Filtre de Kalman

Part 1 : Utilisation du filtre de Kalman sur un système simple et sur quelques pas de simulation

EXERCISE 7.11.– *Three-step Kalman filter*

Let us consider the discrete-time system :

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{u}_k + \alpha_k \\ y_k \quad = \mathbf{C}_k \mathbf{x}_k + \beta_k \end{cases}$$

with $k \in \{0, 1, 2\}$. The values for the quantities $\mathbf{A}_k, \mathbf{C}_k, \mathbf{u}_k, y_k$ are given by :

k	$\mathbf{A}_k$	$\mathbf{u}_k$	$\mathbf{C}_k$	y_k
0	$\begin{pmatrix} 0.5 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 8 \\ 16 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \end{pmatrix}$	7
1	$\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$	$\begin{pmatrix} -6 \\ -18 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \end{pmatrix}$	30
2	$\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$	$\begin{pmatrix} 32 \\ -8 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \end{pmatrix}$	-6

Let us assume that the signals α_k and β_k are white Gaussian signals with a unitary covariance matrix, in other words :

$$\Gamma_\alpha = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \text{ and } \mathbf{\Gamma}_\beta = 1$$

The initial state vector is unknown and is represented by an estimation $\hat{\mathbf{x}}_{0|-1}$ and a covariance matrix $\mathbf{\Gamma}_{0|-1}$. We will take :

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

Draw the confidence ellipses with center $\hat{\mathbf{x}}_{k|k}$ and covariance matrix $\Gamma_{k|k}$ obtained by the Kalman filter, in MATLAB.

Useful files : http://www.ensta-bretagne.fr/lebars/Share/kalman_files_matlab.zip

Part 2 : Utilisation du filtre de Kalman sur un robot simple simulé

Exercise 7.18.– *Dead reckoning*

Dead reckoning corresponds to the problem of localization in which only proprioceptive sensors are available. This type of navigation was used by early navigators who were trying to locate themselves during long journeys. They were able to do this in a very approximative way by measuring the heading of the boat, the speed at various instants and integrating all the corresponding variations in position over the entire journey. In a more general context, we may consider that using a state observer in prediction mode and without correction (in the particular case in which the state is the position of the robot) corresponds to dead reckoning. Let us consider the robot represented on Figure 7.14 and whose state equations are :

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{v} \\ \dot{\delta} \end{pmatrix} = \begin{pmatrix} v \cos\delta \cos\theta \\ v \cos\delta \sin\theta \\ \frac{v \sin\delta}{3} + \alpha_\theta \\ u_1 + \alpha_v \\ u_2 + \alpha_\delta \end{pmatrix}$$

where $\alpha_\theta, \alpha_v, \alpha_\delta$ are independent continuous-time Gaussian white noises. In a more rigorous way, these are random distributions with infinite power, but once they are discretized, the mathematical difficulties disappear.

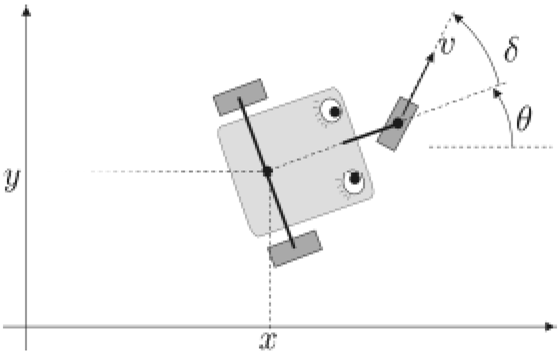

Figure 7.14 – Dead reckoning for a tricycle robot

The robot is equipped with a compass that returns θ with high precision and an angle sensor that returns the angle δ of the front wheel.

1) Discretize this system with an Euler method. Simulate this system in MATLAB for an arbitrary input $\mathbf{u}(t)$ and initial vector. For the variance of the discretized noises $\alpha_\theta, \alpha_v, \alpha_\delta$ we will take $0.01 \cdot dt$, where dt is the discretization step.

2) Express this localization problem in a linear and discretized form.

3) Using a Kalman filter, predict the position of the robot as well as the associated covariance matrix.

4) How does the localization program change if we assume that, using odometers, the robot is capable of measuring its speed v with a variance of 0.01 ?