Chapitre 5

Kalman Filter

The purpose of the Kalman filter [3] is to use measurements that are observed over time that contain noise (random variations) to estimate the state vector of a dynamic system. The Kalman filter has many applications in technology, and is an essential part of the development of space, military technology and mobile robotics.

5.1 Covariance matrices

5.1.1 Definitions and interpretations

Consider two random vectors $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^m$. The esperances of $\mathbf{x}$ and $\mathbf{y}$ are denoted by $\bar{\mathbf{x}} = E(\mathbf{x})$, $\bar{\mathbf{y}} = E(\mathbf{y})$. We define the *variations* of $\mathbf{x}$ and $\mathbf{y}$ by $\widetilde{\mathbf{x}} = \mathbf{x} - \bar{\mathbf{x}}$ and $\widetilde{\mathbf{y}} = \mathbf{y} - \bar{\mathbf{y}}$. The *covariance matrix* is given by

$$\mathbf{\Gamma}_{\mathbf{xy}} = E\left(\widetilde{\mathbf{x}}.\widetilde{\mathbf{y}}^{\mathrm{T}}\right) = E\left((\mathbf{x} - \bar{\mathbf{x}})(\mathbf{y} - \bar{\mathbf{y}})^{\mathrm{T}}\right).$$

The covariance matrices for $\mathbf{x}$ and $\mathbf{y}$ are

$$\begin{array}{rcl} \mathbf{\Gamma}_{\mathbf{x}} & = & \mathbf{\Gamma}_{\mathbf{xx}} = E\left(\widetilde{\mathbf{x}}.\widetilde{\mathbf{x}}^{\mathrm{T}}\right) = E\left((\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^{\mathrm{T}}\right) \\ \mathbf{\Gamma}_{\mathbf{y}} & = & \mathbf{\Gamma}_{\mathbf{yy}} = E\left(\widetilde{\mathbf{y}}.\widetilde{\mathbf{y}}^{\mathrm{T}}\right) = E\left((\mathbf{y} - \bar{\mathbf{y}})(\mathbf{y} - \bar{\mathbf{y}})^{\mathrm{T}}\right). \end{array}$$

Note that $\mathbf{x}, \mathbf{y}, \widetilde{\mathbf{x}}, \widetilde{\mathbf{y}}$ are random vectors whereas $\bar{\mathbf{x}}, \bar{\mathbf{y}}, \mathbf{\Gamma}_{\mathbf{x}}, \mathbf{\Gamma}_{\mathbf{y}}, \mathbf{\Gamma}_{\mathbf{xy}}$ are deterministic. A covariance matrix $\mathbf{\Gamma}_{\mathbf{x}}$ of a random vector $\mathbf{x}$ is always positive definite (we shall write $\mathbf{\Gamma}_{\mathbf{x}} \succ \mathbf{0}$). A random vector can be represented by a cloud of points. Consider the following program

```
rand('normal');
x=2+rand(1000,1);
e=rand(1000,1);
y=2*x.^2+e;
plot2d(x,y,-1);
xbar=mean(x); ybar=mean(y);
```

```
xtilde=x-xbar ; ytilde=y-ybar ;
plot2d(xtilde,ytilde,-1) ;
G_xy=mean(xtilde.*ytilde) ;
```

We get Figure 5.1 which provides a representation of the random variables $x, y, \widetilde{x}, \widetilde{y}$. The program also yields the following evaluations :

$$\bar{x} \simeq 1.99, \bar{y} \simeq 9.983, \Gamma_x \simeq 1.003, \Gamma_y \simeq 74.03, \Gamma_{xy} \simeq 8.082.$$

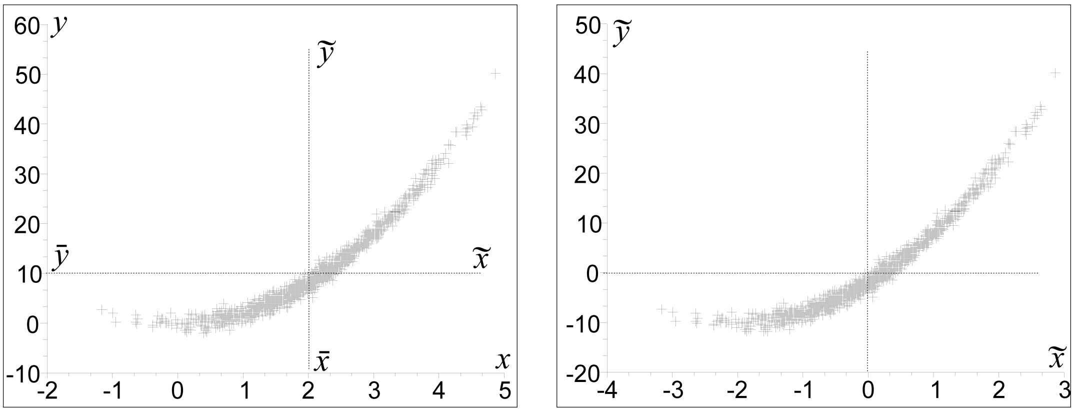


Figure 5.1 – A cloud of points to represent a pair of random variables

The two random vectors $\mathbf{x}$ and $\mathbf{y}$ are linearly independent (or uncorrelated or orthogonal) if $\mathbf{\Gamma_{xy}} = \mathbf{0}$. On Figure 5.2, both clouds of points correspond to uncorreleted random variables. Only the right subfigure corresponds to independent variables.

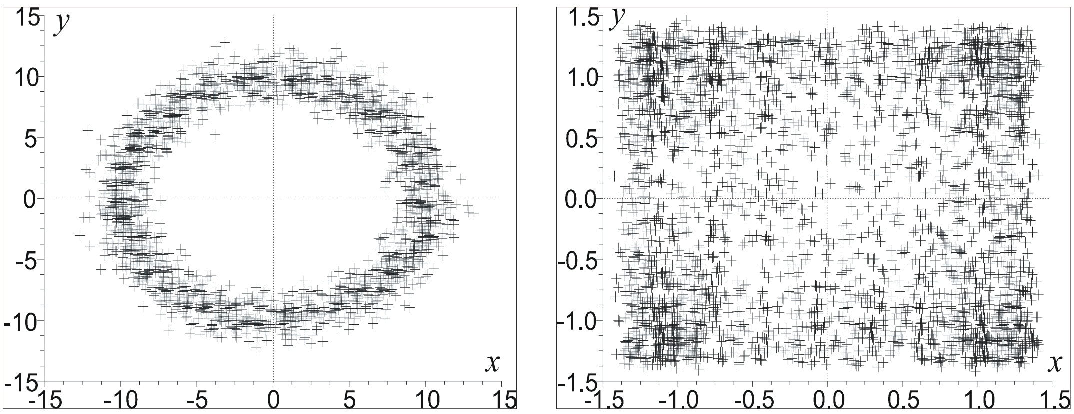


Figure 5.2 – Left : dependent but not correlated ; Right : independent

The left subfigure is generated by the following program

```
rand('normal') ; rho=10+rand(2000,1) ;
```

```
rand('uniform') ; theta=2*%pi*rand(2000,1) ;
x=rho.*sin(theta) ; y=rho.*cos(theta) ;
```

The right subfigure is obtained by the following program

```
rand('normal') ;
x=atan(2*rand(3000,1)) ;
y=atan(2*rand(3000,1)) ;
```

5.1.2 Properties

A covariance matrix is symmetric and positive, i.e., all its eigen values are real and non negative. The set of all covariance matrices of $\mathbb{R}^{n\times n}$ is denoted by $\mathcal{S}^+(\mathbb{R}^n)$.

Decomposition. A covariance matrix can be decomposed as

$$\mathbf{\Gamma} = \mathbf{R}^{-1}.\mathbf{D}.\mathbf{R}$$

where $\mathbf{R}$ is a rotation matrix (i.e. $\mathbf{R}^{\mathrm{T}}\mathbf{R} = \mathbf{I}$ and $\det \mathbf{R} = 1$) and where $\mathbf{D}$ is a diagonal matrix with positive elements on its diagonal.

Power. For $\alpha \in \mathbb{R}$ and $\mathbf{\Gamma}$ defined as above, we define

$$\mathbf{\Gamma}^{\alpha} = \mathbf{R}^{-1}.\mathbf{D}^{\alpha}.\mathbf{R},$$

where $\mathbf{D}^{\alpha}$ is the diagonal matrix whose diagonal elements are the d_{ii}^{α}. The matrix $\mathbf{\Gamma}^{\alpha}$ is also a covariance matrix. The eigen values of $\mathbf{\Gamma}^{\alpha}$ are the d_{ii}^{α}. The matrix $\mathbf{\Gamma}^{\frac{1}{2}}$ is called the *square root* of $\mathbf{\Gamma}$.

Order. The set $\mathcal{S}^+(\mathbb{R}^n)$ is a convex cone in $\mathbb{R}^{n\times n}$. As a consequence, if $\mathbf{\Gamma}_1$ and $\mathbf{\Gamma}_2$ belong to $\mathcal{S}^+(\mathbb{R}^n)$, then $\mathbf{\Gamma} = \mathbf{\Gamma}_1 + \mathbf{\Gamma}_2$ also belongs to $\mathcal{S}^+(\mathbb{R}^n)$. We define the following order

$$\mathbf{\Gamma}_1 \leq \mathbf{\Gamma}_2 \;\Leftrightarrow\; \mathbf{\Gamma}_2 - \mathbf{\Gamma}_1 \in \mathcal{S}^+(\mathbb{R}^n).$$

In such a case, the confidence ellipsoid (see next subsection) of level a of $\mathbf{\Gamma}_1$ is included inside the confidence ellipsoid of $\mathbf{\Gamma}_2$.

5.1.3 Confidence ellipsoid

A random vector $\mathbf{x}$ of $\mathbb{R}^n$ can be characterized by pair $(\bar{\mathbf{x}}, \mathbf{\Gamma}_{\mathbf{x}})$, to which we can associate an ellipsoid of $\mathbb{R}^n$ that is assumed to enclose consistent values for $\mathbf{x}$. In practice, for graphical reasons, we are only interested by two components $\mathbf{w} = (x_i, x_j)$ of $\mathbf{x}$. The mean $\bar{\mathbf{w}}$ can be obtained from $\bar{\mathbf{x}}$ by extracting the ith and jth entries. The covariance matrix $\mathbf{\Gamma}_{\mathbf{w}}$ can be obtained from $\mathbf{\Gamma}_{\mathbf{x}}$ by extracting the ith and jth rows and columns. The *confidence ellipse* associated with $\mathbf{w}$ has the form

$$\mathcal{E}_{\mathbf{w}} : (\mathbf{w} - \bar{\mathbf{w}})^{\mathrm{T}}\, \mathbf{\Gamma}_{\mathbf{w}}^{-1} (\mathbf{w} - \bar{\mathbf{w}}) \leq a^2,$$

where a is a positive threshold. Since $\mathbf{\Gamma}_{\mathbf{w}}^{-1} \succ \mathbf{0}$, it has a square root denoted by $\mathbf{\Gamma}_{\mathbf{w}}^{-\frac{1}{2}}$ which is also positive definite. Thus, we can write

$$\mathcal{E}_{\mathbf{w}} : (\mathbf{w} - \bar{\mathbf{w}})^{\mathrm{T}}\;\; \mathbf{\Gamma}_{\mathbf{w}}^{-\frac{1}{2}}.\mathbf{\Gamma}_{\mathbf{w}}^{-\frac{1}{2}} (\mathbf{w} - \bar{\mathbf{w}}) \leq a^2,$$

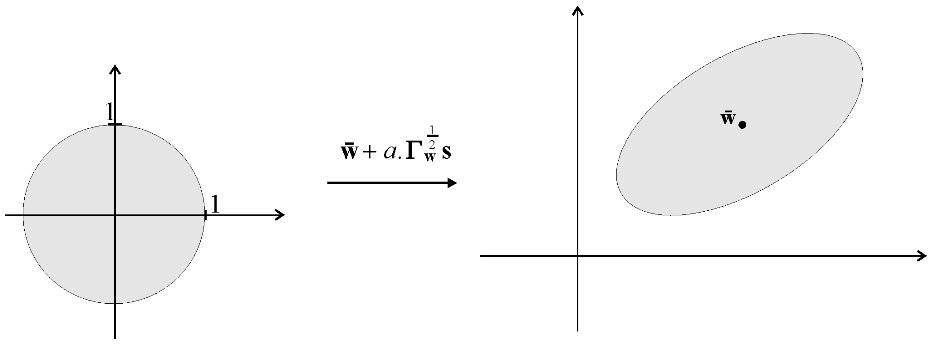


Figure 5.3 – The ellipsoid is the image by a linear application of the unit disk

or equivalently

$$||\mathbf{s}|| \leq 1 \text{ with } \mathbf{s} = \frac{1}{a}.\mathbf{\Gamma}_{\mathbf{w}}^{-\frac{1}{2}} (\mathbf{w} - \bar{\mathbf{w}}).$$

The ellipse $\mathcal{E}_{\mathbf{w}}$ can thus be defined as the image of the unit disk by the affine function $\mathbf{w}(\mathbf{s}) = \bar{\mathbf{w}} + a.\mathbf{\Gamma}_{\mathbf{w}}^{\frac{1}{2}}\mathbf{s}$ (see figure 5.3).

Remark. Consider a two-dimensional normal unary and centered random vector $\mathbf{s}$. The random variable $z = \mathbf{s}^{\mathrm{T}}\mathbf{s}$ follows a 2-dimensional χ^2 low. Now, the probability density function of a m-dimensional χ^2 low, is

$$\pi_z(z) = \begin{cases} \frac{z^{\frac{m-2}{2}} e^{-\frac{z}{2}}}{2^{\frac{m}{2}}\Gamma(\frac{m}{2})} \text{ if } z \geq 0 \\ 0 \text{ otherwise,} \end{cases}$$

where $\Gamma(m) = \int_0^\infty x^{m-1}e^{-x}dx$ is the *gamma* function. Thus, for $m = 2$, we have

$$\pi_z(z) = \begin{cases} \frac{1}{2}\exp\left(-\frac{z}{2}\right) \text{ if } z \geq 0 \\ 0 \text{ otherwise.} \end{cases}$$

For a given $a > 0$,

$$\eta \stackrel{\text{def}}{=} \text{prob}\left(||\mathbf{s}|| \leq a\right) = \text{prob}\left(\mathbf{s}^{\mathrm{T}}\mathbf{s} \leq a^2\right) = \text{prob}\left(z \leq a^2\right) = \int_0^{a^2} \frac{1}{2}\exp\left(-\frac{z}{2}\right) dz = 1 - e^{-\frac{1}{2}a^2}.$$

or equivalently

$$a = \sqrt{-2\ln(1-\eta)}.$$

To plot the boundary of $\mathcal{E}_{\mathbf{w}}$ we should thus use the following SCILAB function

```
function Draw_Ellipse(wbar,G_w,eta) ;

 s=0 :0.05 :2*%pi+0.05 ;

 w=wbar*ones(s)+sqrtm(-2*log(1-eta)*G_w)*[cos(s) ;sin(s)] ;

 xpoly(w(1, :),w(2, :)) ;

endfunction
```

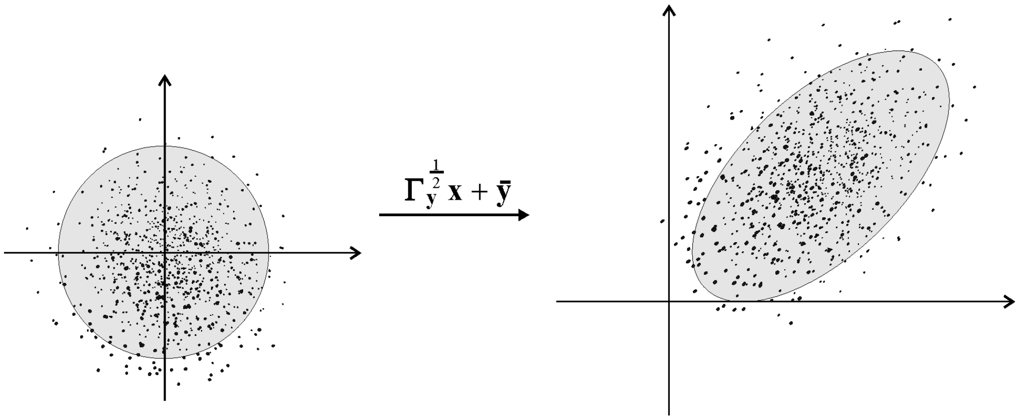


Figure 5.4 – A Gaussian vector is the image by an affine function or a white Gaussian vector

5.1.4 Generating Gaussian vectors

Theorem. If $\mathbf{x}$ and $\mathbf{y}$ are two random vectors related by the relation $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b}$ (where $\mathbf{A}$ and $\mathbf{b}$ are deterministic), we have

$$\begin{array}{rcl} \bar{\mathbf{y}} & = & \mathbf{A}\bar{\mathbf{x}} + \mathbf{b} \\ \mathbf{\Gamma_y} & = & \mathbf{A}.\mathbf{\Gamma_x}.\mathbf{A}^{\mathrm{T}}. \end{array}$$

Proof. We have

$$\bar{\mathbf{y}} = E\left(\mathbf{A}\mathbf{x} + \mathbf{b}\right) = \mathbf{A}E\left(\mathbf{x}\right) + \mathbf{b} = \mathbf{A}\bar{\mathbf{x}} + \mathbf{b}.$$

Moreover

$$\begin{array}{rcl} \mathbf{\Gamma_y} & = & E\left((\mathbf{y} - \bar{\mathbf{y}})(\mathbf{y} - \bar{\mathbf{y}})^{\mathrm{T}}\right) = E\left((\mathbf{A}\mathbf{x} - \mathbf{A}\bar{\mathbf{x}})(\mathbf{A}\mathbf{x} - \mathbf{A}\bar{\mathbf{x}})^{\mathrm{T}}\right) \\ & = & E\left(\mathbf{A}(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\right) = \mathbf{A}E\left((\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^{\mathrm{T}}\right)\mathbf{A}^{\mathrm{T}} = \mathbf{A}.\mathbf{\Gamma_x}.\mathbf{A}^{\mathrm{T}}.\blacksquare \end{array}$$

Since $\mathbf{\Gamma_x}, \mathbf{\Gamma_y}$ are symetric positive definite, they have squared roots that are also symetric positive definite. As a consequence, if $\mathbf{x}$ is a unit random Gaussian vector (i.e., $\bar{\mathbf{x}} = \mathbf{0}$ and $\mathbf{\Gamma_x} = \mathbf{I}$), the random vector $\mathbf{\Gamma}_{\mathbf{y}}^{\frac{1}{2}}\mathbf{x} + \bar{\mathbf{y}}$ will have an esperance of $\bar{\mathbf{y}}$ and a covariance matrix equal to $\mathbf{\Gamma_y}$ (see Figure 5.4). Therefore, to generate a Gaussian random vector with a covariance matrix $\mathbf{\Gamma_y}$ and an esperance of $\bar{\mathbf{y}}$, we have can use the following SCILAB function

```
function y=Randcov(ybar,Gy) ;
rand('normal') ;
x=rand(ybar) // x is Gaussian and has the dimension of ybar
y=ybar+sqrtm(Gy)*x ;
endfunction
```

We can also directly use the SCILAB statement `grand(1,'mn',ybar,Gy)` where `grand` and `mn` stand *Generate random* and *multivariate normal*, respectively.

5.2 Orthogonal unbiased estimator

Consider two random vectors $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^m$. The vector $\mathbf{y}$ corresponds to the vector of measurements, which is not available yet, and $\mathbf{x}$ is the vector to be estimated. An estimator is a function $\boldsymbol{\phi}(\mathbf{y})$ Figure 5.5 shows a non-linear estimator which corresponds to $E(x|y)$.

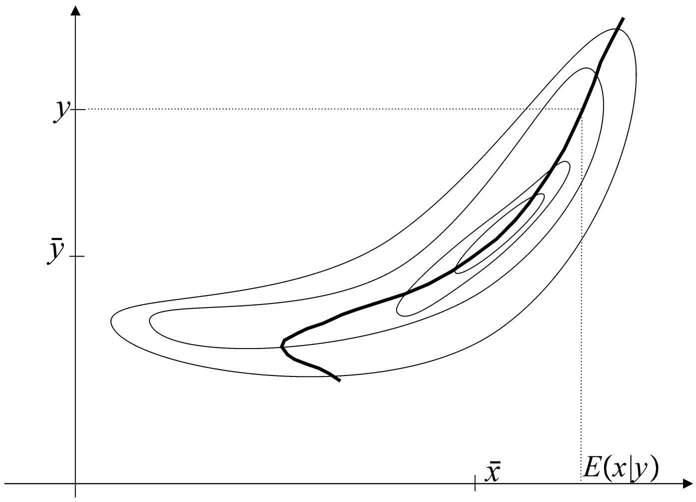


Figure 5.5 – Nonlinear estimator $E(x|y)$

Now getting an analytical expression of a good nonlinear estimator is not easy and we often prefer to deal with linear estimator. A *linear estimator* is a vector function of the form

$$\hat{\mathbf{x}} = \boldsymbol{\phi}(\mathbf{y}) = \mathbf{K}\mathbf{y} + \mathbf{b}, \tag{5.1}$$

where $\mathbf{K} \in \mathbb{R}^{n \times m}$ and $\mathbf{b} \in \mathbb{R}^n$. The *estimation error* is

$$\boldsymbol{\varepsilon} = \hat{\mathbf{x}} - \mathbf{x} = \mathbf{K}\mathbf{y} + \mathbf{b} - \mathbf{x}. \tag{5.2}$$

The estimator is said to be *unbiased* if $E(\boldsymbol{\varepsilon}) = \mathbf{0}$. It is *orthogonal* if $E\left(\boldsymbol{\varepsilon}\widetilde{\mathbf{y}}^{\mathrm{T}}\right) = \mathbf{0}$. Figure 5.6 represents a probability distribution function for the pair (x, y) and a linear estimator. Take any pair (x, y) in the plane randomly with respect to the corresponding probability distribution function. The probability to be below the line is high, i.e., the probability to have $\hat{x} > x$ is small. This amounts to saying that $E(\varepsilon)$ is strictly negative. The estimator is thus biased.

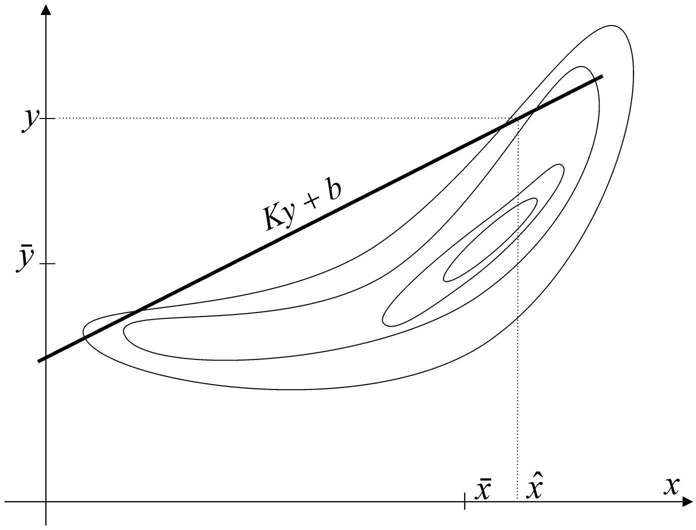


Figure 5.6 – Biased estimator

Figure 5.7 represents 4 different linear estimators. For (a), $E(\varepsilon) < 0$ and for (c), $E(\varepsilon) > 0$. For (b) and (d), $E(\varepsilon) = 0$ and thus the estimator is unbiased. For (d), $E(\varepsilon\widetilde{y}) < 0$, (if $\tilde{y} > 0, \varepsilon$ tends to be negative whereas if $\tilde{y} < 0, \varepsilon$ tends to be positive).

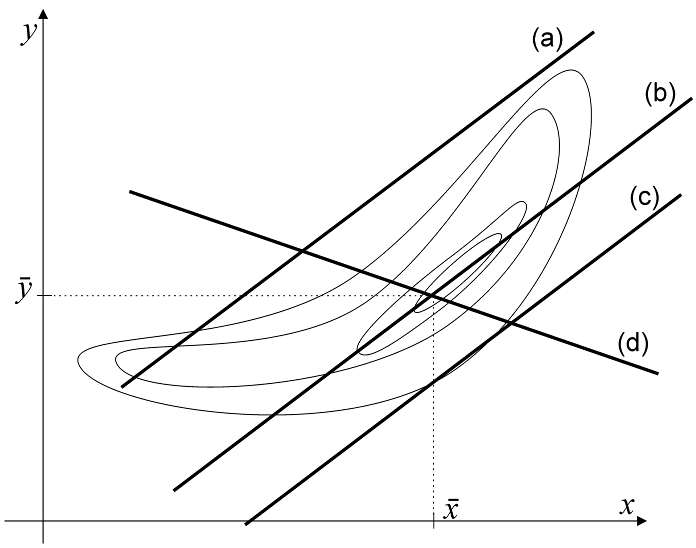


Figure 5.7 – Which one is the best linear estimator ?

Theorem. There exists a unique unbiased orthogonal linear estimator. It is given by

$$\hat{\mathbf{x}} = \bar{\mathbf{x}} + \mathbf{K}.\left(\mathbf{y} - \bar{\mathbf{y}}\right) \tag{5.3}$$

where

$$\mathbf{K} = \boldsymbol{\Gamma}_{\mathbf{xy}}\boldsymbol{\Gamma}_{\mathbf{y}}^{-1} \tag{5.4}$$

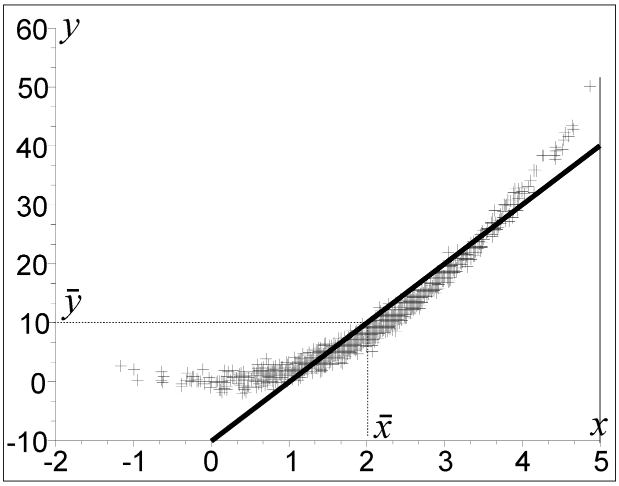


Figure 5.8 – Unbiased orthogonal linear estimator

is called the *Kalman Gain*.

Example. Consider again the example of Section 5.1.1. We get

$$\hat{x} = \bar{x} + \Gamma_{xy}\Gamma_y^{-1}.\left(y - \bar{y}\right) = 2 + 0.1 * \left(y - 10\right).$$

Proof. We have

$$E\left(\boldsymbol{\varepsilon}\right) \overset{(5.2)}{=} E\left(\mathbf{Ky} + \mathbf{b} - \mathbf{x}\right) = \mathbf{K}E\left(\mathbf{y}\right) + \mathbf{b} - E\left(\mathbf{x}\right) = \mathbf{K\bar{y}} + \mathbf{b} - \mathbf{\bar{x}}.$$

The estimator is thus unbiased if $E\left(\boldsymbol{\varepsilon}\right) = \mathbf{0}$, i.e.,

$$\mathbf{b} = \mathbf{\bar{x}} - \mathbf{K\bar{y}}. \tag{5.5}$$

In such a case, we have

$$\boldsymbol{\varepsilon} \overset{(5.2)}{=} \mathbf{Ky} + \mathbf{b} - \mathbf{x} \overset{(5.5)}{=} \mathbf{Ky} + \mathbf{\bar{x}} - \mathbf{K\bar{y}} - \mathbf{x} = \mathbf{K\widetilde{y}} - \mathbf{\widetilde{x}}. \tag{5.6}$$

The estimator is orthogonal if

$$\begin{array}{rcl} E\left(\boldsymbol{\varepsilon}.\mathbf{\widetilde{y}}^{\mathrm{T}}\right) = \mathbf{0} & \overset{(5.6)}{\Leftrightarrow} & E\left(\left(\mathbf{K\widetilde{y}} - \mathbf{\widetilde{x}}\right).\mathbf{\widetilde{y}}^{\mathrm{T}}\right) = \mathbf{0} \\ & \Leftrightarrow & E\left(\mathbf{K\widetilde{y}\widetilde{y}}^{\mathrm{T}} - \mathbf{\widetilde{x}\widetilde{y}}^{\mathrm{T}}\right) = \mathbf{0} \\ & \Leftrightarrow & \mathbf{K\Gamma_y} - \mathbf{\Gamma_{xy}} = \mathbf{0} \\ & \Leftrightarrow & \mathbf{K} = \mathbf{\Gamma_{xy}}.\mathbf{\Gamma_y}^{-1}. \end{array}$$

As a consequence,

$$\begin{array}{rcl} \mathbf{\hat{x}} & \overset{(5.1)}{=} & \mathbf{Ky} + \mathbf{b} \\ & \overset{(5.5)}{=} & \mathbf{Ky} + \mathbf{\bar{x}} - \mathbf{K\bar{y}} \\ & = & \mathbf{\bar{x}} + \mathbf{K}\left(\mathbf{y} - \mathbf{\bar{y}}\right). \end{array}$$

■

Theorem. The updated covariance matrix of the orthogonal linear estimator is

$$\mathbf{\Gamma}_{\boldsymbol{\varepsilon}} = \mathbf{\Gamma}_{\mathbf{x}} - \mathbf{K}.\mathbf{\Gamma}_{\mathbf{yx}} \tag{5.7}$$

and it is minimal (i.e. there is no other linear unbiased estimator which leads to a smaller $\mathbf{\Gamma}_{\boldsymbol{\varepsilon}}$).

Proof. The covariance matrix of $\boldsymbol{\varepsilon}$

$$\begin{array}{rcl}
\mathbf{\Gamma}_{\boldsymbol{\varepsilon}} & = & E\left(\boldsymbol{\varepsilon}.\boldsymbol{\varepsilon}^{\mathrm{T}}\right) \stackrel{(5.6)}{=} E\left((\mathbf{K}\widetilde{\mathbf{y}} - \widetilde{\mathbf{x}}).(\mathbf{K}\widetilde{\mathbf{y}} - \widetilde{\mathbf{x}})^{\mathrm{T}}\right) \\
& = & E\left((\mathbf{K}\widetilde{\mathbf{y}} - \widetilde{\mathbf{x}}).\left(\widetilde{\mathbf{y}}^{\mathrm{T}}\mathbf{K}^{\mathrm{T}} - \widetilde{\mathbf{x}}^{\mathrm{T}}\right)\right) \\
& = & E\left(\mathbf{K}\widetilde{\mathbf{y}}\widetilde{\mathbf{y}}^{\mathrm{T}}\mathbf{K}^{\mathrm{T}} - \widetilde{\mathbf{x}}\widetilde{\mathbf{y}}^{\mathrm{T}}\mathbf{K}^{\mathrm{T}} - \mathbf{K}\widetilde{\mathbf{y}}\widetilde{\mathbf{x}}^{\mathrm{T}} + \widetilde{\mathbf{x}}\widetilde{\mathbf{x}}^{\mathrm{T}}\right) \\
& = & \left(\mathbf{K}\mathbf{\Gamma}_{\mathbf{y}} - \mathbf{\Gamma}_{\mathbf{xy}}\right)\mathbf{K}^{\mathrm{T}} - \mathbf{K}\mathbf{\Gamma}_{\mathbf{yx}} + \mathbf{\Gamma}_{\mathbf{x}}.
\end{array}$$

Assume that $\mathbf{K} = \mathbf{K}_0 + \Delta$ with $\mathbf{K}_0 = \mathbf{\Gamma}_{\mathbf{xy}}\mathbf{\Gamma}_{\mathbf{y}}^{-1}$, we have

$$\begin{array}{rcl}
\mathbf{\Gamma}_{\boldsymbol{\varepsilon}} & = & ((\underbrace{\mathbf{K}_0 + \Delta}_{\mathbf{\Gamma}_{\mathbf{xy}}\mathbf{\Gamma}_{\mathbf{y}}^{-1}+\Delta})\mathbf{\Gamma}_{\mathbf{y}} - \mathbf{\Gamma}_{\mathbf{xy}})\left(\mathbf{K}_0 + \Delta\right)^{\mathrm{T}} - \left(\mathbf{K}_0 + \Delta\right)\mathbf{\Gamma}_{\mathbf{yx}} + \mathbf{\Gamma}_{\mathbf{x}} \\
& = & \Delta\mathbf{\Gamma}_{\mathbf{y}}\left(\mathbf{K}_0 + \Delta\right)^{\mathrm{T}} - \left(\mathbf{K}_0 + \Delta\right)\mathbf{\Gamma}_{\mathbf{yx}} + \mathbf{\Gamma}_{\mathbf{x}} \\
& = & \underbrace{\Delta\mathbf{\Gamma}_{\mathbf{y}}\mathbf{K}_0^{\mathrm{T}}}_{=\mathbf{\Gamma}_{\mathbf{yx}}} + \Delta\mathbf{\Gamma}_{\mathbf{y}}\Delta^{\mathrm{T}} - \mathbf{K}_0\mathbf{\Gamma}_{\mathbf{yx}} - \Delta\mathbf{\Gamma}_{\mathbf{yx}} + \mathbf{\Gamma}_{\mathbf{x}} \\
& = & \Delta\mathbf{\Gamma}_{\mathbf{y}}\Delta^{\mathrm{T}} - \mathbf{K}_0\mathbf{\Gamma}_{\mathbf{yx}} + \mathbf{\Gamma}_{\mathbf{x}}.
\end{array}$$

Since $\Delta\mathbf{\Gamma}_{\mathbf{y}}^{\mathrm{T}}\Delta^{\mathrm{T}}$ is always positive (i.e., its eigen values are positive and real), the covariance matrix is minimal for $\Delta = \mathbf{0}$, i.e., for $\mathbf{K} = \mathbf{\Gamma}_{\mathbf{xy}}\mathbf{\Gamma}_{\mathbf{y}}^{-1}$. In this case, we obviously have $\mathbf{\Gamma}_{\boldsymbol{\varepsilon}} = \mathbf{\Gamma}_{\mathbf{x}} - \mathbf{K}.\mathbf{\Gamma}_{\mathbf{yx}}$. ■

5.3 Application to linear estimation

5.3.1 Principle

Assume that $\mathbf{y}$ and $\mathbf{x}$ are related by a relation of the form

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \boldsymbol{\beta},$$

where $\boldsymbol{\beta}$ is a centered random vector, uncorrelated with $\mathbf{x}$. The covariance matrix of $\mathbf{x}$ and $\boldsymbol{\beta}$ are denoted by $\mathbf{\Gamma}_{\mathbf{x}}$ and $\mathbf{\Gamma}_{\boldsymbol{\beta}}$. Let us find the best linear unbiased estimator for $\mathbf{x}$ (see [4] for more about linear estimation). We have

$$\begin{array}{rcl}
\bar{\mathbf{y}} & = & \mathbf{C}\bar{\mathbf{x}} + \bar{\boldsymbol{\beta}} = \mathbf{C}\bar{\mathbf{x}} \\
\mathbf{\Gamma}_{\mathbf{y}} & = & E\left(\widetilde{\mathbf{y}}.\widetilde{\mathbf{y}}^{\mathrm{T}}\right) = E\left(\left(\mathbf{C}\widetilde{\mathbf{x}} + \widetilde{\boldsymbol{\beta}}\right).\left(\mathbf{C}\widetilde{\mathbf{x}} + \widetilde{\boldsymbol{\beta}}\right)^{\mathrm{T}}\right) = \mathbf{C}\mathbf{\Gamma}_{\mathbf{x}}\mathbf{C}^{\mathrm{T}} + \mathbf{\Gamma}_{\boldsymbol{\beta}} \\
\mathbf{\Gamma}_{\mathbf{xy}} & = & E\left(\widetilde{\mathbf{x}}.\widetilde{\mathbf{y}}^{\mathrm{T}}\right) = E\left(\widetilde{\mathbf{x}}.\left(\mathbf{C}\widetilde{\mathbf{x}} + \widetilde{\boldsymbol{\beta}}\right)^{\mathrm{T}}\right) = E\left(\widetilde{\mathbf{x}}.\widetilde{\mathbf{x}}^{\mathrm{T}}\mathbf{C}^{\mathrm{T}} + \left(\widetilde{\mathbf{x}}.\widetilde{\boldsymbol{\beta}}\right)^{\mathrm{T}}\right) = \mathbf{\Gamma}_{\mathbf{x}}\mathbf{C}^{\mathrm{T}}.
\end{array} \tag{5.8}$$

As a consequence, the unbiased orthogonal linear estimator for $\mathbf{x}$ with its covariance error is obtained from $\mathbf{\Gamma_x}, \mathbf{\Gamma}_\beta, \mathbf{C}, \bar{\mathbf{x}}$ by the following formulas

$$\begin{array}{llcll}
\text{(i)} & \hat{\mathbf{x}} & \overset{(5.3)}{=} & \bar{\mathbf{x}} + \mathbf{K}\widetilde{\mathbf{y}} & \text{(updated estimate)} \\
\text{(ii)} & \mathbf{\Gamma}_\varepsilon & \overset{(5.7)}{=} & \mathbf{\Gamma_x} - \mathbf{K}\mathbf{C}\mathbf{\Gamma_x} & \text{(updated covariance)} \\
\text{(iii)} & \widetilde{\mathbf{y}} & \overset{(5.8)}{=} & \mathbf{y} - \mathbf{C}\bar{\mathbf{x}} & \text{(innovation)} \\
\text{(iv)} & \mathbf{\Gamma_y} & \overset{(5.8)}{=} & \mathbf{C}\mathbf{\Gamma_x}\mathbf{C}^{\mathrm{T}} + \mathbf{\Gamma}_\beta & \text{(covariance innovation)} \\
\text{(v)} & \mathbf{K} & \overset{(5.4,5.8)}{=} & \mathbf{\Gamma_x}\mathbf{C}^{\mathrm{T}}\mathbf{\Gamma_y}^{-1} & \text{(Kalman gain)}
\end{array} \tag{5.9}$$

5.3.2 DC motor

The angular speed Ω of a brushed DC electric motor satisfies the relation :

$$\Omega = x_1 U + x_2 T_r,$$

where U is the input voltage, T_r is the resistive torque and $\mathbf{x} = (x_1, x_2)^{\mathrm{T}}$ are the parameters to be estimated. The table below presents the data that have been collected on the motor for different experimental conditions.

U(V)	10	13	15	15
T_r(Nm)	20	20	10	30
Ω(rad/sec)	109	141	173	163

We assume that the variance of the error measurement is constant and given by 25. Moreover, from the prior knowledge on the motor, we know that $x_1 \simeq 8$ and $x_2 \simeq -0.5$ with a variance equal to 9. We shall thus apply the formulas (5.9) with

$$\bar{\mathbf{x}} = \begin{pmatrix} 8 \\ -0.5 \end{pmatrix}, \mathbf{\Gamma_x} = \begin{pmatrix} 9 & 0 \\ 0 & 9 \end{pmatrix}, \mathbf{C} = \begin{pmatrix} 10 & 20 \\ 13 & 20 \\ 15 & 10 \\ 15 & 30 \end{pmatrix}, \ \mathbf{\Gamma}_\beta = \begin{pmatrix} 25 & 0 & 0 & 0 \\ 0 & 25 & 0 & 0 \\ 0 & 0 & 25 & 0 \\ 0 & 0 & 0 & 25 \end{pmatrix} \text{ and } \mathbf{y} = \begin{pmatrix} 109 \\ 141 \\ 173 \\ 163 \end{pmatrix}.$$

We get

$$\begin{aligned}
\widetilde{\mathbf{y}} &= \mathbf{y}-\mathbf{C}\bar{\mathbf{x}}=\begin{pmatrix}39\\47\\58\\58\end{pmatrix}\\
\mathbf{\Gamma_y} &= \mathbf{C}\mathbf{\Gamma_x}\mathbf{C}^{\mathrm{T}}+\mathbf{\Gamma}_{\boldsymbol{\beta}}=\begin{pmatrix}4525 & 4770 & 3150 & 6750\\ 4770 & 5146 & 3555 & 7155\\ 3150 & 3555 & 2950 & 4725\\ 6750 & 7155 & 4725 & 10\,150\end{pmatrix}\\
\mathbf{K} &= \mathbf{\Gamma_x}\mathbf{C}^{\mathrm{T}}\mathbf{\Gamma_y}^{-1}=\begin{pmatrix}-0.0179 & 0.0126 & 0.0926 & -0.0268\\ 0.0216 & 0.00369 & -0.0489 & 0.0324\end{pmatrix}\\
\hat{\mathbf{x}} &= \bar{\mathbf{x}}+\mathbf{K}\widetilde{\mathbf{y}}=\begin{pmatrix}11.7\\-0.44\end{pmatrix}\\
\mathbf{\Gamma}_{\boldsymbol{\varepsilon}} &= \mathbf{\Gamma_x}-\mathbf{K}\mathbf{C}\mathbf{\Gamma_x}=\begin{pmatrix}0.254 & -0.15\\ -0.15 & 0.102\end{pmatrix}.
\end{aligned}$$

The corresponding SCILAB program is

```
y=[109 ;141 ;173 ;163] ;
C=[100 20 ; 130 20 ;150 10 ;150 30] ;
xbar=[8 ;-0.5] ;
G_x=9*eye(2,2) ;
G_Beta=25*eye(6,6) ;
ytilde=y-C*xbar
G_y=C*G_x*C'+G_Beta ;
K=G_x*C'*inv(G_y) ;
xhat=xbar+K*ytilde
G_eps=G_x-K*C*G_x
```

5.3.3 Linear equations

Let us consider the following linear equations

$$\begin{cases} 2x_1+3x_2 &= 8\\ 3x_1+2x_2 &= 7\\ x_1-x_2 &= 0.\end{cases}$$

The first equation is assumed to be twice more reliable than the others. We translate the problem into

$$\underbrace{\begin{pmatrix}8\\7\\0\end{pmatrix}}_{\mathbf{y}}=\underbrace{\begin{pmatrix}2 & 3\\ 3 & 2\\ 1 & -1\end{pmatrix}}_{\mathbf{C}}\underbrace{\begin{pmatrix}x_1\\x_2\end{pmatrix}}_{\mathbf{x}}+\underbrace{\begin{pmatrix}\beta_1\\ \beta_2\\ \beta_3\end{pmatrix}}_{\boldsymbol{\beta}}.$$

We take

$$\bar{\mathbf{x}} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \mathbf{\Gamma}_{\mathbf{x}} = \begin{pmatrix} 1000 & 0 \\ 0 & 1000 \end{pmatrix}; \mathbf{C} = \begin{pmatrix} 2 & 3 \\ 3 & 2 \\ 1 & -1 \end{pmatrix}, \ \mathbf{\Gamma}_{\boldsymbol{\beta}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{pmatrix} \text{ and } \mathbf{y} = \begin{pmatrix} 8 \\ 7 \\ 0 \end{pmatrix}.$$

We get

$$\begin{array}{rcl} \widetilde{\mathbf{y}} & = & \mathbf{y} - \mathbf{C}\bar{\mathbf{x}} = \begin{pmatrix} 8 \\ 7 \\ 0 \end{pmatrix} \\ \mathbf{\Gamma}_{\mathbf{y}} & = & \mathbf{C}\mathbf{\Gamma}_{\mathbf{x}}\mathbf{C}^{\mathrm{T}} + \mathbf{\Gamma}_{\boldsymbol{\beta}} = \begin{pmatrix} 13\,001 & 12\,000 & -1000 \\ 12\,000 & 13004 & 1000 \\ -1000 & 1000 & 2004 \end{pmatrix} \\ \mathbf{K} & = & \mathbf{\Gamma}_{\mathbf{x}}\mathbf{C}^{\mathrm{T}}\mathbf{\Gamma}_{\mathbf{y}}^{-1} = \begin{pmatrix} -0.0886 & 0.288 & 0.310 \\ 0.355 & -0.155 & -0.244 \end{pmatrix} \\ \hat{\mathbf{x}} & = & \bar{\mathbf{x}} + \mathbf{K}\widetilde{\mathbf{y}} = \begin{pmatrix} 1.311 \\ 1.755 \end{pmatrix} \\ \mathbf{\Gamma}_{\boldsymbol{\varepsilon}} & = & \mathbf{\Gamma}_{\mathbf{x}} - \mathbf{K}\mathbf{C}\mathbf{\Gamma}_{\mathbf{x}} = \begin{pmatrix} 0.722 & -0.517 \\ -0.54 & 0.44 \end{pmatrix}. \end{array}$$

5.4 Kalman filter

This section presents the Kalman filter (see [1] for more information). Consider the system described by the following state equations

$$\begin{cases} \mathbf{x}_{k+1} & = & \mathbf{A}_k\mathbf{x}_k + \mathbf{B}_k\mathbf{u}_k + \boldsymbol{\alpha}_k \\ \mathbf{y}_k & = & \mathbf{C}_k\mathbf{x}_k + \boldsymbol{\beta}_k, \end{cases}$$

where $\boldsymbol{\alpha}_k$ and $\boldsymbol{\beta}_k$ are centered, independent, white random signals. The Kalman filter alternates two steps : the *prediction* and the *update* steps. We now introduce the following notations

$$\begin{array}{rcl} E(\mathbf{v}|\ell) & = & E(\mathbf{v}|\mathbf{y}_0,\dots,\mathbf{y}_\ell), \text{ where } \mathbf{v} \text{ is any random vector} \\ \hat{\mathbf{x}}_{k|\ell} & = & E(\mathbf{x}_k|\ell), \\ \mathbf{\Gamma}_{k|\ell} & = & E\left(\widetilde{\mathbf{x}}_k\widetilde{\mathbf{x}}_k^{\mathrm{T}}|\ell\right). \end{array}$$

Prediction. Assume that we know an estimate $\hat{\mathbf{x}}_{k|k}$ of $\mathbf{x}_k$ deduced from the measurements $\mathbf{y}_0,\dots,\mathbf{y}_k$ and the associated covariance matrix $\mathbf{\Gamma}_{k|k}$. Let us compute an estimate $\hat{\mathbf{x}}_{k+1|k}$ of $\mathbf{x}_k$ deduced from these measurements $\mathbf{y}_0,\dots,\mathbf{y}_k$ with the associated covariance matrix $\mathbf{\Gamma}_{k+1|k}$. We have

$$\begin{array}{rcl} \hat{\mathbf{x}}_{k+1|k} & = & E(\mathbf{x}_{k+1}|k) \\ & = & E(\mathbf{A}_k\mathbf{x}_k + \mathbf{B}_k\mathbf{u}_k + \boldsymbol{\alpha}_k|k) \\ & = & \mathbf{A}_k\underbrace{E(\mathbf{x}_k|k)}_{\hat{\mathbf{x}}_{k|k}} + \mathbf{B}_k\mathbf{u}_k + \underbrace{E(\boldsymbol{\alpha}_k|k)}_{=\mathbf{0}} \end{array}$$

i.e.

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{A}_k \hat{\mathbf{x}}_{k|k} + \mathbf{B}_k \mathbf{u}_k. \tag{5.10}$$

Moreover

$$\begin{aligned}
\boldsymbol{\Gamma}_{k+1|k} &= E\left(\widetilde{\mathbf{x}}_{k+1}\widetilde{\mathbf{x}}_{k+1}^{\mathrm{T}}|k\right) = E\left(\left(\mathbf{A}_k\widetilde{\mathbf{x}}_k + \widetilde{\boldsymbol{\alpha}}_k\right)\left(\mathbf{A}_k\widetilde{\mathbf{x}}_k + \widetilde{\boldsymbol{\alpha}}_k\right)^{\mathrm{T}}|k\right) \\
&= E\left(\left(\mathbf{A}_k\widetilde{\mathbf{x}}_k + \widetilde{\boldsymbol{\alpha}}_k\right)\left(\widetilde{\mathbf{x}}_k^{\mathrm{T}}\mathbf{A}_k^{\mathrm{T}} + \widetilde{\boldsymbol{\alpha}}_k^{\mathrm{T}}\right)|k\right) \\
&= E\left(\mathbf{A}_k\widetilde{\mathbf{x}}_k\widetilde{\mathbf{x}}_k^{\mathrm{T}}\mathbf{A}_k^{\mathrm{T}} + \widetilde{\boldsymbol{\alpha}}_k\widetilde{\mathbf{x}}_k^{\mathrm{T}}\mathbf{A}_k^{\mathrm{T}} + \mathbf{A}_k\widetilde{\mathbf{x}}_k\widetilde{\boldsymbol{\alpha}}_k^{\mathrm{T}} + \widetilde{\boldsymbol{\alpha}}_k\widetilde{\boldsymbol{\alpha}}_k^{\mathrm{T}}|k\right) \\
&= \mathbf{A}_k\underbrace{E\left(\widetilde{\mathbf{x}}_k\widetilde{\mathbf{x}}_k^{\mathrm{T}}|k\right)}_{=\boldsymbol{\Gamma}_{k|k}}\mathbf{A}_k^{\mathrm{T}} + \underbrace{E\left(\widetilde{\boldsymbol{\alpha}}_k\widetilde{\mathbf{x}}_k^{\mathrm{T}}|k\right)}_{=0}\mathbf{A}_k^{\mathrm{T}} + \mathbf{A}_k\underbrace{E\left(\widetilde{\mathbf{x}}_k\widetilde{\boldsymbol{\alpha}}_k^{\mathrm{T}}|k\right)}_{=0} + \underbrace{E\left(\widetilde{\boldsymbol{\alpha}}_k\widetilde{\boldsymbol{\alpha}}_k^{\mathrm{T}}|k\right)}_{=\Gamma_{\alpha_k}}.
\end{aligned}$$

Thus

$$\boldsymbol{\Gamma}_{k+1|k} = \mathbf{A}_k.\boldsymbol{\Gamma}_{k|k}.\mathbf{A}_k^{\mathrm{T}} + \Gamma_{\alpha_k}. \tag{5.11}$$

Kalman filter. The whole Kalman filter is thus given by the following equations

$$\begin{array}{cccc}
\hat{\mathbf{x}}_{k+1|k} & \overset{(5.10)}{=} & \mathbf{A}_k\hat{\mathbf{x}}_{k|k} + \mathbf{B}_k\mathbf{u}_k & \text{(predicted state estimate)} \\
\boldsymbol{\Gamma}_{k+1|k} & \overset{(5.11)}{=} & \mathbf{A}_k.\boldsymbol{\Gamma}_{k|k}.\mathbf{A}_k^{\mathrm{T}} + \boldsymbol{\Gamma}_{\alpha_k} & \text{(predicted state covariance)} \\
\hat{\mathbf{x}}_{k|k} & \overset{(5.9,\text{i})}{=} & \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\widetilde{\mathbf{y}}_k & \text{(updated state estimate)} \\
\boldsymbol{\Gamma}_{k|k} & \overset{(5.9,\text{ii})}{=} & \left(\mathbf{I} - \mathbf{K}_k\mathbf{C}_k\right)\boldsymbol{\Gamma}_{k|k-1} & \text{(updated state covariance)} \\
\widetilde{\mathbf{y}}_k & \overset{(5.9,\text{iii})}{=} & \mathbf{y}_k - \mathbf{C}_k\hat{\mathbf{x}}_{k|k-1} & \text{(innovation)} \\
\mathbf{S}_k & \overset{(5.9,\text{iv})}{=} & \mathbf{C}_k\boldsymbol{\Gamma}_{k|k-1}\mathbf{C}_k^{\mathrm{T}} + \boldsymbol{\Gamma}_{\boldsymbol{\beta}_k} & \text{(covariance innovation)} \\
\mathbf{K}_k & \overset{(5.9,\text{v})}{=} & \boldsymbol{\Gamma}_{k|k-1}\mathbf{C}_k^{\mathrm{T}}\mathbf{S}_k^{-1} & \text{(Kalman gain)}
\end{array}$$

Figure 5.9 shows that the Kalman filter only memorizes the vector $\hat{\mathbf{x}}_{k+1|k}$ and the matrix $\boldsymbol{\Gamma}_{k+1|k}$. Its inputs are $\mathbf{y}_k$, $\mathbf{u}_k$, $\mathbf{A}_k$, $\mathbf{B}_k$, $\mathbf{C}_k$, $\Gamma_{\boldsymbol{\alpha}_k}$ and $\Gamma_{\boldsymbol{\beta}_k}$. The quantities $\hat{\mathbf{x}}_{k|k}$, $\boldsymbol{\Gamma}_{k|k}$, $\widetilde{\mathbf{y}}_k$, $\mathbf{S}_k$, $\mathbf{K}_k$ are auxiliary variables.

The following SCILAB function implements the Kalman filter. In this program, we have the following correspondences : `x_pred1`$\leftrightarrow \hat{\mathbf{x}}_{k+1|k}$, `G_pred1`$\leftrightarrow \boldsymbol{\Gamma}_{k+1|k}$ (prediction), `x_upd`$\leftrightarrow \hat{\mathbf{x}}_{k|k}$, `G_upd`$\leftrightarrow \boldsymbol{\Gamma}_{k|k}$ (update).

```
function [x_pred1,G_pred1]=kalman(x_pred,G_pred,u,y,G_alpha,G_beta,A,B,C) ;
S=C*G_pred*C'+G_beta ;
K=G_pred*C'*inv(S) ;
y_tilde=y-C*x_pred ;
x_upd=x_pred+K*y_tilde ;
x_pred1=A*x_upd + B*u ;
G_upd=G_pred-K*C*G_pred ;
G_pred1=A*G_upd*A'+G_alpha ;
endfunction
```

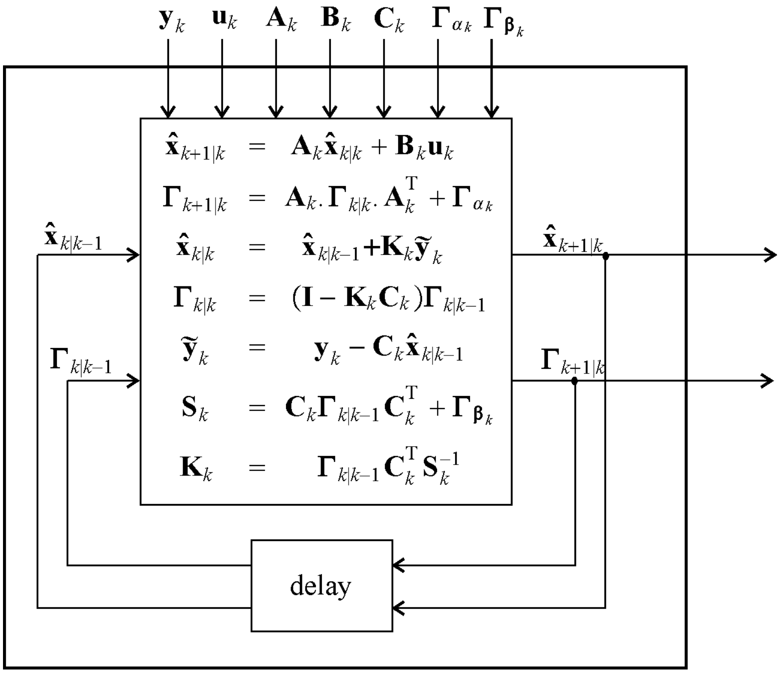


Figure 5.9 – Kalman filter

5.5 Applications

5.5.1 Solving linear equations

Let us consider again the following linear equations

$$\begin{cases} 2x_1 + 3x_2 & = & 8 \\ 3x_1 + 2x_2 & = & 7 \\ x_1 - x_2 & = & 0 \end{cases}$$

As for Section 5.3.3, the first equation is assumed to be twice more reliable than the others. Let us apply a Kalman filter to solve this system.

```
G_alpha=0*eye(2,2) ;
A=[1 0 ;0 1] ;
B=[0 ;0] ;
C0=[2 3] ; C1=[3 2] ; C2=[1 -1] ;
u=0 ;
xhat0=[0 ;0] ; Gx0=1000*eye(2,2) ;
[xhat1,Gx1]=kalman(xhat0,Gx0,u,8,G_alpha,1,A,B,C0)
[xhat2,Gx2]=kalman(xhat1,Gx1,u,7,G_alpha,4,A,B,C1)
[xhat3,Gx3]=kalman(xhat2,Gx2,u,0,G_alpha,4,A,B,C2)
```

We get the same results as in Section 5.3.3.

5.5.2 Simple case

Consider the discrete-time system described by the following equations

$$\begin{cases} \mathbf{x}_{k+1} &= \mathbf{A}_k\mathbf{x}_k + \mathbf{B}_k u_k + \boldsymbol{\alpha}_k \\ y_k &= \mathbf{C}_k\mathbf{x}_k + \boldsymbol{\beta}_k \end{cases}$$

with $k \in \{0,1,2\}$. The values for the quantities $\mathbf{A}_k, \mathbf{B}_k, u_k, y_k$ are given by the following table.

k	$\mathbf{A}_k$	$\mathbf{B}_k$	$\mathbf{C}_k$	u_k	y_k
0	$\begin{pmatrix} 0.5 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 2 \\ 4 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \end{pmatrix}$	4	7
1	$\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 3 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \end{pmatrix}$	−6	30
2	$\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$	$\begin{pmatrix} 4 \\ -1 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \end{pmatrix}$	8	−6

We first assume that $\boldsymbol{\alpha}_k$, β_k are white Gaussian noises with a covariance matrices equal to identity, i.e.,

$$\Gamma_\alpha = 1, \mathbf{\Gamma}_{\boldsymbol{\beta}} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

The initial condition, is unknown and represented by the estimation $\hat{\mathbf{x}}_{0|-1}$ and a covariance matrix $\mathbf{\Gamma}_{0|-1}$, given by

$$\hat{\mathbf{x}}_{0|-1} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \ \mathbf{\Gamma}_{0|-1} = \begin{pmatrix} 100 & 0 \\ 0 & 100 \end{pmatrix}.$$

Denote by $\mathcal{E}_{k|\ell}$ the ellipsoid with center $\hat{\mathbf{x}}_{\ell|k}$ and associated with the covariance matrix $\Gamma_{\ell|k}$. Figure 5.10 provides the ellipsoids obtained by the Kalman Filter.

The corresponding SCILAB program is given below.

```
A0=[0.5 0 ;0 1] ; A1=[1 -1 ;1 1] ; A2=[1 -1 ;1 1] ;
B0=[2 ;4] ; B1=[1 ;3] ; B2=[4 ;-1] ;
C0=[1 1] ;C1=[1 1] ;C2=[1 1] ;
u0=4 ; u1=-6 ; u2=8 ;
y0=7 ; y1=30 ; y2=-6 ;
G_alpha=1*eye(2,2) ;
G_beta=1*eye(1,1) ;
xhat0=[0 ;0] ; Gx0=100*eye(2,2) ;
[xhat1,Gx1]=kalman(xhat0,Gx0,u0,y0,G_alpha,G_beta,A0,B0,C0) ;
[xhat2,Gx2]=kalman(xhat1,Gx1,u1,y1,G_alpha,G_beta,A1,B1,C1) ;
[xhat3,Gx3]=kalman(xhat2,Gx2,u2,y2,G_alpha,G_beta,A2,B2,C2) ;
```

5.5.3 Parameter estimation

Consider again the brushed DC electric motor with an angular speed Ω. We have the relation :

$$\Omega = x_1 U + x_2 T_r,$$

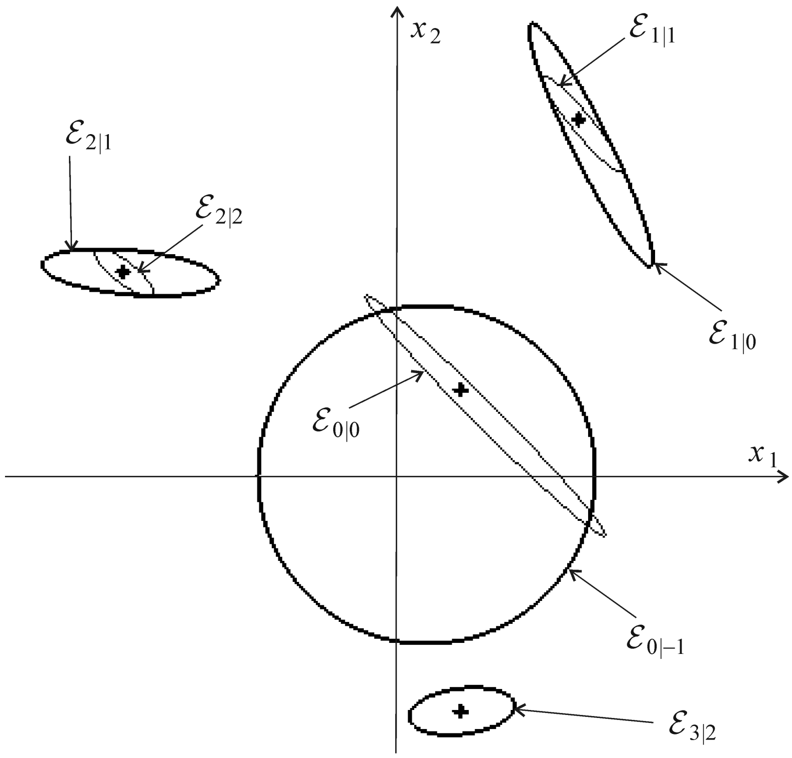


Figure 5.10 – Graphical illustation of the Kalman filter on a simple example

where U is the input voltage, T_r is the resistive torque and $\mathbf{x} = (x_1, x_2)^{\mathrm{T}}$ are the parameters to be estimated. The table below presents the data that have been collected on the motor for different experimental conditions.

k	0	1	2	3
U(V)	10	13	15	15
T_r(Nm)	20	20	10	30
Ω(rad/ sec)	109	141	173	163

We assume that the variance of the error measurement is constant and given by 25. Moreover, from some prior knowledge on the motor, we know that $x_1 \simeq 8$ and $x_2 \simeq -0.5$ with a variance error equal to 4. To use a Kalman filter, we shall take the following state space model

$$\left\{ \begin{array}{lcl} \mathbf{x}_{k+1} & = & \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}}_{\mathbf{A}_k} \mathbf{x}_k + \underbrace{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}_{\mathbf{B}_k} u_k + \underbrace{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}_{\boldsymbol{\alpha}_k} \\ y_k & = & \underbrace{\begin{pmatrix} U(k) & T_r(k) \end{pmatrix}}_{\mathbf{C}_k} \mathbf{x}_k \end{array} \right.$$

Using a Kalman filter, we get the same results as the linear estimator presented at Section 5.3.2. The corresponding SCILAB program is

```
y=[109 ;141 ;173 ;163] ;
C=[10 20 ;  13 20 ;15 10 ;15 30]
```

```
xhat=[8 ;-0.5] ;
G_x=9*eye(2,2) ;
G_alpha=zeros(2,2) ;
G_beta=25 ;
A=eye(2,2) ;
B=zeros(2,1) ;
for k=1 :4,
[xhat,G_x]=kalman(xhat,G_x,0,y(k),G_alpha,G_beta,A,B,C(k, :)) ;
end ;
```

5.5.4 Temperature sensor

The temperature of room x can be modeled, after some discretization, by the following state equations

$$\begin{cases} x_{k+1} &= x_k + \alpha_k \\ y_k &= x_k + \beta_k. \end{cases}$$

We assume that both α_k and β_k are Gaussian with variances given by $\Gamma_\alpha = 4$ and $\Gamma_\beta = 3$. Let us use a Kalman filter top estimate the temperature.

$$\begin{cases} \hat{x}_{k+1|k} = \hat{x}_{k|k} \\ \Gamma_{k+1|k} = \Gamma_{k|k} + \Gamma_{\alpha_k} \\ \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \widetilde{y}_k \\ \Gamma_{k|k} = (1 - K_k)\,\Gamma_{k|k-1} \\ \widetilde{y}_k = y_k - \hat{x}_{k|k-1} \\ S_k = \Gamma_{k|k-1} + \Gamma_{\boldsymbol{\beta}_k} \\ K_k = \Gamma_{k|k-1} S_k^{-1} \end{cases}$$

i.e.

$$\begin{cases} \hat{x}_{k+1|k} = \hat{x}_{k|k-1} + \frac{\Gamma_{k|k-1}}{\Gamma_{k|k-1}+\Gamma_{\boldsymbol{\beta}_k}}\left(y_k - \hat{x}_{k|k-1}\right) \\ \Gamma_{k+1|k} = \left(1 - \frac{\Gamma_{k|k-1}}{\Gamma_{k|k-1}+\Gamma_{\boldsymbol{\beta}_k}}\right)\Gamma_{k|k-1} + \Gamma_{\alpha_k} \end{cases}$$

When $k \to \infty$, we have $\Gamma_{k+1|k} - \Gamma_{k|k-1} \to 0$, i.e., $\Gamma_{k+1|k} \to \Gamma_\infty$. Thus

$$\Gamma_\infty = \left(1 - \frac{\Gamma_\infty}{\Gamma_\infty + \Gamma_{\boldsymbol{\beta}_k}}\right)\Gamma_\infty + \Gamma_{\alpha_k},$$

i.e.

$$\Gamma_\infty^2 - \Gamma_{\alpha_k}\Gamma_\infty - \Gamma_{\alpha_k}\Gamma_{\boldsymbol{\beta}_k} = 0.$$

There exists a unique positive solution given by

$$\Gamma_\infty = \frac{\Gamma_{\alpha_k} + \sqrt{\Gamma_{\alpha_k}^2 + 4\Gamma_{\alpha_k}\Gamma_{\boldsymbol{\beta}_k}}}{2} = \frac{4 + \sqrt{16 + 4 * 4 * 3}}{2} = 6$$

and thus the estimator is given by

$$\hat{x}_{k+1} = \hat{x}_k + \frac{2}{3}\left(y_k - \hat{x}_k\right).$$

The accuracy of the estimator is characterized by the variance $\Gamma_\infty = 6$.

Consider now, the situation where $\Gamma_{\alpha_k} = 0$. We get $\Gamma_\infty = 0$. This means that after a long period, the Kalman filter returns the exact temperature.

5.5.5 Dead reckoning

The robot represented on Figure 5.11 is described by the following state equations

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{v} \\ \dot{\delta} \end{pmatrix} = \begin{pmatrix} v\cos\delta\cos\theta \\ v\cos\delta\sin\theta \\ v\sin\delta + \alpha_1 \\ u_1 + \alpha_2 \\ u_2 + \alpha_3 \end{pmatrix}$$

where $\alpha_1, \alpha_2, \alpha_3$ are white Gaussian continuous-time noises.

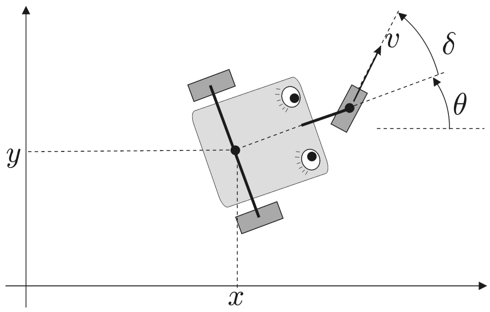


Figure 5.11 – Tricycle

The robot is equipped with a compass which returns θ with a very high precision and an angle sensor which provides the front wheel angle δ. At time $t = 0$, we know that $x = 0, y = 0, v = 1$. Using a Kalman filter, let us build a predictor for this system. If we set $z = (x, y, v)$, we get

$$\dot{\mathbf{z}} = \begin{pmatrix} 0 & 0 & \cos\delta\cos\theta \\ 0 & 0 & \cos\delta\sin\theta \\ 0 & 0 & 0 \end{pmatrix} \mathbf{z} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix} \mathbf{u} + \begin{pmatrix} 0 \\ 0 \\ \alpha_2 \end{pmatrix}.$$

After discretization, we obtain

$$\mathbf{z}_{k+1} = \begin{pmatrix} 1 & 0 & dt\cos\delta\cos\theta \\ 0 & 1 & dt\cos\delta\sin\theta \\ 0 & 0 & 1 \end{pmatrix} \mathbf{z}_k + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ dt & 0 \end{pmatrix} \mathbf{u}_k + \begin{pmatrix} 0 \\ 0 \\ dt.\alpha_2 \end{pmatrix}.$$

```
x=[0 ;0 ;%pi/3 ;1 ;0] ;
zhat=x([1 ;2 ;4]) ;
G_z=0*eye(3,3) ;
```

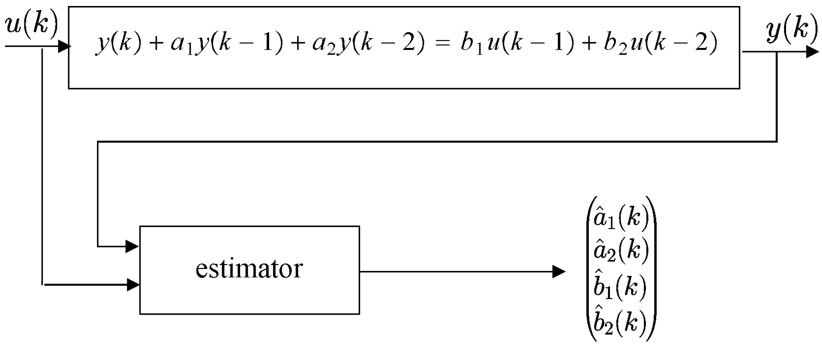


Figure 5.12 – Estimator of the parameters of a linear system

```
Bk=[0 ;0 ;dt] ;
G_alpha=dt*diag([0.01 ;0.01 ;0.1]) ;
while sortie==%F,
draw_voiture(x) ;
//y=x(4) ; Ck=[0 0 1] ; G_beta=0.1 ; // if odometers exist
y=[] ; Ck=[] ; G_beta=[] ; // if no odometer exists
Ak=[1 0 dt*cos(x(5))*cos(x(3)) ; 0 1 dt*cos(x(5))*sin(x(3)) ; 0 0 1 ] ;
[zhat,G_z]=kalman(zhat,G_z,uk(1),y,G_alpha,G_beta,Ak,Bk,Ck) ;
noise_x=zeros(x) ; noise_x([1 ;2 ;4])=Randcov(G_alpha) ;
x=x+f(x,u)*dt+noise_x ;
end ;
```

5.5.6 Parameter estimation of a transfer function

Consider a system with a scalar input $u(k)$ and a scalar output $y(k)$ described by the following equation

$$y(k) + a_1 y(k-1) + a_2 y(k-2) = b_1 u(k-1) + b_2 u(k-2).$$

The problem to be considered is the online estimation of the coefficients a_i and b_j (see Figure 5.12). We only know that these coefficients are moving slowly with respect to k. Define the parameter vector $\mathbf{p} = (a_1, a_2, b_1, b_2)^{\mathrm{T}}$. To take into account the fact that $\mathbf{p}$ is almost constant, we shall model its evolution by the following state equation

$$\mathbf{p}(k+1) = \mathbf{p}(k) + \boldsymbol{\alpha}(k),$$

where $\boldsymbol{\alpha}(k)$ is a white noise with a covariance matrix $\mathbf{\Gamma}_\alpha$. A Kalman filter based on the following linear system

$$\begin{cases} \mathbf{p}\,(k+1) &= \mathbf{p}(k) + \boldsymbol{\alpha}(k) \\ y(k) &= \mathbf{C}(k)\mathbf{p}(k) + \beta(k) \end{cases}$$

with

$$\mathbf{C}(k) = (-y(k-1), -y(k-2), u(k-1), u(k-2))$$

where $\beta(k)$ is a measurement noise with variance $\mathbf{\Gamma}_\beta$, can be used to estimate $\mathbf{p}(k)$. Such a filter can be used for instance to recognize vowels in a sound-based signal.

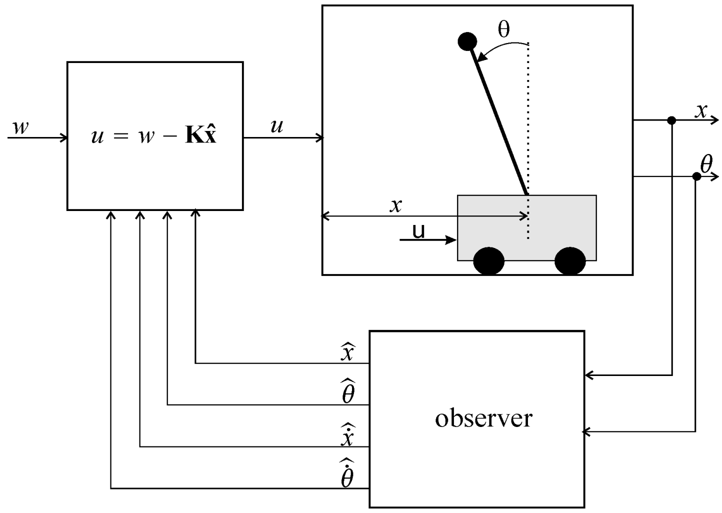


Figure 5.13 – Kalman filter to estimate the state of the inverted pendulum

5.5.7 Inverted pendulum

Consider the inverted pendulum described by the following state equations

$$\frac{d}{dt}\begin{pmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \dot{x} \\ \dot{\theta} \\ \frac{-\sin\theta(\dot{\theta}^2 - g\cos\theta)}{1+\sin^2\theta} \\ \frac{\sin\theta(2g-\dot{\theta}^2\cos\theta)}{1+\sin^2\theta} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \frac{1}{1+\sin^2\theta} \\ \frac{\cos\theta}{1+\sin^2\theta} \end{pmatrix} u.$$

or equivalently

$$\dot{\mathbf{x}} = \underbrace{\begin{pmatrix} x_3 \\ x_2 \\ \frac{-\sin x_2(x_4^2 - g\cos x_2)}{1+\sin^2 x_2} \\ \frac{\sin x_2(2g-x_4^2\cos x_2)}{1+\sin^2 x_2} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \frac{1}{1+\sin^2 x_2} \\ \frac{\cos x_2}{1+\sin^2 x_2} \end{pmatrix} u}_{\mathbf{f}(\mathbf{x},u)}$$

The controller needs an estimation $\hat{\mathbf{x}}$ of the state vector $\mathbf{x}$ that will be estimated using a Kalman filter (see Figure 5.13). Around $\mathbf{x} = \mathbf{0}$, we can linearize the system using the first order approximation method

$$\begin{cases} \frac{\sin x_2(g\cos x_2 - x_4^2)+u}{1+\sin^2 x_2} = \frac{(x_2+\varepsilon)(g(1+\varepsilon)-\ell\varepsilon)+u}{1+(x_2+\varepsilon)^2} = \frac{(x_2+\varepsilon)(g+\varepsilon)+u}{1+\varepsilon} = gx_2 + u + \varepsilon \\ \frac{\sin x_2(2g-x_4^2\cos x_2)+\cos x_2 u}{1+\sin^2 x_2} = \frac{(x_2+\varepsilon)(2g-\varepsilon(1+\varepsilon))+(1+\varepsilon)u}{1+(x_2+\varepsilon)^2} = 2x_2 g + u + \varepsilon. \end{cases}$$

We thus get the following linear approximation :

$$\dot{\mathbf{x}} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & g & 0 & 0 \\ 0 & 2g & 0 & 0 \end{pmatrix} \mathbf{x} + \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} u.$$

An Euler discretization with a sampling time of dt yields

$$\mathbf{x}(k+1) = \underbrace{\begin{pmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ 0 & dt.g & 1 & 0 \\ 0 & 2dt.g & 0 & 1 \end{pmatrix}}_{=\mathbf{A}} \mathbf{x}(k) + \underbrace{\begin{pmatrix} 0 \\ 0 \\ dt. \\ dt. \end{pmatrix}}_{=\mathbf{B}} u(k) + \begin{pmatrix} \alpha_1(k) \\ \alpha_2(k) \\ \alpha_3(k) \\ \alpha_4(k) \end{pmatrix}$$

where the vector $\alpha(k)$ is noise vector which takes into account the model errors as well as the discretization noise. The observation equation is

$$\mathbf{y}(k) = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}}_{\mathbf{C}} \mathbf{x}(k) + \begin{pmatrix} \beta_1(k) \\ \beta_2(k) \end{pmatrix}.$$

A Kalman filter can thus be used as an observer to estimate the state of the pendulum. If now, we replace in the prediction step of the Kalman filter, the statement

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{A}_k \hat{\mathbf{x}}_{k|k} + \mathbf{B}_k \mathbf{u}_k$$

by

$$\hat{\mathbf{x}}_{k+1|k} = \hat{\mathbf{x}}_{k|k} + \mathbf{f}\left(\hat{\mathbf{x}}_{k|k}, \mathbf{u}_k\right).dt$$

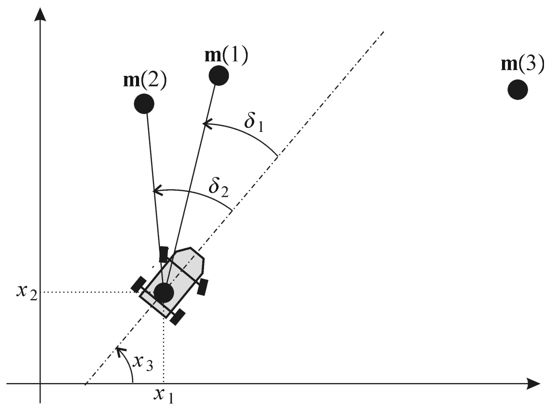
which corresponds to an Euler prediction, then, the prediction will be more accurate. This principle yields the *extended Kalman filter* (EKF).

Kalman filter for localization

Consider a car robot described by the following state equations

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{pmatrix} = \begin{pmatrix} x_4 \cos x_5 \cos x_3 \\ x_4 \cos x_5 \sin x_3 \\ \frac{x_4 \sin x_5}{3} \\ u_1 \\ u_2 \end{pmatrix}$$

where (x_1, x_2) represents the position the coordinates of the center of the robot, x_3 is its heading, x_4 its speed and x_5 the angle of the front wheels. The robot is surrounded by landmarks $\mathbf{m}(1), \mathbf{m}(2), \ldots$ with known positions. The robot is able to see the landmark $\mathbf{m}(i)$ only if its distance to the landmark is less than 15m. In this case, the robot measures the angle δ_i with a high precision. Moreover, at each time t, the robot is able to collect angles x_3 and x_5 without any error. It is also able to measure it speed x_4 with an error of variance 1. The figure below illustrates the situation where two landmarks $\mathbf{m}(1)$ and $\mathbf{m}(2)$ are seen by the robot.



To use a Kalman filter for localization, we need linear state equations, which is not the case here. Since x_3 and x_5 are assumed to be known, the nonlinearity can be cast into linear time-dependent equations, by setting $\mathbf{z} = (x_1, x_2, x_4)$. We get

$$\begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{pmatrix} = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_4 \end{pmatrix} = \begin{pmatrix} x_4 \cos x_5 \cos x_3 \\ x_4 \cos x_5 \sin x_3 \\ u_1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & \cos x_5 \cos x_3 \\ 0 & 0 & \cos x_5 \sin x_3 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix} \mathbf{u}.$$

An Euler discretization yields

$$\mathbf{z}(k+1) = \underbrace{\begin{pmatrix} 1 & 0 & dt.\cos x_5.\cos x_3 \\ 0 & 1 & dt.\cos x_5.\sin x_3 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathbf{A}(k)} .\mathbf{z}_k + \underbrace{\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ dt & 0 \end{pmatrix}}_{\mathbf{B}(k)} \mathbf{u}_k + \alpha_k$$

which is a linear evolution equation. We shall take $dt = 0.03$s. When the robot sees the ith landmark $\mathbf{m}(i) = (x_m(i), y_m(i))^{\mathrm{T}}$ with an angle δ_i, we have

$$(x_m(i) - x_1)\sin(x_3 + \delta_i) - (y_m(i) - x_2)\cos(x_3 + \delta_i) = 0.$$

i.e.,

$$\underbrace{-x_m(i)\sin(x_3 + \delta_i) + y_m(i)\cos(x_3 + \delta_i)}_{\text{known}} = \underbrace{\begin{pmatrix} -\sin(x_3 + \delta_i) & \cos(x_3 + \delta_i) \end{pmatrix}}_{\text{known}} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \beta_i.$$

where β_i is a Gaussian noise with a variance 1, witch has been inserted to take into account the uncertainties on the measured angles. If $\{i_1, i_2, \dots\}$ are the numbers of the landmarks that are seen by the robot, we have the following linear observation equation

$$\mathbf{y}(k) = \underbrace{\begin{pmatrix} 0 & 0 & 1 \\ -\sin(x_3 + \delta_{i_1}) & \cos(x_3 + \delta_{i_1}) & 0 \\ -\sin(x_3 + \delta_{i_2}) & \cos(x_3 + \delta_{i_2}) & 0 \\ \vdots & \vdots & \vdots \end{pmatrix}}_{\mathbf{C}(k)} .\mathbf{z}(k)$$

Note that the dimension of $\mathbf{y}$ depends on k. The first equation is provided by the odometers which return the speed. All other equations correspond to the detected landmarks..

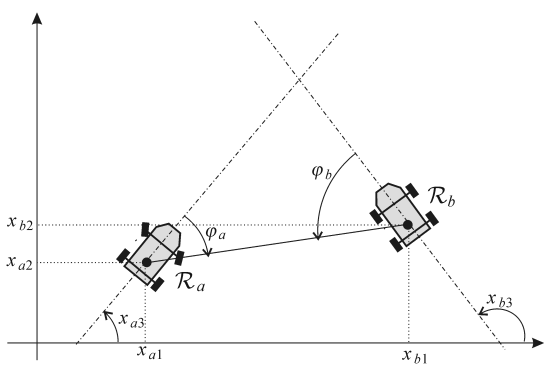
1) Download the following program

`www.ensta-bretagne.fr/jaulin/ensi_kalman_gonio.sce`

and propose a Kalman filter for localization. The initial state is assumed to be unknown.

2) After a simulated experiment, draw the volume of the confidence ellipse with respect to time. Give an interpretation of the results.

3) Consider now the situation with two robots $\mathcal{R}_a$ and $\mathcal{R}_b$ that can communicate via wifi. When the distance between the two robots is less than 15m, using cameras, they can measure the angles φ_a and φ_b with a good precision (see the figure below).



We can see the two robots as a single system the state of which is given by

$$\mathbf{x} = (x_{a1}, x_{a2}, x_{a3}, x_{a4}, x_{a5}, x_{b1}, x_{b2}, x_{b3}, x_{b4}, x_{b5})^{\mathrm{T}}.$$

When the robots are able to detect each other, we have the relation

$$(x_{b1} - x_{a1}) \sin(x_{a3} + \varphi_a) - (x_{b2} - x_{a2}) \cos(x_{a3} + \varphi_a) = 0.$$

Define the vector

$$\mathbf{z} = (x_{a1}, x_{a2}, x_{a4}, x_{b1}, x_{b2}, x_{b4})^{\mathrm{T}}$$

and show that $\mathbf{z}$ satisfies a linear state-space equation. Show that

$$\begin{pmatrix} -\sin(x_{a3} + \varphi_a) & \cos(x_{a3} + \varphi_a) & 0 & \sin(x_{a3} + \varphi_a) & -\cos(x_{a3} + \varphi_a) & 0 \end{pmatrix} . \mathbf{z} = 0$$

4) Propose a centralized Kalman filter for the localization of the two robots.