Delft University of Technology

Delft Center for Systems and Control

Technical report 07-017

Distributed Kalman Filtering for Cascaded Systems*

Zs. Lendek, R. Babuška, and B. De Schutter

To cite this work, please refer to the published version:
Zs. Lendek, R. Babuška, and B. De Schutter, "Distributed Kalman filtering for cascaded systems," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 3, pp. 457–469, Apr. 2008. doi:10.1016/j.engappai.2007.05.002

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: https://www.dcsc.tudelft.nl

* This report can also be downloaded via https://dspub.eu/07-017

Distributed Kalman Filtering for Cascaded Systems

Zs. Lendek [a,*] R. Babuška [a] B. De Schutter [a,b]

[a] *Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands, {zs.lendek, r.babuska}@tudelft.nl*

[b] *Marine and Transport Technology Department, Delft University of Technology, b@deschutter.info*

Abstract

The Kalman filter provides an efficient means to estimate the state of a linear process, so that it minimizes the mean of the squared estimation error. However, for naturally distributed applications, the construction and tuning of a centralized observer may present difficulties. Therefore, we propose the decomposition of a linear process model into a cascade of simpler subsystems and the use of a Kalman filter to individually estimate the states of these subsystems. Both a theoretical comparison and simulation examples are presented. The theoretical results show that the distributed observers, except for special cases, do not minimize the overall error covariance, and the distributed observer system is therefore suboptimal. However, in practice, the performance achieved by the cascaded observers is comparable and in certain cases even better than the performance of the centralized observer. A distributed observer system also leads to increased modularity, reduced complexity, and lower computational costs.

Key words: State estimation, Kalman filters, cascaded systems

1 Introduction

Many problems in decision making, control, and monitoring require the estimation of states and possibly uncertain parameters, based on a dynamic system model

* Corresponding author. Tel.:+31 (0)15 27 88573, Fax:+31 (0)15 27 86679

and a sequence of noisy measurements. For such a purpose, dynamic systems are often modeled in the state-space framework, using a state-transition model, which describes the evolution of states over time and a measurement model, which relates the measurement to the states. In state-estimation problems, these models may also be given in a probabilistic form.

The most well-known and widely used probabilistic estimation methods are the Kalman filter and its extension to nonlinear systems, the Extended Kalman Filter (Kalman, 1960; Welch and Bishop, 2002). While the Kalman filter has severe limitations and becomes unstable for highly nonlinear processes, for a linear process, it provides an efficient means to estimate the states so that it also minimizes the mean of the squared error. The filter supports the estimation of past, present and future states, even if a precise model of the system considered is unknown.

Since the publication of the Kalman's seminal paper (Kalman, 1960), the Kalman filter has been the subject of extensive research and applications, particularly in the area of autonomous robots, assisted navigation and sensor data fusion (Lee et al., 1995; Dorfmüller-Ulhaas, 2003; Caron et al., 2006). A wide variety of Kalman filters have also been developed from the Kalman's original formulation: the extended Kalman filter, the information filter and the family of sigma-point Kalman filters (van der Merwe and Wan, 2003).

The Kalman filter is also extensively used in combination with fuzzy systems. Since the Takagi-Sugeno fuzzy model (Wang et al., 2000) is a nonlinear combination of local linear models, Kalman filters have been used to develop state estimators for nonlinear systems which can be represented by models in this form (McGinnity and Irwin, 1997; Simon, 2003; Zhang and Wei, 2003). Fuzzy systems can also be used to tune parameters in a Kalman filter (Aja-Fernandez et al., 2003).

In multi-agent applications, each agent should be able to observe at least some local states and make decisions based on these observations. This can be achieved by using local observers, i.e., each agent implements an individual observer. However, currently, no results are available on the performance analysis of the local observers versus a centralized observer. In this paper, we provide such results and study the conditions under which Kalman-type observers can be used in a cascaded setting.

We propose the decomposition of a system model into cascaded subsystems, and use separate estimators for the subsystems. The idea behind this type of estimation is that many systems can be represented as cascaded, observable subsystems, which are less complex than the original system. Separate observers can be designed for the individual subsystems. This makes the tuning easier. Moreover, different types of observers can be combined, depending on the subsystems considered. Most importantly, such a setting is well suited for a cooperative multi-agent system. Each agent has the task of observing one of the subsystems, possibly using different methods and relying on its own measurements and the information gathered from

other agents. In turn, each agent communicates its own results to other agents. If all the agents in a system use the same estimation method, then such a distributed observer system can be designed and implemented in a modular manner.

In this paper, Kalman-type filters are designed for cascaded subsystems and the performance of the cascaded filters is studied. We present a theoretical comparison of the centralized and cascaded Kalman filter and also compare their performance both on academic examples and simulated real-world problems. These examples illustrate the possible application of the proposed distributed setting not only for large-scale systems, but also a class of nonlinear systems.

The paper is organized as follows. Section 2 presents the proposed cascaded observer setting, Section 3 reviews the Kalman Filter methodology. The distributed Kalman filters are presented in Section 4, with three illustrative examples in Section 5. Section 6 presents the application of the cascaded Kalman filters to a large-scale system and Section 7 presents the application of the proposed approach to a nonlinear system. Finally, Section 8 concludes the paper.

2 Cascaded Subsystems

Consider the following observable linear MIMO system:

$$\begin{aligned} \mathbf{x}(k) &= A\mathbf{x}(k-1) + B\mathbf{u}(k-1) \\ \mathbf{y}(k) &= C\mathbf{x}(k) \end{aligned} \tag{1}$$

and assume that this system can be partitioned into subsystems. For the ease of notation, and without a loss of generality, only two subsystems are considered, $\mathbf{x} = [\mathbf{x}^{1^T}\ \mathbf{x}^{2^T}]^T$ and $\mathbf{y} = [\mathbf{y}^{1^T}\ \mathbf{y}^{2^T}]^T$:

$$\begin{aligned} \mathbf{x}^1(k) &= A_{11}\mathbf{x}^1(k-1) + B_1\mathbf{u}(k-1) \\ \mathbf{y}^1(k) &= C_{11}\mathbf{x}^1(k) \end{aligned} \tag{2}$$

and

$$\begin{aligned} \mathbf{x}^2(k) &= A_{22}\mathbf{x}^2(k-1) + B_2\mathbf{u}(k-1) + A_{21}\mathbf{x}^1(k-1) \\ \mathbf{y}^2(k) &= C_{22}\mathbf{x}^2(k) + C_{21}\mathbf{x}^1(k) \end{aligned} \tag{3}$$

so that (2) is observable. Note that, since both systems (1) and (2) are observable, this also means that the subsystem (3) is observable for given $\mathbf{x}^1(k)$ and $\mathbf{x}^1(k-1)$. In fact, for subsystem (3), $\mathbf{x}^1(k-1)$ is an input.

In general, such a partition of the model does not necessarily exist. The necessary and sufficient condition for the existence of a partition is that the A and C matrices can be transformed into block lower-triangular forms. If the partition exists, it might

not be unique. Consider, for instance, the system

$$\begin{aligned} x_1(k) &= x_1(k-1) + x_3(k-1) & y_1(k) &= x_1(k) \\ x_2(k) &= x_2(k-1) + x_3(k-1) & y_2(k) &= x_2(k) \\ x_3(k) &= u(k-1) \end{aligned}$$

This system is observable, and there are two possible ways to partition it: by using as the first subsystem

$$\begin{aligned} x_1(k) &= x_1(k-1) + x_3(k-1) & y_1(k) &= x_1(k) \\ x_3(k) &= u(k-1) \end{aligned}$$

or, by using as the first subsystem

$$\begin{aligned} x_2(k) &= x_2(k-1) + x_3(k-1) & y_2(k) &= x_2(k) \\ x_3(k) &= u(k-1) \end{aligned}$$

both being observable.

Given the above partitioning, state estimators can be designed for the two subsystems separately, with the second observer using the results of the first observer. Such a structure is depicted in Figure 1.

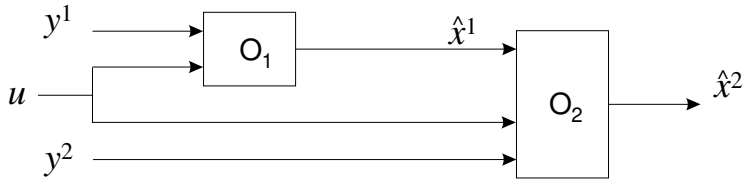


Fig. 1. Cascaded observers.

Although the framework is presented for linear, time-invariant systems, the proposed distributed observer approach is applicable also for cascaded general nonlinear systems. In general, the distributed estimation approach is applicable when the system considered can be represented by a directed acyclic graph structure, where each node corresponds to an observable subsystem. In many cases, these subsystems are much simpler than the global system (e.g., some subsystems may be linear). Therefore, simpler observer can be designed.

Currently, a general analysis of the joint performance (convergence, convergence speed, optimality) of the cascaded observers and a centralized observer designed for a cascaded system does not exist. In the remainder of the paper we study the conditions under which Kalman-type filters can be designed for the two subsystems of the form (2)–(3), so that the performance of the cascaded filters is the same as that of a single Kalman filter for system (1).

3 Kalman Filter

The Kalman filter addresses the problem of estimating the state $\mathbf{x} \in \mathbb{R}^n$ of a linear discrete-time process:

$$\begin{aligned} \mathbf{x}(k) &= A\mathbf{x}(k-1) + B\mathbf{u}(k-1) + \mathbf{w}(k-1) \\ \mathbf{y}(k) &= C\mathbf{x}(k) + \mathbf{v}(k) \end{aligned} \tag{4}$$

with $\mathbf{x}_0$ (initial state) and P_0 (initial covariance of the states) known or previously estimated.

The inputs **w** and **v** are random variables, representing the process and measurement noise, respectively. These random variables are assumed to be independent, white and with normal probability distributions $\mathbf{w}(k) \sim \mathcal{N}(0, Q)$ and $\mathbf{v}(k) \sim \mathcal{N}(0, R)$. In general, the process and measurement noise covariance matrices (Q and R), the state transition matrix A and the measurement matrix C can change at every time step; however, here, they are assumed constant to simplify the notation. The objective is to recursively estimate or filter the state $\mathbf{x}_k$ based on the available measurements.

The Kalman filter works in two steps: prediction:

$$\begin{aligned} \widehat{\mathbf{x}}(k|k-1) &= A\mathbf{x}(k-1) + B\mathbf{u}(k-1) \\ P(k|k-1) &= AP(k-1)A^T + Q \end{aligned} \tag{5}$$

and update or correction:

$$\begin{aligned} \widehat{\mathbf{x}}(k) &= \widehat{\mathbf{x}}(k|k-1) + K(k)(\mathbf{y}(k) - C\widehat{\mathbf{x}}(k|k-1)) \\ P(k) &= (I - K(k)C)P(k|k-1)(I - K(k)C)^T + K(k)RK^T(k) \end{aligned} \tag{6}$$

where $\widehat{\mathbf{x}}(k)$ ($P(k)$) refers to the estimate of the states (covariance) obtained by using all the measurements up to k. The Kalman gain $K(k)$ is computed at each step k so that it minimizes the error covariance $P(k)$. This is obtained by minimizing the trace of $P(k)$ at every step:

$$\begin{aligned} &\frac{\partial(\text{tr}(P(k)))}{\partial K(k)} = -2CP(k|k-1) + 2(CP(k|k-1)C^T + R)K^T(k) = 0 \\ &\implies K(k) = P(k|k-1)C^T(CP(k|k-1)C^T + R)^{-1} \end{aligned} \tag{7}$$

Then, assuming that at step $k-1$ the error covariance is $P(k-1)$, the covariance and the Kalman gain at step k can be expressed as:

$$\begin{aligned} P(k) &= (I - K(k)C)(AP(k-1)A^T + Q)(I - K(k)C)^T + K(k)RK^T(k) \\ K(k) &= (AP(k-1)A^T + Q)C^T(C(AP(k-1)A^T + Q)C^T + R)^{-1} \end{aligned} \tag{8}$$

A generic Kalman filter algorithm is summarized in Algorithm 1.

Algorithm 1 Kalman filter

Input: u, y, Q, R, A, B, C, $\mathbf{x}_0$, P_0

Output: $\widehat{\mathbf{x}}$, P

for $k = 1, 2, \ldots$ **do**

Prediction:

$\widehat{\mathbf{x}}(k|k-1) = A\mathbf{x}(k-1) + B\mathbf{u}(k-1)$ {predict states}

$P(k|k-1) = AP(k-1)A^T + Q$ {predict covariance}

Update:

$K(k) = P(k|k-1)C^T(CP(k|k-1)C^T + R)^{-1}$ {compute Kalman gain}

$\widehat{\mathbf{x}}(k) = \widehat{\mathbf{x}}(k|k-1) + K(k)(\mathbf{y}(k) - C\widehat{\mathbf{x}}(k|k-1))$ {correct states}

$P(k) = (I - K(k)C)P(k|k-1)(I - K(k)C)^T + K(k)RK^T(k)$ {correct covariance}

end for

4 Distributed Kalman Filters

Consider the linear system (4), corrupted with zero-mean Gaussian noise and assume that the system can be written in the following form:

$$\begin{aligned}
\begin{pmatrix} \mathbf{x}^1(k) \\ \mathbf{x}^2(k) \end{pmatrix} &= \begin{pmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} \mathbf{x}^1(k-1) \\ \mathbf{x}^2(k-1) \end{pmatrix} + \begin{pmatrix} B_1 \\ B_2 \end{pmatrix} \mathbf{u}(k-1) + \begin{pmatrix} \mathbf{w}^1(k-1) \\ \mathbf{w}^2(k-1) \end{pmatrix} \\
\begin{pmatrix} \mathbf{y}^1(k) \\ \mathbf{y}^2(k) \end{pmatrix} &= \begin{pmatrix} C_{11} & 0 \\ C_{21} & C_{22} \end{pmatrix} \begin{pmatrix} \mathbf{x}^1(k) \\ \mathbf{x}^2(k) \end{pmatrix} + \begin{pmatrix} \mathbf{v}^1(k) \\ \mathbf{v}^2(k) \end{pmatrix}
\end{aligned} \tag{9}$$

i.e., as two cascaded subsystems. Our goal is to design separate observers for the two subsystems, so that the cascaded observers have the same performance (error covariance) as the Kalman filter designed for the joint system. Note that for the system to be cascaded without losing available information (e.g. cross-covariances of states belonging to different subsystems), it is also necessary that the covariance matrices are block-diagonal, i.e., $Q = \begin{pmatrix} Q_1 & 0 \\ 0 & Q_2 \end{pmatrix}$ and $R = \begin{pmatrix} R_1 & 0 \\ 0 & R_2 \end{pmatrix}$. While this condition appears restrictive, in practice one rarely knows the true cross-covariances and it is often assumed that the covariance matrix is diagonal (Hue et al., 2002; Aja-Fernandez et al., 2003).

Since our goal is to design separate observers for the two subsystems, but still minimize the error covariance, it is logical to use separate Kalman filters, one for each subsystem. The first subsystem can be expressed as:

$$\begin{aligned}\mathbf{x}^1(k) &= A_{11}\mathbf{x}^1(k-1) + B_1\mathbf{u}(k-1) + \mathbf{w}^1(k-1)\\ \mathbf{y}^1(k) &= C_{11}\mathbf{x}^1(k) + \mathbf{v}^1(k)\end{aligned} \tag{10}$$

which is a linear system, with $\mathbf{w}^1(k) \sim \mathcal{N}(0, Q_1)$ and $\mathbf{v}^1(k) \sim \mathcal{N}(0, R_1)$ and the deterministic input $\mathbf{u}$. In order to minimize the error covariance for this subsystem, the Kalman filter presented in Section 3 is used. Then, for the first subsystem (with the deterministic input $\mathbf{u}$), the covariance and the gain at each time step can be written as:

$$\begin{aligned}P_1(k) &= (I - K_1(k)C_{11})(A_{11}P_1(k-1)A_{11}^T + Q_1)(I - K_1(k)C_{11})^T + K_1(k)R_1K_1^T(k)\\ K_1(k) &= (A_{11}P_1(k-1)A_{11}^T + Q_1)C_{11}^T(C_{11}(A_{11}P_1(k-1)A_{11}^T + Q_1)C_{11}^T + R_1)^{-1}\end{aligned} \tag{11}$$

The second subsystem can be expressed as:

$$\begin{aligned}\mathbf{x}^2(k) &= A_{22}\mathbf{x}^2(k-1) + B_2\mathbf{u}(k-1) + A_{21}\mathbf{x}^1(k-1) + \mathbf{w}^1(k-1)\\ \mathbf{y}^2(k) &= C_{22}\mathbf{x}^2(k) + C_{21}\mathbf{x}^1(k) + \mathbf{v}^2(k)\end{aligned} \tag{12}$$

with $\mathbf{w}^2(k) \sim \mathcal{N}(0, Q_2)$ and $\mathbf{v}^2(k) \sim \mathcal{N}(0, R_2)$, the deterministic input $\mathbf{u}$ and the stochastic variable $\mathbf{x}^1$. In a multi-agent setting, agents may communicate only the state estimate, and not the covariance. In such a case, $\mathbf{x}^1$ can also be considered as a deterministic input. Thus, two cases can be distinguished.

Case 1: Use $\mathbf{x}^1$ as another **deterministic input** besides $\mathbf{u}$ for the second subsystem. This will be the case in a multi-agent system, if the agent entirely trusts the estimate of another agent, considering it correct and not taking into account possible errors, or a distribution of the estimate. In this case, the Kalman filter can be used also for this subsystem, and the expression for covariance the and the gain are:

$$\begin{aligned}P_2(k) &= (I - K_2(k)C_{22})(A_{22}P_2(k-1)A_{22}^T + Q_2)(I - K_2(k)C_{22})^T + K_2(k)R_2K_2^T(k)\\ K_2(k) &= (A_{22}P_2(k-1)A_{22}^T + Q_2)C_{22}^T(C_{22}(A_{22}P_2(k-1)A_{22}^T + Q_2)C_{22}^T + R_2)^{-1}\end{aligned} \tag{13}$$

However, in this case, the computed error covariance is not equal to the true error covariance for the second subsystem.

Case 2: If the covariance of the estimates is also available, then $\mathbf{x}^1$ can be considered as a **stochastic input**, with computed covariance $P_1(k)$, for the second subsystem. For this case, a Kalman-type gain can be computed by minimizing the trace of the error covariance for the second subsystem, assuming that $\mathbf{x}^1$ is a stochastic

variable with a known covariance matrix P_1:

$$\begin{aligned}0 = &-2C_{22}(A_{22}P_2(k-1)A_{22}^T + A_{21}P_1(k-1)A_{21}^T + Q_2)+\\ &2(C_{22}(A_{22}P_2(k-1)A_{22}^T + A_{21}P_1(k-1)A_{21}^T + Q_{22})C_{22}^T + R_2)K_2(k)^T+\\ &2C_{21}P_1(k-1)C_{21}^T K_2^T(k)\\ K_2(k) = &(C_{22}(A_{22}P_2(k-1)A_{22}^T + A_{21}P_1(k-1)A_{21}^T + Q_2))^T \cdot ((C_{22}(A_{22}P_2(k-1)A_{22}^T+\\ &A_{21}P_1(k-1)A_{21}^T + Q_{22})C_{22}^T + R_2 + C_{21}P_1(k-1)C_{21}^T)^{-1})^T\end{aligned} \tag{14}$$

The solution of the above equation leads to the following expression of $P_2(k)$:

$$\begin{aligned}P_2(k) = &(I - K_2(k)C_{22})(A_{22}P_2(k-1)A_{22}^T + A_{21}P_1(k-1)A_{21}^T+\\ &Q_2)(I - K_2(k)C_{22})^T + K_2(k)R_2K_2^T(k) + K_2(k)C_{21}P_1(k-1)(K_2(k)C_{21})^T\end{aligned} \tag{15}$$

and $P_2(k)$ is the true covariance obtained for the states of the second subsystem.

In both cases, the observer gain and the covariance matrix for the whole system are expressed as:

$$K = \begin{pmatrix} K_1 & 0 \\ 0 & K_2 \end{pmatrix} \qquad P = \begin{pmatrix} P_1 & 0 \\ 0 & P_2 \end{pmatrix} \tag{16}$$

However, only in the second case (if $\mathbf{x}^1$ is considered a stochastic input), the covariance matrix for the joint system equals the true covariance obtained by the observers.

Proposition 1: The cascaded setting achieves the same error covariance as the centralized Kalman filter if and only if the subsystems are independent, i.e., in (9), $A_{21} = 0$, $C_{21} = 0$, $R_{12} = 0$ and $Q_{12} = 0$.

Proof: Assume that the joint form of the cascaded Kalman filters is equivalent to that of the centralized Kalman filter. If this assumption holds, then it is also possible to decompose the error system and the Kalman gain obtained for the joint system. In order to study this possibility, let

$$P(k|k-1) = \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{pmatrix} \tag{17}$$

Then,

$$\begin{aligned}&CP(k|k-1)C^T + R =\\ &\quad = \begin{pmatrix} C_{11}P_{11}C_{11}^T + R_{11} & C_{11}P_{11}C_{21}^T + C_{11}P_{12}C_{22}^T + R_{12} \\ C_{21}P_{11}C_{11}^T + C_{22}P_{21}C_{11}^T + R_{21} & C_{21}(P_{11}C_{21}^T + P_{12}C_{22}^T) + C_{22}(P_{21}C_{21}^T + P_{22}C_{22}^T) + R_{22} \end{pmatrix}\end{aligned}$$

The conditions for the observer to be partitioned without losing optimality, are:

$$\begin{aligned}
P_{11}C_{11}^T &= K_1(k)(C_{11}P_{11}C_{11}^T + R_{11}) \\
P_{21}C_{21}^T + P_{22}C_{22}^T &= K_2(k)(C_{21}(P_{11}C_{21}^T + P_{12}C_{22}^T) + C_{22}(P_{21}C_{21}^T + P_{22}C_{22}^T) + R_{22}) \\
P_{11}C_{21}^T + P_{12}C_{22}^T &= K_1(k)(C_{11}P_{11}C_{21}^T + C_{11}P_{12}C_{22}^T + R_{12}) \\
P_{21}C_{11}^T &= K_2(k)(C_{11}P_{11}C_{21}^T + C_{11}P_{12}C_{22}^T + R_{12})^T
\end{aligned} \tag{18}$$

Moreover,

$$\begin{aligned}
P(k|k-1) &= \begin{pmatrix} P_{11} & P_{12} \\ P_{12}^T & P_{22} \end{pmatrix} \\
&= \begin{pmatrix} A_{11}P_{11}(k-1)A_{11}^T + Q_{11} & A_{11}P_{11}(k-1)A_{21}^T + A_{11}P_{12}(k-1)A_{22}^T + Q_{12} \\ \begin{pmatrix} A_{11}P_{11}(k-1)A_{21}^T + \\ +A_{11}P_{12}(k-1)A_{22}^T + Q_{12})^T \end{pmatrix} & \begin{pmatrix} A_{21}(P_{11}(k-1)A_{21}^T + P_{12}(k-1)A_{22}^T) \\ +A_{22}(P_{21}(k-1)A_{21}^T + P_{22}(k-1)A_{22}^T) + Q_{22} \end{pmatrix} \end{pmatrix}
\end{aligned} \tag{19}$$

and it is also required that $P_{21} = P_{12}^T = 0$ (due to the form of the covariance matrix obtained in (16)). Under these conditions, the requirements expressed by (18) will only be fulfilled if the two subsystems are independent, i.e. $A_{21} = 0$, $C_{21} = 0$, $R_{12} = 0$ and $Q_{12} = 0$. Only in this case, the cross-covariances $P_{12}(k|k-1)$ and $P_{12}(k)$ and their transpose will also be zero. □

Since the distributed filters obtain the same performance as the Kalman filter if and only if the subsystems are independent, in general, the distributed observers will not minimize the joint covariance. However, in practice, the performance of the centralized and distributed observers is comparable, as demonstrated in the following sections.

Although the framework was presented for linear time-invariant Gaussian systems, the proposed distributed filtering approach is also applicable also the system considered can be cascaded so that the subsystems are linear time-varying. Therefore, the deterministic version of the proposed cascaded Kalman filter can be applied even for a class of nonlinear systems, where the individual subsystems are linear time-varying. When only two subsystems are considered, the nonlinear system has to be bilinear, so that the states present in the bilinear terms are contained in different subsystems. A real-world application example to illustrate this feature, is presented in Section 8.

5 Examples

In the previous sections, the basic form of the Kalman filter and the proposed distributed version were given. Here, three examples are presented to compare the performance of the distributed and centralized observers, both in open-loop and closed-loop control.

1 Distributed Kalman Filter in Open-Loop

Consider the following, randomly generated discrete-time system:

$$\begin{aligned}\mathbf{x}(k) &= A\mathbf{x}(k-1) + Bu(k-1) + \mathbf{w}(k-1)\\ \mathbf{y}(k) &= C\mathbf{x}(k) + \mathbf{v}(k)\end{aligned}$$

with

$$A = \begin{pmatrix} -0.2034 & 0 & 0 \\ -0.8520 & -0.3182 & -1.2951 \\ 0.0218 & 0.5776 & 0.9522 \end{pmatrix} \qquad B = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \qquad C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

$$\mathbf{w}(k) \sim \mathcal{N}(0, Q) \qquad \mathbf{v}(k) \sim \mathcal{N}(0, R)$$

$$Q = \begin{pmatrix} 0.6818 & 0.2244 & 0.0577 \\ 0.2244 & 0.2796 & 0.1039 \\ 0.0577 & 0.1039 & 0.2263 \end{pmatrix} \qquad R = \begin{pmatrix} 0.1679 & 0.0616 \\ 0.0616 & 0.1204 \end{pmatrix} \tag{20}$$

It can be easily seen that the deterministic part of the system can be cascaded. Two cases are distinguished:

a) **Approximate noise covariance:** Since the cascaded filters do not take into account the cross-covariance between the subsystems, in order to ensure the exact same conditions for both types of filters, consider for both the Kalman filter and the cascaded filters the following approximate noise covariances:

$$\bar{Q} = \begin{pmatrix} 0.6818 & 0 & 0 \\ 0 & 0.2796 & 0.1039 \\ 0 & 0.1039 & 0.2263 \end{pmatrix} \qquad \bar{R} = \begin{pmatrix} 0.1679 & 0 \\ 0 & 0.1204 \end{pmatrix} \tag{21}$$

The input signal is presented in Figure 2. Using the centralized Kalman filter,

after 300 steps, we obtain:

$$P = \begin{pmatrix} 0.1349 & 0.0004 & 0.0015 \\ 0.0004 & 0.1091 & -0.0438 \\ 0.0015 & -0.0438 & 0.4804 \end{pmatrix} \qquad K = \begin{pmatrix} 0.8036 & 0.0036 \\ 0.0026 & 0.9060 \\ 0.0090 & -0.3640 \end{pmatrix}$$

while for the cascaded subsystems:

$$P_c = \begin{pmatrix} 0.1350 & 0 & 0 \\ 0 & 0.1078 & -0.0461 \\ 0 & -0.0461 & 0.4646 \end{pmatrix} \qquad K_c = \begin{pmatrix} 0.8037 & 0 \\ 0 & 0.8982 \\ 0 & -0.3921 \end{pmatrix}$$

if $\mathbf{x}^1$ is considered to be a deterministic input (Case 1) and

$$P_c = \begin{pmatrix} 0.1350 & 0 & 0 \\ 0 & 0.1091 & -0.0438 \\ 0 & -0.0438 & 0.4812 \end{pmatrix} \qquad K_c = \begin{pmatrix} 0.8037 & 0 \\ 0 & 0.9059 \\ 0 & -0.3921 \end{pmatrix}$$

if $\mathbf{x}^1$ is considered to be a stochastic input (Case 2).

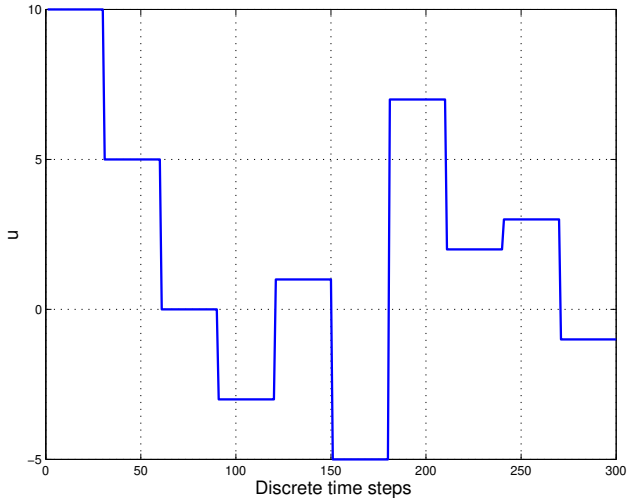


Fig. 2. Input used for the distributed filters in open-loop.

Histograms of the residuals obtained for x_3 (the state which is not measured) with the centralized Kalman filter, and for both cases of the distributed filters are presented in Figure 3. The statistics of the distributions of the residuals for all states and observers are given in Table 1. It can be seen that the performance of the cascaded observers is comparable with that obtained with the centralized observer.

Table 1
Statistics of the residuals when the centralized and distributed observers use the same covariance matrix.

State	Method	Mean	Standard deviation
x_1	centralized	-0.0032	0.1890
	cascaded	-0.0033	0.1889
x_2	centralized	-0.0105	0.1246
	cascaded deterministic	-0.0103	0.1262
	cascaded stochastic	-0.0113	0.1318
x_3	centralized	0.0420	0.4022
	cascaded deterministic	0.0397	0.4035
	cascaded stochastic	0.0420	0.4024

b) **True noise covariance:** The Kalman filter uses the true noise covariances (20), while the cascaded filters neglect the cross covariance between the subsystems and consider only (21). The same input is used as in the previous case. The centralized filter performs slightly better than the cascaded one.

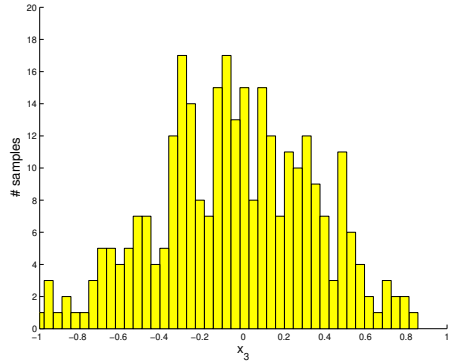
The histogram of the residuals obtained for x_3 is presented in Figure 4. The statistics of the distributions of the residuals for all states and observers are given in Table 2.

For this case, the final covariance and the Kalman gain obtained after 300 steps by the centralized Kalman filter are

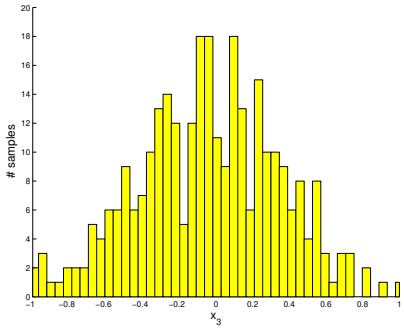
$$P = \begin{pmatrix} 0.1350 & 0.0496 & 0.0098 \\ 0.0496 & 0.1074 & -0.0359 \\ 0.0098 & -0.0359 & 0.4214 \end{pmatrix} \qquad K = \begin{pmatrix} 0.8036 & 0.0002 \\ -0.0399 & 0.9126 \\ 0.2064 & -0.4036 \end{pmatrix}$$

while those obtained by the cascaded observers are the same as in item 1.

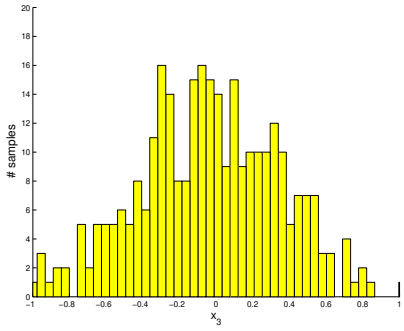
The statistics of the residuals confirm that the cascaded filters are suboptimal. However, the difference between the residuals is minimal, even if $\mathbf{x}^1$ obtained from the first subsystem is considered as a deterministic input, and the computed covariance is not the correct one.



(a) Residuals for x_3 with the centralized Kalman filter.



(b) Residuals for x_3 with the cascaded Kalman filter and deterministic input (case 1).



(c) Residuals for x_3 with the cascaded Kalman filter and stochastic input (case 2).

Fig. 3. Results when the centralized and cascaded filters use the same covariance matrix.

2 Distributed Kalman Filter in Closed-Loop

In this section, two examples are presented to compare the performance of the distributed and centralized observers, in closed-loop control. For this purpose, a state-feedback control is designed based on the system model. However, not all the states are measured, and the control input is computed based on the estimated states. Such a setting is depicted in Figure 5.

Example 2: Consider the following, randomly generated discrete-time system:

$$\mathbf{x}(k) = A\mathbf{x}(k-1) + Bu(k-1) + \mathbf{w}(k-1)$$
$$\mathbf{y}(k) = C\mathbf{x}(k) + \mathbf{v}(k)$$

Table 2

Statistics of residuals with the cascaded Kalman filter disregarding the cross-covariance.

State	Method	Mean	Standard deviation
x_1	centralized	-0.0077	0.2058
	cascaded	-0.0076	0.2058
x_2	centralized	-0.0074	0.1406
	deterministic	-0.0106	0.1444
	stochastic	-0.0106	0.1522
x_3	centralized	-0.0070	0.3757
	deterministic	0.0072	0.4393
	stochastic	0.0077	0.4365

$$A = \begin{pmatrix} 1.1274 & 0 & 0 \\ 0.0639 & 0.9091 & 0.0391 \\ 0.1381 & -0.2306 & 1.0020 \end{pmatrix} \qquad B = \begin{pmatrix} 0.1 \\ 0 \\ 0 \end{pmatrix} \qquad C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

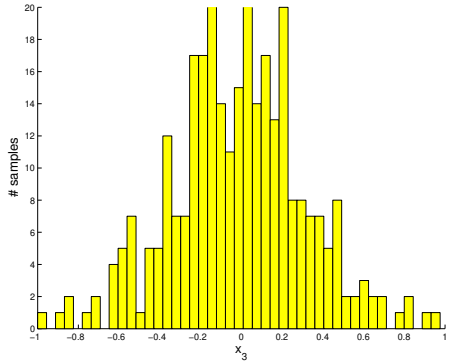
$$\mathbf{w}(k) \sim \mathcal{N}(0, Q) \qquad \mathbf{v}(k) \sim \mathcal{N}(0, R)$$

$$Q = \begin{pmatrix} 0.0097 & 0.0026 & 0.0032 \\ 0.0026 & 0.0066 & 0.0002 \\ 0.0032 & 0.0002 & 0.0128 \end{pmatrix} \qquad R = \begin{pmatrix} 0.0035 & 0.0078 \\ 0.0078 & 0.0118 \end{pmatrix}$$

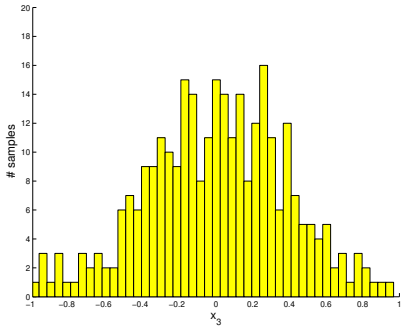
for which a state feedback control with constant gain $L = [9.3853 \ 16.2397 \ 3.2858]$ has been computed.

The deterministic part of the system is decomposed. The cascaded filters do not take into account the noise covariances between the subsystems. Now the control is applied for four different cases:

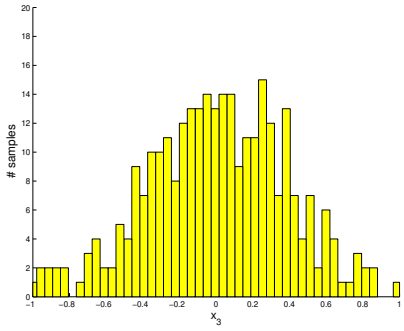
(1) the states are known, and the controller is applied directly;
(2) the first two states are measured, and the control input is computed based on the estimate given by a centralized Kalman filter;
(3) the first two states are measured, and the control input is computed based on the estimate given by a cascaded Kalman-type filter, with the second subsystem considering the estimates of the first subsystem as stochastic inputs;
(4) the first two states are measured, and the control is computed based on the estimate given by a cascaded Kalman-type filter, with the second subsystem



(a) Residuals for x_3 with centralized Kalman filter.



(b) Residuals for x_3 with cascaded Kalman filter and deterministic input (Case 1).



(c) Residuals for x_3 with cascaded Kalman filter and stochastic input (Case 2).

Fig. 4. Results with discarded cross-covariances.

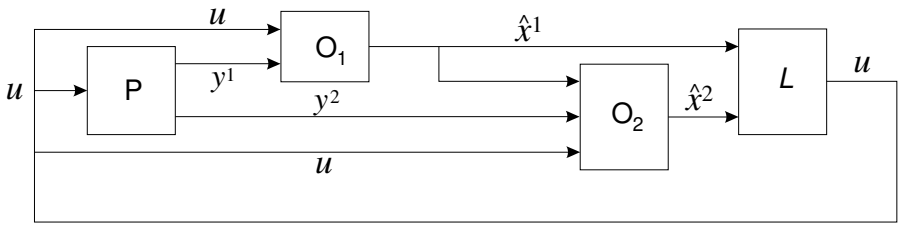
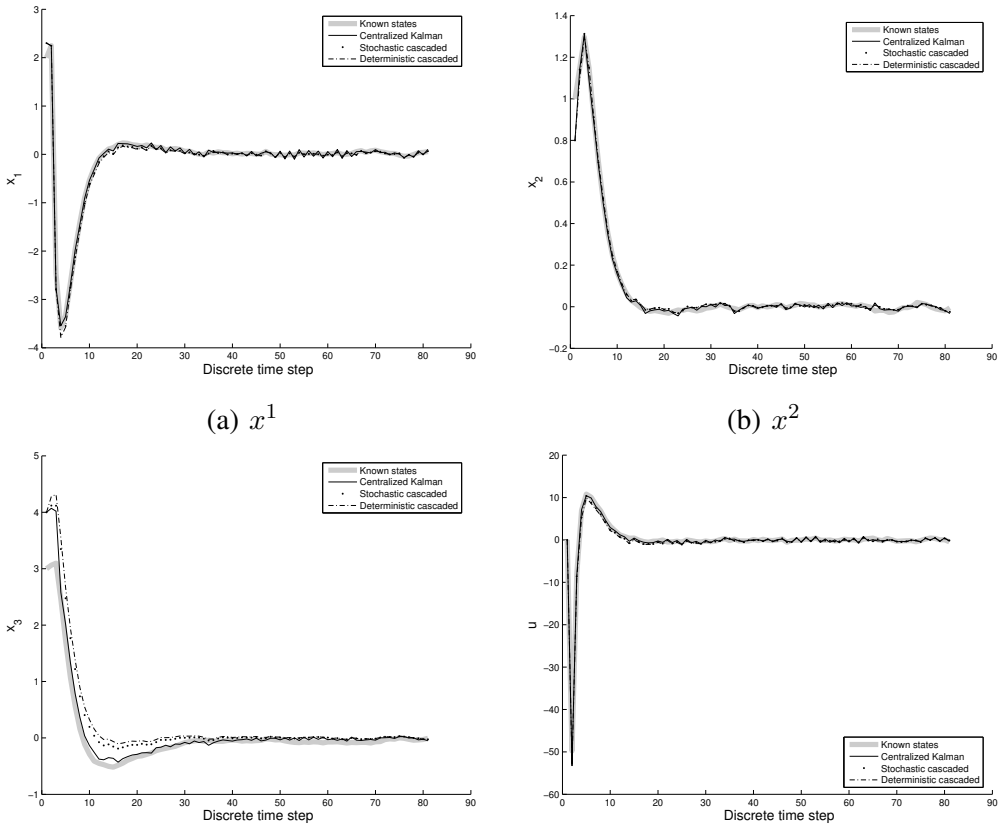


Fig. 5. Cascaded observers in closed-loop.

considering the estimates of the first subsystem as deterministic inputs.

The results obtained can be seen in Figure 6. The estimation error for the first two states, which are measured, is very small. However, for the third state the estimate of the centralized observers converges more slowly than the estimate of the cascaded one.



(a) x^1 (b) x^2

(c) x^3 (d) u

Fig. 6. Example 2 state estimates in closed-loop with different observers (state feedback without observer, Kalman, stochastic cascaded, deterministic cascaded).

Example 3: Consider the following, randomly generated discrete-time system:

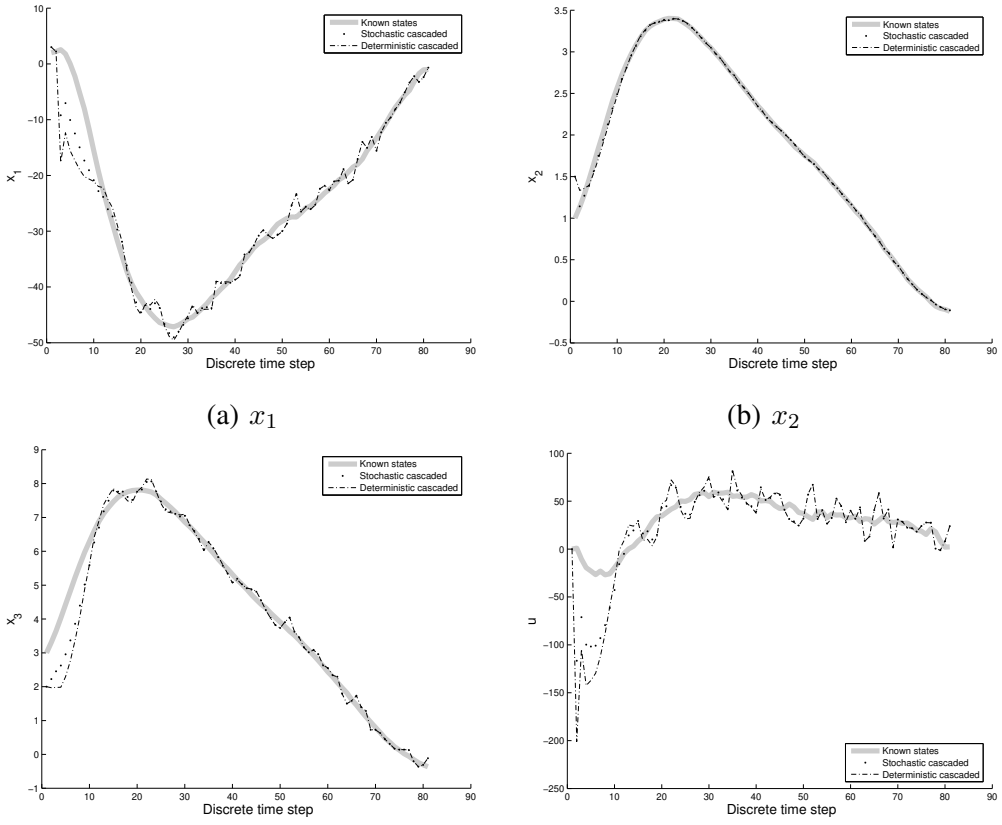
$$\mathbf{x}(k) = A\mathbf{x}(k-1) + Bu(k-1) + \mathbf{w}(k-1)$$
$$\mathbf{y}(k) = C\mathbf{x}(k) + \mathbf{v}(k)$$

$$A = \begin{pmatrix} 1.1137 & 0 & 0 \\ 0.0087 & 1.0829 & 0.0117 \\ 0.0170 & -0.0009 & 1.0909 \end{pmatrix} \qquad B = \begin{pmatrix} 0.1 \\ 0 \\ 0 \end{pmatrix} \qquad C = \begin{pmatrix} 1\ 0\ 0 \\ 0\ 1\ 0 \end{pmatrix}$$

for which a state feedback control with constant gain $L = [5.1743\ 298.9764\ -106.2475]$ has been computed. The state and measurement noises have the same covariance as in the previous example. Note that the closed-loop system with the centralized Kalman filter becomes unstable.

The estimates of the states using the distributed observers can be seen in Figure 7. While the estimate is noisy, and this noise is also reflected in the control law, the system does not become unstable. A possible explanation of this result is the following: though the Kalman filter is optimal, its convergence is not guaranteed,

in particular when it is used in closed loop with a linear state feedback controller. This is why the systems may become unstable.



(a) x_1 (b) x_2

(c) x_3 (d) u

Fig. 7. Example 3 state estimates in closed-loop with different observers (state feedback without observer, stochastic cascaded, deterministic cascaded). The system using the centralized Kalman filter is unstable in this case, and the results are not shown.

Based on theoretical considerations, the situation when the centralized system is stable and the distributed one is unstable is also possible. However, simulations indicate that in most cases, if the cascaded system becomes unstable, then also the centralized does. To quantify the occurrences, batches of simulations were run, using randomly generated system matrices. Two cases were studied:

(1) the Kalman filter uses the full covariance matrix: the centralized filter failed in 2327 cases out of 3000 cases, while the cascaded only in 69. Both filters failed simultaneously in 69 cases.
(2) exactly the same covariance matrices are used: the failure rate for the centralized Kalman filter is 882/3000, versus 78/3000 for the cascaded filters, and both fail simultaneously in 77 cases, which means than in only one case from 3000 the centralized Kalman filter is stable while the cascaded is not.

The system matrices were independently generated in each case. It can be seen that

the centralized Kalman filter becomes unstable more often than the cascaded filter.

A possible explanation is that the error for the states in the two subsystems is not the same for the centralized and cascaded case. For instance, a large overshoot of the error for a state from the second subsystem does not influence the states of the first subsystem in the cascaded setting. However, this can also be a shortcoming in cases, when the second subsystem could be used to dampen the error of the first subsystem. This is why, in one trial the centralized system is stable while the cascaded is not.

6 Cascaded Kalman Filtering for a Water Treatment Plant

This example demonstrates the application of the proposed Kalman filter approach to a linear large-scale system. Since the process consists of several treatment stages, separate observer design is convenient in this application.

Drinking water is usually collected from a natural water source, such as a river or a lake and undergoes several treatment stages to remove chemicals and organic materials.

A simple model of a treatment plant describes the effect of different chemical dosages and reactions through the so-called M and P numbers (concentrations), related to the pH. These numbers need to be estimated through 12 treatment steps: coagulation, mixing, HCl dosage, rapid sand filtration, transportation, ozonization, second HCl dosage, activated carbon filtration, NaOH dosage, slow sand filtration and storage before distribution. In each treatment step, the dynamic model can be expressed as:

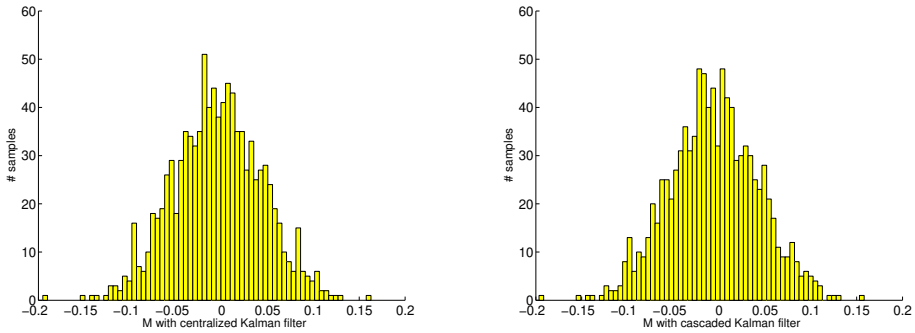
$$\begin{aligned}
\dot{M} &= \frac{F}{V}(M_{\text{prev}} - M) + \frac{F}{V} f_M(r_{\text{in}}) - R_M(M, P, r, \tau) \\
\dot{P} &= \frac{F}{V}(P_{\text{prev}} - P) + \frac{F}{V} f_P(r_{\text{in}}) - R_P(M, P, r, \tau) \\
\dot{r} &= \frac{F}{V}(r_{\text{in}} - r) - R_r(M, P, r, \tau)
\end{aligned}$$

where F is the flow, V is the water volume in the corresponding process step, r is the reactant, R_M, R_P and R_r are the reaction kinetics in the treatment step, dependent of the temperature τ, f_M and f_P are the instantaneous changes in M and P due to the dosage of chemicals and M_{prev} and P_{prev} are the M and P numbers from the previous treatment step.

In this paper, we consider f_M and f_P are linear in r_{in}, and R_M, R_P and R_r as known linear combinations of M, P and r. The parameters of these functions also change in each treatment step, depending on the reaction type. The measurements are linear

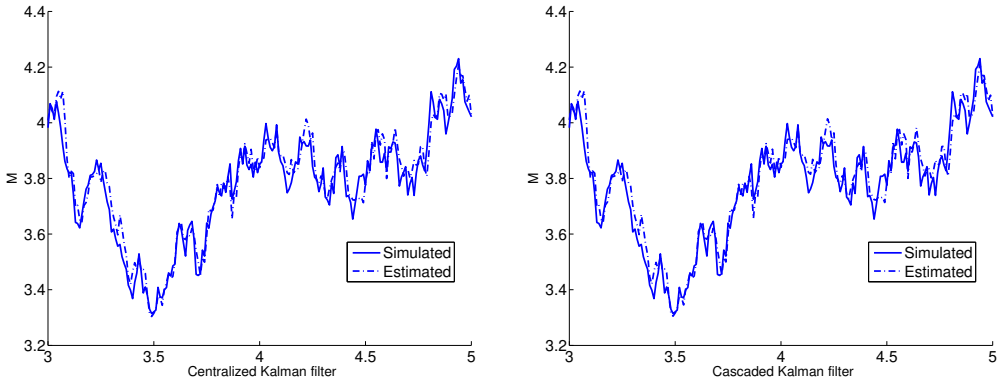
combinations of M, P and r, different for each treatment step. For estimation purposes, the model was first discretized with the Euler method, using a sample period of $T = 0.01$ s, to preserve the accuracy of the simulation. The model was simulated using randomly generated reactant data, and both the states and the measurements were corrupted by zero-mean, Gaussian noises. We compare the estimates of the centralized Kalman filter and a cascaded deterministic Kalman filter (case 1). The total number of states is 36 (3 in each stage), and in each stage a different linear combination of the states is measured.

The histogram of the residuals for the 12th step for the state M and two types of filters are presented in Figure 8, a section of the true and estimated trajectories are given in Figure 9, while the means and standard deviations of all the states, computed for 1000 time steps, are given in Table 3. Based on the data presented, one can easily see, that though the standard deviations obtained by the cascaded filter are indeed slightly larger than those obtained by the centralized Kalman filter, the difference is minimal.



(a) M with centralized KF (b) M with cascaded KF

Fig. 8. Histograms of the residuals of M using centralized and cascaded Kalman filters.



(a) Trajectory of M with centralized KF (b) Trajectory of M with cascaded KF

Fig. 9. Section of the trajectory of M, estimated by centralized and cascaded Kalman filters.

Table 3
Statistics of residuals of the states.

State	Mean (centralized)	Std (centralized)	Mean (distributed)	Std (distributed)
M	-0.0047	0.0482	-0.0060	0.0483
P	0.0056	0.0506	0.0065	0.0509
r	-0.0041	0.0407	-0.0051	0.0408

7 Cascaded Kalman Filtering for Estimating Overflow Losses in a Hopper Dredger

Prior to stating the estimation problem, the principle of the dredging process is briefly explained. The dredger is a large ship using a drag head to excavate soil from the sea bottom. A mixture of soil and water is transported through a pipe to the hopper, which is a large storage tank inside the ship (see Figure 10).

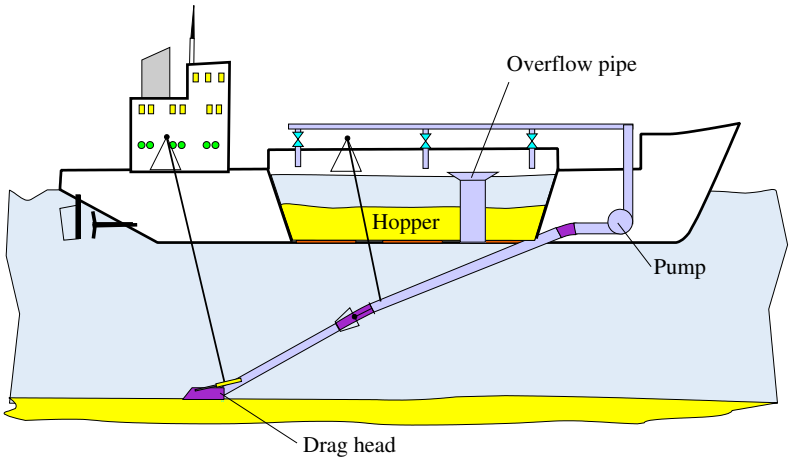


Fig. 10. Schematic drawing of a hopper dredger.

The soil gradually settles at the bottom of the hopper, while excessive water (in fact low-density mixture) is discharged through an overflow pipe whose level can be adjusted (see Figure 11). As the height of the settled sand bed rises, so does the concentration of the overflow mixture and eventually the losses become so high that it is no longer economical to continue dredging. The ship then sails back to deliver the load. After the sand is discharged, the ship sails again to the dredging location and the whole cycle repeats.

The efficiency of the sedimentation process heavily depends on the type of soil and is influenced by the flow-rate and density of the incoming mixture and the manner the overflow pipe is controlled. An important factor in the optimization of

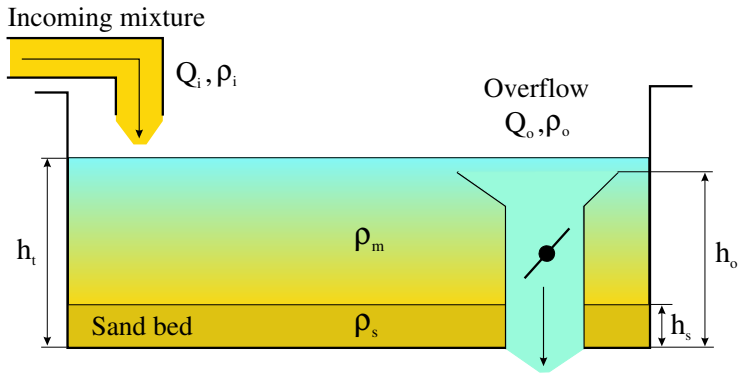


Fig. 11. The sedimentation process in the hopper.

the dredging performance is the minimization of the overflow losses.

Using nonlinear volume and mass balance equations, the losses can be estimated. The discrete volume and mass balance equations can be written as :

$$V_{\mathrm{t},k} = V_{\mathrm{t},k-1} + T(Q_{\mathrm{i},k-1} - Q_{\mathrm{o},k-1}) \tag{22}$$

$$m_{\mathrm{t},k} = m_{\mathrm{t},k-1} + T(Q_{\mathrm{i},k-1}\rho_{\mathrm{i},k-1} - Q_{\mathrm{o},k-1}\rho_{\mathrm{o},k-1}) \tag{23}$$

where V_t and m_t are the total volume and mass of the mixture in the tank, Q_i is the inflow rate, ρ_i is the density of the incoming mixture, Q_o and ρ_o are the flow-rate and the density of overflowing mixture and $T = 5\,\mathrm{s}$ is the sampling period. To estimate Q_o and ρ_o, the state equations are augmented with a random-walk model for Q_o and ρ_o:

$$Q_{\mathrm{o},k} = Q_{\mathrm{o},k-1} + \epsilon_{Q,k-1} \tag{24}$$

$$\rho_{\mathrm{o},k} = \rho_{\mathrm{o},k-1} + \epsilon_{\rho,k-1} \tag{25}$$

The augmented state, input and output vectors are:

$$x = \begin{pmatrix} V_\mathrm{t} \\ m_\mathrm{t} \\ Q_\mathrm{o} \\ \rho_\mathrm{o} \end{pmatrix}, \qquad u = \begin{pmatrix} Q_\mathrm{i} \\ \rho_\mathrm{i} \end{pmatrix}, \qquad y = \begin{pmatrix} V_\mathrm{t} \\ m_\mathrm{t} \end{pmatrix}$$

The inputs Q_i and ρ_i and the outputs V_t and m_t are measured, Q_o and ρ_o need to be estimated. The corrupting noises are considered zero-mean Gaussians ($\epsilon_{xi,k} \sim \mathcal{N}(0, \nu_{xi})$), and their standard deviations are determined experimentally.

The system (22)–(25) is nonlinear, therefore a centralized Kalman filter cannot be used, and for an efficient estimation of Q_o and ρ_o, a particle filter (Doucet et al., 2000; Chen et al., 2005) is needed. However, the model can be decomposed into two subsystems, and observers may be designed separately for the subsystems. The

first observer uses the model

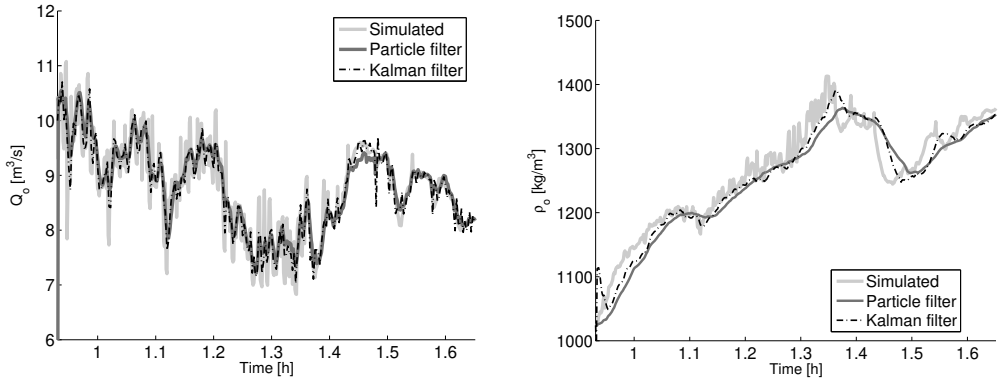
$$\begin{aligned} V_{\mathrm{t},k} &= V_{\mathrm{t},k-1} + T(Q_{\mathrm{i},k-1} - Q_{\mathrm{o},k-1}) + \epsilon_{V,k-1} \\ Q_{\mathrm{o},k} &= Q_{\mathrm{o},k-1} + \epsilon_{Q,k-1} \end{aligned} \tag{26}$$

where V_{t} is the measured output. The second observer uses the model

$$\begin{aligned} m_{\mathrm{t},k} &= m_{\mathrm{t},k-1} + T(Q_{\mathrm{i},k-1}\rho_{\mathrm{i},k-1} - Q_{\mathrm{o},k-1}\rho_{\mathrm{o},k-1}) + \epsilon_{m,k-1} \\ \rho_{\mathrm{o},k} &= \rho_{\mathrm{o},k-1} + \epsilon_{\rho,k-1} \end{aligned} \tag{27}$$

where m_{t} is the measured output and Q_{o} was obtained by the first observer. The individual subsystems are linear time-varying and assumed to be corrupted by Gaussian noise, and therefore, cascaded Kalman filters can be used.

To test the observer, the data were generated using the first-principle sedimentation model described by Babuška et al. (2006), i.e., not the model used by the cascaded Kalman filter. The results are presented in Figure 12. As can be seen, the cascaded Kalman filter obtains a good estimate of both Q_{o} and ρ_{o}, even better than the results obtained by the particle filter, which requires extensive tuning and a considerable computational power. However, when the data is measured, and the noise is no longer Gaussian, a particle filter is beneficial.



(a) Q_{o} (b) ρ_{o}

Fig. 12. Estimates of the overflow losses with different observers (particle filter, distributed Kalman).

8 Conclusions

We have proposed a distributed setting for state estimation. In many real-life applications, a complex process model can be decomposed into simpler subsystems, and observers can be designed for these individual subsystems. This partitioning of a process and observer leads to increased modularity and reduced complexity of the estimation problem, with reduced computational costs and easier tuning.

For cascaded systems, distributed, Kalman-like filters can be designed. The observers are optimal for the individual subsystems, and the error system will converge to a zero-mean Gaussian. However, the overall filter will not necessarily be optimal. The theoretical results show that the distributed Kalman filters can be jointly optimal, if and only if the subsystems are independent. However, based on several simulation examples, we conclude that the performance of the cascaded Kalman filter is comparable to that of the centralized Kalman filter. Moreover, the simulations show that for certain cases, especially in closed-loop, the cascaded observers perform better than the Kalman filter. Two application examples were presented to illustrate the distributed Kalman filters in a high dimensional and even nonlinear system. The results were comparable with those obtained by the centralized observer, while obtaining increased modularity.

In our future research, we will investigate the conditions under which such a decomposition of the process and the estimation problem is possible for other types of observers while maintaining the performance (convergence, rate of convergence) of the centralized one.

Acknowledgments: This research is partly funded by Senter, Ministry of Economic Affairs of the Netherlands within the BSIK-ICIS project "Interactive Collaborative Information Systems" (grant no. BSIK03024). The sedimentation model is studied within the project Artificial Intelligence for the Control of a Hopper Dredger (grant no. TSMA 2017). The water treatment process model is studied within the Promicit project, sponsored by Senter-Novem, Netherlands. We are grateful to Kim van Schagen for his contribution.

References

Aja-Fernandez, S., Alberola-Lopez, C., Ruiz-Alzola, J., 2003. A fuzzy-controlled Kalman filter applied to stereo-visual tracking schemes. Signal Processing 83, 101–120.

Babuška, R., Lendek, Z., Braaksma, J., de Keizer, C., 2006. Particle filtering for on-line estimation of overflow losses in a hopper dredger. In: American Control Conference. pp. 5751–5756.

Caron, F., Duflos, E., Pomorski, D., Vanheeghe, P., 2006. GPS/IMU data fusion using multisensor Kalman filtering: introduction of contextual aspects. Information Fusion 7, 221–230.

Chen, T., Morris, J., Martin, E., 2005. Particle filters for state and parameter estimation in batch processes. Journal of Process Control 15, 665–673.

Dorfmüller-Ulhaas, K., May 2003. Robust optical user motion tracking using a Kalman filter. Tech. Rep. 2003-6, Klaus Dorfmüller-Ulhaas, Augsburg University, mm-werkstatt.informatik.uni-augsburg.de/files/publications/38/squint2.pdf.

Doucet, A., Godsill, S., Andrieu, C., 2000. On sequential Monte Carlo sampling methods for Bayesian filtering. Statistics and Computing 10, 197–208.

Hue, C., Cadre, J. L., Perez, P., 2002. Tracking multiple objects with particle filtering. IEEE Transactions on Aerospace and Electronic Systems 38 (3), 791–812.

Kalman, R. E., 1960. A new approach to linear filtering and prediction problems. Transactions of the ASME–Journal of Basic Engineering 82, 35–45.

Lee, J. W., Kim, M. S., Kweon, I. S., 1995. A Kalman filter based visual tracking algorithm for an object moving in 3D. In: IEEE/RSJ International Conference on Intelligent Robots and Systems 95. Vol. 1. pp. 342–347.

McGinnity, S., Irwin, G., 1997. Nonlinear Kalman filtering using fuzzy local linear models. In: Proceedings of the 1997 American Control Conference. Denver, CO, pp. 3299–3300.

Simon, D., 2003. Kalman filtering for fuzzy discrete time dynamic systems. Applied Soft Computing 3 (3), 191–207.

van der Merwe, R., Wan, E., June 2003. Sigma-point Kalman filters for probabilistic inference in dynamic state-space models. In: Workshop on Advances in Machine Learning. Montreal.

Wang, H., Li, J., Niemann, D., Tanaka, K., 2000. T-S fuzzy model with linear rule consequence and PDC controller: a universal framework for nonlinear control systems. In: The Ninth IEEE International Conference on Fuzzy Systems, 2000. FUZZ IEEE 2000. Vol. 2. pp. 549–554.

Welch, G., Bishop, G., 2002. An introduction to the Kalman filter. Tech. Rep. TR 95-041, Department of Computer Science, University of North Carolina, NC, USA.

Zhang, S.-T., Wei, X.-Y., 2003. Fuzzy adaptive Kalman filtering for DR/GPS. In: International Conference on Machine Learning and Cybernetics. Vol. 5. pp. 2634–2637.