Kalman Filters: A Tutorial

Joris De Schutter[1], Jan De Geeter[2], Tine Lefebvre[1]*and Herman Bruyninckx[1]†
[1] Katholieke Universiteit Leuven
Division of Production Engineering, Machine Design and Automation (PMA)
Celestijnenlaan 300 B, B-3001 Heverlee, Belgium
[2] SCK•CEN Belgian Nuclear Energy Research Centre
Boeretang 200, B-2400 Mol, Belgium

October 29, 1999

Abstract

A Kalman filter is a stochastic, recursive estimator, which estimates the state of a system based on the knowledge of the system input, the measurement of the system output, and a model of the relation between input and output. The Kalman filter equations are well known, but often little effort is spent to explain or understand how the Kalman filter really works, and what its assets and limitations are.

This paper introduces the Kalman filter in a very intuitive way, and looks at its properties: what information can be extracted from it, and what are its limitations. This intuitive understanding should help to develop successful practical applications. The paper also points at further reading.

1 Introduction

The purpose of this tutorial is to provide a very intuitive introduction to Kalman filters (KFs). KFs are used to estimate the state of a system or part of it, based on the measurement of the system inputs and outputs, and based on a model of the relation between them. Many textbooks treat KFs in a very concise and formal way, concentrating on the mathematical derivation. When lacking an intuitive understanding, students often have difficulty in applying KFs to practical situations, and they are unable to take full advantage of them. Similarly, practising engineers may be reluctant to apply KFs to their practical problem at hand, because the lack of intuitive understanding makes them feel unconfident.

This tutorial tries to respond to the need for a more intuitive treatment of KFs. First, later in this Section, we briefly introduce a KF in general terms as a tool to fuse all kinds of information in order to obtain the most accurate estimate of the state of a system. Sections 2, 3 and 5 formulate the estimation problem and provide its solution for systems with increasing complexity: first static systems (Section 2), then dynamic systems (Section 3), and finally extensions to nonlinear systems and to systems with constrained state variables. (Section 5). Section 4 shows how the convergence of the KF can be monitored and how the KF can be used to test hypotheses about the nature of the system. Section 6 describes a practical application with particular emphasis on the interpretation of the results. Section 7 briefly introduces the analogy between a KF and a system of springs. For some readers, this might further develop the intuitive understanding. Finally, Section 8 points at further reading.

*Tine Lefebvre is a Doctoral Research Fellow with the Fund for Scientific Research-Flanders.
†Herman Bruyninckx is a Postdoctoral Research Fellow with the Fund for Scientific Research-Flanders.

The Kalman filter

A Kalman filter is a linear, model based, stochastic, recursive, weighted, least squares estimator. *Estimator:* As stated above, the KF estimates the state of a system, or part of it, based on the knowledge (measurement) of the system inputs and outputs. *Model based & linear:* The KF is based on a system model consisting of a state equation and an output (measurement) equation, which are all *linear*. *Least squares:* Usually the estimation process is overdetermined, i.e. it is not possible to produce an estimate that is perfectly consistent with all collected information about the system (measurements). Noise is one important source of error. The KF then provides the estimate that tries to minimize the inconsistencies with all pieces of information in the least squares sense. In this respect, the KF is an *optimal* estimator. *Weighted:* When minimizing the sum of their least squares, the inconsistencies with the different pieces of information are weighted with a measure of the certainty of the information. Uncertain information is given low weight, whereas highly certain information is given a very high weight. *Recursive:* When all information is available at once, it can be processed in batch, as a classical weighted least squares problem. If, however, the information becomes available incrementally, as is the case for an on-line estimator, a recursive formulation of the estimation process is necessary. The KF does nothing more than that. *Stochastic:* The confidence about pieces of information is expressed in terms of probability distributions. The KF works with Gaussian distributions for both measurements and state estimates.

An intuitive example

As introduced above, the KF is a tool for stochastic data fusion. This is illustrated with the following example. Consider a quantity (x), for example a length, that is measured twice with the same or with different measurement equipment, for example a mechanical ruler and a laser system. The two measurements are noted x_1 and x_2; they are characterised by Gaussian probability distributions with means $\bar{x}_1$ and $\bar{x}_2$ and standard deviations σ_1 and σ_2:

$$p(x_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left[-\left(\frac{x_i - \bar{x}_i}{\sigma_i}\right)^2\right] \qquad (i = 1, 2). \tag{1}$$

The standard deviations are interpreted as a measure of uncertainty. The two measurements are combined to give an estimate of the length:

$$\hat{x} = wx_1 + (1 - w)x_2 \ , \tag{2}$$

where ($\hat{\cdot}$) means estimate, and w is a weight which is still to be determined. Because x_1 and x_2 have Gaussian distributions, $\hat{x}$ has also a Gaussian distribution with standard deviation $\hat{\sigma}$ given by:

$$\hat{\sigma}^2 = w^2\sigma_1^2 + (1 - w)^2\sigma_2^2 \ . \tag{3}$$

The weight that minimizes the uncertainty, expressed by $\hat{\sigma}$, is found as (put $\frac{\partial}{\partial w} = 0$):

$$w_{opt} = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} \ . \tag{4}$$

Hence, eqs. (2),(3) become:

$$\hat{x} = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}x_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}x_2 \ ; \tag{5}$$

$$\hat{\sigma}^2 = \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2 + \sigma_2^2} \ . \tag{6}$$

The estimate $\hat{x}$ minimizes the sum of the distances to x_1 and x_2, weighted by the respective standard deviations:

$$\hat{x} = \arg \min_x \left[\left(\frac{x_1 - x}{\sigma_1} \right)^2 + \left(\frac{x_2 - x}{\sigma_2} \right)^2 \right] . \tag{7}$$

This is again shown by putting $\frac{\partial}{\partial x} = 0$.

Suppose the two measurements become available sequentially. At time step 1 measurement x_1 becomes available. Since this is the only information, the state estimate and its variance are $\hat{x}_1 = x_1$; $\hat{\sigma}_1^2 = \sigma_1^2$ (the subscripts refer to the time step). Then, at time step 2, measurement x_2 becomes available, and the estimate is now taken as in eq. (5), which is rewritten in recursive form:

$$\hat{x}_2 = \hat{x}_1 + \frac{\hat{\sigma}_1^2}{\hat{\sigma}_1^2 + \sigma_2^2}(x_2 - \hat{x}_1) ; \tag{8}$$

$$\hat{\sigma}_2^2 = \left(1 - \frac{\hat{\sigma}_1^2}{\hat{\sigma}_1^2 + \sigma_2^2} \right) \hat{\sigma}_1^2 . \tag{9}$$

$x_2 - \hat{x}_1$ represents the new information received at time step 2, and is called the *innovation*. Furthermore,

$$K = \frac{\hat{\sigma}_1^2}{\hat{\sigma}_1^2 + \sigma_2^2} \tag{10}$$

is called the *update gain*. Hence eqs. (8), (9) become:

$$\hat{x}_2 = \hat{x}_1 + K (x_2 - \hat{x}_1) ; \tag{11}$$

$$\hat{\sigma}_2^2 = (1 - K) \hat{\sigma}_1^2 . \tag{12}$$

The structure of eqs. (11),(12) reappears in later sections, where a similar problem is solved for more complex systems. Notice that, in the sequential case, the subscripts of x and $\hat{x}$ refer to the time step at which the measurement is taken or the estimate is computed.

2 Static systems

A linear *static* system has a constant state vector x of dimension n and a linear output (measurement) equation:

$$z = Hx + \rho_m , \tag{13}$$

where z is an l-dimensional measurement vector. ρ_m represents Gaussian measurement uncertainty with zero mean ($\bar{\rho}_m = 0$) and covariance matrix $R = E\left[(\rho_m - \bar{\rho}_m) (\rho_m - \bar{\rho}_m)^T \right]$, where $E[.]$ means expected value. Hence, compared to the example in Section 1, we do not measure the state variables directly, but only linear combinations, as expressed by eq. (13). Suppose N measurements z_k are available. Each measurement may have its own measurement relation H_k and covariance matrix R_k. Then, the state estimate $\hat{x}$ that minimizes the weighted least squares distance between the measurements z_k and the *expected* measurement, $\hat{z}_k = H_k \hat{x}$, is given by:

$$\hat{x} = \arg \min_x \sum_{k=1}^{N} (z_k - H_k x)^T R_k^{-1} (z_k - H_k x) . \tag{14}$$

In the example of Section 1 the measurement space coincides with the state space. Hence, eq. (7), follows from eq. (14) by: *1)* replacing measurements z_k by x_k, and *2)* putting $H_k = 1$.

Initial information about the state vector and its covariance matrix, $\hat{x}_0$ and $\hat{P}_0$ can be added to eq. (14) by treating it as a first measurement at time step 0, i.e. $z_0 = \hat{x}_0$, $H_0 = I$ and $R_0 = \hat{P}_0$):

$$\hat{x} = \arg \min_x \left[(\hat{x}_0 - x)^T \hat{P}_0^{-1} (\hat{x}_0 - x) + \sum_{k=1}^{N} (z_k - H_k x)^T R_k^{-1} (z_k - H_k x) \right] . \tag{15}$$

If the measurements become available sequentially, and at each time instant k we add the new information to obtain a new estimate $\hat{x}_k$ and corresponding covariance matrix $\hat{P}_k$, then eq. (15) is rewritten in recursive form:

$$\hat{x}_k = \arg \min_x \left[(\hat{x}_{k-1} - x)^T \hat{P}_{k-1}^{-1} (\hat{x}_{k-1} - x) + (z_k - H_k x)^T R_k^{-1} (z_k - H_k x) \right] . \tag{16}$$

The solution of this minimization problem is called *static* Kalman filter. It computes the new estimate $\hat{x}_k$ and its covariance matrix $\hat{P}_k$ from $\hat{x}_{k-1}$, $\hat{P}_{k-1}$, z_k and R_k. (Section 3 finds this solution as a special case of the *dynamic* Kalman filter.)

3 Dynamic systems

Suppose the state of the system evolves according to the linear state equation:

$$x_k = A x_{k-1} + B u_{k-1} + \rho_p , \tag{17}$$

where A is the state matrix, B the input matrix, u_{k-1} is the input vector of dimension r at time step $k-1$, and ρ_p represents the process uncertainty, with covariance matrix Q (or Q_{k-1} if the process uncertainty is time variant).

The state vector evolves between time instants $k-1$ and k. Hence, the first term in eq. (16) has to be changed, based on a prediction of this evolution. The *state prediction* $\tilde{x}_k$ follows from the state equation (17):

$$\tilde{x}_k = A \hat{x}_{k-1} + B u_{k-1} . \tag{18}$$

If the input u_k is perfectly known, then the prediction error is found by subtracting eqs. (18) and (17):

$$\tilde{x}_k - x_k = A (\tilde{x}_{k-1} - x_{k-1}) - \rho_p . \tag{19}$$

Hence the covariance matrix of $\tilde{x}_k$ is:

$$\tilde{P}_k = A \hat{P}_{k-1} A^T + Q_{k-1} . \tag{20}$$

Substituting $\tilde{x}_k$ and $\tilde{P}_k$ for $\hat{x}_{k-1}$ and $\hat{P}_{k-1}$ in eq. (16) yields:

$$\hat{x}_k = \arg \min_x \left[(\tilde{x}_k - x)^T \tilde{P}_k^{-1} (\tilde{x}_k - x) + (z_k - H_k x)^T R_k^{-1} (z_k - H_k x) \right] . \tag{21}$$

The minimum is again found by putting the partial derivative with respect to x equal to zero. This yields:[1]

$$\hat{x}_k = \tilde{x}_k + K_k (z_k - H_k \tilde{x}_k) ; \tag{22}$$

$$\hat{P}_k = (I - K_k H_k) \tilde{P}_k ; \tag{23}$$

with

$$K_k = \tilde{P}_k H_k^T S_k^{-1} ; \tag{24}$$

$$S_k = R_k + H_k \tilde{P}_k H_k^T . \tag{25}$$

[1] To prove this result we need to apply the matrix inversion lemma, given by: $(A + BCB^T)^{-1} = A^{-1} - A^{-1}B (B^T A^{-1} B + C^{-1})^{-1} B^T A^{-1}$.

Eqs. (18),(20), (22)-(25) are called the *dynamic* Kalman filter. $\nu_k = z_k - H_k\tilde{x}_k$ is the new information at time step k and is called the *innovation*. It is the difference between the real and the predicted measurements. S_k is its covariance matrix. K_k is called the *Kalman gain*.

The static KF in Section 2 follows as a special case, by putting $A = I$, $B = 0$ and $Q = 0$ in eqs. (18),(20).

The similarity with eqs. (10)-(12) is revealed by substituting in eqs. (22)-(25): $\hat{x}_k \leftarrow \hat{x}_2$, $\tilde{x}_k \leftarrow \hat{x}_1$, $K_k \leftarrow K$, $z_k \leftarrow x_2$, $H_k \leftarrow 1$, $\hat{P}_k \leftarrow \hat{\sigma}_2^2$, $I \leftarrow 1$, $\tilde{P}_k \leftarrow \hat{\sigma}_1^2$, $R_k \leftarrow \sigma_2^2$.

4 Convergence and consistency

Convergence monitoring

The covariance matrix $\hat{P}_k$ is a measure for the uncertainty of state estimate $\hat{x}_k$. The uncertainty in different state space directions is graphically represented by the ellipsoid $(x - \hat{x}_k)^T \hat{P}_k^{-1} (x - \hat{x}_k) = 1$, figure 1. Due

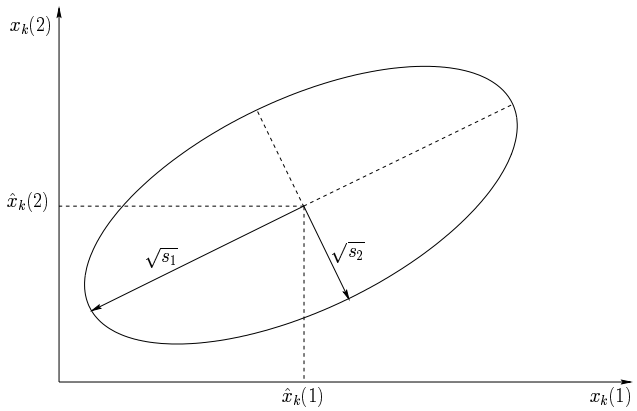

Figure 1: The uncertainty ellipsoid $(x - \hat{x}_k)^T \hat{P}_k^{-1} (x - \hat{x}_k) = 1$ in a 2D state space. s_1 and s_2 are the singular values of the matrix $\hat{P}_k$.

to process uncertainty, the state estimate becomes more uncertain over time. This is reflected by a growing uncertainty ellipsoid. On the other hand, measurements reduce the uncertainty of the state estimate, hence the ellipsoid shrinks.

The axes of the ellipsoid are oriented along the singular directions of $\hat{P}_k$. These are found by performing a singular value decomposition:

$$\hat{P}_k = U \Sigma V^T, \tag{26}$$

where Σ is a diagonal matrix which contains the singular values. $V = U$ because $\hat{P}_k$ is a symmetric matrix. The columns of U are the singular directions. Small singular values correspond to directions in state space with little uncertainty, while large singular values correspond to directions with large uncertainty. Convergence of the estimate means that the ellipsoid shrinks in all directions.

Warning. The state covariance matrix shows whether the Kalman filter converges or not, *not* whether the Kalman filter converges to the *correct* value: the state covariance matrix is determined solely by the system state model (A, B, Q), the measurement models (H_k, R_k), and the initial state covariance P_0, *not* by the measurement values z_k and the initial estimate $\hat{x}_0$ themselves. If the system model or one of the measurement models is not correct, the calculated state covariance is not correct. Hence, one should be very critical about

any conclusion on the accuracy of the state estimate that is drawn solely from the state covariance matrix. It is essential to verify the correctness of the calculated state covariance matrix using the consistency checks listed in the next Subsection.

Active sensing. The convergence process is influenced by the system input sequence u_k in eq. (17) and by the measurement relations H_k. Convergence is accelerated or made selective in some state space direction *1)* by designing an appropriate input sequence u_k, or *2)* by proper placement of the sensors and hence choosing proper measurement equations. Such procedures are called *experiment design* or *active sensing.*

Consistency checking

The KF strongly relies on a model of the system. This model consists of: the state equation (17), the measurement equation (13), the measurement and process covariance matrices R_k and Q_k, and the initial state estimate, $\hat{x}_0$, with its covariance matrix, $\hat{P}_0$. Wrong models result in estimation errors and innovations that cannot be explained statistically.

Estimation errors. The Normalised Estimation Error Squared $\text{NEES}_k = (\hat{x}_k - \bar{x})^T \hat{P}_k^{-1} (\hat{x}_k - \bar{x})$, where $\bar{x}$ is the true value of x, is a measure to verify if the estimation errors are consistent with the model. The NEES is χ^2-distributed with n DOF (degrees of freedom), n being the dimension of the state vector. Of course, the NEES can only be calculated when $\bar{x}$ is known, e.g. in simulations.

Innovations. The Normalised Innovation Squared $\text{NIS}_k = \nu_k^T S_k^{-1} \nu_k$ is a measure that indicates whether the innovations are consistent with the model. The NIS is χ^2-distibuted with l_k DOF, l_k being the number of statistically independent measurements in z_k. Another measure is the SNIS (Summed NIS) which is the sum of the M latest NIS values:

$$\text{SNIS}_k = \sum_{j=k-M+1}^{k} \nu_j^T S_j^{-1} \nu_j \ . \tag{27}$$

The SNIS is χ^2-distributed with $\sum_{j=k-M+1}^{k} l_j$ DOF.

Testing for consistency is possible by checking whether the NEES, NIS or SNIS are within a given confidence interval. If the initial state estimate and all covariance matrices are sufficiently accurate, then any inconsistency detected by the NIS or SNIS is due to erroneous system or measurement equations. Hence, in this case the NIS and SNIS can be used to detect sudden changes in the nature of the system.

5 Extended Kalman Filter

Nonlinear systems

Measurement equation (13) and state equation (17) calculate respectively the measurement and the predicted state as a *linear* function of the state x. Unfortunately, in practice these functions are often *nonlinear*:

$$z = h(x) + \rho_m \ , \tag{28}$$

$$x_k = f(x_{k-1}, u_{k-1}) + \rho_p \ , \tag{29}$$

where ρ_m and ρ_p are as defined in the previous sections.

In order to apply the KF, these equations are linearised in the most recent estimate: eq. (28) is linearised in $\tilde{x}_k$, yielding

$$z = h(\tilde{x}_k) + \left.\frac{\delta h}{\delta x}\right|_{\tilde{x}_k} (x - \tilde{x}_k) + \rho_m \ , \tag{30}$$

while eq. (29) is linearised in $\hat{x}_{k-1}$, yielding

$$x_k = f(\hat{x}_{k-1}, u_{k-1}) + \left.\frac{\delta f}{\delta x}\right|_{\hat{x}_{k-1}} (x_{k-1} - \hat{x}_{k-1}) + \rho_p \ . \tag{31}$$

The so called *Extended Kalman Filter* (EKF) is then obtained by

- Calculating $\tilde{x}_k$ with the nonlinear state equation:

$$\tilde{x}_k = f(\hat{x}_{k-1}, u_{k-1}) \ ; \tag{32}$$

- Predicting the measurement with the nonlinear measurement equation, such that eq. (22) becomes:

$$\hat{x}_k = \tilde{x}_k + K_k(z_k - h(\tilde{x}_k)) \ ; \tag{33}$$

- Substituting in eqs. (23)- (25): $H_k \leftarrow \left.\frac{\delta h}{\delta x}\right|_{\tilde{x}_k}$;
- Substituting in eq. (20): $A \leftarrow \left.\frac{\delta f}{\delta x}\right|_{\hat{x}_{k-1}}$.

Convergence. The linearized state and measurement models are only approximations of the true models, which invalidates all properties of optimality and convergence of the KF (Denham and Pines 1966), (Sorenson 1985). The correct convergence of an EKF depends on several factors, such as the initial estimate, the nonlinearity of the equations, the order in which the measurements are processed, and the measurements themselves. Hence, no formal proof of convergence exists; only consistency checks during Monte Carlo simulations can assess the performance of the filter.

Implicit measurement equation

Measurement equation (28) cannot always be made explicit in the measurement z, i.e., when only an implicit measurement equation is available:

$$h(x, z) + \rho_m = c, \tag{34}$$

where c is a vector of constants. For this case, the EKF equations are adapted as follows:

- eq. (33) becomes:

$$\hat{x}_k = \tilde{x}_k + K_k(c - h(\tilde{x}_k, z_k)) \ ; \tag{35}$$

- Substitute in eq. (25): $R_k \leftarrow \left(\left.\frac{\delta h}{\delta z}\right|_{\tilde{x}_k, z_k}\right) R_k \left(\left.\frac{\delta h}{\delta z}\right|_{\tilde{x}_k, z_k}\right)^T$;
- Substitute in eqs. (23)- (25): $H_k \leftarrow \left.\frac{\delta h}{\delta x}\right|_{\tilde{x}_k, z_k}$.

Constraint equations

Sometimes the system equations are much simpler when expressed in terms of a *nonminimal* state vector. The cost for this simpler expression is an additional set of algebraic equations which express relations between the state variables. These relations are called *constraints*. Since they are not influenced by measurements, they are just a special case of the previous case, i.e.:

$$h(x) = c \ . \tag{36}$$

In this case R corresponds to the covariance matrix of the constants c. If these constants are not uncertain, then $R = 0$.

6 Example: a mobile robot with ultrasonic sensor

A mobile robot navigates freely in a planar world. It uses an ultrasonic sensor in order to locate itself on a world map. Beside the ultrasonic sensor, the robot is equipped with so called *internal* sensors: encoders at the driving wheels and a gyroscope. These internal sensors are used to calculate the tangential velocity v and the angular velocity ω of the robot. v and ω are supposed to be perfectly known (ideal encoders and gyroscope, no wheel slip, ...).

In all simulations the robot moves on a straight line with orientation $\theta = -0.52$ rad, and with velocity $v = 0.5$ m/s. Since $\omega = 0$ rad/s, the state equations are:

$$X_k = X_{k-1} + v_{k-1}\cos(\theta_{k-1})\Delta t; \tag{37}$$

$$Y_k = Y_{k-1} + v_{k-1}\sin(\theta_{k-1})\Delta t; \tag{38}$$

$$\theta_k = \theta_{k-1}; \tag{39}$$

where the state vector consists of X_k and Y_k, the robot coordinates in the world map, and θ_k, the robot orientation. Δt is the time step. In the first simulations the robot orientation is supposed to be perfectly known. This reduces the state equations to eqs. (37),(38), linear functions of the state $[X_k, Y_k]^T$. Afterwards, a more realistic simulation is given where θ_k is also estimated.

Known orientation

Deadreckoning In the first simulation the robot does not use its ultrasonic sensor, only its internal sensors. This is called *deadreckoning*. Figure 2 shows the real and the estimated trajectory of the robot (estimate 1). The initial state is $\bar{x}_0 = [\text{-1 m}, 0 \text{ m}]^T$, while the initial state estimate is $\hat{x}_0 = [7 \text{ m}, \text{-5 m}]^T$ with covariance:

$$\hat{P}_0 = \begin{bmatrix} 100\ m^2 & 0\ m^2 \\ 0\ m^2 & 100\ m^2 \end{bmatrix}.$$

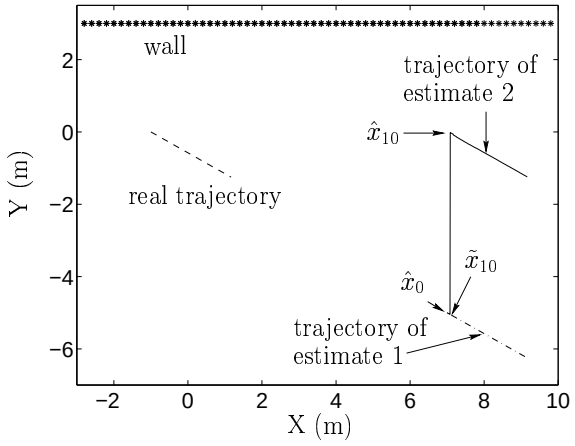

Figure 2: *Known orientation.* Estimated trajectory during deadreckoning (estimate 1) and when observing a wall (estimate 2).

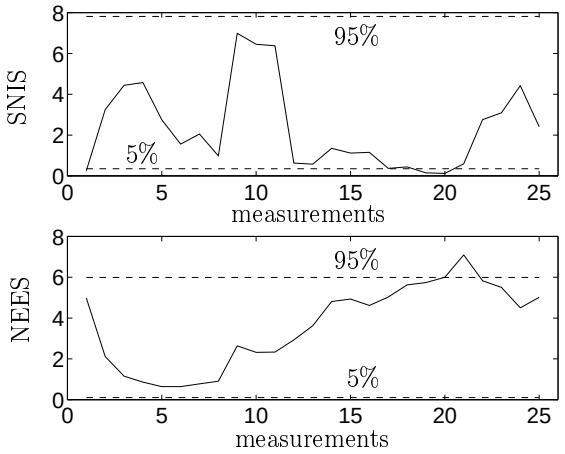

Figure 3: *Known orientation.* SNIS and NEES consistency tests when observing a wall.

Every 0.02 seconds, the robot updates its state corresponding to the only information it has: it is moving on a straight line with $v = 0.5$ m/s. Hence, deadreckoning corresponds to applying state equations (37)-(38). In the absence of process uncertainty, and since state matrix $A = I$, the uncertainty on the state does not grow: $\forall k, \hat{P}_k = \hat{P}_0$.

Measuring a wall To obtain a more precise estimate of its location, the robot observes a known wall ($y = ax + b$) with its ultrasonic sensor. The measurement value is twice the orthogonal distance between the sensor and the wall. The measurement uncertainty is $R = (0.03 \text{ m})^2$, the sensor measurements are taken every 0.2 seconds. Hence, a *linear* measurement equation,

$$z_k = (\frac{2a}{\sqrt{a^2+1}})X_k + (\frac{-2}{\sqrt{a^2+1}})Y_k + \frac{2b}{\sqrt{a^2+1}} + \rho_m, \tag{40}$$

is applied after every ten deadreckoning steps. $\bar{x}_0$, $\hat{x}_0$ and $\hat{P}_0$ are the same as in previous simulation. The wall is situated at $y = 3$ ($a = 0$, $b = 3$).

Figure 2 shows the estimated trajectory (estimate 2). $\tilde{x}_{10}$ is the result of ten deadreckoning steps starting from $\hat{x}_0$. The first measurement changes the estimate to $\hat{x}_{10}$. The robot gives a high importance to this measurement, because its uncertainty is much smaller than the state uncertainty.

The final covariance matrix after five seconds is:

$$\hat{P}_{250} = \begin{bmatrix} 100\ m^2 & 0\ m^2 \\ 0\ m^2 & 9e-06\ m^2 \end{bmatrix}.$$

This means that the robot made a precise estimation of Y, while the uncertainty of X remains exactly the same. This is obvious as the robot has no information about its position in the direction of the wall.

Figure 3 plots the SNIS and NEES consistency tests. The SNIS is summed over three NIS ($M = 3$). As statistical tests, SNIS and NEES have a 10% chance not to be in the 5 to 95% confidence interval when the model is perfect.

Measuring a table leg Next, the robot observes a point beacon, e.g. a table leg, with position $(p_X, p_Y) = (3, 0)$. The measurement value is twice the distance between robot and beacon, corresponding to measurement equation

$$z_k = 2\sqrt{(X_k - p_X)^2 + (Y_k - p_Y)^2} + \rho_m. \tag{41}$$

This is a *nonlinear* function of the state. Figure 4 shows the results for two different initial state estimates $\hat{x}_0$: [-0.5 m, -1.5 m]^T and [1.5 m, 0.5 m]^T. $\hat{P}_0$ is as in previous simulations, $\bar{x}_0 = [-1 \text{ m}, -2 \text{ m}]^T$.

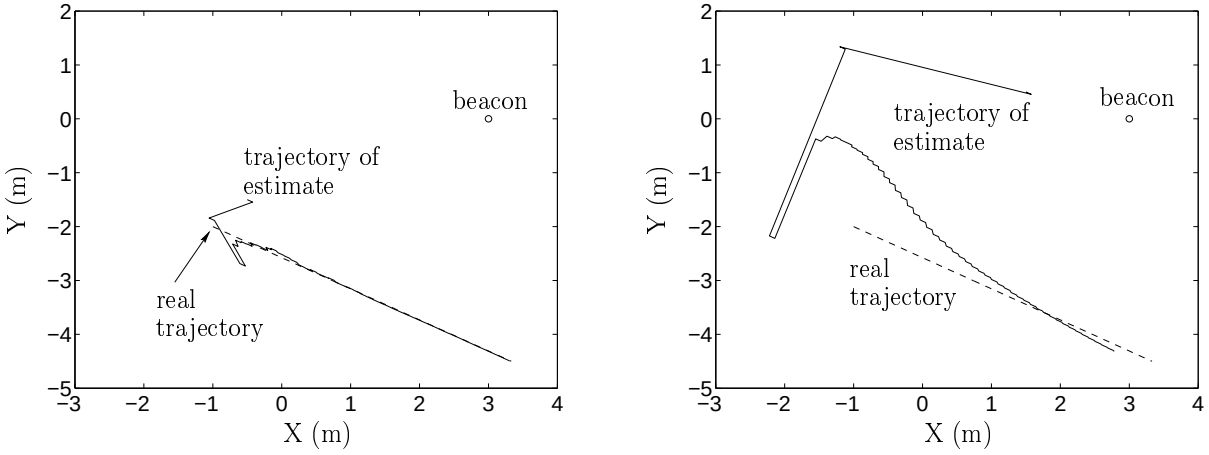

Figure 4: Estimated trajectory when observing a table leg. Left: first simulation. Right: second simulation.

After ten seconds, the covariance matrices are:
first simulation:

$$\hat{P}_{500} = \begin{bmatrix} 1.2e-04\ m^2 & -5.5e-05 m^2 \\ -5.5e-05\ m^2 & 5.2e-05\ m^2 \end{bmatrix};$$

second simulation:

$$\hat{P}_{500} = \begin{bmatrix} 5.1e-05\ m^2 & -2.9e-05\ m^2 \\ -2.9e-05\ m^2 & 5.4e-05\ m^2 \end{bmatrix}.$$

These covariance matrices illustrate that, although there is just one measurement equation available at each time step, the whole 2D state space is observed (all singular values become smaller). This is due to the fact that a *different* combination of states is measured at different time steps (different H_k).[2]

Because of linearisation, H_k depends on the state estimate. This explains the different values of $\hat{P}_{500}$ for the two simulations.

Moreover, the higher order terms in the Taylor expansion of the measurement equation contain powers of the uncertain state estimate, but these are not taken into account in the calculation of $\hat{P}_k$. Consequently, the calculated state uncertainty is always underestimated. Figure 5 shows the SNIS ($M = 3$) and NEES for both simulations. A too small covariance matrix $\hat{P}_k$ leads to too high SNIS and NEES values. In the first simulation linearisation errors are small thanks to a good initial estimate. But even in this case the covariances are no good measures for the difference between real state and estimate! (All NEES values are around the 95% boundary here). On the other hand, the bigger linearisation errors in the second simulation lead to very high SNIS and NEES values, which indicate something is wrong with the model.

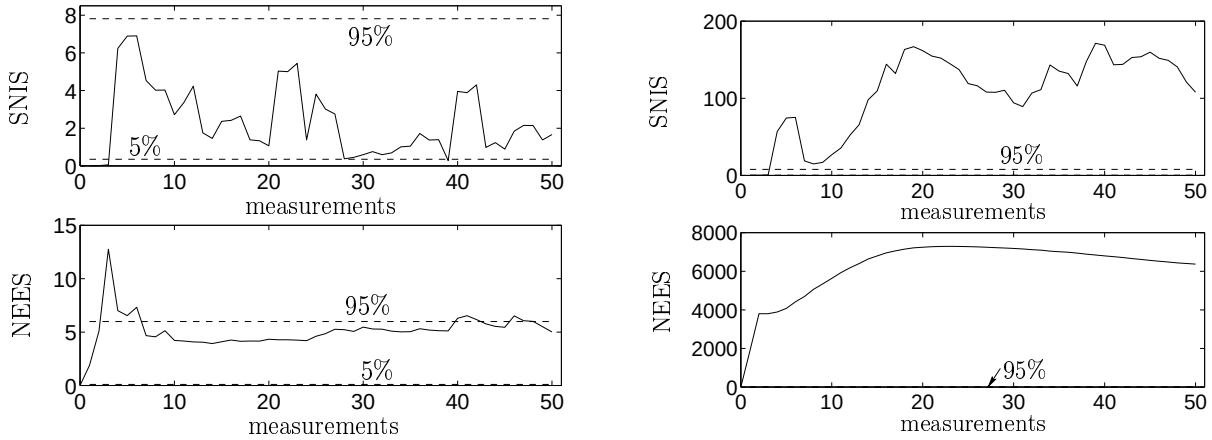

Figure 5: SNIS and NEES consistency tests when observing a table leg. Left: first simulation. Right: second simulation.

Unknown orientation

In a more realistic approach, the robot orientation θ_k is also estimated. The state of the robot at time step k is now represented by $[X_k,\ Y_k,\ \theta_k]^T$.

Deadreckoning Figure 6 shows the real and the estimated trajectory based only on deadreckoning (estimate 1). $\bar{x}_0 = [\text{-1 m, 0 m, -0.52 rad}]^T$, $\hat{x}_0 = [\text{7 m, -5 m, 0 rad}]^T$,

$$\hat{P}_0 = \begin{bmatrix} 100\ m^2 & 0\ m^2 & 0\ m\ rad \\ 0\ m^2 & 100\ m^2 & 0\ m\ rad \\ 0\ m\ rad & 0\ m\ rad & 0.76\ rad^2 \end{bmatrix}.$$

[2] From eqs. (37) and (38) it follows that the state matrix $A = I_{2\times 2}$. Furthermore, there is only one measurement equation, i.e. the dimensions of H are 1×2. Hence, in order to have full observability, at least two different combinations of the states have to be measured. Clearly, if $A \neq I$ a time invariant measurement equation may be sufficient to guarantee observability.

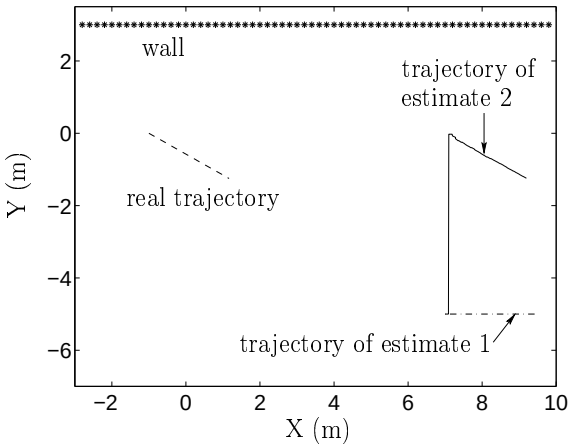

Figure 6: *Unknown orientation.* Estimated trajectory during deadreckoning (estimate 1) and when observing a wall (estimate 2).

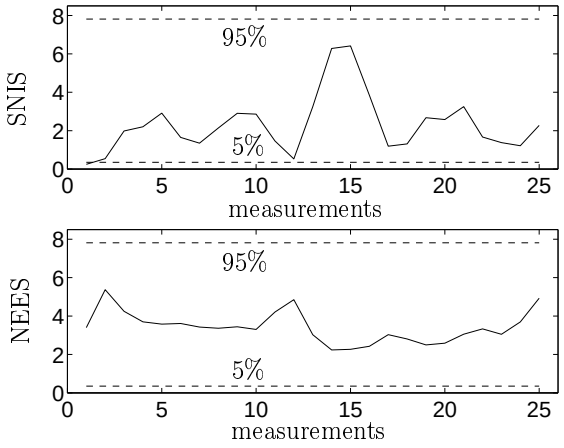

Figure 7: *Unknown orientation.* SNIS and NEES consistency tests when observing a wall.

After five seconds the state covariance matrix is:

$$\hat{P}_{250} = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 104.8 & 1.90 \\ 0 & 1.90 & 0.76 \end{bmatrix} = U\Sigma U^T;$$

$$\Sigma = \begin{bmatrix} 104.8 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 0.73 \end{bmatrix}; \qquad U = \begin{bmatrix} 0 & 1 & 0 \\ 0.9998 & 0 & -0.0183 \\ 0.0183 & 0 & 0.9998 \end{bmatrix}.$$

The uncertainty on $\hat{\theta}$ leads to *1)* the estimate perpendicular to the estimated motion direction (Y) becoming more uncertain in time, as expressed by the second diagonal element of the covariance matrix, and *2)* correlations between that estimate and $\hat{\theta}$, as expressed by the off-diagonal elements.

Measuring a wall $\bar{x}_0$, $\hat{x}_0$ and $\hat{P}_0$ are the same as in the last experiment. Figure 6 shows the estimated trajectory (estimate 2).

After five seconds the state estimate covariance matrix is:

$$\hat{P}_{250} = \begin{bmatrix} 100\ m^2 & 0\ m^2 & 0\ m\ rad \\ 0\ m^2 & 3.4e-05\ m^2 & 2.4e-05\ m\ rad \\ 0\ m\ rad & 2.4e-05\ m\ rad & 2.4e-05\ rad^2 \end{bmatrix} = U\Sigma U^T;$$

$$\Sigma = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 5.4e-05 & 0 \\ 0 & 0 & 3.9e-06 \end{bmatrix}; \qquad U = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.78 & 0.63 \\ 0 & 0.63 & -0.78 \end{bmatrix}.$$

This means that the robot made a precise estimation of Y and θ, while X remains not observed.

Figure 7 plots the SNIS ($M = 3$) and NEES consistency tests.

7 Mechanical spring analogy

The KF can be represented by an analogous model consisting of a set of interconnected springs. The equilibrium position of the interconnection point of the springs behaves in exactly the same way as the estimate of the

KF. This model is particularly instructive for engineers who like to reason about physical quantities like deformations, forces and energy, rather than abstract concepts like states, covariances and SNIS.

An intuitive example

Consider again the simple example of Section 1: the state of the system is observed directly rather than through a linear combination of the states, and the state x is one-dimensional, i.e. $H = 1$; the system is static and there is no process uncertainty, i.e. $A = 1$ and $Q = 0$; there are two measurements available, x_1 and x_2 with standard deviation σ_1 and σ_2. For this case, the spring model consists of two springs, one with compliance[3] σ_1^2 and one with compliance σ_2^2. Both springs have rest length zero. They are connected together at one end, and they are fixed to the environment with their other end, at the location x_1 and x_2 respectively, see fig. 8. Then:

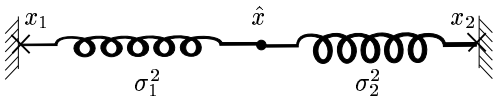

Figure 8: The spring model of the Kalman filter: an intuitive example. The equilibrium position of the interconnection point of the two springs is equal to the KF estimate $\hat{x}$; the compliance of the combination of the two springs equals the variance $\hat{\sigma}^2$.

1. The estimate $\hat{x}$ of the Kalman filter is equal to the location of the interconnection point of the two springs in equilibrium.
   This is shown as follows. The total potential energy is equal to the sum of the potential energy in each of the two springs. Since, in equilibrium, the potential energy in the spring system is minimal, the location of the interconnection point $\hat{x}$ is given by eq. (7).

2. The variance $\hat{\sigma}^2$ of $\hat{x}$ equals the compliance of the complete spring system.
   This is shown as follows. Seen from the interconnection point x of the springs, both springs are connected in parallel. The stiffness of a system of two parallel springs is equal to sum of the stiffness of the two springs, yielding exactly eq. (6).

   With these two properties, it is clear that at each time step, the set of two springs can be replaced by one equivalent spring, anchored in $\hat{x}$, with rest length zero and compliance $\hat{\sigma}^2$.

3. The NIS is equal to twice the energy needed to add to the spring system one extra spring corresponding to a new measurement. Accordingly, the SNIS, eq. (27) with $M = k$, is equal to twice the total potential energy present in the spring system.

4. The NEES is equal to twice the energy needed to move the interconnection point of all springs to the true state value.

Generalisation

The equivalence of the spring model and the KF holds equally well for the most general case of a dynamic system with a multi-dimensional state x and nonzero process uncertainty (i.e. $A \neq I$ and $Q \neq O$), multi-dimensional measurement vectors z, with different measurement space at each time step, i.e. $H_k \neq H_l \neq I$ for $k \neq l$. For proofs of these properties the reader is referred to (De Geeter 1998).

[3] Compliance is the inverse of stiffness.

8 Further reading

(Sorenson 1985) contains an excellent collection of papers on the history of Kalman filtering. This book includes the paper of Kalman (Kalman 1960), where he proves the optimality of a recursive filter that was later named after him. Although Kalman's solution to the recursive estimation problem is the best known, several other authors have published similar results approximately simultaneously, see e.g. (Swerling 1959)[4]. The introductory paper in this book by Sorenson is particularly helpful before reading and comparing the papers in the collection.

Numerically reliable implementation of the KF (square root form) is addressed in (Kailath 1981).

A classical book is (Jazwinski 1970). It contains, among other things, a study of stability, sensitivity to modelling errors for linear KF's, and various approaches to nonlinear filtering.

Bar-Shalom and Li (1993) have written an excellent textbook on Kalman filtering for engineers. This book contains many examples illustrating the usefulness of the consistency checks.

Linearisation errors are a terrible nuisance, especially when the measurement uncertainty is relatively small. The effects of such errors are well described by Denham and Pines (1966). The linearisation errors of nonlinear constraints have the most dramatic impact on the Kalman filter convergence, since there is no measurement error at all. See (De Geeter 1998) or (De Geeter et al. 1997) for a detailed analysis of these errors, and an algorithm to reduce their effect on the filter convergence.

A historical reference on optimal experiment design is the book by Fedorov (1972). More recent but not easy to read is the book by Pukelsheim (1993). (De Geeter et al. 1998) contains an intuitive geometric comparison of the most popular strategies for optimal experiment design.

Interpretation of the Kalman Filter from an information theory point of view can be found in (Meinhold and Singpurwalla 1983) and (Zellner 1988).

More details on the spring model can be found in (De Geeter 1998).

References

Bar-Shalom, Y. and X.-R. Li (1993). *Estimation and Tracking: Principles, Techniques and Software.* Artech House, Norwood.

De Geeter, J. (1998, May). *Constrained system state estimation and task-directed sensing. Application to the local modelling of a structured nuclear environment.* Ph. D. thesis, Dept. of Mechanical Engineering, Katholieke Universiteit Leuven, Heverlee, Belgium.

De Geeter, J., J. De Schutter, H. Bruyninckx, H. Van Brussel, and M. Decréton (1998). Tolerance-weighted L-optimal experiment design for active sensing. In *Proceedings of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Victoria B.C., Canada, pp. 1670–1675.

De Geeter, J., J. De Schutter, H. Van Brussel, and M. Decréton (1997, October). A smoothly constrained Kalman filter. *IEEE Transactions on Pattern Analysis and Machine Intelligence 19*(10), 1171–1177.

Denham, W. F. and S. Pines (1966). Sequential estimation when measurement function nonlinearity is comparable to measurement error. *AIAA Journal 4*(6), 1071–1076.

Fedorov, V. V. (1972). *Theory of optimal experiments.* New York and London: Academic Press.

Jazwinski, A. H. (1970). *Stochastic processes and filtering theory.* Academic Press.

Kailath, T. (2nd print, 1981). *Lectures on Wiener and Kalman filtering, CISM Courses and Lectures, No. 140.* Springer-Verlag.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME, Journal of Basic Engineering*, 35–45.

[4] "The reason why the Kalman filter is named after Kalman and not after Swerling, is that Swerling is not a good name for a filter, it sounds too turbulent. Kalman sounds smoother, calmer" (free after Prof. M. Mintz in a Lecture at the Univ. of Pennsylvania, 1996).

Meinhold, R. J. and N. D. Singpurwalla (1983). Understanding the Kalman-Filter. *The American Statistician 37*, 123–127.

Pukelsheim, F. (1993). *Optimal design of experiments.* John Wiley & Sons.

Sorenson, H. W. (1985). *Kalman Filtering: Theory and Application.* IEEE Press.

Swerling, P. (1959). First order error propagation in a stagewise smoothing procedure for satellite observations. *Journal of the Astronautical Sciences 6*(3), 46–52.

Zellner, A. (1988). Optimal information processing and Bayes's theorem. *The American Statistician 42*, 278–284.

Joris De Schutter received the degree of mechanical engineer from the Katholieke Universiteit (K.U.) Leuven, Belgium, in 1980, the M.S. degree from the Massachusetts Institute of Technology, in 1981, and the Ph.D. degree in mechanical engineering, also from the K.U.Leuven, in 1986. He then worked as a control systems engineer in industry. In 1986 he became Lecturer with the Department of Mechanical Engineering, Division PMA (Production Engineering, Machine Design and Automation), K.U.Leuven, where he is now full professor since 1995. He teaches courses in kinematics and dynamics of machinery, control, and robotics.

Jan De Geeter obtained the degree of Burgerlijk Ingenieur ("Masters") in Mechanical Engineering (1991) from the Katholieke Universiteit Leuven, Belgium. In 1998 he obtained his Doctoral Degree in Applied Sciences from the same university, with a thesis entitled "Constrained system state estimation and task-directed sensing. Application to the local modelling of a structured nuclear environment". Since October 1991, he is a scientific staff member of the Dept. Instrumentation within the Division Reactor Safety of SCK•CEN, the Belgian research centre for nuclear energy. His research interests include teleoperation, man-machine interfacing, sensing in highly radioactive environments and reasoning with uncertainty.

Tine Lefebvre received the degree of mechanical engineer ("Masters"), specialisation Mechatronics, from the Katholieke Universiteit Leuven, Belgium, in 1999. She is currently PhD student in Applied Sciences at the same university, with the Fund for Scientific Research-Flanders (F.W.O) in Belgium. Her research focuses on active sensing for on-line estimation of model uncertainties in sensor-based robot tasks.

Herman Bruyninckx obtained the degrees of Licentiate ("Bachelors") in Mathematics (1984), Burgerlijk Ingenieur ("Masters") in Computer Science (1987) and in Mechatronics (1988), all from the Katholieke Universiteit Leuven, Belgium. In 1995 he got his Doctoral Degree in Applied Sciences from the same university, with a thesis entitled "Kinematic Models for Robot Compliant Motion with Identification of Uncertainties." Since October 1994 he is a Post-Doctoral Researcher with the Fund for Scientific Research-Flanders (F.W.O.) in Belgium, and since 1998 he is Assistant Professor at the K.U.Leuven. He held visiting research positions at the Grasp Lab of the University of Pennsylvania, Philadelphia (1996), and the Robotics Lab of Stanford University (1999). His research include on-line estimation of model uncertainties in sensor-based robot tasks, kinematics of serial and parallel manipulators, geometric foundations of robotics, and Bayesian probability for robotics.