}}\left(x - P_{\boldsymbol{x}\boldsymbol{y}}P_{\boldsymbol{y}}^{-1}(y - m_{\boldsymbol{y}})\right) f_{\boldsymbol{y}}(y)}{f_{\boldsymbol{y}}(y)} \\ &= f_{\hat{\boldsymbol{x}}}\left(x - P_{\boldsymbol{x}\boldsymbol{y}}P_{\boldsymbol{y}}^{-1}(y - m_{\boldsymbol{y}})\right) \\ &= \frac{1}{(2\pi)^{n/2}(\det P_{\hat{\boldsymbol{x}}})^{1/2}} e^{-\frac{1}{2}(x - P_{\boldsymbol{x}\boldsymbol{y}}P_{\boldsymbol{y}}^{-1}(y-m_{\boldsymbol{y}}) - m_{\hat{\boldsymbol{x}}})^T P_{\hat{\boldsymbol{x}}}^{-1}(x - P_{\boldsymbol{x}\boldsymbol{y}}P_{\boldsymbol{y}}^{-1}(y-m_{\boldsymbol{y}}) - m_{\hat{\boldsymbol{x}}})}. \end{aligned}$$

By recalling the mean and covariance matrix of $\hat{\boldsymbol{x}}$, one immediately has the result. □

According to Theorem 1.3, we can conclude that: *if $\boldsymbol{x}, \boldsymbol{y}$ are Gaussian variables, the LMSE estimator coincides with the minimum MSE estimator, i.e.,* $\hat{\boldsymbol{x}}_{MSE} = \hat{\boldsymbol{x}}_{LMSE}$. In other words, in the Gaussian case the minimum MSE estimator is a linear function of the observations $\boldsymbol{y}$.

1.6 Exercises

1.1. Consider two random variables $\boldsymbol{x}$ and $\boldsymbol{y}$, whose joint pdf is

$$f_{\boldsymbol{x},\boldsymbol{y}}(x,y) = \begin{cases} -\frac{3}{2}x^2 + 2xy & 0 \le x \le 1,\ 1 \le y \le 2 \\ 0 & \text{elsewhere} \end{cases}$$

Find the LMSE estimate $\hat{\boldsymbol{x}}_{LMSE}$ of $\boldsymbol{x}$, based on one observation of $\boldsymbol{y}$.
Plot the estimate $\hat{\boldsymbol{x}}_{LMSE}$ computed above and the minimum MSE estimate $\hat{\boldsymbol{x}}_{MSE}$ derived in Example 1.1, as functions of y (the realization of $\boldsymbol{y}$).
Compute the expected values of both estimates and compare them with the *a priori* mean $\mathbf{E}[\boldsymbol{x}]$.

1.2. Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be two random variables with joint pdf

$$f_{\boldsymbol{x},\boldsymbol{y}}(x,y) = \begin{cases} \frac{1}{12}(x+y)e^{-y} & 0 \le x \le 4,\quad y \ge 0 \\ 0 & \text{elsewhere} \end{cases}$$

Assume that an observation y of $\boldsymbol{y}$ is available.

a) Find the estimators $\hat{\boldsymbol{x}}_{MSE}$ and $\hat{\boldsymbol{x}}_{LMSE}$ of $\boldsymbol{x}$, and plot them as functions of the observation y.

b) Compute the MSE of the estimators obtained in item a) [Hint: use MATLAB to compute the integrals].

1.3. Let $\boldsymbol{x}$ be a random variable and assume the following measurement is available

$$\boldsymbol{y} = \ln\left(\frac{1}{\boldsymbol{x}}\right) + \boldsymbol{v}$$

where $\boldsymbol{v}$ is a random variable, whose pdf is given by $f_{\boldsymbol{v}}(v) = \begin{cases} e^{-v} & v \geq 0 \\ 0 & v < 0 \end{cases}$.

a) Assume that $\boldsymbol{x}$ is independent from $\boldsymbol{v}$, and its *a priori* pdf is given by

$$f_{\boldsymbol{x}}(x) = \begin{cases} 1 & 0 \leq x \leq 1 \\ 0 & \text{altrimenti} \end{cases}.$$

Find the MSE and LMSE estimators of $\boldsymbol{x}$.

b) Plot the estimates obtained in items a) as functions of the observation y of $\boldsymbol{y}$.

Chapter 2

State estimation for linear systems

This chapter addresses the problem of state estimation for linear stochastic systems. First, the general framework of stochastic dynamic systems is introduced and the state estimation problem is formulated. In the second part of the chapter, the solution of the state estimation problem for linear systems is derived and its properties are discussed in detail.

2.1 Notation

Let us briefly recall some basic notations that will be used in this chapter. A *discrete-time stochastic process* (herafter, often abbreviated as s.p.) is a sequence of random variables $x(t)$, with $t \in \{t_1, t_2, \dots, t_k, \dots\}$. For simplicity of notation, we will write $x(k)$ instead of $x(t_k)$, with $k \in \mathbb{Z}$. Hence, summing up, the s.p. $x(t)$ is a sequence of real random variables, indexed by the discrete time $t \in \mathbb{Z}$. The pdf of a s.p. $x(t)$ is denoted by $f_{\boldsymbol{x}}(x(t))$. Its mean and covariance function are defined as $m_{\boldsymbol{x}}(t) = \mathbf{E}[x(t)]$ and $R_{\boldsymbol{x}}(t,s) = \mathbf{E}\left[(x(t) - m_{\boldsymbol{x}}(t))(x(s) - m_{\boldsymbol{x}}(s))^T\right]$, respectively. For a (weakly) stationary stochastic process, the mean is constant , $m_{\boldsymbol{x}}(t) = m_{\boldsymbol{x}}\ \forall t$, and the covariance function depends only on the lag between the time instants $\tau = t - s$, i.e.,

$R_{\boldsymbol{x}}(t,s) = R_{\boldsymbol{x}}(t-s) = R_{\boldsymbol{x}}(\tau)$

A *white stochastic process* is a sequence of independent random variables, hence it is completely defined by its mean $m_{\boldsymbol{x}}(t)$ and its variance matrix $R_{\boldsymbol{x}}(t,t)$ (being $R_{\boldsymbol{x}}(t,s) = 0$ for $t \neq s$, due to independence). If $m_{\boldsymbol{x}}(t) = m_{\boldsymbol{x}}$ and $R_{\boldsymbol{x}}(t,t) = R_{\boldsymbol{x}}$, $\forall t$, then the white process is also stationary. We briefly denote a white stochastic process as $x(t) \sim WP(m_{\boldsymbol{x}}, R_{\boldsymbol{x}})$.

2.2 State space representation of stochastic systems

The *state variables* (or simply *the state*) of a dynamic system are the variables that one needs to know at a generic time t, to determine the evolution of the system at all future times $\tau > t$, provided that the input signal $u(\tau)$, $\tau > t$, is known. For deterministic discrete-time systems, this allows one to write the input-state-output (i/s/o) representation

$$\begin{cases} x(t+1) = f(x(t), u(t), t) \\ y(t) = h(x(t), u(t), t) \end{cases} \tag{2.1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is the input and $y(t) \in \mathbb{R}^p$ is the output. For linear time-invariant (LTI) systems, model (2.1) boils down to

$$\begin{cases} x(t+1) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}$$

where A, B, C, D are constant matrices of suitable dimensions.

When the involved signals are stochastic processes, the system becomes a *stochastic system* and the state can be interpreted in terms of an important class of stochastic processes.

Definition 2.1 (Markov process)**.** The s.p. $x(t)$ is a Markov process if, at every time $t \in \mathbb{Z}$, the conditional pdf of $x(t)$ is such that

$$f_{\boldsymbol{x}}\left(x(t+1) | x(t), x(t-1), \cdots, x(1)\right) = f_{\boldsymbol{x}}\left(x(t+1) | x(t)\right)$$

The above definition states that for a Markov process the conditional pdf with respect to past observations is equal to the conditional pdf with respect to the most recent one. In other words, it is not necessary to keep track of all past observations, because the state vector $x(t_k)$ contains all the necessary information to compute the a posteriori pdf of the next state $x(t_{k+1})$.

Hereafter, we consider the following general class of systems:

$$\begin{cases} x(t+1) = f(x(t), u(t), w(t)) \\ y(t) = h(x(t), u(t)) + v(t) \end{cases} \tag{2.2}$$

where $w(t) \in \mathbb{R}^d$ and $v(t) \in \mathbb{R}^p$ are stochastic processes, on which we make the following assumption.

Assumption 2.1. *For system* (2.2)*, we assume that:*

i) $\mathbf{E}\left[\begin{pmatrix} w(t) \\ v(t) \end{pmatrix}\right] = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$;

ii) $\mathbf{E}\left[\begin{pmatrix} w(t) \\ v(t) \end{pmatrix}\begin{pmatrix} w(t) \\ v(t) \end{pmatrix}^T\right] = \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix}$ *where* Q *and* R *are the covariance matrices of* $w(t)$ *and* $v(t)$*, respectively;*

iii) $w(t)$ *and* $v(t)$ *are independent white processes;*

iv) $x(0)$ *is a random vector, independent from* $w(t)$ *and* $v(t)$*, with mean* m_0 *and covariance matrix* P_0*;*

v) $u(t)$ *is a deterministic (known) signal.*

The s.p. $w(t)$ is often referred to as *disturbance process* and models the stochastic component of the dynamic model $f(\cdot)$ (unmodeled dynamics, disturbances, etc.). The s.p. $v(t)$ is the so-called *measurement noise*, which represents the error of the sensor measuring the output function $h(\cdot)$. It is easy to see that under Assumption 2.1, the state $x(t)$ of model (2.2) is a Markov process.

For an LTI system, model (2.2) can be written as

$$\begin{cases} x(t+1) = Ax(t) + Bu(t) + Gw(t) \\ y(t) = Cx(t) + Du(t) + v(t) \end{cases} \tag{2.3}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $G \in \mathbb{R}^{n\times d}$, $C \in \mathbb{R}^{p\times n}$, $D \in \mathbb{R}^{p\times m}$, and

$$\begin{pmatrix} w(t) \\ v(t) \end{pmatrix} \sim WP\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} Q & 0 \\ 0 & R \end{pmatrix}\right) \tag{2.4}$$

with $Q \in \mathbb{R}^{d\times d}$, $R \in \mathbb{R}^{p\times p}$. Hereafter, it is assumed for simplicity $D = 0$, as it occurs in many real systems[1]. We will refer to the 6-tuple $\mathcal{M} = \{A, B, G, C, Q, R\}$ as the *model* of the stochastic system (2.3)-(2.4).

Let us now state a result concerning the mean and covariance functions of the processes $x(t)$ and $y(t)$, defined by system (2.3)-(2.4).

Theorem 2.1. *Consider system* (2.3) *and assume for simplicity* $D = 0$. *Then, under Assumption 2.1,one has*

$$m_{\boldsymbol{x}}(t) = A^t m_0 + \sum_{i=0}^{t-1} A^{t-1-i} B u(i) \tag{2.5}$$

$$m_{\boldsymbol{y}}(t) = C m_{\boldsymbol{x}}(t) \tag{2.6}$$

$$R_{\boldsymbol{x}}(t+\tau, t) = A^\tau R_x(t,t) \triangleq A^\tau P(t) \tag{2.7}$$

$$R_{\boldsymbol{y}}(t+\tau, t) = \begin{cases} CA^\tau P(t) C^T & \text{if } \tau > 0 \\ CP(t)C^T + R & \text{if } \tau = 0 \end{cases} \tag{2.8}$$

where

$$P(t+1) = AP(t)A^T + GQG^T. \tag{2.9}$$

Proof

By taking the expected value of the first equation in (2.3), one gets

$$\mathbf{E}\left[x(t+1)\right] = A\mathbf{E}\left[x(t)\right] + Bu(t) + G\mathbf{E}\left[w(t)\right]$$

[1]The extension to the case $D \neq 0$ is trivial, as one can replace $y(t)$ with $y(t) - Du(t)$, being $u(t)$ known.

which results in

$$m_{\boldsymbol{x}}(t+1) = Am_{\boldsymbol{x}}(t) + Bu(t). \tag{2.10}$$

Then, (2.5) follows from the total response of the LTI deterministic system (2.10). Similarly, one obtains (2.6).

Let us define $\tilde{x}(t) = x(t) - m_{\boldsymbol{x}}(t)$. By using (2.10), one has

$$\begin{aligned}
R_{\boldsymbol{x}}(t+\tau,t) =& \mathbf{E}\left[(x(t+\tau) - m_{\boldsymbol{x}}(t+\tau))(x(t) - m_{\boldsymbol{x}}(t))^T\right] = \\
=& \mathbf{E}\left[(Ax(t+\tau-1) + Gw(t+\tau-1) - Am_{\boldsymbol{x}}(t+\tau-1))\tilde{x}(t)^T\right] = \\
=& \mathbf{E}\left[(A\tilde{x}(t+\tau-1) + Gw(t+\tau-1))\tilde{x}(t)^T\right] = \\
=& A\mathbf{E}\left[\tilde{x}(t+\tau-1)\tilde{x}(t)^T\right] + G\underbrace{\mathbf{E}\left[w(t+\tau-1)\tilde{x}(t)^T\right]}_{0 \text{ for } \tau\geq 1} = \\
=& AR_x(t+\tau-1,t)
\end{aligned}$$

where we have exploited the fact that $w(t)$ is a white process and hence $\tilde{x}(t)$, which depends on samples of w up to time $t-1$, is uncorrelated with $w(t+\tau-1)$, $\forall\tau > 0$. By iterating backwards, one gets

$$\begin{aligned}
R_x(t+\tau,t) =& AR_x(t+\tau-1,t) = \\
=& A^2R_x(t+\tau-2,t) = \\
& \vdots \\
=& A^\tau R_x(t,t), \quad \text{for } \tau \geq 1.
\end{aligned}$$

For $\tau = 0$ one has

$$\begin{aligned}
P(t+1) = R_x(t+1,t+1) &= \mathbf{E}\left[\tilde{x}(t+1)\tilde{x}(t+1)^T\right] = \\
&= \mathbf{E}\left[(A\tilde{x}(t) + Gw(t))(A\tilde{x}(t) + Gw(t))^T\right] = \\
&= AP(t)A^T + GQG^T
\end{aligned}$$

where once again we exploited $\mathbf{E}\left[\tilde{x}(t)w(t)^T\right] = 0$.

Finally, (2.8) can be proven in the same way, by exploiting (2.7) and the fact that $v(t)$ is uncorrelated with $w(t)$ and $x(0)$, and hence also with $\tilde{x}(t)$. □

Remark 2.1. It is worth stressing that all the results in Theorem 2.1 can be easily extended to the case of a linear time-varying system, in which the matrices of model $\mathcal{M}$ change with time. For example, if the first equation in (2.3) is $x(t+1) = A(t)x(t) + G(t)w(t)$ (we set $B = 0$ for simplicity), (2.5) becomes

$$m_{\boldsymbol{x}}(t) = A(t-1)A(t-2)\cdots A(0)m_0 = \prod_{i=0}^{t-1} A(i)\, m_0$$

The other equations can be amended in a similar way.

The matrix $P(t) = R_{\boldsymbol{x}}(t,t)$ is the autocovariance matrix of process $x(t)$ and equation (2.9), which describes its evolution in time, is called *recursive Lyapunov equation*. The fact that both $m_{\boldsymbol{x}}(t)$ and $P(t)$ are not constant in time implies that in general both $x(t)$ and $y(t)$ are not stationary process. The following result provides conditions under which they are asymptotically stationary.

Theorem 2.2. *Let λ_i $i = 1\cdots n$, be the eigenvalues of A and assume that $|\lambda_i| < 1$, $\forall i$, (i.e., the system is asymptotically stable). Then:*

$$\lim_{t\to\infty} P(t) = \bar{P} \qquad \forall\, P(0) = P_0 > 0$$

where $\bar{P} = \bar{P}^T \geq 0$ is the unique solution of the Lyapunov equation

$$\bar{P} = A\bar{P}A^T + GQG^T.$$

Moreover, $x(t)$ and $y(t)$ are asymptotically stationary stochastic processes with zero mean and covariance matrices

$$\begin{aligned} R_{\boldsymbol{x}}(\tau) &= A^\tau \bar{P}\,, \qquad \tau \geq 0, \\ R_{\boldsymbol{y}}(\tau) &= \begin{cases} CA^\tau \bar{P}C^T & \text{if } \tau > 0 \\ C\bar{P}C^T + R & \text{if } \tau = 0 \end{cases} . \end{aligned}$$

2.3 The state estimation problem

Consider the linear stochastic system

$$\begin{cases} x(t+1) = Ax(t) + Bu(t) + Gw(t) \\ y(t) = Cx(t) + v(t) \end{cases} \tag{2.11}$$

and let Assumption 2.1 hold. We are now ready to formulate the state estimation problem for system (2.11).

Problem 2.1. *(State estimation problem). At each time t, find an estimate of $x(t)$, based on the knowledge of the input sequence $\{u(0), u(1), \ldots, u(t-1)\}$ and the output measurements*

$$\mathcal{Y}^t = \{y(0), y(1), \cdots, y(t)\}.$$

Being both the data set $\mathcal{Y}^t$ and the quantity to be estimated $x(t)$ random variables, this is clearly a Bayesian estimation problem. Therefore, the MSE estimate $\hat{x}(t)$, minimizing the mean square error $\mathbf{E}\left[||x(t) - \hat{x}(t)||^2\right]$, is the conditional mean with respect to the data, i.e., $\hat{x}_{MSE}(t) = \mathbf{E}\left[x(t)|\mathcal{Y}^t\right]$.

In order to compute the MSE solution, it is necessary to know the joint pdf of the state $x(t)$ and the data $\mathcal{Y}^t$. If we restrict our attention to linear estimator, we can compute the linear MSE solution, which requires only knowledge of the mean and covariance functions of the involved processes. In fact, one has

$$\hat{x}_{LMSE}(t) = m_{\boldsymbol{x}}(t) + P_{x(t),Y^t}[P_{Y^t}]^{-1}(Y^t - m_{\boldsymbol{Y^t}}), \tag{2.12}$$

where P_{Y^t} is the covariance matrix of the data vector

$$Y^t = \begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(t) \end{bmatrix},$$

while $P_{x(t),Y^t}$ is the cross-covariance matrix between $x(t)$ and Y^t. Although such quantities can be computed from model (2.11), by using Assumption 2.1, equation (2.12) cannot be employed in practice because the dimension of the involved covariance matrices grows as time passes. In fact, $P_{Y^t} \in \mathbb{R}^{(t+1)p\times(t+1)p}$ and for large values of t the computation of its inverse is practically infeasible.

In order to provide a computationally efficient approach to solve Problem 2.1, we aim at finding a *recursive solution* of the form

$$\hat{x}(t+1) = \Phi_t\, \hat{x}(t) + \Psi_t\, y(t+1)$$

where Φ_t and Ψ_t are suitable time-varying matrices which are used to compute a linear combination of the current estimate $\hat{x}(t)$ and the next measurement $y(t+1)$, providing the new estimate $\hat{x}(t+1)$, based on the data set $\mathcal{Y}^{t+1}$. The key idea is that the current estimate $\hat{x}(t)$ embeds all the information provided by the data up to time t, $\mathcal{Y}^t$. Clearly, the gains Φ_t and Ψ_t must be computed in order to minimize the mean square estimation error. The solution is based on a 2-step procedure and is known as the *Kalman Filter* (KF).

2.4 The Kalman Filter

Let us first introduce the notation that will be used in the construction of the LMSE state estimator. We denote by:

- $\hat{x}(t|t)$ the LMSE estimate of $x(t)$ based on Y^t;
- $\hat{x}(t+1|t)$ the LMSE estimate of $x(t+1)$ based on Y^t (LMSE 1-step ahead prediction);
- $P(t|t) = \mathbf{E}\left[(x(t) - \hat{x}(t|t))(x(t) - \hat{x}(t|t))^T\right] \in \mathbb{R}^{n\times n}$ the covariance matrix of the estimation error at time t;
- $P(t+1|t) = \mathbf{E}\left[(x(t+1) - \hat{x}(t+1|t))(x(t+1) - \hat{x}(t+1|t))^T\right] \in \mathbb{R}^{n\times n}$ the covariance matrix of the 1-step ahead prediction error at time t.

We aim at constructing the LMSE estimate of $x(t)$ through a two-step recursive procedure, also known as *prediction-correction algorithm*:

1. *Prediction*: Given $\hat{x}(t|t), P(t|t)$ and the model $\mathcal{M}$, compute $\hat{x}(t+1|t), P(t+1|t)$

2. *Correction*: Given $\hat{x}(t+1|t), P(t+1|t)$, the new measurement $y(t+1)$ and the model $\mathcal{M}$, compute $\hat{x}(t+1|t+1), P(t+1|t+1)$

3. Set $t \leftarrow t+1$ and go to step 1.

2.4.1 The prediction step

Given $\hat{x}(t|t), P(t|t)$ and the model

$$x(t+1) = Ax(t) + Bu(t) + Gw(t)$$

we want to compute the LMSE estimate $\hat{x}(t+1|t)$ of $x(t+1)$, based on data up to time t. The objective is to minimze $\mathbf{E}\left[||x(t+1) - \hat{x}(t+1|t)||^2\right]$ which is equivalent to minimize

$$\mathbf{E}\left[(x(t+1) - \hat{x}(t+1|t))(x(t+1) - \hat{x}(t+1|t))^T\right]$$

in the matricial sense. Let us introduce the notation for the estimation error $\tilde{x}(t|t) = x(t) - \hat{x}(t|t)$ and the prediction error $\tilde{x}(t+1|t) = x(t+1) - \hat{x}(t+1|t)$. Then one has

$$\begin{aligned} & \min_{\hat{x}} \mathbf{E}\left[(x(t+1) - \hat{x}(t+1|t))(x(t+1) - \hat{x}(t+1|t))^T\right] = \\ = & \min_{\hat{x}} \mathbf{E}\left[(Ax(t) + Bu(t) + Gw(t) - \hat{x}(t+1|t))(\cdots)^T\right] = \\ = & \min_{\hat{x}} \mathbf{E}\left[(A\hat{x}(t|t) + A\tilde{x}(t|t) + Bu(t) + Gw(t) - \hat{x}(t+1|t))(\cdots)^T\right] \quad (2.13) \end{aligned}$$

where the notation $(\cdots)^T$ is used to denote the transpose of the same term that appears on the left. Let

$$\begin{aligned} ① &= A\hat{x}(t|t) + Bu(t) - \hat{x}(t+1|t), \\ ② &= A\tilde{x}(t|t) + Gw(t). \end{aligned}$$

It is easy to see that $\mathbf{E}\left[①②^T\right] = 0$. In fact, a property of the LMSE estimate is that its estimation error is uncorrelated from the data, i.e. $\mathbf{E}\left[\tilde{x}(t|t)\, Y^t\right] = 0$. The term ① is a linear combination of the data up to time t, namely Y^t and

$u(t)$. On the other hand, in ② both $\tilde{x}(t|t)$ and $w(t)$ are uncorrelated from Y^t. Hence, (2.13) becomes

$$\begin{aligned}\min_{\hat{x}(t+1|t)} \quad & \mathbf{E}\left[(A\hat{x}(t|t) + Bu(t) - \hat{x}(t+1|t))(\cdots)^T\right] \\ & + \mathbf{E}\left[(A\tilde{x}(t|t) + Gw(t))(\cdots)^T\right].\end{aligned} \tag{2.14}$$

While the second term in (2.14) does not depend on the prediction $\hat{x}(t+1|t)$, the first term can be minimized by setting

$$\hat{x}(t+1|t) = A\hat{x}(t|t) + Bu(t) \tag{2.15}$$

which turns out to be the sought LMSE 1-step ahead state prediction. The corresponding error covariance is given by the second term in (2.14)

$$\begin{aligned}P(t+1|t) &= \mathbf{E}\left[(A\tilde{x}(t|t) + Gw(t))(\cdots)^T\right] = \\ &= \mathbf{E}\left[A\tilde{x}(t|t))\tilde{x}^T(t|t)A^T\right] + \mathbf{E}\left[Gw(t)w^T(t)G^T\right] = \\ &= A\mathbf{E}\left[\tilde{x}(t|t))\tilde{x}^T(t|t)\right]A^T + G\mathbf{E}\left[w(t)w^T(t)\right]G^T = \\ &= AP(t|t)A^T + GQG^T\end{aligned} \tag{2.16}$$

where we have exploited the fact that $\mathbf{E}\left[\tilde{x}(t|t)w(t)\right] = 0$, being $w(t)$ uncorrelated with both $x(t)$ and Y^t.

2.4.2 The correction step

Now assume that $\hat{x}(t+1|t)$, $P(t+1|t)$ and $y(t+1)$ are available. In order to derive the corrected estimate $\hat{x}(t+1|t+1)$, which incorporates also the new information provided by $y(t+1)$, we need to introduce a decoupling property of the LMSE estimate. Consider the expression of the LMSE estimate of a random variable $\boldsymbol{x}$ based on data $\boldsymbol{y}$

$$\hat{\boldsymbol{x}}_{LMSE} = m_{\boldsymbol{x}} + P_{\boldsymbol{xy}}P_{\boldsymbol{y}}^{-1}(\boldsymbol{y} - m_{\boldsymbol{y}}). \tag{2.17}$$

and assume that $\boldsymbol{y}$ is partitioned in two subvectors, i.e.

$$\boldsymbol{y} = \begin{bmatrix}\boldsymbol{y}_1 \\ \boldsymbol{y}_2\end{bmatrix}.$$

Then, (2.17) becomes

$$\hat{\boldsymbol{x}}_{\text{LMSE}} = m_{\boldsymbol{x}} + \begin{bmatrix} P_{\boldsymbol{x}\boldsymbol{y_1}} & P_{\boldsymbol{x}\boldsymbol{y_2}} \end{bmatrix} \begin{bmatrix} P_{\boldsymbol{y_1}} & P_{\boldsymbol{y_1}\boldsymbol{y_2}} \\ P_{\boldsymbol{y_2}\boldsymbol{y_1}} & P_{\boldsymbol{y_2}} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{y_1} - m_{\boldsymbol{y_1}} \\ \boldsymbol{y_2} - m_{\boldsymbol{y_2}} \end{bmatrix}$$

The computation of the matrix inverse simplifies if $\boldsymbol{y_1}$ and $\boldsymbol{y_2}$ are uncorrelated. Indeed, if $P_{\boldsymbol{y_1}\boldsymbol{y_2}} = 0$ one has

$$\begin{aligned} \hat{\boldsymbol{x}}_{\text{LMSE}} =& m_{\boldsymbol{x}} + \begin{bmatrix} P_{\boldsymbol{x}\boldsymbol{y_1}} & P_{\boldsymbol{x}\boldsymbol{y_2}} \end{bmatrix} \begin{bmatrix} P_{\boldsymbol{y_1}} & 0 \\ 0 & P_{\boldsymbol{y_2}} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{y}_1 - m_{\boldsymbol{y_1}} \\ \boldsymbol{y}_2 - m_{\boldsymbol{y_2}} \end{bmatrix} = \\ =& m_{\boldsymbol{x}} + \begin{bmatrix} P_{\boldsymbol{x}\boldsymbol{y_1}} & P_{\boldsymbol{x}\boldsymbol{y_2}} \end{bmatrix} \begin{bmatrix} P_{\boldsymbol{y_1}}^{-1} & 0 \\ 0 & P_{\boldsymbol{y_2}}^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{y}_1 - m_{\boldsymbol{y_1}} \\ \boldsymbol{y}_2 - m_{\boldsymbol{y_2}} \end{bmatrix} = \\ =& m_{\boldsymbol{x}} + P_{\boldsymbol{x}\boldsymbol{y_1}} P_{\boldsymbol{y_1}}^{-1} (y_1 - m_{\boldsymbol{y_1}}) + P_{\boldsymbol{x}\boldsymbol{y_2}} P_{\boldsymbol{y_2}}^{-1} (y_2 - m_{\boldsymbol{y_2}}) \\ =& \hat{\boldsymbol{x}}_{\text{LMSE}} | \boldsymbol{y_1} + P_{\boldsymbol{x}\boldsymbol{y_2}} P_{\boldsymbol{y_2}}^{-1} (y_2 - m_{\boldsymbol{y_2}}) \end{aligned} \tag{2.18}$$

where $\hat{\boldsymbol{x}}_{\text{LMSE}} | \boldsymbol{y_1}$ is the LMSE estimate of $\boldsymbol{x}$ based on the observation of $\boldsymbol{y_1}$. This suggests that if one is able to decompose the available data in two subsets that are uncorrelated with each other, it is possible to first compute the LMSE estimate based on the first data subset and then update it when the second data subset becomes available, by just adding the correction term $P_{\boldsymbol{x}\boldsymbol{y_2}} P_{\boldsymbol{y_2}}^{-1} (y_2 - m_{\boldsymbol{y_2}})$.

Since in general it is not true that $y(t+1)$ is uncorrelated with Y^t (notice that if this occurs for every time t, y is an uncorrelated process), we want to find a new process that contains only the information provided by $y(t+1)$ which is not correlated with Y^t. This turns out to be the so called *innovation process* $e(t)$, which is defined as

$$\begin{aligned} e(t+1) &\triangleq y(t+1) - C\hat{x}(t+1|t) = \\ &= Cx(t+1) + v(t+1) - C\hat{x}(t+1|t) = \\ &= C\tilde{x}(t+1|t) + v(t+1). \end{aligned} \tag{2.19}$$

Proposition 2.1. *The innovation process $e(t)$ defined in (2.19) has the following properties.*

1. *$e(t+1)$ is a linear combination of the data $Y^{t+1} = \begin{bmatrix} Y^t \\ y(t+1) \end{bmatrix}$;*

2. The process $e(t)$ is a sequence of uncorrelated random variables.

Proof

1. Being $e(t+1) = y(t+1) - C\hat{x}(t+1|t)$, where $\hat{x}(t+1|t)$ is a linear combination of data Y^t, one has that $e(t+1)$ is a linear combination of Y^{t+1}.
2. Being $e(t+1) = C\tilde{x}(t+1|t)+v(t+1)$, we have that $\tilde{x}(t+1|t)$ is uncorrelated with Y^t because the LMSE estimation error is uncorrelated with the data used to compute the estimate, while $v(t+1)$ is uncorrelated with Y^t because it is white and independent from $w(t)$ and $x(0)$ (and hence also from $x(t)$). Hence, $e(t+1)$ is uncorrelated with Y^t and therefore also with $e(i)$, $i = 0, 1, \ldots, t$, which are linear combinations of Y^i. □

By exploiting Proposition 2.1, we compute the LMSE estimate of $x(t+1)$ based on

$$\begin{bmatrix} Y^t \\ e(t+1) \end{bmatrix} = \begin{bmatrix} I & 0 \\ * & 1 \end{bmatrix} \begin{bmatrix} Y^t \\ y(t+1) \end{bmatrix}$$

i.e., on a nonsingular linear transformation of the original dataset Y^{t+1}. By assuming $\boldsymbol{y_1} = Y^t$ and $\boldsymbol{y_2} = e(t+1)$, from (2.18) one gets

$$\hat{x}(t+1|t+1) = \hat{x}(t+1|t) + P_{x(t+1)e(t+1)} P^{-1}_{e(t+1)} (e(t+1) - m_{e(t+1)}) \quad (2.20)$$

where we need to compute the mean and covariance function of $e(t+1)$ and its cross covariance with $x(t+1)$. The mean is given by

$$\begin{aligned} m_{e(t+1)} &= \mathbf{E}\left[C\tilde{x}(t+1|t) + v(t+1)\right] = \\ &= C\mathbf{E}\left[\tilde{x}(t+1|t)\right] + \mathbf{E}\left[v(t+1)\right] = 0 \end{aligned} \quad (2.21)$$

in which the first term is zero because the LMSE estimate is unbiased. The cross-covariance between $x(t+1)$ and $e(t+1)$ is given by

$$\begin{aligned} &P_{x(t+1)e(t+1)} = \mathbf{E}\left[(x(t+1) - m_{x(t+1)})e^T(t+1)\right] = \\ &= \mathbf{E}\left[(x(t+1) - m_{x(t+1)})(C\tilde{x}(t+1|t) + v(t+1))^T\right] = \\ &= \mathbf{E}[\{\underbrace{\tilde{x}(t+1|t)}_{①} + \underbrace{\hat{x}(t+1|t) - m_{x(t+1)}}_{②}\}\{\underbrace{C\tilde{x}(t+1|t)}_{③} + \underbrace{v(t+1)}_{④}\}^T]. \end{aligned}$$

We claim that the only product which is not zero is the one between the terms ① and ③. In fact, $\tilde{x}(t+1|t)$ is uncorrelated with $\hat{x}(t+1|t)$, which is a linear combination of Y^t, while $v(t+1)$ is uncorrelated with both $\hat{x}(t+1|t)$ and $\tilde{x}(t+1|t)$, which depend on y and w up to time t. Therefore,

$$P_{x(t+1)e(t+1)} = \mathbf{E}\left[\tilde{x}(t+1|t)\tilde{x}^T(t+1|t)\right] C^T = P(t+1|t)C^T \tag{2.22}$$

Then, let us compute the covariance matrix of $e(t+1)$

$$\begin{aligned} P_{e(t+1)} &= \mathbf{E}\left[(C\tilde{x}(t+1|t) + v(t+1))(C\tilde{x}(t+1|t) + v(t+1))^T\right] = \\ &= C\mathbf{E}\left[(\tilde{x}(t+1|t)(\tilde{x}(t+1|t))^T\right] C^T + \mathbf{E}\left[v(t+1)v^T(t+1)\right] = \\ &= CP(t+1|t)C^T + R \end{aligned} \tag{2.23}$$

By substituting (2.19), (2.21), (2.22) and (2.23) into (2.20), one gets

$$\begin{aligned} \hat{x}(t+1|t+1) = \hat{x}(t+1|t) + P(t+1|t)C^T[CP(t+1|t)C^T + R]^{-1} \cdot \\ (y(t+1) - C\hat{x}(t+1|t)) \end{aligned}$$

which can be written in a shorter form as

$$\hat{x}(t+1|t+1) = \hat{x}(t+1|t) + K(t+1)(y(t+1) - C\hat{x}(t+1|t)) \tag{2.24}$$

where we have introduced the *Kalman gain*

$$K(t+1) = P(t+1|t)C^T[CP(t+1|t)C^T + R]^{-1}.$$

Finally, we need to compute the covariance of the estimation error at time $t+1$, which is given by

$$\begin{aligned} &P(t+1|t+1) = \mathbf{E}\left[\tilde{x}(t+1|t+1)\tilde{x}^T(t+1|t+1)\right] = \\ &= \mathbf{E}\left[(x(t+1) - \hat{x}(t+1|t) - K(t+1)[C\tilde{x}(t+1|t) + v(t+1)])(\cdots)^T\right] = \\ &= \mathbf{E}\left[((I - K(t+1)C)\tilde{x}(t+1|t) - K(t+1)v(t+1))(\cdots)^T\right] \\ &= (I - K(t+1)C)P(t+1|t)(I - K(t+1)C)^T + K(t+1)RK(t+1)^T \\ &= P(t+1|t) - P(t+1|t)C^T[CP(t+1|t)C^T + R]^{-1}CP(t+1|t) \\ &= P(t+1|t)[I - C^T K(t+1)^T]. \end{aligned} \tag{2.25}$$

2.4.3 Initialization

In order to start the iteration of the Kalman filter, one needs to choose $\hat{x}(0|-1)$ and $P(0|-1)$, i.e., the state estimate and the corresponding covariance error *before* the first measurement $y(0)$ is processed. The natural choice are clearly the mean and covariance matrix of $x(0)$, if these quantities are known, i.e., $\hat{x}(0|-1) = m_0$ and $P(0|-1) = P_0$.

Otherwise, if such quantities are not available, one can choose $\hat{x}(0|-1)$ as any vector which is compatible with the a priori information on $x(0)$, and set

$$P(0|-1) = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix}$$

with $\lambda_i > 0$, "big enough" so that the resulting confidence interval of $x_i(0)$ covers the initial uncertainty associated to the i-th element of the state vector.

2.5 Properties of the Kalman Filter

Summing up the derivation carried out in the previous section, the Kalman Filter algorithm is defined as follows.

Initialization: $\hat{x}(0|-1) = m_0,\ P(0|-1) = P_0$

For $t = 0, 1, 2, \cdots$

$$K(t) = P(t|t-1)C^T[CP(t|t-1)C^T + R]^{-1} \tag{2.26}$$

$$\hat{x}(t|t) = \hat{x}(t|t-1) + K(t)(y(t) - C\hat{x}(t|t-1)) \tag{2.27}$$

$$P(t|t) = P(t|t-1)[I - C^T K(t)^T] \tag{2.28}$$

$$\hat{x}(t+1|t) = A\hat{x}(t|t) + Bu(t) \tag{2.29}$$

$$P(t+1|t) = AP(t|t)A^T + GQG^T \tag{2.30}$$

end

It is worth stressing that the derivation of the Kalman filter does not change if the model $\mathcal{M}$ is time-varying. Hence, the above equation can be easily extended to the time-varying case by just setting $A = A(t)$, $B = B(t)$, $C = C(t)$, $G = G(t)$, $Q = Q(t)$, and $R = R(t)$.

Another interesting observation concerns the fact that the evolution of the matrices $P(t|t)$, $P(t+1|t)$ and $K(t)$ does not depend on the data set $\{u(t), y(t)\}$ which is processed. In fact, the input-output data affect only the estimates $\hat{x}(t|t)$, $\hat{x}(t+1|t)$. This means that the *quality* of the estimates (which is determined by the covariances of the estimation errors) depends only on the model $\mathcal{M}$ and not on the actual data realization. Moreover, the sequences $P(t|t)$, $P(t+1|t)$, $K(t)$, for $t = 0, 1, \ldots$, can be computed offline, *before* the filter is applied to a data stream, and they remain the same for all data sets. This is particularly useful in those applications in which the computational burden at each iteration is critical and must be kept as low as possible.

In the following, we analyze other useful properties of the KF algorithm.

2.5.1 The information matrix

Let us rewrite the correction equation for the covariance of the estimation error as

$$P(t|t) = P(t|t-1) - P(t|t-1)C^T[CP(t|t-1)C^T + R]^{-1}CP(t|t-1) \quad (2.31)$$

and define the *Information matrix*

$$I(t|t) \triangleq P(t|t)^{-1}.$$

The information matrix is the inverse of the covariance matrix, so the larger is $I(t|t)$, the smaller is the uncertainty associated to the estimate $\hat{x}(t|t)$ of $x(t)$ (i.e., the more accurate is the estimate). By recalling the Matrix Inversion Lemma

$$(\bar{A} - \bar{B}\bar{C}\bar{D})^{-1} = \bar{A}^{-1} + \bar{A}^{-1}\bar{B}(\bar{C}^{-1} - \bar{D}\bar{A}^{-1}\bar{B})^{-1}\bar{D}\bar{A}^{-1} \quad (2.32)$$

and applying it to (2.31) with $\bar{A} = P(t|t-1)$, $\bar{B} = P(t|t-1)C^T$, $\bar{C} = [CP(t|t-1)C^T + R]^{-1}$ and $\bar{D} = CP(t|t-1)$, one gets

$$I(t|t) = I(t|t-1) + C^T R^{-1} C$$

where $I(t|t-1) = P(t|t-1)^{-1}$ and $C^T R^{-1} C$ can be interpreted as the quantity of information provided by the new measurement $y(t)$. In the scalar case ($n = p = 1$), one has $y(t) = c\,x(t) + v(t)$, with $c \in \mathbb{R}$ and

$$C^T R^{-1} C = \frac{c^2}{\sigma_v^2}$$

which can be seen as a sort of signal-to-noise ratio, between the *signal* $c\,x(t)$ measured by the output sensor and the measurement noise $v(t)$.

2.5.2 The Kalman Filter as a dynamic system

It is easy to see that the KF is a dynamic system processing the input and output data to produce state estimates. By substituting (2.27) into (2.29), one gets

$$\begin{aligned}\hat{x}(t+1|t) &= A\{\hat{x}(t|t-1) + K(t)(y(t) - C\hat{x}(t|t-1))\} + Bu(t) = \\ &= (A - AK(t)C)\hat{x}(t|t-1) + AK(t)y(t) + Bu(t) \\ &= (A - AK(t)C)\,\hat{x}(t|t-1) + \begin{bmatrix} AK(t) & B \end{bmatrix} \begin{bmatrix} y(t) \\ u(t) \end{bmatrix} \end{aligned} \tag{2.33}$$

which is the equation of a linear time-varying system, whose state vector is $\hat{x}(t+1|t)$ and the input vector is $\begin{bmatrix} y(t) \\ u(t) \end{bmatrix}$. Notice that the KF is an inherently time-varying system, due to the fact that the Kalman gain $K(t)$ changes at every time instant. The corresponding equation of the prediction error turns

out to be

$$\begin{aligned}\tilde{x}(t+1|t) &= x(t+1) - \hat{x}(t+1|t) \\ &= Ax(t) + Bu(t) + Gw(t) \\ &\quad -A\{\hat{x}(t|t-1) + K(t)(y(t) - C\hat{x}(t|t-1))\} - Bu(t) = \\ &= A\tilde{x}(t|t-1) + Gw(t) - AK(t)\{Cx(t) + v(t) - C\hat{x}(t|t-1)\} \\ &= (A - AK(t)C)\,\tilde{x}(t|t-1) + Gw(t) - AK(t)v(t) \qquad (2.34)\end{aligned}$$

which leads to the evolution of the average prediction error

$$\mathbf{E}\left[\tilde{x}(t+1|t)\right] = (A - AK(t)C)\,\mathbf{E}\left[\tilde{x}(t|t-1)\right]. \qquad (2.35)$$

For the covariance of the 1-step ahead prediction error, by substituing (2.28) into (2.30) one gets

$$\begin{aligned}P(t+1|t) &= AP(t|t-1)A^T + GQG^T \\ &\quad - AP(t|t-1)C^T[CP(t|t-1)C^T + R]^{-1}CP(t|t-1)A^T \qquad (2.36)\end{aligned}$$

which can be seen as a dynamic system whose state is the matrix $P(t|t-1)$. Equation (2.36) is known as the *Discrete Riccati equation (DRE)*.

2.5.3 The Kalman Filter as a state observer

Being the aim of the KF the computation of a state estimate, there is clearly a connection with the classical Luenberger state observer for deterministic systems. Consider the deterministic system

$$\begin{cases} x(t+1) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases}$$

then, the Luenberger observer is given by

$$\hat{x}(t+1) = A\hat{x}(t) + Bu(t) + L(y(t) - C\hat{x}(t)). \qquad (2.37)$$

If we introduce the state estimation error $\tilde{x}(t) = x(t) - \hat{x}(t)$, we obtain the error dynamics

$$\tilde{x}(t+1) = (A - LC)\tilde{x}(t) \qquad (2.38)$$

If the pair (A, C) is detectable, it is always possible to find a matrix L such that $(A - LC)$ has all its eigenvalues inside the unit circle. This implies that system (2.38) is asymptotically stable and hence $\lim_{t\to\infty} \tilde{x}(t) = 0$ for every initial condition $\tilde{x}(0)$. Equation (2.37) can be rewritten as

$$\hat{x}(t+1) = (A - LC)\hat{x}(t) + \begin{bmatrix} L & B \end{bmatrix} \begin{bmatrix} y(t) \\ u(t) \end{bmatrix}$$

in which it is easy to recognize the same structure as in (2.33). Indeed, they turn out to be the same equation if we set $L = AK(t)$. A similar analogy can be observed between equations (2.38) and (2.35). This means that the KF can be seen as a time-varying state observer for the linear stochastic system (2.11).

2.5.4 The innovation process

Let us consider the innovation process $e(t) = y(t) - C\hat{x}(t|t-1)$. It is possible to reformulate the KF as a dynamic system that processes the signals $y(t)$ and $u(t)$, to generate $e(t)$ as the output. In fact, by recalling (2.33) one can write

$$\begin{cases} \hat{x}(t+1|t) = (A - AK(t)C)\hat{x}(t|t-1) + \begin{bmatrix} AK(t) & B \end{bmatrix} \begin{bmatrix} y(t) \\ u(t) \end{bmatrix} \\ e(t) = -C\hat{x}(t|t-1) + \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} y(t) \\ u(t) \end{bmatrix} \end{cases}$$

This can be seen as a *whitening filter*, i.e., a system which takes the (usually) correlated process $y(t)$ and transforms it into the sequence of uncorrelated random variables $e(t)$ (which is also a white process in the Gaussian case, i.e., when $x(0)$, $w(t)$ and $v(t)$ are all normally distributed).

2.5.5 Extension to non white disturbance and noise processes

So far, we have supposed that the disturbance process $w(t)$ and the measurement noise $v(t)$ are white. This assumption can be relaxed, provided that a model for such processes is known. For example, assume that $w(t)$ is a stochastic process generated by the system

$$\begin{aligned} x_w(t+1) &= A_w x_w(t) + B_w \xi(t) \\ w(t) &= C_w x_w(t) + D_w \xi(t) \end{aligned}$$

where $\xi(t) \sim WP(0, Q_\xi)$ and it is assumed to be uncorrelated with the measurement noise $v(t) \sim WP(0, R)$. Notice that for an asymptotically stationary s.p., matrices A_w, B_w, C_w, D_w can be derived by computing a state space realization of the canonical spectral factor of process $w(t)$. Hence, we can write

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) + Gw(t) \\ &= Ax(t) + Bu(t) + G(C_w x_w(t) + D_w \xi(t)). \end{aligned}$$

By defining the extended state vector

$$\bar{x}(t) = \begin{bmatrix} x(t) \\ x_w(t) \end{bmatrix}$$

one can write the system

$$\begin{bmatrix} x(t+1) \\ x_w(t+1) \end{bmatrix} = \begin{bmatrix} A & GC_w \\ 0 & A_w \end{bmatrix} \begin{bmatrix} x(t) \\ x_w(t) \end{bmatrix} + \begin{bmatrix} GD_w \\ B_w \end{bmatrix} \xi(t) + \begin{bmatrix} B \\ 0 \end{bmatrix} u(t) \tag{2.39}$$

$$y(t) = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ x_w(t) \end{bmatrix} + v(t) \tag{2.40}$$

Hence, one can apply the standard KF to system (2.39)-(2.40), thus obtaining an estimate $\hat{\bar{x}}(t|t) = \begin{bmatrix} \hat{x}(t|t) \\ \hat{x}_w(t|t) \end{bmatrix}$ of the extended state. Notice that, besides

the desired estimate of the state of the original system, this contains also the estimate of the disturbance state $x_w(t)$ as a byproduct.

Similarly, assume that $v(t)$ is not white and it is generated by the system

$$\begin{aligned} x_v(t+1) &= A_v x_v(t) + B_v \xi(t) \\ v(t) &= C_v x_v(t) + D_v \xi(t) \end{aligned}$$

where $\xi(t) \sim WP(0, Q_\xi)$ and is uncorrelated with the process disturbance $w(t) \sim WP(0, Q)$. By defining the extended state vector

$$\bar{x}(t) = \begin{bmatrix} x(t) \\ x_v(t) \end{bmatrix}$$

one obtains the extended system equations

$$\begin{bmatrix} x(t+1) \\ x_v(t+1) \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & A_v \end{bmatrix} \begin{bmatrix} x(t) \\ x_v(t) \end{bmatrix} + \begin{bmatrix} G & 0 \\ 0 & B_v \end{bmatrix} \begin{bmatrix} w(t) \\ \xi(t) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(t) \qquad (2.41)$$

$$y(t) = \begin{bmatrix} C & C_v \end{bmatrix} \begin{bmatrix} x(t) \\ x_v(t) \end{bmatrix} + D_v \, \xi(t) \qquad (2.42)$$

where the extended process $\bar{w}(t) = \begin{bmatrix} w(t) \\ \xi(t) \end{bmatrix}$ is a white process with zero mean and covariance matrix $\begin{bmatrix} Q & 0 \\ 0 & Q_\xi \end{bmatrix}$.
Also in this case, one may think to apply the standard KF to system (2.41)-(2.42), thus obtaining an estimate $\hat{\bar{x}}(t|t) = \begin{bmatrix} \hat{x}(t|t) \\ \hat{x}_v(t|t) \end{bmatrix}$ of the extended state, which contains the desired state estimate, along with the estimate of the noise state $x_v(t)$. However, it should be noticed that it is not true that the extended process $\bar{w}(t)$ and the new output noise $D_v\xi(t)$ are uncorrelated. In fact, one has $\mathbf{E}\left[\bar{w}(t)(D_v\xi(t))^T\right] = [0 \quad Q_\xi D_v^T]$. Therefore, it is customary to use the version of the Kalman Filter in which item ii) in Assumption 2.1 is replaced by

$$\mathbf{E}\left[\begin{pmatrix} w(t) \\ v(t) \end{pmatrix} \begin{pmatrix} w(t) \\ v(t) \end{pmatrix}^T \right] = \begin{bmatrix} Q & N \\ N^T & R \end{bmatrix}$$

where $N \in \mathbb{R}^{d\times p}$ is the cross-covariance between $w(t)$ and $v(t)$. The equations of such a version of the Kalman Filter turn out to be as follows.

$$\hat{x}(t|t) = \hat{x}(t|t-1) + K(t)(y(t) - C\hat{x}(t|t-1)) \tag{2.43}$$

$$P(t|t) = P(t|t-1)[I - C^T K^T(t)] \tag{2.44}$$

$$K(t) = P(t|t-1)C^T[CP(t|t-1)C^T + R]^{-1} \tag{2.45}$$

$$\begin{aligned}\hat{x}(t+1|t) = {} & A\hat{x}(t|t) + Bu(t) \\ & + GN[CP(t|t-1)C^T + R]^{-1}(y(t) - C\hat{x}(t|t-1))\end{aligned} \tag{2.46}$$

$$\begin{aligned}P(t+1|t) = {} & AP(t|t)A^T + GQG^T - GN[CP(t|t-1)C^T + R]^{-1}N^T G^T \\ & - AP(t|t-1)C^T[CP(t|t-1)C^T + R]^{-1}N^T G^T \\ & - GN[CP(t|t-1)C^T + R]^{-1}CP(t|t-1)A^T\end{aligned} \tag{2.47}$$

which in predictor form takes the more coincise form

$$\hat{x}(t+1|t) = A\hat{x}(t|t-1) + Bu(t) + K_c(t)(y(t) - C\hat{x}(t|t-1)) \tag{2.48}$$

$$P(t+1|t) = AP(t|t-1)A^T + GQG^T - K_c(t)[CP(t|t-1)C^T + R]K_c(t)^T \tag{2.49}$$

with

$$K_c(t) = (AP(t|t-1)C^T + GN)[CP(t|t-1)C^T + R]^{-1}. \tag{2.50}$$

Finally, the case in which both $w(t)$ and $v(t)$ are not white can be treated in the same way, by combining the two approaches outlined above.

2.6 Asymptotic behavior of the KF

A fundamental question on the performance of the Kalman Filter concerns its asymptotic behavior. In particular, we would like to answer the following questions.

1. What is the asymptotic expected value of the estimation error, i,e.

$$\lim_{t\to+\infty} \mathbf{E}\left[\tilde{x}(t|t)\right] = \lim_{t\to+\infty} \mathbf{E}\left[x(t) - \hat{x}(t|t)\right] \tag{2.51}$$

and how does it depend on the initialization of the filter? If the limit in (2.51) is equal to zero, this means that the KF is an asymptotically unbiased estimator.

2. What is the asymptotic behavior of the DRE (2.36)? Does the limit $\lim_{t\to+\infty} P(t+1|t)$ exist? How does it depend on the initial condition $P(0|-1)$?

3. Does the Kalman gain $K(t)$ converge to a constant matrix? And if this is the case, what are the properties of the asymptotic version of system (2.33)?

Clearly, all such questions make sense only when the dynamic model we are considering is time-invariant and the stochastic processes $w(t)$ and $v(t)$ are stationary. The next result provide an answer to the above questions.

Theorem 2.3. *Consider system* (2.11) *and let Assumtpion 2.1 hold. Let* A, B, C, G, Q, R *be constant matrices. Define the matrix* $H \in \mathbb{R}^{n\times d}$ *such that* $HH^T = GQG^T$ *and assume that the pair* (A, C) *is detectable and the pair* (A, H) *is stabilizable. Then, the following results hold.*

1. $\lim_{t\to+\infty} \mathbf{E}\left[\tilde{x}(t|t)\right] = \lim_{t\to+\infty} \mathbf{E}\left[\tilde{x}(t+1|t)\right] = 0\ ,\ \forall \hat{x}(0|-1) \in \mathbb{R}^n$

2. $\lim_{t\to+\infty} P(t+1|t) = P_\infty < +\infty,\ \forall\, P(0|-1) > 0$ *where* P_∞ *is the unique positive semidefinite solution of the Algebraic Riccati Equation (ARE)*

$$P_\infty = AP_\infty A^T + GQG^T - AP_\infty C^T[CP_\infty C^T + R]^{-1}CP_\infty A^T \quad (2.52)$$

3. *Let*

$$K_\infty = \lim_{t\to\infty} K(t) = P_\infty C^T[CP_\infty C^T + R]^{-1}.$$

Then, the matrix $(A - AK_\infty C)$ *has all its eigenvalues inside the unit circle.*

The first item in Theorem 2.3 states that the state estimates provided by the KF are asymptotically unbiased. Even more important is the result in

item 2: the covariance of the 1-step ahead prediction error always converges to the same constant matrix P_∞, whatever is the initial covariance matrix $P(0|-1)$ chosen to start the KF iterations. Notice that the same occurs for the covariance matrix of the state estimation error

$$\lim_{t\to+\infty} P(t|t) = P_\infty - P_\infty C^T[CP_\infty C^T + R]^{-1}CP_\infty = P_\infty(I - C^T K_\infty^T)$$

It is also worth pointing out that the above results hold under quite mild assumptions. The most important one concerns detectability of the pair (A, C): in fact, if the system is not detectable, it means that there is a subsystem which is not asymptotically stable and it is also unobservable. Hence, the covariance of the state estimation error for such subsystem will eventually grow to infinity. Recall that if a system is fully observable, it is also detectable.

The other assumption in Theorem 2.3, namely stabilizability of the pair (A, H), is essentially technical and it is necessary to guarantee that the ARE (2.52) has a unique positive semidefinite solution. Recall that if (A, H) is fully reachable, it is also stabilizable. This assumption can be further relaxed to exclude only unreachable eigenvalues of A with modulus exactly equal to 1: in such a case the ARE may have multiple positive semidefinite solutions, but they will be ordered (in matrical sense) and $P(t+1|t)$ will converge to the largest one.

Finally, the third item in Theorem 2.3 suggests that the time-varying evolution of the average prediction error in equation (2.35), converges to an asymptotically stable time-invariant system. This suggests that one may want to use the constant gain K_∞ right from the start, in place of the time-varying Kalman gain $K(t)$, as explained next.

2.6.1 The asymptotic Kalman Filter

Let us assume that instead of changing the Kalman gain $K(t)$ at every KF iteration, we want to use a constant gain. A natural choice is to use K_∞, because we know that it minimizes the asymptotic mean square estimation

error. Then, the recursion (2.27)-(2.30) simplifies to

$$\hat{x}(t+1|t) = A\hat{x}(t|t) + Bu(t) \tag{2.53}$$

$$\hat{x}(t+1|t+1) = \hat{x}(t+1|t) + K_\infty(y(t+1) - C\hat{x}(t+1|t)) \tag{2.54}$$

which we will refer to as the *Asymptotic Kalman Filter*. Clearly, while (2.53)-(2.54) has the advantage of being a linear time-invariant system, not requiring the computation of the covariance matrices $P(t|t)$, $P(t+1|t)$ at every time step, it does not guarantee anymore to minimize the estimation MSE for every t, but only as time approaches infinity.

In order to understand which is the trade-off between using a time-invariant filter and the optimal KF, let us consider the following simple example involving a scalar state ($n = 1$)

$$\begin{cases} x(t+1) = a\,x(t) \\ y(t) = x(t) + v(t) \end{cases} \tag{2.55}$$

where $a \in \mathbb{R}$ and $v(t) \sim WP(0, r)$. The associated KF equations are

$$\begin{aligned} \hat{x}(t+1|t) &= a\,\hat{x}(t|t) \\ P(t+1|t) &= a^2 P(t|t) \\ \hat{x}(t+1|t+1) &= \hat{x}(t+1|t) + K(t+1)(y(t+1) - \hat{x}(t+1|t)) \\ P(t+1|t+1) &= P(t+1|t) - \frac{P^2(t+1|t)}{P(t+1|t)+r}. \end{aligned}$$

In prediction form, they become

$$\hat{x}(t+1|t) = a\hat{x}(t|t-1) + aK(t)(y(t) - \hat{x}(t|t-1)) \tag{2.56}$$

$$P(t+1|t) = a^2 P(t|t-1) - \frac{a^2 P^2(t|t-1)}{P(t|t-1)+r} = \frac{a^2 P(t|t-1) r}{P(t|t-1)+r} \tag{2.57}$$

The estimation error is given by

$$\begin{aligned} \tilde{x}(t+1|t) &= x(t+1) - \hat{x}(t+1|t) \\ &= ax(t) - a\hat{x}(t|t-1) - aK(t)(x(t) + v(t) - \hat{x}(t|t-1)) \\ &= a\tilde{x}(t|t-1) - aK(t)\tilde{x}(t|t-1) - aK(t)v(t) \\ &= a(1 - K(t))\tilde{x}(t|t-1) - aK(t)v(t) \end{aligned} \tag{2.58}$$

where

$$K(t) = \frac{P(t|t-1)}{P(t|t-1)+r}.$$

Now, let us assume that we want to use an LTI filter with constant gain K. If we denote by $\hat{x}(t)$ the estimate of the LTI filter and by $\tilde{x}(t)$ the corresponding estimation error, from (2.58) one has

$$\tilde{x}(t+1) = a(1-K)\tilde{x}(t) - aKv(t).$$

By taking the expected value, being $\mathbf{E}[v(t)] = 0$, one gets

$$\mathbf{E}[\tilde{x}(t+1)] = a(1-K)\mathbf{E}[\tilde{x}(t)]$$

and hence the *bias error* will tend to zero whenever $|a(1-K)| < 1$. In particular, convergence will be faster as K approaches 1. On the other hand, if we consider the variance of the estimation error

$$P(t) = \mathbf{E}\left[(\tilde{x}(t) - m_{\tilde{\boldsymbol{x}}(\boldsymbol{t})})^2\right]$$

by exploiting the fact that $\tilde{x}(t)$ and $v(t)$ are uncorrelated, we get

$$\begin{aligned} P(t+1) &= \mathbf{E}\left[\left\{a(1-K)(\tilde{x}(t) - m_{\tilde{\boldsymbol{x}}(\boldsymbol{t})}) - aKv(t)\right\}^2\right] \\ &= a^2(1-K)^2\mathbf{E}\left[(\tilde{x}(t) - m_{\tilde{\boldsymbol{x}}(\boldsymbol{t})})^2\right] + a^2K^2\mathbf{E}\left[v^2(t)\right] \\ &= a^2(1-K)^2P(t) + a^2K^2r \end{aligned}$$

which is a first order dynamic system in the variable $P(t)$, forced by the constant input a^2K^2r. If $a^2(1-K)^2 < 1$ such a system is asymptotically stable and one has

$$\lim_{t\to\infty} P(t) = \frac{1}{1-a^2(1-K)^2}a^2K^2r.$$

Therefore, it is easy to see that the asymptotic value of the variance of the estimation error will tend to zero as K approaches zero. Summing up, we are faced to the typical bias-variance trade off: we need a "large" filter gain ($K \to 1$) to reduce the bias error as fast as possible, and a "small" gain

($K \to 0$) to reduce the asymptotic variance (uncertainty) associated to the estimate. This is the reason why the MSE filter is indeed a time-varying one: it employs a large gain during the transient, in order to rapidly reduce the bias error, and a small gain asymptotically, to reduce the variance of the estimation error.

Figures 2.1-2.3 report the results of a numerical test performed on system (2.55) with $a = 0.9$, $r = 0.04$, and $x(0) = 1$. Fig. 2.1 shows the evolution of the true state $x(t)$ and the noisy observation $y(t)$. Fig. 2.2 reports on top the estimates of two LTI filters with constant gain $K = 0.9$ and $K = 0.1$, respectively. It can be seen that the former quickly reduces the bias error but shows a remarkable uncertainty in the estimates; conversely, the latter has a small asymptotic variance of the estimate but it is quite slow in tracking the true state evolution. The bottom plot reports the estimate provided by the Kalman Filter, initialized with $\hat{x}(0|-1) = 0$ and $P(0|-1) = 1$. It can be observed that the KF succeeds in both reducing the initial bias error and keeping small the asymptotic error variance. This is clearly due to the time varying gain $K(t)$, shown in Fig. 2.3, along with the error variance $P(t|t)$.

The considered one-dimensional example is also helpful to provide an insight in the asymptotic results of Theorem 2.3. Since there is no process disturbance in system (2.55), one has $Q = 0$ and the ARE (2.52) reduces to

$$P_\infty = \frac{a^2 P_\infty r}{P_\infty + r}$$

which can be written as

$$P_\infty^2 + P_\infty r(1 - a^2) = 0$$

and therefore it has two solutions

$$\begin{cases} P_\infty = 0 \\ P_\infty = r(a^2 - 1) \end{cases} \tag{2.59}$$

If $a^2 < 1$, then the only positive semidefinite solution is $P_\infty = 0$. This is consistent with the fact that, being $H = 0$, the pair $(A, H) = (a, 0)$ is not

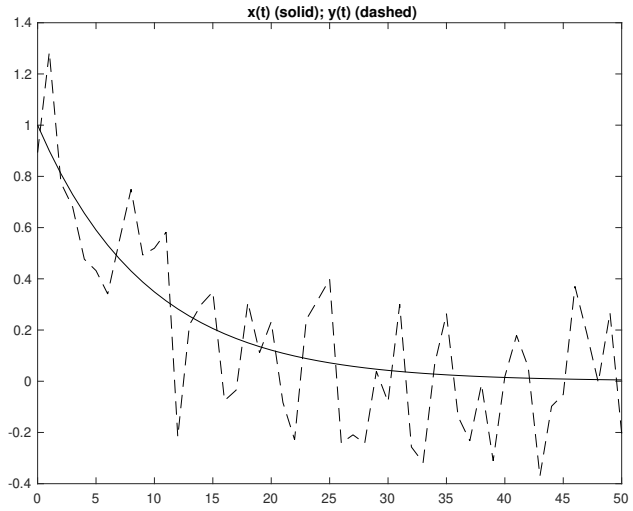


Figure 2.1: Evolution of system (2.55) with $a = 0.9$, $r = 0.04$, and $x(0) = 1$: $x(t)$ (solid); $y(t)$ (dashed).

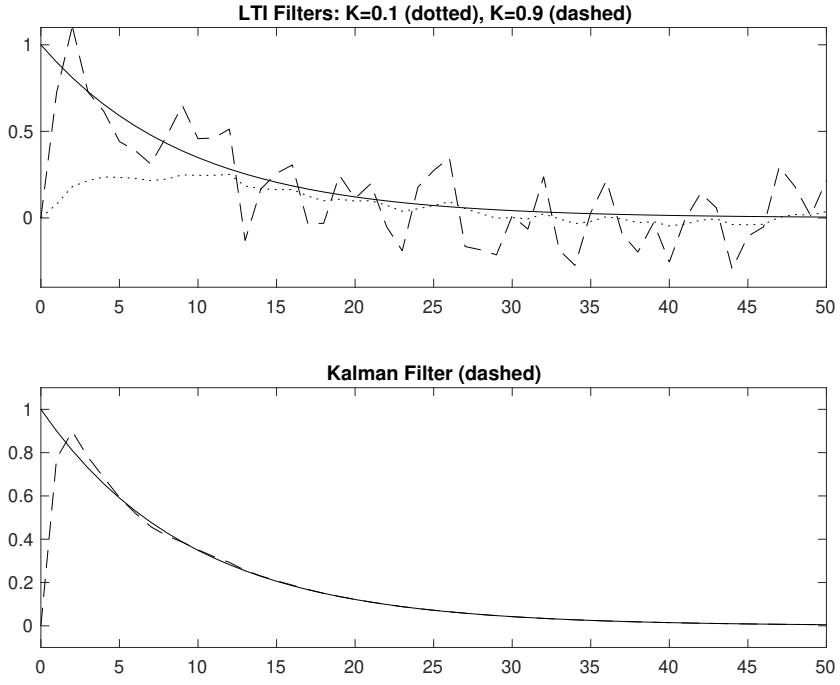


Figure 2.2: Top: performance of LTI filters with $K = 0.9$ (dashed) and $K = 0.1$ (dotted), with respect to the true state $x(t)$ (solid). Bottom: KF estimate $\hat{x}(t|t)$ (dashed) compared to true state $x(t)$ (solid).

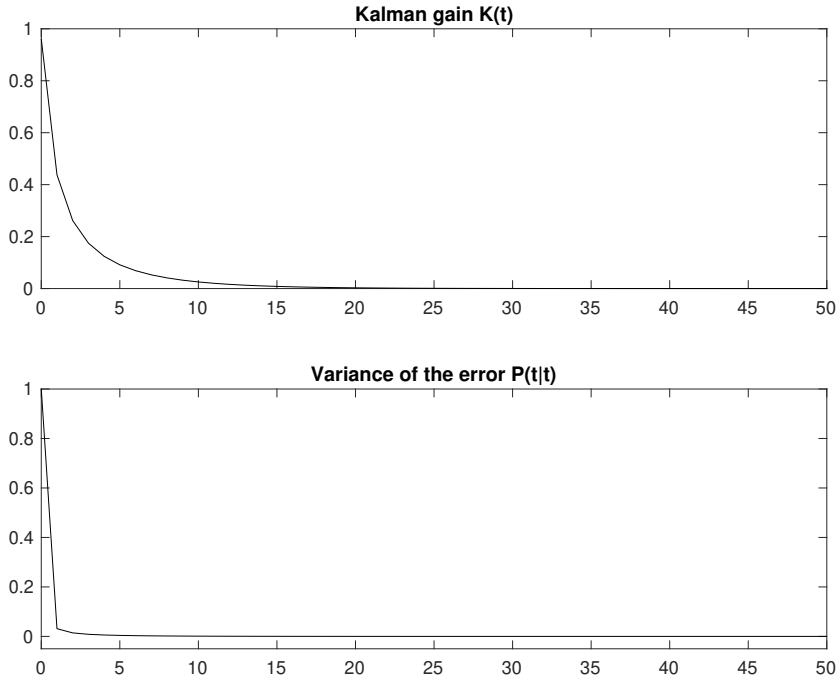


Figure 2.3: Top: Kalman gain $K(t)$. Bottom: variance of estimation error $P(t|t)$.

reachable but it is indeed stabilizable (the system is already asymptotically stable!). Hence, Theorem 2.3 guarantees that

$$\lim_{t\to\infty} P(t|t-1) = P_\infty = 0$$
$$\lim_{t\to\infty} K(t) = \frac{P_\infty}{P_\infty + r} = K_\infty = 0$$

This is precisely what can be obtained by studying the one-dimensional (nonlinear) system (2.57), which describes the behavior of the variance $P(t|t-1)$. Notice that in this example the asymptotic Kalman filter simply ignores the output data ($K_\infty = 0$) and it just waits that the state estimate converges to zero, like the true state does.

Conversely, if $a^2 > 1$, the pair $(a, 0)$ is not stabilizable and hence we cannot apply Theorem 2.3. Indeed, both solutions (2.59) are positive semidefi-

nite. Nevertheless, by analyzing again system (2.57), it can be shown that:

$$\lim_{t\to\infty} P(t|t-1) = r(a^2-1)$$
$$\lim_{t\to\infty} K(t) = \frac{P_\infty}{P_\infty + r} = \frac{r(a^2-1)}{r(a^2-1)+r} = \frac{a^2-1}{a^2}$$

i.e., the covariance converges to the largest solution of the ARE. We stress that the solution $K_\infty = 0$ this time is unacceptable because the system is unstable and ignoring the output data would lead to divergence of the error variance. Instead, $K_\infty = \frac{a^2-1}{a^2}$ leads to

$$\lim_{t\to\infty}(A - AK(t)C) = a - aK_\infty = a - a\frac{a^2-1}{a^2} = \frac{1}{a}$$

which is indeed inside the unit circle, as $a^2 > 1$ implies $\left|\frac{1}{a}\right| < 1$. This confirms that the asymptotic KF is an asymptotically stable LTI filter, guaranteeing that the expected value of the state estimation error converges to zero also for unstable systems.

It is also interesting to analyze the case in which the process disturbance is present. Let

$$x(t+1) = ax(t) + w(t)$$

with $w(t) \sim WP(0, q)$ and $h = \sqrt{q} \neq 0$, which implies that the pair (a, h) is always reachable and hence also stabilizable. The ARE becomes

$$P_\infty = a^2 P_\infty + q - \frac{a^2 P_\infty^2}{P_\infty + r} = \frac{a^2 P_\infty r}{P_\infty + r} + q$$

or equivalently

$$P_\infty^2 + P_\infty[r(1-a^2) - q] - rq = 0$$

which has always a unique positive solution $P_\infty > 0$, for every $a \in \mathbb{R}$, $q > 0$, $r > 0$. Hence, Theorem 2.3 guarantees that

$$\lim_{t\to\infty} P(t|t-1) = P_\infty.$$

Notice that in this case P_∞ cannot be equal to zero because $w(t)$ steadily injects uncertainty into the system.

2.7 Applications of the Kalman Filter

The Kalman Filter is widely employed in a number of applications. Hereafter, we briefly review some of them.

Observer for stochastic systems and LQG control

As it has been previously remarked, the KF can be used as a state observer for stochastic systems. When coupled with a suitable state feedback, it allows one to design an output feedback control law, according to the scheme in Figure 2.4.

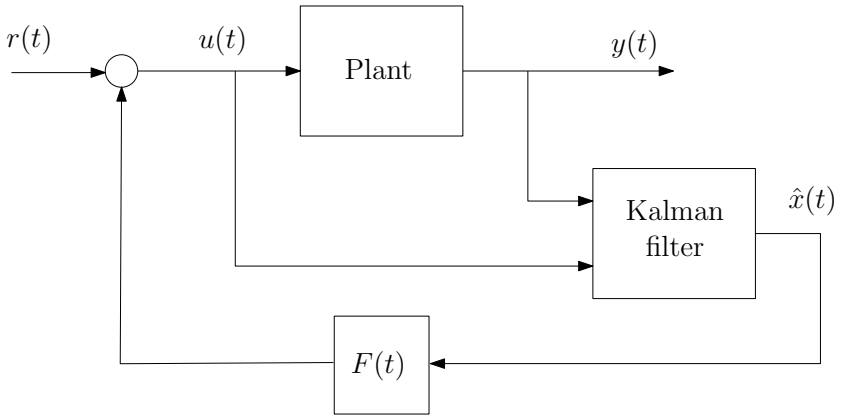


Figure 2.4: KF as a state observer in an output feedback control loop.

A special case is that of *Linear Quadratic Gaussian (LQG)* control, in which the aim is to find a control signal $u(t)$ which minimizes the cost function

$$\mathbf{E}\left[\sum_{t=0}^{N-1}(x^T(t)\bar{Q}x(t)+u^T(t)\bar{R}u(t))+x^T(N)\bar{S}x(N)\right] \tag{2.60}$$

where $\bar{Q}$, $\bar{R}$ and $\bar{S}$ are positive definite matrices. It can be shown that the solution of such problem is provided by the scheme in Figure 2.4, in which matrix $F(t)$ is designed to minimize cost (2.60) as the true state $x(t)$ were available. This is known as *separation principle*, as it allows to design the

state feedback and the observer separately. The resulting control input turns out to be

$$u(t) = F(t)\hat{x}(t) + r(t)$$

where

$$F(t) = -(B^T S(t+1)B + \bar{R})^{-1} B^T S(t+1)A$$

and $S(t)$ is a sequence of matrices computed via the backward Riccati iteration

$$\begin{aligned} S(N) &= \bar{S}, \\ S(t) &= A^T S(t+1)A + \bar{Q} \\ &\quad -A^T S(t+1)B(B^T S(t+1)B + \bar{R})^{-1} B^T S(t+1)A, \end{aligned}$$

for $t = N-1, N-2, \ldots, 1, 0$.

Sensor fusion

Sensor fusion is a problem in which information provided by multiple sensors has to be processed in order to estimate an unknown quantity. The problem is particularly important in networked systems, where a large number of sensors is deployed within the network and their measurements have to be "fused" either by a central unit or in a decentralized way (i.e., distributing the estimation task among multiple nodes of the network).

The KF provides the natural framework for facing such a problem. Assume that at time t the i-th sensor provides a measurement $y_i(t) = C_i\boldsymbol{x} + v_i(t)$ of the quantity $\boldsymbol{x}$ to be estimated, corrupted by noise $v_i(t)$. All measurements can be stack in the unique vector

$$y(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_p(t) \end{bmatrix} = \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_p \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} v_1(t) \\ v_2(t) \\ \vdots \\ v_p(t) \end{bmatrix}$$

where p is the total number of sensors. If $\boldsymbol{x}$ is constant in time, one can use directly the KF to estimate it, by setting $A = I$, $B = G = 0$, in the

state dynamic model. Clearly, if $\boldsymbol{x}$ is not constant but its dynamics is known (possibly with uncertainty modeled as a process disturbance), the KF setting can be readily applied.

Estimation of sensor bias and drit

Let us consider system (2.11) and assume that the sensor providing the scalar measurement $y(t) = Cx(t)+v(t)$ is affected by unknown bias and drift errors. This can be modeled by assuming that $v(t) \sim WP(b + d\,t, R)$, where b and d are, respectively, the bias and drift to be estimated. This can be done by augmenting the state of the system to include such quantities, i.e.

$$\bar{x}(t) = \begin{bmatrix} x(t) \\ x_b(t) \\ x_d(t) \end{bmatrix}$$

where $x_b(t)$ and $x_d(t)$ represent the bias and drift, respectively. Their dynamic model can be chosen as a random walk, i.e.

$$\begin{aligned} x_b(t+1) &= x_b(t) + w_b(t) \\ x_d(t+1) &= x_d(t) + w_d(t) \end{aligned}$$

in which $w_b(t) \sim WP(0, \sigma_b^2)$, $w_d(t) \sim WP(0, \sigma_d^2)$ are independent white processes. Their variances σ_b^2, σ_d^2 are tunable parameters whose aim is to allow the filter to move from the initial (possibly wrong) estimate of b and d and converge to their actual values. Then, by defining $\tilde{v}(t) = v(t) - (b + d\,t)$, one can write the output equation as

$$y(t) = Cx(t) + b + d\,t + \tilde{v}(t) = [C \;\; 1 \;\; t]\,\bar{x}(t) + \tilde{v}(t).$$

Finally, by observing that $\tilde{v}(t) \sim WP(0, R)$, one can use the standard KF to estimate the extended state $\bar{x}(t)$, that corresponds to simultaneoulsy estimate the original state $x(t)$ along with the sensor bias b and drift d.

Fault detection

The aim of fault detection is to detect undesired behaviors of dynamical systems, by using observations of the input and output signal. The KF is among the standard tools employed to this purpose. Assume that a model of the dynamic system is available, in the form (2.11). The idea is to exploit one of the key properties of the KF: the fact that the innovation process $e(t) = y(t) - C\hat{x}(t|t-1)$ is white, with zero mean and covariance matrix equal to $CP(t|t-1)C^T + R$, i.e.

$$e(t) \sim WP(0, CP(t|t-1)C^T + R). \tag{2.61}$$

Therefore, one can run the KF as a dynamic system with $e(t)$ as output signal, like in Figure 2.5, and then apply statistical tests to check whether $e(t)$ is consistent with the hypothesis (2.61). If there is statistical evidence that such an hypothesis is violated, this suggests a possible malfunctioning of the system.

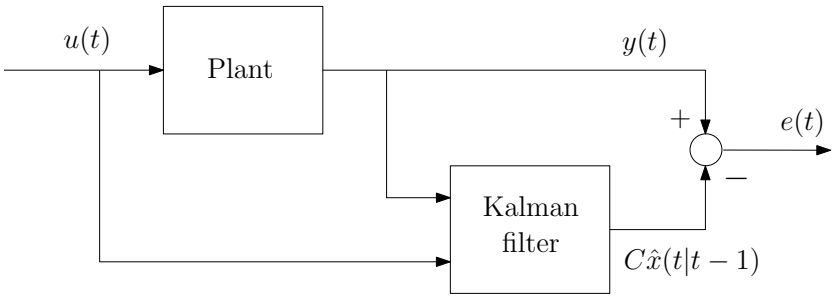


Figure 2.5: KF in innovation form for fault detection.

Recursive system identification

Consider a parametric system identification problem, in which the model has the linear regression form

$$y(t) = \varphi^T(t)\theta + e(t) \tag{2.62}$$

where $y(t)$ is the measured output at time t, $\phi(t)$ is the (known) regressor vector, containing input and output values at times before t, $e(t)$ is an error term modeled as a white stochastic process with zero mean and variance σ_e^2, and θ is the parameter vector to be estimated. Assume that at every time t, a new output measurement $y(t)$ and regressor $\varphi(t)$ is collected. The aim is to compute the *least squares estimate* of θ based on the measurements up to time t, which is defined as

$$\hat{\theta}_t^{LS} = \arg\min_{\theta} \sum_{k=1}^{t} (y(k) - \varphi^T(k)\theta)^2. \tag{2.63}$$

The solution of (2.63) is given by

$$\hat{\theta}_t^{LS} = \left(\sum_{k=1}^{t} \varphi(k)\varphi^T(k) \right)^{-1} \sum_{k=1}^{t} \varphi(k)y(k) \tag{2.64}$$

but it would be clearly impractical to use equation (2.64) to compute the parameter estimates at each time t, as one needs to process a steadily increasing number of data as t grows. An efficient solution is provided by the *recursive least squares (RLS)* algorithm

$$\begin{aligned} \hat{\theta}_t &= \hat{\theta}_{t-1} + L(t)[y(t) - \varphi^T(t)\hat{\theta}_{t-1}] \\ L(t) &= \frac{P(t-1)\varphi(t)}{1 + \varphi^T(t)P(t-1)\varphi(t)} \\ P(t) &= P(t-1) - \frac{P(t-1)\varphi(t)\varphi^T(t)P(t-1)}{1 + \varphi^T(t)P(t-1)\varphi(t)}. \end{aligned} \tag{2.65}$$

It can be shown that if the RLS algorithm is initialized at a certain time t_0 with $\hat{\theta}_{t_0} = \hat{\theta}_{t_0}^{LS}$ given by (2.64) and $P(t_0) = \left(\sum_{k=1}^{t_0} \varphi(k)\varphi^T(k)\right)^{-1}$, then one has that the RLS algorithm returns the same parameter estimate as the least squares estimate given by (2.64), i.e., $\hat{\theta}_t = \hat{\theta}_t^{LS}$, $\forall t \geq t_0$.

It is interesting to highlight that the RLS algorithm can be seen as an application of the Kalman filter to a time-varying linear system. In fact, define the state vector $x(t)$ as the unknown parameter vector θ to be estimated. Being θ constant, for a linear regression model $y(t) = \varphi^T(t)\theta + e(t)$ one can

write the state space model

$$\begin{aligned} x(t+1) &= x(t) \\ y(t) &= \varphi^T(t)x(t) + e(t). \end{aligned} \tag{2.66}$$

This clearly falls within the general framework of the LMSE state estimation problem (2.11), by setting $A = I$, $B = G = Q = 0$, $C = \varphi^T(t)$ and $v(t) = e(t) \sim WP(0, \sigma_e^2)$, i.e., $R = \sigma_e^2$. If we apply to system (2.66) the KF algorithm, we get the recursions

$$\begin{aligned} \hat{x}(t) &= \hat{x}(t-1) + K(t)[y(t) - \varphi^T(t)\hat{x}(t-1)] \\ K(t) &= \frac{P(t-1)\varphi(t)}{\sigma_e^2 + \varphi^T(t)P(t-1)\varphi(t)} \\ P(t) &= P(t-1) - \frac{P(t-1)\varphi(t)\varphi^T(t)P(t-1)}{\sigma_e^2 + \varphi^T(t)P(t-1)\varphi(t)} \end{aligned} \tag{2.67}$$

where we used the shorthand notations $\hat{x}(t)$ and $P(t)$, in place of $\hat{x}(t|t) = \hat{x}(t|t-1)$ and $P(t|t) = P(t|t-1)$, respectively. It is easy to see that equations (2.67) coincide with those in (2.65), if we set $\sigma_e^2 = 1$. Notice that this can be done without loss of generality, as one can always scale the linear regression model by the standard deviation σ_e thus obtaining a new output equation

$$\tilde{y}(t) = \tilde{\varphi}^T(t)\theta + \tilde{e}(t) \tag{2.68}$$

where $\tilde{y}(t) = \frac{y(t)}{\sigma_e}$, $\tilde{\varphi}(t) = \frac{\varphi(t)}{\sigma_e}$, $\tilde{e}(t) = \frac{e(t)}{\sigma_e}$ and $\mathbf{E}\left[\tilde{e}^2(t)\right] = 1$. If we apply the RLS algorithm (2.65) to the linear regression model (2.68) , the resulting sequence of estimates $\hat{\theta}_t$ is exactly the same as the sequence $\hat{x}(t)$ returned by the recursion (2.67).

Chapter 3

State estimation for nonlinear systems

This chapter addresses the problem of state estimation for nonlinear stochastic systems.

3.1 Nonlinear State Estimation

Let us consider the system

$$\begin{cases} x(t+1) = f\left(x(t), u(t), w(t)\right) \\ y(t) = h\left(x(t)\right) + v(t) \end{cases} \tag{3.1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is a known deterministic input, $y(t) \in \mathbb{R}^p$ is the output, $w(t) \in \mathbb{R}^d$ is the process disturbance and $v(t) \in \mathbb{R}^p$ is the measurement noise. The functions $f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R}^n$ and $h : \mathbb{R}^n \to \mathbb{R}^p$ are assumed to be known, continuous and differentiable. For the process disturbance $w(t)$, measurement noise $v(t)$ and initial state $x(0)$ we enforce the same Assumption 2.1 as in the linear setting.

The problem we want to solve is the same as Problem 2.1: find an estimate of $x(t)$ based on input-output measurements up to time t. As we know, this is a Beyesian estimation problem, being both the unknown vector $x(t)$ and

the data Y^t random variables. With respect to the linear case, the main difficulty lies in the fact that the functions $f(\cdot)$ and $h(\cdot)$ can be any $\mathcal{C}^1$ nonlinear mapping. Therefore, even if we restrict our attention to the LMSE estimate, the relevant covariance matrices at time t cannot be expressed only in terms of covariance matrices at previous times, because they will depend also on higher order moments (through the nonlinear model functions).

Since the exact solution of the LMSE problem is intractable, a wide variety of approximations have been proposed in the literature. It is important to keep in mind that all these techniques do not provide the actual LMSE state estimate, and hence they must be tested in simulations and real-world experiments in order to assess their performance in the specific application at hand. The most popular approach is the one based on the linearization of the model equations, which is known as *Extended Kalman Filter (EKF)*.

3.1.1 The Extended Kalman Filter

The EKF is a recursive procedure based on the same prediction-correction approach adopted in the linear KF. The notation adopted is the same as in Section 2.4, although both $\hat{x}(t|t)$ and $P(t|t)$ will end up to be only *approximations* of the true LMSE state estimate and covariance of the estimation error, respectively. We start by deriving the equations of the prediction step.

3.1.2 The EKF prediction step

Let $\hat{x}(t|t)$ and $P(t|t)$ be given. The objective is to compute the 1-step ahead prediction $\hat{x}(t+1|t)$ and the corresponding error covariance $P(t+1|t)$. To do this, we write the Taylor expansion of $f(x, u, w)$ around the nominal values $x = \hat{x}(t|t)$, $u = u(t)$ (recall that $u(t)$ is known at every time t) and $w = 0$

(being $\mathbf{E}[w(t)] = 0$). Then, one has

$$f(x,u,w) = f(\hat{x}(t|t),u(t),0) + \left.\frac{\partial f}{\partial x}\right|_{\substack{x=\hat{x}(t|t)\\u=u(t)\\w=0}} (x-\hat{x}(t|t)) + \left.\frac{\partial f}{\partial u}\right|_{\substack{x=\hat{x}(t|t)\\u=u(t)\\w=0}} (u-u(t)) + \left.\frac{\partial f}{\partial w}\right|_{\substack{x=\hat{x}(t|t)\\u=u(t)\\w=0}} (w-0) + O\left(\left|\left|\begin{smallmatrix} x-\hat{x}(t|t)\\ u-u(t)\\ w-0\end{smallmatrix}\right|\right|^2\right)$$

where the last term contains all the terms of degree equal or higher than 2 in the involved variables. The main idea is that, *if these terms are small*, they can be neglected with respect to the linear terms. Hence, by doing so, we obtain the following approximated version of the first equation in (3.1)

$$x(t+1) \simeq f(\hat{x}(t|t),u(t),0) + \left.\frac{\partial f}{\partial x}\right|_{\substack{x=\hat{x}(t|t)\\u=u(t)\\w=0}} (x(t)-\hat{x}(t|t)) + \left.\frac{\partial f}{\partial w}\right|_{\substack{x=\hat{x}(t|t)\\u=u(t)\\w=0}} w(t) \tag{3.2}$$

In order to obtain $\hat{x}(t+1|t)$, we have to replace $x(t)$ and $w(t)$ in (3.2) with their best estimates based on the data available up to time t, which are clearly $\hat{x}(t|t)$ and 0, respectively (recall that being $w(t)$ white, its best prediction is equal to the a priori expected value). Hence, one gets

$$\hat{x}(t+1|t) = f(\hat{x}(t|t),u(t),0) \tag{3.3}$$

which turns out to be the first equation of the EKF prediction step. In order to update the covariance matrix, let us introduce the following notation for the Jacobian matrices of $f(\cdot)$ with respect to x and w

$$F(t) = \left.\frac{\partial f}{\partial x}\right|_{\substack{x=\hat{x}(t|t)\\u=u(t)\\w=0}} \in \mathbb{R}^{n\times n}$$

$$G(t) = \left.\frac{\partial f}{\partial w}\right|_{\substack{x=\hat{x}(t|t)\\u=u(t)\\w=0}} \in \mathbb{R}^{n\times d}$$

Then, by exploiting again (3.2) and (3.3), one gets

$$\begin{aligned} P(t+1|t) &= E\left[(x(t+1)-\hat{x}(t+1|t))\,(x(t+1)-\hat{x}(t+1|t))^T\right] \\ &\simeq E\left[\{f(\hat{x}(t|t),u(t),0)+F(t)(x(t)-\hat{x}(t|t))+G(t)w(t)\right. \\ &\qquad \left.-f(\hat{x}(t|t),u(t),0)\}\,\{\cdots\}^T\right] = \\ &= E\left[\{F(t)(x(t)-\hat{x}(t|t))+G(t)w(t)\}\,\{\cdots\}^T\right] = \\ &= F(t)P(t|t)F(t)^T+G(t)QG(t)^T. \end{aligned}$$

3.1.3 The EKF correction step

Let $\hat{x}(t+1|t), P(t+1|t)$ and $y(t+1)$ be given. Recall that

$$y(t+1) = h(x(t+1)) + v(t+1) \tag{3.4}$$

Let us write the Taylor expansion of $h(x)$ around the nominal value $x = \hat{x}(t+1|t)$ (the best available guess of $x(t+1)$ at time t). One has

$$\begin{aligned} h(x) = h(\hat{x}(t+1|t)) &+ \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}(t+1|t)} (x-\hat{x}(t+1|t)) \\ &+ O\left(\left|\left|x-\hat{x}(t+1|t)\right|\right|^2\right) \end{aligned} \tag{3.5}$$

where the last term contains all the terms of order equal or higher than 2 in the error $x-\hat{x}(t+1|t)$. As in the prediction step, we define the Jacobian matrix

$$H(t+1) = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}(t+1|t)} \in \mathbb{R}^{p\times n}.$$

By using the expansion (3.5) into (3.4) and neglecting higher order terms, we get

$$y(t+1) \simeq h(\hat{x}(t+1|t)) + H(t+1)(x(t+1)-\hat{x}(t+1|t)) + v(t+1) \tag{3.6}$$

Now, denote the prediction error as

$$d(t+1) = x(t+1) - \hat{x}(t+1|t) \tag{3.7}$$

and set $m(t+1) = y(t+1) - h(\hat{x}(t+1|t))$. Then, (3.6) becomes

$$m(t+1) = H(t+1)d(t+1) + v(t+1). \tag{3.8}$$

By noticing that $\hat{d}(t+1|t) = \hat{x}(t+1|t) - \hat{x}(t+1|t) = 0$ and

$$\begin{aligned} &E\left[(d(t+1) - \hat{d}(t+1|t))(d(t+1) - \hat{d}(t+1|t))^T\right] \\ &= E\left[(x(t+1) - \hat{x}(t+1|t))(x(t+1) - \hat{x}(t+1|t))^T\right] = P(t+1|t) \end{aligned}$$

one can apply the correction step of the standard Kalman filter to the linear output equation (3.8) (observe that $m(t+1)$ is known). Therefore, one gets

$$\begin{aligned} \hat{d}(t+1|t+1) &= \hat{d}(t+1|t) + K(t+1)\left(m(t+1) - H(t+1)\hat{d}(t+1|t)\right) \\ &= K(t+1)(y(t+1) - h(\hat{x}(t+1|t))). \end{aligned}$$

From (3.7), one has $\hat{d}(t+1|t+1) = \hat{x}(t+1|t+1) - \hat{x}(t+1|t)$, which leads to the first EKF correction equation

$$\hat{x}(t+1|t+1) = \hat{x}(t+1|t) + K(t+1)(y(t+1) - h(\hat{x}(t+1|t))) \tag{3.9}$$

where $K(t+1)$ is defined according to the standard KF applied to (3.8), i.e.

$$K(t+1) = P(t+1|t)H(t+1)^T\left[H(t+1)P(t+1|t)H(t+1)^T + R\right]^{-1} \tag{3.10}$$

and similarly one has that the update of the covariance matrix $P(t+1|t+1)$ is given by

$$P(t+1|t+1) = P(t+1|t)\left[I - H(t+1)^T K(t+1)^T\right]. \tag{3.11}$$

Hence, by assuming to initialize the EKF recursion in the same way as in the KF, the equations of the EKF algorithm can be summarized as follows.

Initialization: $\hat{x}(0|-1) = m_0,\ P(0|-1) = P_0$

For $t = 0, 1, 2, \cdots$

$$K(t) = P(t|t-1)H(t)^T[H(t)P(t|t-1)H(t)^T + R]^{-1} \tag{3.12}$$

$$\hat{x}(t|t) = \hat{x}(t|t-1) + K(t)(y(t) - h(\hat{x}(t|t-1))) \tag{3.13}$$

$$H(t) = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}(t|t-1)} \tag{3.14}$$

$$P(t|t) = P(t|t-1)[I - H(t)^T K(t)^T] \tag{3.15}$$

$$\hat{x}(t+1|t) = f(\hat{x}(t|t), u(t), 0) \tag{3.16}$$

$$F(t) = \left.\frac{\partial f}{\partial x}\right|_{\substack{x=\hat{x}(t|t)\\u=u(t)\\w=0}}, \quad G(t) = \left.\frac{\partial f}{\partial w}\right|_{\substack{x=\hat{x}(t|t)\\u=u(t)\\w=0}} \tag{3.17}$$

$$P(t+1|t) = F(t)P(t|t)F(t)^T + G(t)QG(t)^T \tag{3.18}$$

end

3.1.4 Properties of the EKF

It is worth pointing out the main differences between the EKF and the KF derived for linear systems.

First, it is necessary to stress that the estimates $\hat{x}(t|t)$ and $\hat{x}(t+1|t)$ provided by the EKF are not the LMSE estimates of $x(t)$ and $x(t+1)$, respectively, based on Y^t. This is due to the approximations introduced in the linearization of the functions $f(x, u, w)$ and $h(x)$. Similarly, $P(t|t)$ and $P(t+1|t)$ are not the covariance matrices of the estimation errors, but only their approximations. How good such approximations are may depend on several factors, including the initial conditions $\hat{x}(0|-1)$, $P(0|-1)$. Therefore, special care must be taken in the choice of such initial values, by exploiting the available a priori knowledge on the variables to be estimated.

Another main difference with the linear case is that the matrices $F(t)$,

$G(t)$ and $H(t)$ depend on the current estimates $\hat{x}(t|t)$ and $\hat{x}(t+1|t)$ (that are the values at which the Jacobian matrice are computed), which in turn depend on the data Y^t. As a consequence, the matrices $K(t), P(t|t), P(t+1|t)$ cannot be precomputed as in the linear case. Moreover, they depend on the specific data realization processed, which means that also the uncertainty associated to the estimates actually depend on the data set. In other words, we cannot assess the quality of the estimates before computing them.

In general, there are no guarantees that the estimates provided by the EKF are satisfactory. Loosely speaking, neglecting the higher order terms is reasonable only if the estimation errors are "small", which clearly leads to a circular reasoning. The possible undesired behaviors that can be observed can be divided in two types: *divergence* occurs when the error grows arbitrarily, i.e.

$$\lim_{t\to\infty} ||x(t) - \hat{x}(t|t)|| = +\infty.$$

On the other hand, even if the error remains bounded, one may face *inconsistency* of the estimates. This happens if

$$E\left[(x(t) - \hat{x}(t|t))(x(t) - \hat{x}(t|t))^T\right] \gg P(t|t)$$

that is, if the actual uncertainty is much larger than the one evaluated by the EKF. In such a case, one may make overoptimistic statements about the quality of the estimates delivered by the filter. Said another way, the true state values may be significantly faraway from the confidence intervals derived from the filter estimates.

State estimation for nonlinear dynamic systems is still an active research area. Whenever the solution provided by the EKF is not satisfactory, one may resort to a variety of alternatives that have been proposed in the literature. Some of them are briefly presented in the next sections.

3.2 The Continuous-Discrete Kalman Filter

In many real-world applications, the system dynamics are described by a set of continuous-time differential equations, while the output measurements

are available only at discrete time instants. For example, this is the case of sampled-data systems, in which the physical behavior is usually described by a state-space continuous-time model, while the sensor measurements are functions of the state vector, sampled at a given clock rate or made available at asynchronous time instants. For a linear time-invariant system, this setting is captured by the equations

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + Gw(t) \\ y(t_k) = Cx(t_k) + v(t_k) \qquad k = 0, 1, 2, \dots \end{cases}$$

where $t_0 < t_1 < t_2 < \dots$ are the discrete time instants at which measurements are collected. It is worth observing that in this case $w(t)$ is a continuous-time stochastic process which is assumed to be stationary and white. This means that its covariance function is given by $\mathbf{E}\left[w(t+\tau)w^T(t)\right] = Q\delta(\tau)$, where Q is the spectral density matrix. We also set $\mathbf{E}\left[w(t)\right] = 0$ and make the usual assumptions on the output noise, i.e., $v(t_k) \sim WP\left(0, R\right)$ and it is independent from $w(t)$.

The LMSE state estimates for system (3.2) is given by the following recursive equations:

Initialization: $\hat{x}(t_0|t_{-1}) = m_0,\ P(t_0|t_{-1}) = P_0$

For $k = 0, 1, 2, \cdots$

$$K(t_k) = P(t_k|t_{k-1})C^T[CP(t_k|t_{k-1})C^T + R]^{-1} \tag{3.19}$$

$$\hat{x}(t_k|t_k) = \hat{x}(t_k|t_{k-1}) + K(t_k)(y(t_k) - C\hat{x}(t_k|t_{k-1})) \tag{3.20}$$

$$P(t_k|t_k) = P(t_k|t_{k-1})[I - C^T K(t_k)^T] \tag{3.21}$$

$$\frac{d}{dt}\hat{x}(t|t_k) = A\hat{x}(t|t_k) + Bu(t)\ , \qquad \text{for } t \in [t_k, t_{k+1}) \tag{3.22}$$

$$\frac{d}{dt}P(t|t_k) = AP(t|t_k) + P(t|t_k)A^T + GQG^T\ , \quad \text{for } t \in [t_k, t_{k+1}) \tag{3.23}$$

end

Algorithm (3.19)-(3.23) is known as the *Continuous-Discrete Kalman Filter (CDKF)*. While equations (3.19)-(3.21) are the same as in the standard

discrete-time Kalman filter, the equations of the prediction step (3.22)-(3.23) are a set of differential equations which allow one to compute the prediction of the state and the associated covariance matrix of the prediction error along the time interval $t \in [t_k, t_{k+1})$, until the next measurements $y(t_{k+1})$ becomes available.

It is worth observing that (3.22) are n first-order linear differential equations in the state components, while (3.23) correspond to $\frac{n(n+1)}{2}$ first-order linear differential equations in the independent entries of $P(t|t_k)$ (recall that P is symmetric). Hence the solutions of these two systems of differential equations can be computed analytically. It must be also stressed that the resulting state estimate trajectories are piecewise continuous, as $\hat{x}(t|t_k)$ is continuous in the interval $t \in (t_k, t_{k+1})$, while jumps occur at the discrete time instants t_k due to the correction step (the same occurs also for the entries of P).

Clearly, a possible alternative to the CDKF described above is to first discretize the continuous-time dynamics of system (3.2) and then apply the discrete-time Kalman filter to the discretized system. This usually works well for linear systems, which can be discretized *exactly*, i.e., without introducing errors in the dynamics due to the discretization. On the other hand, the state estimation problem becomes much more challenging when dealing with nonlinear systems. In particular, if the time interval between two time samples t_k and t_{k+1} is long (compared to the time constants of the system dynamics), a poor discretization of the system dynamics may lead to significant errors, which in turn can generate inconsistency of the estimates or even divergence of the filter.

A possible solution of the state estimation problems in the case of sampled-data nonlinear systems is provided by the *Continuous-Discrete Extended Kalman Filter (CDEKF)* described next.

Consider the system

$$\begin{cases} \dot{x}(t) = f\left(x(t), u(t), w(t)\right) \\ y(t_k) = h\left(x(t_k)\right) + v(t_k) \end{cases} \tag{3.24}$$

in which $w(t)$ and $v(t_k)$ satisfy the same assumptions as in the linear case treated above. Then, the equations of the CDEKF algorithm are given by

Initialization: $\hat{x}(t_0|t_{-1}) = m_0,\ P(t_0|t_{-1}) = P_0$

For $k = 0, 1, 2, \cdots$

$$K(t_k) = P(t_k|t_{k-1})H^T(t_k)[H(t_k)P(t_k|t_{k-1})H^T(t_k) + R]^{-1} \tag{3.25}$$

$$\hat{x}(t_k|t_k) = \hat{x}(t_k|t_{k-1}) + K(t_k)(y(t_k) - h(\hat{x}(t_k|t_{k-1}))) \tag{3.26}$$

$$P(t_k|t_k) = P(t_k|t_{k-1})[I - H^T(t_k)K(t_k)^T] \tag{3.27}$$

$$\frac{d}{dt}\hat{x}(t|t_k) = f(\hat{x}(t|t_k), u(t), 0)\ , \qquad \text{for } t \in [t_k, t_{k+1}) \tag{3.28}$$

$$\frac{d}{dt}P(t|t_k) = F(t_k)P(t|t_k) + P(t|t_k)F^T(t_k) + G(t_k)QG^T(t_k)\ , \qquad \text{for } t \in [t_k, t_{k+1}) \tag{3.29}$$

end

where

$$F(t_k) = \left.\frac{\partial f}{\partial x}\right|_{\substack{x=\hat{x}(t_k|t_k)\\ u=u(t_k)\\ w=0}} \in \mathbb{R}^{n\times n} \tag{3.30}$$

$$G(t_k) = \left.\frac{\partial f}{\partial w}\right|_{\substack{x=\hat{x}(t_k|t_k)\\ u=u(t_k)\\ w=0}} \in \mathbb{R}^{n\times d} \tag{3.31}$$

$$H(t_k) = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}(t_k|t_k-1)} \in \mathbb{R}^{p\times n} \tag{3.32}$$

As in the linear case, the correction step composed by equations (3.25)-(3.27) are analogous to those of the discrete-time EKF. The prediction step (3.28)-(3.29) consists of n nonlinear differential equations in the state predictions $\hat{x}(t|t_k)$ and $\frac{n(n+1)}{2}$ linear differential equations in the entries of $P(t|t_k)$. Such equations must be integrated over the interval $t \in [t_k, t_{k+1})$. In particular, the crucial task is the integration of the nonlinear equations (3.28). In fact, the main reason why the CDEKF is often successfully employed in applications involving sampled-data systems is that it allows one to precisely evolve

the state estimates between two consecutive measurements, through precise numerical integration of the prediction step equations.

Clearly, the same observations made for the EKF in the discrete-time setting, apply also to the CDEKF. In particular, there is no guarantee that the expected value of the state estimation error converge to zero, nor that $P(t|t_k)$ be equal to the covariance matrix of the estimation error. In this respect, the role of the initial conditions $\hat{x}(t_0|t_{-1})$ and $P(t_0|t_{-1})$ may be crucial to promote a satisfactory behavior of the filter.

Remark 3.1. In principle, the Jacobian matrices (3.30) and (3.31) could be computed by linearizing the system around the current prediction $\hat{x}(t|t_k)$ and the current input $u(t)$, instead of their corresponding values at the beginning of the integration interval $[t_k, t_{k+1})$. Notice however that this would make the matrices F and G time-varying and, more importantly, dependent on the variables $\hat{x}(t|t_k)$ themselves! As a consequence, the equations (3.28)-(3.29) would be coupled, that is they would become a unique system of nonlinear differential equations in $n + \frac{n(n+1)}{2}$ variables. The resulting increase of the computational burden may not be worth the advantage provided by a more precise approximation of the covariance matrix of the estimation errors.

3.3 The Unscented Kalman Filter

The Unscented Kalman Filter (UKF) is an algorithm for state estimation of nonlinear systems, based on the *unscented transform*. The key idea is to use a set of points in the state space, in order do match some relevant statistics of the a posteriori pdf of the state $f_x(x(t)|Y^t)$, and then suitably propagate and update such points according to the equations of the nonlinear dynamic system.

3.3.1 The Unscented Transform

Let us consider a set of p points in $\mathbb{R}^n$, hereafter referred to as *sigma points* and denoted by $X^{(i)} \in \mathbb{R}^n$, for $i = 1, \ldots, p$. To each sigma point we associate

a scalar weight $W^{(i)} \in \mathbb{R}$, $i = 1, \ldots, p$, such that $\sum_{i=1}^{p} W^{(i)} = 1$. The sigma points are chosen in such a way that their (weighted) sample statistics are equal to the selected statistics of the target a posteriori pdf.

Example 3.1. Let $x \in \mathbb{R}^n$ be a Gaussian random variable, $x \sim \mathcal{N}(m, P)$. Let us choose $p = 2n$ sigma points as

$$X^{(i)} = m + (\sqrt{nP})_i \quad i = 1, \cdots, n \tag{3.33}$$

$$X^{(i)} = m - (\sqrt{nP})_{i-n} \quad i = n+1, \cdots, 2n \tag{3.34}$$

$$W^{(i)} = \frac{1}{2n} \quad i = 1, \cdots, 2n \tag{3.35}$$

where $(M)_i$ denotes the i-th column of matrix M, while for M and U square matrices, $\sqrt{M} = U$ means that $M = UU^T$. It is easy to show that the weighted mean and covariance matrix of the sigma points match those of the original Gaussian pdf, i.e.

$$\sum_{i=1}^{2n} W^{(i)} X^{(i)} = m,$$

$$\sum_{i=1}^{2n} W^{(i)} (X^{(i)} - m)(X^{(i)} - m)^T = P.$$

We aim to use the sigma points to propagate the relevant statistics of a random variable that undergoes a nonlinear transformation. Consider the non linear function $h : \mathbb{R}^n \to \mathbb{R}^n$ such that

$$z = h(x)$$

Let $X^{(i)}, W^{(i)}$ be a set of sigma points for x. If we apply the nonlinear function $h(\cdot)$ to the sigma points, we obtain the new set of sigma points

$$Z^{(i)} = h(X^{(i)}) \quad i = 1, \cdots, p.$$

Then, we approximate the statistic of z with the new sigma points. For example, the mean

$$\mathbf{E}[z] \simeq \sum_{i=1}^{p} W^{(i)} Z^{(i)} = m_z$$

and the covariance matrix

$$\mathrm{Cov}(z) \simeq \sum_{i=1}^{p} W^{(i)}(Z^{(i)} - m_z)(Z^{(i)} - m_z)^T$$

It is worth stressing that the set of sigma points and weights (3.33)-(3.35) is just one of the many possible choices. In particular, by enriching the set of sigma points it is possible to match other moments of the target pdf, thus improving the quality of the approximation. For instance, considering again Example 3.1, by choosing the set of $2n+1$ sigma points

$$X^{(i)} = m + (\sqrt{3P})_i \quad i = 1, \cdots, n \tag{3.36}$$

$$X^{(i)} = m - (\sqrt{3P})_{i-n} \quad i = n+1, \cdots, 2n \tag{3.37}$$

$$X^{(2n+1)} = m \tag{3.38}$$

$$W^{(i)} = \frac{1}{6} \quad i = 1, \cdots, 2n \tag{3.39}$$

$$W^{(2n+1)} = 1 - \frac{n}{3} \tag{3.40}$$

it is possible to match not only the mean and covariance of the pdf $\mathcal{N}(m, P)$, but also the 4-th order moments $\mathbf{E}\left[(x_i - m_i)^4\right] = 3P_{ii}^2$, $i = 1, \ldots, n$. Notice that this comes at the price of adding only one further sigma point with respect to the set (3.33)-(3.35).

Example 3.2. Let us consider the mapping from polar coordinates $\xi = [\rho\ \theta]^T$ to cartesian coordinates $z = [x\ y]^T$, defined as

$$z = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \rho\cos\theta \\ \rho\sin\theta \end{bmatrix} = h\left(\begin{bmatrix} \rho \\ \theta \end{bmatrix}\right) = h(\xi)$$

Assume that M samples of polar coordinates are drawn from the pdf

$$\begin{bmatrix} \rho \\ \theta \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m_\rho \\ m_\theta \end{bmatrix}, \begin{bmatrix} \sigma_\rho^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix}\right) \tag{3.41}$$

By using the linearization approach, as in the EKF, the mean and covariance

of the distribution of the cartesian coordinates can be approximated as

$$\mathbf{E}\left[z\right] \simeq h\left(\begin{bmatrix} m_\rho \\ m_\theta \end{bmatrix}\right) = \begin{bmatrix} m_\rho \cos(m_\theta) \\ m_\rho \sin(m_\theta) \end{bmatrix}$$
$$\mathrm{Cov}(z) \simeq H \begin{bmatrix} \sigma_\rho^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix} H^T$$

where

$$H = \left.\frac{\partial h}{\partial \xi}\right|_{\substack{\rho=m_\rho \\ \theta=m_\theta}} = \begin{bmatrix} \cos(m_\theta) & -m_\rho \sin(m_\theta) \\ \sin(m_\theta) & m_\rho \cos(m_\theta) \end{bmatrix}$$

. In order to compare the approximations provided by the linearization approach to that based on the unscent transform, we consider the estimated confidence ellipses, defined as $(z-\hat{z})^T P_z^{-1}(z-\hat{z}) \leq r_\chi$, in which $\hat{z}$ and P_z are the estimates of $\mathbf{E}\left[z\right]$ and $\mathrm{Cov}(z)$, respectively. We set $r_\chi = 9.21$, which corresponds to a 99% confidence level. We generated $M = 1000$ points $\xi = [\rho\ \theta]^T$ distributed according to the pdf (3.41), with $m_\rho = 1$, $m_\theta = \frac{\pi}{2}$, $\sigma_\rho = 0.02$, $\sigma_\theta = \frac{15\pi}{180}$. In Figure 3.1, such points are shown (in blue) together with different estimates of the 99% confidence ellipses. The dashed black line corresponds to the ellipse generated by the sample mean and covariances of the 1000 points and can be considered as a ground truth. The confidence ellipse provided by the linearization-based approach is shown in red. The ellipses generated from the unscented transforms using the sets of sigma points (3.33)-(3.35) and (3.36)-(3.40) are shown respectively in light blue and in green. It can be observed that the approximation based on the linearization underestimates significantly the covariance of the transformed vector $z = [x\ y]^T$. Conversely, the ellipse resulting from the first set of sigma point is much closer to that based on sample estimates, which is almost perfectly matched by the ellipse corresponding to the extended set of sigma points.

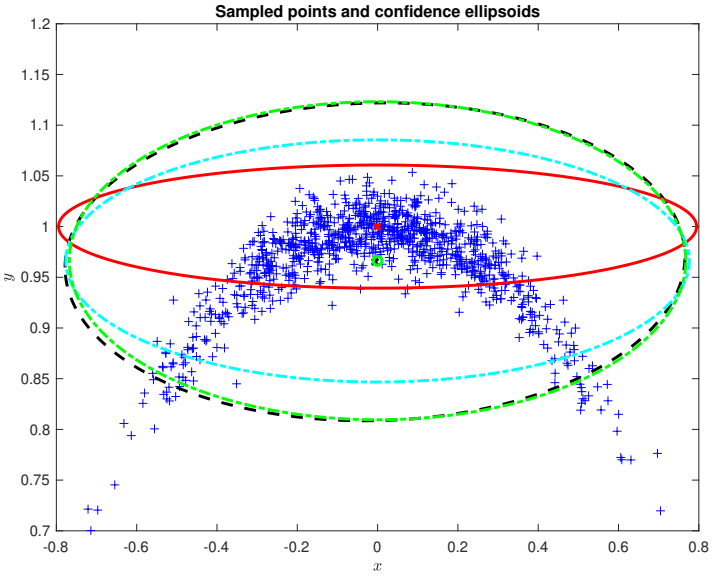


Figure 3.1: Example 3.2. Comparison of confidence ellipses: linearization (red), unscented transform based on (3.33)-(3.35) (light blue); unscented transform based on (3.36)-(3.40) (green). The blue crosses are the points drawn from the pdf (3.41).

3.3.2 The UKF Algorithm

The main idea behind the UKF algorithm is to use the sigma points to propagate the statistics of the posterior pdf $f_x(x(t)|Y^t)$, using a prediction-correction structure similar to that of the EKF. Consider system (3.1) and let Assumption 2.1 hold. Let $\hat{x}(t|t)$ and $P(t|t)$ have the same meaning as in the EKF and assume they are given at time t. The prediction and correction steps of the UKF are summarized next. For simplicity, it is assumed that the same number p of sigma points, with the same set of weights $W^{(i)}$, are generated for the distributions of $x(t)$, $w(t)$ and $v(t)$.

Prediction

- Generate the sigma points $\{X^{(i)}, W^{(i)}\}$ so that they have mean $\hat{x}(t|t)$

and covariance $P(t|t)$.

- Generate the sigma points $\{\Xi^{(i)}, W^{(i)}\}$ so that they match some relevant statistics of the pdf of $w(t)$ (e.g., the mean 0 and the covariance Q).

- Compute the predicted sigma points

$$\hat{X}^{(i)} = f(X^{(i)}, u(t), \Xi^{(i)}) \quad i = 1, \ldots, p$$

- Compute the (approximated) mean and covariance of the a posteriori pdf $f_x(x(t+1)|Y^t)$ as

$$\hat{x}(t+1|t) = \sum_{i=1}^{p} W^{(i)} \hat{X}^{(i)}$$
$$P(t+1|t) = \sum_{i=1}^{p} W^{(i)} (\hat{X}^{(i)} - \hat{x}(t+1|t))(\hat{X}^{(i)} - \hat{x}(t+1|t))^T$$

Correction

- Generate the sigma points $\{V^{(i)}, W^{(i)}\}$ so that they match some relevant statistics of the pdf of $v(t)$ (e.g., the mean 0 and the covariance R).

- Compute the output sigma points

$$\hat{Y}^{(i)} = h(\hat{X}^{(i)}) + V^{(i)} \quad i = 1, \cdots, p$$

- Set

$$\hat{y}(t+1) = \sum_{i=1}^{p} W^{(i)} \hat{Y}^{(i)}$$
$$S(t+1) = \sum_{i=1}^{p} W^{(i)} (\hat{Y}^{(i)} - \hat{y}(t+1))(\hat{Y}^{(i)} - \hat{y}(t+1))^T$$
$$P_{xy}(t+1) = \sum_{i=1}^{p} W^{(i)} (\hat{X}^{(i)} - \hat{x}(t+1|t))(\hat{Y}^{(i)} - \hat{y}(t+1))^T$$

- Compute the (approximated) mean and covariance of the a posteriori pdf $f_x(x(t+1)|Y^{t+1})$ as

$$\hat{x}(t+1|t+1) = \hat{x}(t+1|t) + P_{xy}(t+1)S(t+1)^{-1}\left(y(t+1) - \hat{y}(t+1)\right)$$
$$P(t+1|t+1) = P(t+1|t) - P_{xy}(t+1)S(t+1)^{-1}P_{xy}(t+1)^T.$$

It is apparent that the last two equations derive from the LMSE estimation expressions, in which the relevant covariances are obtained from the corresponding sample statistics of the sigma points.

It is worth stressing that many different versions of the UKF have been proposed in the literature. The one presented above contains only the basic features, but the tool is flexible enough to allow several interesting extensions. For example, one may take into account cross-correlations between state variables, process disturbances and measurement noise, by generating a set of sigma points representative of the joint statistics of an extended vector including $x(t)$, $w(t)$ and $v(t)$. Moreover, one may exploit a priori knowledge on the functional form of the involved distributions and generate and extended set of sigma points matching higher order statistics, beyond the mean and the covariance. For further details, see the references (Julier and Uhlmann, 2004; Wan *et al.*, 2001).

3.4 The Particle Filter

Ideally, the aim of the state estimation problem is to find an estimate of the a posteriori pdf of the state, $f_x(x(t)|Y^t)$. Then, one can compute the MSE estimate of $x(t)$ as

$$\mathbf{E}\left[x(t)|Y^t\right] = \int x f(x|Y^t)dx.$$

In the linear Gaussian case (i.e., if the system is linear and the stochastic processes $w(t)$ and $v(t)$ have a Gaussian pdf), it turns out that

$$f(x(t)|Y^t) = N(\hat{x}(t|t), P(t|t))$$

where $\hat{x}(t|t)$ and $P(t|t)$ can be computed through the KF iterations. If the Gaussian assumption is not satisfied, the KF still provides the LMSE estimate of the state. However, in the nonlinear case, the a posteriori pdf of $x(t)$ can be significantly different from a Gaussian pdf. In such cases, the pdf $N(\hat{x}(t|t), P(t|t))$, with $\hat{x}(t|t)$ and $P(t|t)$ provided by the EKF, can be a very coarse approximation of the true pdf $f_x(x(t)|Y^t)$.

In principle, it is possible to recursively update the exact a posteriori pdf of the state, by using a prediction-correction iterative procedure, as explained next.

3.4.1 Recursive computation of the a posteriori state distribution

Consider again system (3.1) and assume that $w(t)$ and $v(t)$ are independent and distributed according to $f_w(w(t))$ and $f_v(v(t))$, respectively.

Prediction

Assume that $f_x(x(t)|Y^t)$ is known. From the theory of joint and conditional pdfs, one has

$$\begin{aligned} f_x(x(t+1)) &= \int f_x(x(t+1), x(t))dx(t) \\ &= \int f_x(x(t+1)|x(t))f_x(x(t))dx(t) \end{aligned}$$

The above equation still holds if we condition all the involved pdfs also to the data Y^t, thus giving

$$\begin{aligned} f_x(x(t+1)|Y^t) &= \int f_x(x(t+1)|x(t), Y^t)f_x(x(t)|Y^t)dx(t) \\ &= \int f_x(x(t+1)|x(t))f_x(x(t)|Y^t)dx(t) \end{aligned} \tag{3.42}$$

where the last equality is due to the fact that $x(t)$ is a Markov process. Therefore, in order to compute the predicted pdf $f_x(x(t+1)|Y^t)$, we need to know $f_x(x(t+1)|x(t))$, which in turn can be derived from the first equation in model (3.1) and the knowledge of $f_w(w(t))$.

Correction

From Bayes's formula, one has

$$\begin{aligned} f_x(x(t+1)|Y^{t+1}) &= f_x(x(t+1)|y(t+1),Y^t) \\ &= \frac{f_{xy}(x(t+1),y(t+1)|Y^t)}{f_y(y(t+1)|Y^t)} \\ &= \frac{f_y(y(t+1)|x(t+1))f_x(x(t+1)|Y^t)}{f_y(y(t+1)|Y^t)} \end{aligned} \tag{3.43}$$

where in the last equality we have exploited the fact that

$$f_y(y(t+1)|x(t+1),Y^t) = f_y(y(t+1)|x(t+1))$$

thanks once again to the fact that $x(t)$ is a Markov process. Now, being

$$\begin{aligned} f_y(y(t+1)|Y^t) &= \int f_y(y(t+1)|x(t+1),Y^t) f_x(x(t+1)|Y^t) dx(t+1) \\ &= \int f_y(y(t+1)|x(t+1)) f_x(x(t+1)|Y^t) dx(t+1) \end{aligned}$$

by substituting in (3.43) one gets

$$\begin{aligned} &f_x(x(t+1)|Y^{t+1}) = \\ &= \frac{f_y(y(t+1)|x(t+1))}{\int f_y(y(t+1)|x(t+1)) f_x(x(t+1)|Y^t) dx(t+1)} f_x(x(t+1)|Y^t). \end{aligned} \tag{3.44}$$

Therefore, in order to compute the corrected pdf $f_x(x(t+1)|Y^{t+1})$, we need to know $f_y(y(t+1)|x(t+1))$, which can be derived from the second equation in model (3.1) and the knowledge of $f_v(v(t))$. In particular, one has

$$f_y(y(t+1)|x(t+1)) = f_v(y(t+1) - h(x(t+1))). \tag{3.45}$$

The equations (3.42) and (3.44) can be used in principle to compute the exact a posteriori pdf of the state vector. However, such computations can be intractable even by numerical approximation techniques, especially when the dimension of the state vector is large. One way to practically approximate the a posterior pdf is through the so-called Monte Carlo sequential methods. The key idea is to use a relatively large set of points in the state space ("particles") whose sample distribution is as close as possible to the true a posteriori pdf $f_x(x(t)|Y^t)$. The resulting estimator, known as *Particle Filter (PF)*, is described next.

3.4.2 The Particle Filter Algorithm

Consider system (3.1) and let $f_w(w(t))$ and $f_v(v(t))$ be known. Assume that a set of particles x_t^i, for $i = 1, \cdots, N$, is available and that they are (approximately) distributed according to $f_x(x(t)|Y^t)$.

Prediction

- Generate N particles w_t^i, $i = 1, \cdots, N$, distributed according to $f_w(w(t))$.
- Compute the predicted particles

$$\hat{x}_{t+1}^i = f(x_t^i, u(t), w_t^i) \qquad i = 1, \cdots, N$$

  The particles $\hat{x}_{t+1}^i$ provide an approximation of $f_x(x(t+1)|Y^t)$.

Correction

- Generate the weights

$$q_i = \frac{f_y(y(t+1)|\hat{x}_{t+1}^i)}{\sum_{j=1}^{N} f_y(y(t+1)|\hat{x}_{t+1}^j)} \qquad i = 1, \cdots, N$$

  One clearly has $q_i \geq 0$, $\forall i$, and $\sum_{i=1}^N q_i = 1$. We use q_i as the probability mass function of $\hat{x}_{t+1}^i$ in the a posteriori pdf $f_x(x(t+1)|Y^{t+1})$. Notice that, according to (3.45), $f_y(y(t+1)|\hat{x}_{t+1}^i) = f_v(y(t+1) - h(\hat{x}_{t+1}^i))$.

- *Re-sampling.* Sample N-times from the set of particles $\hat{x}_{t+1}^i$ in such a way that the probability of extracting $\hat{x}_{t+1}^i$ is equal to q_i. This amounts to generate a new set of particles x_{t+1}^j, $j = 1, \ldots, N$, such that

$$P\{x_{t+1}^j = \hat{x}_{t+1}^i\} = q_i \ , \qquad j = 1, \cdots, N.$$

  The new set of particles x_{t+1}^j, $j = 1, \cdots, N$, is approximately distributed according to $f_x(x(t+1)|Y^{t+1})$.

We illustrate the application of the PF algorithm with some examples.

Example 3.3. Consider the system

$$\begin{cases} x(t+1) = x(t) + w(t) \\ y(t) = \frac{1}{20}x^2(t) + v(t) \end{cases}$$

where $w(t)$ and $v(t)$ satisfy Assumption 2.1 with $Q = 0.5$ and $R = 0.05$, and they are normally distributed. The EKF and a particle filter with 5000 particles have been run on a realization of the output $y(t)$, for $t = 0, 1, \ldots, 50$. Figure 3.2 shows the obtained results at time $t = 50$. The histogram of the particles, shown in blue, represents an approximation of the state a posteriori pdf $f_x(x(t)|Y^t)$. It can be noticed that it is a bimodal pdf, with maxima close to 4 and -4. This is due to the output nonlinearity which essentially provides the same nominal output, irrespectively of the sign of $x(t)$. The Gaussian pdf corresponding to the EKF mean and covariance estimates is depicted in red. It is apparent that it matches only one lobe of the a posteriori pdf. The true state $x(t)$ is marked with a black cross, while the estimate $\hat{x}(t|t)$ provided by the EKF with a red cross. In this case, the estimate of the EKF is affected by a large error, due to the fact that the true state is located in the lobe of the a posteriori pdf which is not matched by the EKF.

Example 3.4. Let a consider a 2D localization problem in which an agent of coordinates $(x(t), y(t))$ is tracked by two radars, placed respectively at locations $(0, 0)$ and $(1, 1)$. The radars return only distance measurements, given by

$$\begin{aligned} d_1(t) &= \sqrt{x^2(t) + y^2(t)} + v_1(t), \\ d_2(t) &= \sqrt{(x(t)-1)^2 + (y(t)-1)^2} + v_2(t). \end{aligned}$$

Assume the agent is fixed at position $(0, 1)$. Both radars measure a nominal (i.e., noiseless) distance equal to 1, which leads to an ambiguity in the position estimate, as both $(0, 1)$ and $(1, 0)$ are locations compatible with the available measurements. By adopting the dynamic model

$$\begin{aligned} x(t+1) &= x(t) \\ y(t+1) &= y(t) \end{aligned}$$

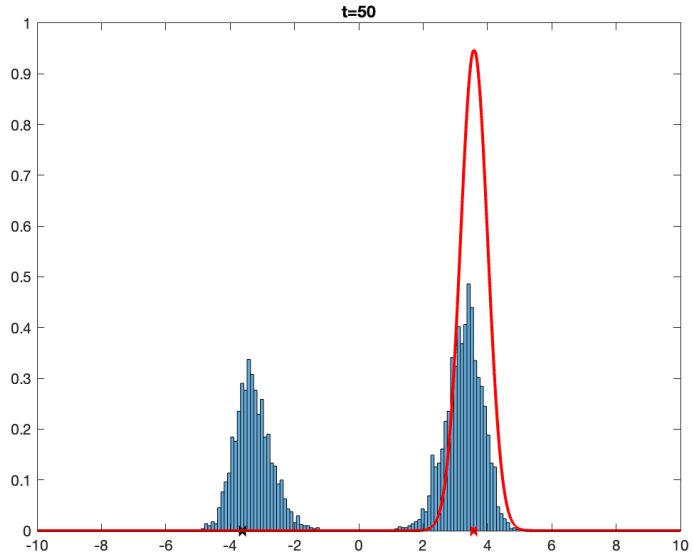


Figure 3.2: Example 3.3. Histogram of a posteriori pdf provided by PF (blue); estimate of a posteriori pdf returned by EKF (red); true state (black cross); EKF state estimate (red cross).

and running a PF with 3000 particles for 100 time instants, the situation depicted in Figure 3.3(left) is obtained. The blue circles represent the radar locations, while the red dotted circles are the positions compatible with their nominal measurements. The intersections of the circles (red crosses) are the positions compatible with the nominal measurements. The black cross is the estimate provided by the EKF, which is approximately halfway the two admissible positions. The particles of the PF are shown in green. It can be observed that they all collapsed to a single particle, located close to one of the admissible positions. Clearly, this is not a good approximation of the actual a posteriori pdf of the state.

This awkward behavior is due to the fact that the particles are initially chosen at random and then are progressively resampled in the correction steps, but always among the initial set of particles. Hence, the correction step only modifies the probabilities q_i associated to each particle, but not their values.

In the shown experiment, one particle was eventually resampled 3000 times, most likely because the probability of all the other particles became too small. In these situations, it is useful to inject a process disturbance in the state dynamics, with the purpose to "spread" the particles and promote the exploration of the state space. In this example, we adopt the random walk model

$$\begin{aligned} x(t+1) &= x(t) + w_x(t) \\ y(t+1) &= y(t) + w_y(t) \end{aligned}$$

where the $[w_x(t)\ w_y(t)]^T$ is generated as a with process with zero mean and variance $Q = \sigma_w^2 \cdot I$. By choosing $\sigma_w = 0.1$, one gets the particle distribution at time $t = 100$ shown in Figure 3.3(right). At it can be seen, now the particles are concentrated around the two admissible positions, thus providing a reliable approximations of the state pdf. Clearly, once the agent will move, the measurements will allow to discriminate between the two positions and the particles will be located with high probability close to the unique location compatible with the measurements.

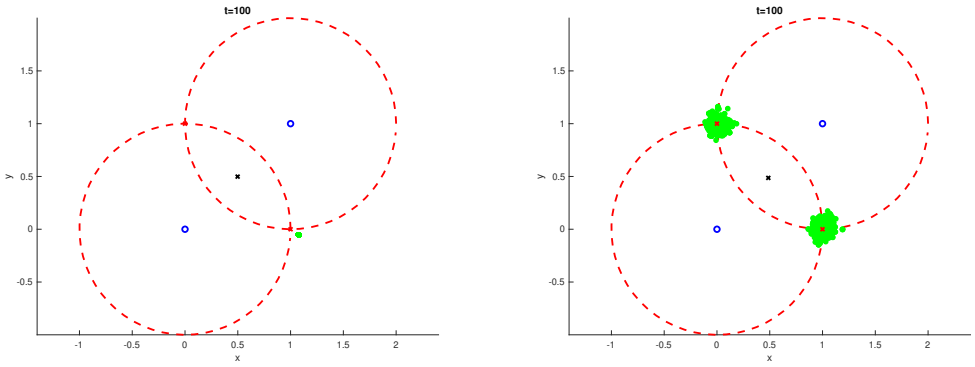


Figure 3.3: Example 3.4. Left: particle distribution without process disturbance. Right: particle distribution with process disturbance of standard deviation $\sigma_w = 0.1$. Particles (green); admissible positions (red crosses); EKF estimate (black cross).

As for the UKF, the version of the particle filter presented in this notes

is a basic one. A number of variations have been proposed in the literature which allow the user to exploit the potential of this approach to carefully approximate probability density functions even in state spaces of very high dimensions. We also hint to the fact that there are theoretical results showing that, under suitable technical assumptions, by increasing the number of particles N one can approximate the true pdf $f_x(x(t)|Y^t)$ with arbitrary precision. For a detailed treatment of the subject, see the references (Gordon *et al.*, 1993; Doucet *et al.*, 2001).

Bibliography

Doucet, Arnaud, Nando De Freitas and Neil Gordon (2001). An introduction to sequential Monte Carlo methods. In: *Sequential Monte Carlo methods in practice*. pp. 3–14. Springer.

Gordon, Neil J., David J. Salmond and Adrian F.M. Smith (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In: *IEE proceedings F (radar and signal processing)*. Vol. 140. IET. pp. 107–113.

Julier, Simon J. and Jeffrey K. Uhlmann (2004). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE* **92**(3), 401–422.

Wan, Eric A., Rudolph Van Der Merwe and Simon Haykin (2001). The unscented Kalman filter. *Kalman filtering and neural networks* **5**(2007), 221–280.