Article

Optimally Distributed Kalman Filtering with Data-Driven Communication [†]

Katharina Dormann [1,‡], **Benjamin Noack** [2,‡,*] [iD] **and Uwe D. Hanebeck** [2] [iD]

[1]   Robert Bosch GmbH, 71636 Ludwigsburg, Germany; katharina.dormann@gmx.de
[2]   Intelligent Sensor-Actuator-Systems Laboratory (ISAS), Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology (KIT), 76131 Karlsruhe, Germany; benjamin.noack@ieee.org or uwe.hanebeck@ieee.org
[*]   Correspondence: noack@kit.edu or benjamin.noack@ieee.org; Tel.: +49-721-6084-4024
[†]   This paper is an extended version of our paper published in Dormann, K.; Noack, B.; Hanebeck, U.D. Distributed Kalman Filtering With Reduced Transmission Rate, In Proceedings of the 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2017), Daegu, South Korea, 16–18 November 2017.
[‡]   These authors contributed equally to this work.

Abstract: For multisensor data fusion, distributed state estimation techniques that enable a local processing of sensor data are the means of choice in order to minimize storage and communication costs. In particular, a distributed implementation of the optimal Kalman filter has recently been developed. A significant disadvantage of this algorithm is that the fusion center needs access to each node so as to compute a consistent state estimate, which requires full communication each time an estimate is requested. In this article, different extensions of the optimally distributed Kalman filter are proposed that employ data-driven transmission schemes in order to reduce communication expenses. As a first relaxation of the full-rate communication scheme, it can be shown that each node only has to transmit every second time step without endangering consistency of the fusion result. Also, two data-driven algorithms are introduced that even allow for lower transmission rates, and bounds are derived to guarantee consistent fusion results. Simulations demonstrate that the data-driven distributed filtering schemes can outperform a centralized Kalman filter that requires each measurement to be sent to the center node.

Keywords: distributed Kalman Filtering; data-driven communication; distributed data fusion; sensor networks

1. Introduction

The efficient processing of sensor data is a central topic in a wide variety of research areas, which is underlined by advances in sensor technology and capabilities, e.g., for odor [1] and taste recognition [2] or by advances in visual information processing [3] as well as by applications in robotics [4] and sensor networks [5,6]. In particular to process data of multiple sensors, the well-known Kalman filter [7] has evolved into a key component of data fusion algorithms. Multisensor data can directly be transmitted to a data sink that employs a *centralized Kalman filter* to process all the accrued sensor readings. Such a simple filter design stands in stark contrast to the communication costs expended to transmit the data. The idea behind *distributed Kalman filter* implementations is to use local processing power to combine and condense sensor data locally so that the processing results can be transmitted more efficiently and less frequently. As the local processing results comprise the information from all past observations, the results can be sent to the data sink after arbitrarily many time steps without losing information from the past measurements. As compared to a centralized Kalman filter, distributed implementations

not only offer the advantage that the communication load in the network can be reduced significantly but also increase robustness to packet losses and drops. The information in a lost package will automatically be handed in later through a subsequent data transmission.

Most distributed estimation architectures employ local Kalman filters at the sensor nodes. Although the local processing can then be deemed to be optimal in terms of the mean squared estimation error, fusion of the local estimates becomes a challenging task due to correlations between the estimates to be fused [8–10]. In particular, a naïve or too simplistic fusion method may render the fusion results inconsistent, i.e., the fusion method reports an estimation error that is too small to account for the actual error. In general, the correlations required to fuse the local estimates optimally are difficult to reconstruct [11,12]. In the case of unknown correlations, conservative fusion algorithms, like *Covariance Intersection* [13,14] or *Ellipsoidal Intersection* [15–17] and its further development *Inverse Covariance Intersection* [18,19], can be employed to obtain consistent fusion results. However, these algorithms often provide too conservative assessments of the actual estimation error.

An alternative approach to distributed state estimation has been pursued in [20–23]. Here, a distributed implementation of a single centralized Kalman filter has been derived. In doing so, the need to keep track of correlations or to derive conservative bounds can be circumvented. The main advantage of this approach is its optimality with respect to the mean squared estimation error; even optimal fusion algorithms, like the Bar-Shalom/Campo fusion rule, cannot achieve the same estimation quality [11]. Therefore, this approach constitutes an *optimally distributed Kalman filter* (ODKF). An important aspect of this filter is that the local estimates are neither consistent nor unbiased, nor does the error covariance matrix describe the actual estimation error. The local estimates have to be fused in the data sink to obtain a consistent estimate, which is optimal and equal to the result of a centralized Kalman filter. Beside this article, several extensions of this filter have been proposed—assumptions on the available information can be relaxed [24,25], it can be implemented in information form [26], or combinations with ellipsoidal state estimation are possible [27,28].

Although communication can take place at arbitrary time steps, a critical drawback of the ODKF is that the data sink requires access to all local estimates in order to compute an estimate, i.e., all nodes have to transmit their estimates to the center node. This limitation has the following two severe implications:

1. *All* nodes have to send their data at the *same* time, and
2. the central cannot infer any information about the state between the sending times.

Hence, the standard implementation of the ODKF implies transmissions of either all or none of the nodes. As a further consequence, full-rate communication of the nodes is required if the data sink needs an estimate at every time step. In this article, extensions of the ODKF are proposed that can operate under lower communication rates. This is achieved by introducing data-driven transmission strategies [29,30]. In particular, the local estimates can asynchronously be transmitted to the data sink. In order to guarantee consistency, bounds on the estimation errors are provided. These bounds are only required in situations when not every local estimate is available at the data sink; the optimal estimate as provided by a central Kalman filter is still obtained each time when all local estimate have been sent to the data sink.

As compared to the standard formulation of the ODKF, the advantage of the proposed extension is that the data sink can now compute an estimate of the state based on a subset of local estimates. This article continues the work in [31] by introducing an additional criterion for the data-driven transmission strategy, providing more details, and extending the evaluation and discussions.

The paper is structured as follows. Section 2 provides a description of the centralized and the optimally distributed Kalman filter as well as a problem formulation. In Section 3, the first extension is introduced which enables the data sink to treat missing estimates. In Section 4 and in Section 5, we describe the second and the third new distributed algorithm, respectively, which implement data-driven transmission schemes and allow for omitted estimates at the fusion center over multiple

time steps. The results of an experimental evaluation are discussed in Section 6. Finally, the article is concluded in Section 7.

2. Centralized and Optimally Distributed Kalman Filtering

We consider a sensor network with N local sensor nodes and a central node, which serves as a data sink and computes an estimate on the state. The true state of the system at time step k is denoted by $\underline{x}_k$, which evolves according to the discrete-time linear dynamic system

$$\underline{x}_{k+1} = A_k \underline{x}_k + \underline{w}_k \,,$$

where A_k is the system matrix and $\underline{w}_k$ denotes the process noise, which is assumed to be zero-mean Gaussian noise, $\underline{w}_k \sim \mathcal{N}(\underline{0}, C_k^w)$ with covariance matrix C_k^w. At each time step k, each sensor $i \in \{1, \ldots, N\}$ observes the state through the model

$$\underline{z}_k^i = H_k^i \underline{x}_k + \underline{v}_k^i \,,$$

where H_k^i is the measurement matrix and $\underline{v}_k^i$ the measurement noise, which is assumed to be Gaussian noise with zero mean, $\underline{v}_k \sim \mathcal{N}(\underline{0}, C_k^{z,i})$ with covariance matrix $C_k^{z,i}$. The measurement noise terms of different local sensors are assumed to be mutually uncorrelated. Also, the process and measurement noise terms for different time steps are uncorrelated.

For the *centralized Kalman filter* (CKF), each measurement is sent to the data sink, and fused by means of the formulas

$$
\begin{aligned}
\left(C_k^{e,c}\right)^{-1} \hat{\underline{x}}_k^{e,c} &= \left(C_k^{p,c}\right)^{-1} \hat{\underline{x}}_k^{p,c} + \sum_{i=1}^{N} \left(H_k^i\right)^T (C_k^{z,i})^{-1} \underline{z}_k^i \,, \\
\left(C_k^{e,c}\right)^{-1} &= \left(C_k^{p,c}\right)^{-1} + \sum_{i=1}^{N} \left(H_k^i\right)^T \left(C_k^{z,i}\right)^{-1} H_k^i \,.
\end{aligned}
\tag{1}
$$

These equations correspond to the information form [32,33] of the measurement update of the standard Kalman filter. $\hat{\underline{x}}_k^{e,c}$ and $C_k^{e,c}$ denote the state estimate and the corresponding error covariance matrix after the fusion step, respectively. $\hat{\underline{x}}_k^{p,c}$ and $C_k^{p,c}$ denote the state estimate and the corresponding error covariance matrix computed by the prediction step of the Kalman filter. The prediction step is carried out at the center node by

$$
\begin{aligned}
\left(C_{k+1}^{p,c}\right)^{-1} \hat{\underline{x}}_{k+1}^{p,c} &= \left(C_{k+1}^{p,c}\right)^{-1} A_k \hat{\underline{x}}_k^{e,c} \,, \\
\left(C_{k+1}^{p,c}\right)^{-1} &= \left(A_k C_k^{e,c} A_k^T + C_k^w\right)^{-1} \,,
\end{aligned}
\tag{2}
$$

where these formulas are also given in the information form [26,32]. Since these equations correspond to the standard Kalman filter, the CKF is unbiased and optimal with respect to the minimum mean squared error. In particular, the computed error covariance matrix is equal to the actual estimation error, i.e.,

$$C_k^{e,c} = E\left(\left(\hat{\underline{x}}_k^{e,c} - \underline{x}_k\right)\left(\hat{\underline{x}}_k^{e,c} - \underline{x}_k\right)^T\right) \,. \tag{3}$$

In [20–23], a distributed implementation of the Kalman filter algorithm has been derived, which is algebraically equivalent to the centralized scheme, i.e., which is also unbiased, minimizes the mean squared error, and which fulfills (3). This is achieved by defining a local filtering scheme such that the fusion result is equal to results (1) and (2) of the CKF. We will describe this algorithm—the *optimally distributed Kalman filter* (ODKF)—in the following.

The local sensor nodes run modified versions of the Kalman filtering algorithm. They use so called *globalized* local states estimates and error covariance matrices (Although, strictly speaking,

the local estimate and covariance matrix do not represent consistent estimates of the state, we denote them as local estimates). To initialize the ODKF, the local initial estimates and covariance matrices $(\underline{\hat{x}}_0^{e,i}, C_0^{e,i})$ at the sensor nodes $i \in \{1, \ldots, N\}$, which are usually identical, have to be replaced by the globalized estimates

$$\underline{\bar{x}}_0^{e,i} = C_0^e \sum_{i=1}^{N} \left(C_0^{e,i} \right)^{-1} \underline{\hat{x}}_0^{e,i} ,$$

$$\bar{C}_0^e = N \left(\sum_{i=1}^{N} \left(C_0^{e,i} \right)^{-1} \right)^{-1} .$$

Since the globalized error covariance matrix is equal for each sensor, the sensor index i is omitted. This equality also applies to all future time steps. The local prediction step is replaced by the globalized prediction equations

$$\underline{\bar{x}}_{k+1}^{p,i} = A_k \underline{\bar{x}}_k^{e,i} ,$$
$$\bar{C}_{k+1}^p = A_k \bar{C}_k^e A_k^T + N C_k^w . \tag{4}$$

The local filtering steps are globalized by

$$\underline{\bar{x}}_k^{e,i} = \bar{C}_k^e \left(\left(\bar{C}_k^p \right)^{-1} \underline{\bar{x}}_k^{p,i} + \left(H_k^i \right)^T \left(C_k^{z,i} \right)^{-1} \underline{z}_k^i \right) ,$$

$$\bar{C}_k^e = \left(\left(\bar{C}_k^p \right)^{-1} + \frac{1}{N} \sum_{i=1}^{N} \left(H_k^i \right)^T \left(C_k^{z,i} \right)^{-1} H_k^i \right)^{-1} . \tag{5}$$

The processing steps (4) and (5) are computed locally on each sensor node. Hence, measurements are not directly transmitted to the central node—instead, the local estimates are sent to the central node. As the globalized covariance matrix is equal for each node, it can also be computed in the central node.

In order to compute an estimate at an arbitrary time step k, the central node receives the globalized estimates $(\underline{\bar{x}}_k^{e,i}, \bar{C}_k^e)$ from each sensor i and fuses them according to

$$\left(C_k^{e,d} \right)^{-1} = \sum_{i=1}^{N} \left(\bar{C}_k^e \right)^{-1} , \tag{6}$$

$$\left(C_k^{e,d} \right)^{-1} \underline{\hat{x}}_k^{e,d} = \sum_{i=1}^{N} \left(\bar{C}_k^e \right)^{-1} \underline{\bar{x}}_k^{e,i} . \tag{7}$$

$\underline{\hat{x}}_k^{e,d}$ denotes the state estimate after the fusion step at the data sink with corresponding error covariance matrix $C_k^{e,d}$. From Equations (6) and (7), we can easily accept that

$$C_k^{e,d} = \frac{1}{N} \bar{C}_k^e ,$$

$$\underline{\hat{x}}_k^{e,d} = \frac{1}{N} \sum_{i=1}^{N} \underline{\bar{x}}_k^{e,i} .$$

The same equations apply to the fusion of the predicted estimates and error covariance matrices in (4). In [20], it has been shown that the results are optimal, i.e.,

$$\underline{\hat{x}}_k^{e,d} = \underline{\hat{x}}_k^{e,c} , \tag{8}$$

$$C_k^{e,d} = C_k^{e,c} , \tag{9}$$

where $\hat{\underline{x}}_k^{e,c}$ and $C_k^{e,c}$ are computed by the CKF, i.e., (1) and (2). The disadvantage of the centralized Kalman filter is that each sensor node has to transmit measurements of each time step to the central node. For the ODKF, we observe that communication in past time steps does not influence $\hat{\underline{x}}_k^{e,d}$ and $C_k^{e,d}$, i.e., the equalities hold independently of the past communication pattern in the distributed network. As a consequence of (8) and (9), we can see that

$$C_k^{e,d} = E\left(\left(\hat{\underline{x}}_k^{e,d} - \underline{x}_k\right)\left(\hat{\underline{x}}_k^{e,d} - \underline{x}_k\right)^T\right) \tag{10}$$

is equal to (3)—the ODKF is optimal.

A significant drawback of the ODKF implementation becomes apparent in the following situation. If only $m < N$ sensors transmit their estimates to the fusion center at time step k, Equations (6) and (7) become

$$\left(C_k^{e,d}\right)^{-1} = \sum_{i=1}^{m} \left(\bar{C}_k^e\right)^{-1} , \tag{11}$$

$$\left(C_k^{e,d}\right)^{-1}\hat{\underline{x}}_k^{e,d} = \sum_{i=1}^{m} \left(\bar{C}_k^e\right)^{-1} \bar{\underline{x}}_k^{e,i}. $$

The resulting ODKF estimate $\hat{\underline{x}}_k^{e,d}$ and error covariance matrix $C_k^{e,d}$ are different from the centralized estimate $\hat{\underline{x}}_k^{e,c}$ and the error covariance matrix $C_k^{e,c}$, which are

$$\left(C_k^{e,c}\right)^{-1}\hat{\underline{x}}_k^{e,c} = \left(C_k^{p,c}\right)^{-1}\hat{\underline{x}}_k^{p,c} + \sum_{i=1}^{m}\left(H_k^i\right)^T\left(C_k^{z,i}\right)^{-1}\underline{z}_k^i , \tag{12}$$

$$\left(C_k^{e,c}\right)^{-1} = \left(C_k^{p,c}\right)^{-1} + \sum_{i=1}^{m}\left(H_k^i\right)^T\left(C_k^{z,i}\right)^{-1}H_k^i . \tag{13}$$

In particular, we notice that the covariance matrix (13) differs from the ODKF covariance matrix in (11). A consequence of this mismatch is a possible bias in the fused estimate as discussed, e.g., in [24]—hence, the ODKF may provide inconsistent estimates in case of missing transmissions. This issue will be addressed in the following sections. Although the CKF does not suffer from inconsistency, (12) and (13) unveil a critical downside of the CKF: Missing measurements at time step k are lost for all future time steps. By contrast, the local estimates of the ODKF incorporate past measurements, which states the reason why the ODKF may outperform the CKF if we can solve the inconsistency problem of the ODKF.

In this section, the ODKF has been revisited; it provides the same results as the CKF but offers the advantage that transmissions can take place at arbitrary instants of time. However, the ODKF still requires that all nodes send their local estimates at the transmission times to compute (6) and (7). As a consequence, the data sink typically operates at a lower rate than the local nodes as it is idle between transmission times. In the following sections, extensions are provided that enable the nodes to transmit their local estimates asynchronously. The data sink can then operate at the same rate as the sensor nodes, i.e., it is able to provide an estimate at every time step k. By employing data-driven strategies, the communication rate of each node can be significantly lower than 1, where the value 1 corresponds to transmissions at every time step k.

In Section 3, we develop a consistent ODKF extension than can cope with situations where sensor nodes may send their estimates at every second time step. This algorithm still provides results equal to the CKF. With this algorithm, we are able to reduce the communication rate by half. Sections 4 and 5 introduce a second and third algorithm that can even reach a lower communication rate by applying bounds on the missing pieces of information.

3. Distributed Kalman Filtering with Omitted Estimates

We consider the ODKF algorithm as described in the previous section. At time step k, only sensor nodes $1, \dots, m$ send their estimates to the fusion center, but the estimates of sensor nodes $m + 1, \dots, N$ are not. In this section, we assume that the data from the nodes $m + 1, \dots, N$ had been available in the fusion center at time step $k - 1$. Thus, the fusion center can compute the predicted estimates $\underline{\bar{x}}_k^{p,m+1}, \dots, \underline{\bar{x}}_k^{p,N}$ for time step k by using (4). In place of the ODKF fusion Equations (6) and (7), the fusion result is now computed by

$$\left(C_k^{e,d'} \right)^{-1} = \sum_{i=1}^{N} (\bar{C}_k^e)^{-1} - \sum_{i=m+1}^{N} (H_k^i)^T (C_k^{z,i})^{-1} H_k^i \, , \tag{14}$$

$$\left(C_k^{e,d'} \right)^{-1} \underline{\hat{x}}_k^{e,d'} = \sum_{i=1}^{m} (\bar{C}_k^e)^{-1} \, \underline{\bar{x}}_k^{e,i} + \sum_{i=m+1}^{N} (\bar{C}_k^p)^{-1} \underline{\bar{x}}_k^{p,i} \tag{15}$$

The resulting estimate $\underline{\hat{x}}_k^{e,d}$ and the error covariance matrix $C_k^{e,d}$ are equal to the estimate $\underline{\hat{x}}_k^{e,c}$ and the error covariance matrix $C_k^{e,c}$ computed by a centralized Kalman filter according to (12) and (13). A proof for the equality is provided in Appendix A.

Since (14) and (15) are equivalent to the CKF, unbiasedness, optimality, and (10) are accordingly inherited from the CKF. We have generalized the original ODKF algorithm such that full-rate communication is not required anymore. The novel fusion algorithm merely requires that if a particular sensor does not communicate with the center at time k, it has sent its data at time $k - 1$, i.e., each sensor has to communicate with the center at least every second time step. Hence, the required communication rate can be reduced by half to 0.5.

However, a higher communication rate—and thus, the incorporation of the information contained in additional measurements—will always result in a lower mean squared error (MSE). Thus, we have to deal with the trade-off between a low communication rate and a low MSE. Nevertheless, it is possible to achieve a smaller MSE while keeping the same communication rate by using a data-driven communication strategy and thus, scheduling the data according to the information contained. Valuable results have already been achieved using data-driven communication in distributed sensor networks [34–42]. The idea is that each local sensor evaluates the distance between the predicted estimate $\underline{\bar{x}}_k^{p,i}$ and the filtered estimate $\underline{\bar{x}}_k^{e,i}$. If the distance is large, the measurement $\underline{z}_k^i$ adds much new information to the prediction. Only in this case, the sensor should send its current estimate $\underline{\bar{x}}_k^{e,i}$ to the center node.

It is important to emphasize that the globalized parameters $\underline{\bar{x}}_k^{p,i}$ and $\underline{\bar{x}}_k^{e,i}$ are not unbiased estimates of the actual state. It can be shown [24] that, in contrast to the difference between the standard Kalman filter estimates, $\underline{\hat{x}}_k^{p,i} - \underline{\hat{x}}_k^{e,i}$, the difference $\underline{\bar{x}}_k^{p,i} - \underline{\bar{x}}_k^{e,i}$ is not zero on average, but may even diverge. Therefore, in order to evaluate the influence of a measurement $\underline{z}_k^i$, we study the difference between the predicted and updated estimates of the standard Kalman filter, which is related to the weighted difference between the measurement and the prediction, i.e.,

$$\underline{\hat{x}}_k^{p,i} - \underline{\hat{x}}_k^{e,i} = K_k^i \left(\underline{z}_k^i - H_k^i \underline{\hat{x}}_k^{p,i} \right) \, ,$$

where K_k^i denotes the standard Kalman gain. For this purpose, the standard Kalman filtering algorithm has to run in parallel to the globalized version of the Kalman filter at each sensor node. The following data-driven communication strategy can be applied:

$$\text{If } \| \underline{\hat{x}}_k^{p,i} - \underline{\hat{x}}_k^{e,i} \| \leq \alpha \tag{16}$$

do not send estimate to the fusion center

else

send estimate to the fusion center,

where α denotes a scalar parameter. We can achieve any communication rate in range $[0.5, 1]$ by varying the parameter α.

By using this communication strategy we can only evaluate the relevance of the measurement to the local estimate $\hat{\underline{x}}_k^{p,i}$, and not to the fused estimate $\hat{\underline{x}}_k^{p,d}$. Still, experiments (see Section 6) will show that by using the data-driven communication strategy instead of random communication, for a fixed communication rate an improvement of the MSE of the fused estimate $\hat{\underline{x}}_k^{e,d'}$ can be achieved. However, a drawback of the algorithm is the assumption that if a particular sensor does not communicate with the center at time k, it has communicated at time $k-1$, i.e., each sensor communicates with the center at least every other time step. Thus, communication rates lower than 0.5 cannot be achieved. This will be addressed by the following extensions.

4. Data-Driven Distributed Kalman Filtering with Omitted Estimates over Multiple Time Steps–Version 1

If we want to achieve communication rates lower than 0.5 in the sensor network, we have to allow that a particular sensor does not send its estimates to the fusion center over multiple time steps. In this case, the fusion center has to perform multiple consecutive predictions. Let us assume that the last communication of sensor i with the center occurred at time $k - l$. The predicted estimate for time step k is computed as shown in the following scheme.

$$\underline{x}_{k-l}^{e,i} \xrightarrow{\textit{Prediction}} \underline{x}_{k-l+1}^{pp,i} \xrightarrow{\textit{Prediction}} \ldots \xrightarrow{\textit{Prediction}} \underline{x}_{k}^{pp,i}$$

Prediction refers to the application of Equation (4). Note that the predicted estimates are now marked with "*pp*" instead of "*p*" to emphasize that possibly multiple prediction steps were applied consecutively. In case that prediction has been performed only once, we have $\underline{x}_k^{pp,i} = \underline{x}_k^{p,i}$.

In fusion Equation (15). the predictions $\underline{x}_k^{pp,i}$ are now used for the missing local estimates of the nodes $i = m+1, \ldots, N$, i.e.,

$$\left(C_k^{e,d'} \right)^{-1} \hat{\underline{x}}_k^{e,d''} = \sum_{i=1}^{m} (\bar{C}_k^e)^{-1} \underline{x}_k^{e,i} + \sum_{i=m+1}^{N} (\bar{C}_k^p)^{-1} \underline{x}_k^{pp,i}. \tag{17}$$

The new estimate $\hat{\underline{x}}_k^{e,d''}$ can be expressed in terms of the estimate $\hat{\underline{x}}_k^{e,d'}$ from (15) as follows:

$$
\begin{aligned}
\hat{\underline{x}}_k^{e,d''} &= C_k^{e,d'} \left(\sum_{i=1}^{m} (\bar{C}_k^e)^{-1} \underline{x}_k^{e,i} + \sum_{i=m+1}^{N} (\bar{C}_k^p)^{-1} \underline{x}_k^{pp,i} \right) \\
&= C_k^{e,d'} \left(\left(C_k^{e,d'} \right)^{-1} \hat{\underline{x}}_k^{e,d'} - \sum_{i=m+1}^{N} \left(\bar{C}_k^p \right)^{-1} \underline{x}_k^{p,i} \right. \\
&\quad \left. + \sum_{i=m+1}^{N} (\bar{C}_k^p)^{-1} \underline{x}_k^{pp,i} \right) \\
&= \hat{\underline{x}}_k^{e,d'} - C_k^{e,d'} \left(\bar{C}_k^p \right)^{-1} \sum_{i=m+1}^{N} \left(\underline{x}_k^{pp,i} - \underline{x}_k^{p,i} \right).
\end{aligned}
$$

For the yet to be defined triggering criterion, we consider the distance

$$d_k^i := \underline{x}_k^{pp,i} - \underline{x}_k^{p,i}. \tag{18}$$

The expected estimation error covariance matrix is then given by

$$E\left(\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d''}\right)\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d''}\right)^T\right)$$

$$= E\left(\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d'} - \boldsymbol{C}_k^{e,d'}\left(\bar{\boldsymbol{C}}_k^p\right)^{-1}\sum_{i=m+1}^{N} d_k^i\right)\right.$$

$$\left.\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d'} - \boldsymbol{C}_k^{e,d'}\left(\bar{\boldsymbol{C}}_k^p\right)^{-1}\sum_{i=m+1}^{N} d_k^i\right)^T\right).$$

Obviously, the expected estimation error cannot be computed exactly at the fusion center, since the difference d_k^i is not available. Nevertheless, it is possible to obtain an upper bound on the estimation error, if we alter the communication test (16). We ensure that in case of communication the matrix $d_k^i\left(d_k^i\right)^T$ is bounded by $\boldsymbol{B}$. The communication strategy becomes

$$\text{If } \|\hat{\underline{x}}_k^{pp,i} - \hat{\underline{x}}_k^{e,i}\| \leq \alpha \text{ and } d_k^i\left(d_k^i\right)^T \leq \boldsymbol{B} \tag{19}$$

$$\text{do not send estimate to the fusion center}$$

$$\text{else}$$

$$\text{send estimate to the fusion center,}$$

where $\hat{\underline{x}}_k^{pp,i}$ denotes the estimate which has been computed by applying the Kalman filtering prediction step multiple times and $\boldsymbol{B}$ denotes a user-defined symmetric positive definite matrix. For any square matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$, $\boldsymbol{X} \leq \boldsymbol{Y}$ denotes that $\boldsymbol{Y} - \boldsymbol{X}$ is positive semi-definite.

For the bound, the relationship

$$d_k^i\left(d_k^i\right)^T \leq \boldsymbol{B} \iff \left(d_k^i\right)^T \boldsymbol{B}^{-1} d_k^i \leq 1 \tag{20}$$

holds. Thus, the communication criterion $d_k^i\left(d_k^i\right)^T \leq \boldsymbol{B}$ can be replaced by $\left(d_k^i\right)^T \boldsymbol{B}^{-1} d_k^i \leq 1$, which reduces the computational cost. Assuming that d_k^i is an n-dimensional vector and $\boldsymbol{B}$ an $n \times n$-matrix, the cost for computing the product $d_k^i(d_k^i)^T$ is in $\mathcal{O}(n^2)$. The computation of the difference $\boldsymbol{B} - d_k^i\left(d_k^i\right)^T$ is also in $\mathcal{O}(n^2)$. The computational cost for computing the eigenvalues of the $n \times n$-matrix $\boldsymbol{B} - d_k^i\left(d_k^i\right)^T$ is in $\mathcal{O}(n^3)$. All in all, the computational cost for the decision if $d_k^i\left(d_k^i\right)^T \leq \boldsymbol{B}$ is in $\mathcal{O}(n^3 + n^2)$. Since the inverse matrix $\boldsymbol{B}^{-1}$ has to be computed only once beforehand, the cost for computing the product $\left(d_k^i\right)^T \boldsymbol{B}^{-1} d_k^i$ is in $\mathcal{O}(n^2)$. Therefore, testing if $\left(d_k^i\right)^T \boldsymbol{B}^{-1} d_k^i \leq 1$ only requires $\mathcal{O}(n^2)$ computations. Since $\|\hat{\underline{x}}_k^{pp,i} - \hat{\underline{x}}_k^{e,i}\| \leq \alpha$ only requires the computation of the Euclidean norm, which is in $\mathcal{O}(n)$, the computational cost of the communication test (19) is in $\mathcal{O}(n^2)$. The employed relationship (20) is proven in Appendix B.

As in the previous communication strategy, we have to take into consideration that the globalized estimates $\underline{\bar{x}}_k^{pp,i}$ and $\underline{\bar{x}}_k^{p,i}$ are *biased* and thus, the difference d_k^i may diverge. As a consequence, the matrix $d_k^i\left(d_k^i\right)^T$ may also diverge. In contrast to the previous fusion algorithm, using the standard Kalman filtering estimates $\hat{\underline{x}}_k^{pp,i}$, $\hat{\underline{x}}_k^{p,i}$ is not a reasonable solution, since the communication strategy is used to get an upper bound on d_k^i. An alternative possibility to avoid the divergence of the difference is to *debias* the local globalized estimates. A strategy to debias the estimates using *debiasing matrices* has been proposed in [24,25]. In each prediction and filtering step, each local node computes a new debiasing matrix. This matrix is initialized by $\Delta_0^{p,i} = \boldsymbol{I}$. In the filtering step, the debiasing matrix is computed by

$$\Delta_k^{e,i} = \bar{\boldsymbol{C}}_k^{e,i}(\bar{\boldsymbol{C}}_k^{p,i})^{-1}\Delta_k^{p,i} + \bar{\boldsymbol{C}}_k^{e,i}\left(\boldsymbol{H}_k^i\right)^T (\boldsymbol{C}_k^{z,i})^{-1}\boldsymbol{H}_k^i.$$

In the prediction step, the debiasing matrix is computed by

$$\Delta_k^{p,i} = A_k \Delta_k^{e,i} A_k^{-1} . \tag{21}$$

$\Delta_k^{pp,i}$ is computed by applying Equation (21) multiple times, until the next communication with the center node occurs. By multiplying the inverse of the debiasing matrix with the globalized estimates, we can debias the estimates [24,25], i.e.,

$$E\left((\Delta_k^{e,i})^{-1} \tilde{\underline{x}}_k^{e,i} \right) = E\left(\underline{x}_k \right),$$

$$E\left((\Delta_k^{p,i})^{-1} \tilde{\underline{x}}_k^{p,i} \right) = E\left(\underline{x}_k \right) .$$

It can be easily shown that the same applies to the predicted estimate over multiple time steps, i.e.,

$$E\left((\Delta_k^{pp,i})^{-1} \tilde{\underline{x}}_k^{pp,i} \right) = E\left(\underline{x}_{k+n} \right) .$$

We new define

$$\tilde{\underline{x}}_k^{pp,i} := \Delta_k^{p,i} (\Delta_k^{pp,i})^{-1} \tilde{\underline{x}}_k^{pp,i}$$

and then have

$$E\left(\tilde{\underline{x}}_k^{pp,i} - \tilde{\underline{x}}_k^{p,i} \right) = E\left(\Delta_k^{p,i} (\Delta_k^{pp,i})^{-1} \tilde{\underline{x}}_k^{pp,i} - \tilde{\underline{x}}_k^{p,i} \right)$$

$$= \Delta_k^{p,i} E\left((\Delta_k^{pp,i})^{-1} \tilde{\underline{x}}_k^{pp,i} - (\Delta_k^{p,i})^{-1} \tilde{\underline{x}}_k^{p,i} \right)$$

$$= \Delta_k^{p,i} E\left(\underline{x}_{k+n} - \underline{x}_k \right) .$$

Thus, in general the difference $\tilde{\underline{x}}_k^{pp,i} - \tilde{\underline{x}}_k^{p,i}$ does not diverge. With this result, we adapt (18), and use

$$\tilde{d}_k^i := \tilde{\underline{x}}_k^{pp,i} - \tilde{\underline{x}}_k^{p,i}$$

to replace the second inequality in (19) by

$$\tilde{d}_k^i \left(\tilde{d}_k^i \right)^T \leq B .$$

We can now define the new fusion equations as

$$C_k^{e,d''} = C_k^{e,d'}$$
$$+ (N - m - l)^2 C_k^{e,d'} \left(\bar{C}_k^p \right)^{-1} B \left(\bar{C}_k^p \right)^{-1} \left(C_k^{e,d'} \right)^T$$

$$\hat{\underline{x}}_k^{e,d''} = C_k^{e,d'} \left(\sum_{i=1}^m (\bar{C}_k^e)^{-1} \tilde{\underline{x}}_k^{e,i} + \sum_{i=m+1}^N (\bar{C}_k^p)^{-1} \tilde{\underline{x}}_k^{pp,i} \right),$$

where m is the number of sensors which communicate with the center at time k and l is the number of sensors which do not communicate with the center at time k, but for which $\tilde{\underline{x}}_k^{pp,i} - \tilde{\underline{x}}_k^{p,i} = 0$ hold. Note that the fusion formulas are equal to (14) and (15) for $N = m + l$, i.e., for the case that each sensor sends its estimate to the center at least every other time step. The resulting estimate is consistent, i.e.,

$$E\left(\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d''} \right) \left(\underline{x}_k - \hat{\underline{x}}_k^{e,d''} \right)^T \right) \leq C_k^{e,d''} . \tag{22}$$

A proof for the consistency condition (22) is provided in Appendix C.

The drawback of the presented algorithm is that it needs two parameters B and α to perform the communication test. Both parameters influence the communication rate. Thus, it is difficult to find the parameters which ensure the desired balance between a small communication rate and a small estimation error. Experiments using the particular dynamic system are needed to find the best combination of both parameters. Thus, we will now present another algorithm which only uses one parameter B for the communication strategy.

5. Data-Driven Distributed Kalman Filtering with Omitted Estimates over Multiple Time Steps–Version 2

For the second data-driven algorithm, fusion Equation (15) is now generalized by

$$\left(C_k^{e,d'}\right)^{-1}\hat{\underline{x}}_k^{e,d''} = \sum_{i=1}^{m}(\bar{C}_k^e)^{-1}\bar{\underline{x}}_k^{e,i} + \sum_{i=m+1}^{N}(\bar{C}_k^e)^{-1}\underline{\bar{x}}_k^{pp,i}\,.$$

The difference to the previous Equation (17) is the covariance matrix $(\bar{C}_k^e)^{-1}$ in the second sum. The new estimate $\hat{\underline{x}}_k^{e,d''}$ can be expressed in terms of the estimate $\hat{\underline{x}}_k^{e,d'}$ from (15) as follows:

$$\hat{\underline{x}}_k^{e,d''} = C_k^{e,d'}\left(\sum_{i=1}^{m}(\bar{C}_k^e)^{-1}\bar{\underline{x}}_k^{e,i} + \sum_{i=m+1}^{N}(\bar{C}_k^e)^{-1}\underline{\bar{x}}_k^{pp,i}\right)$$

$$= C_k^{e,d'}\left(\sum_{i=1}^{N}(\bar{C}_k^e)^{-1}\bar{\underline{x}}_k^{e,i} - \sum_{i=m+1}^{N}(\bar{C}_k^e)^{-1}\bar{\underline{x}}_k^{e,i}\right.$$

$$\left. + \sum_{i=m+1}^{N}(\bar{C}_k^e)^{-1}\underline{\bar{x}}_k^{pp,i}\right)$$

$$= C_k^{e,d'}\sum_{i=1}^{N}(\bar{C}_k^e)^{-1}\bar{\underline{x}}_k^{e,i} + C_k^{e,d'}(\bar{C}_k^e)^{-1}\sum_{i=m+1}^{N}\left(\underline{\bar{x}}_k^{pp,i} - \underline{\bar{x}}_k^{e,i}\right)$$

$$= \hat{\underline{x}}_k^{e,d'} + \frac{1}{N}\sum_{i=m+1}^{N}\left(\underline{\bar{x}}_k^{pp,i} - \underline{\bar{x}}_k^{e,i}\right)\,.$$

In order to define a communication strategy, we consider the difference

$$d_k^i = \underline{\bar{x}}_k^{pp,i} - \underline{\bar{x}}_k^{e,i},$$

which is compared against the matrix B by

$$\text{If } d_k^i\left(d_k^i\right)^T \le B$$

do not send estimate to the fusion center

else

send estimate to the fusion center.

B denotes a user-defined symmetric positive definite matrix. This time we do not need the Euclidean distance $\|\hat{\underline{x}}_k^{pp,i} - \hat{\underline{x}}_k^{e,i}\| \le \alpha$ since the distance between predicted and filtered estimate is included in d_k^i. We can now define the new fusion equations as

$$C_k^{e,d''} = C_k^{e,d'} + \frac{1}{N^2}(N-m)^2 B$$

$$\hat{\underline{x}}_k^{e,d''} = C_k^{e,d'}\left(\sum_{i=1}^{m}(\bar{C}_k^e)^{-1}\bar{\underline{x}}_k^{e,i} + \sum_{i=m+1}^{N}(\bar{C}_k^e)^{-1}\underline{\bar{x}}_k^{pp,i}\right),$$

where m is the number of sensors which communicate with the center at time k. With the same arguments as in Section 4 it can be shown that the resulting estimate is consistent, i.e.,

$$E\left(\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d''}\right)\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d''}\right)^T\right) \leq C_k^{e,d''}.$$

In the experimental evaluation of the algorithms we will see that although this version of fusing the estimates has the advantage that only one parameter has to be chosen, the estimate of the fused error covariance matrix is not as good as as in the previous version.

6. Simulations and Evaluation

We apply the CKF algorithm as well as the three data-driven ODKF algorithms to a single-target tracking problem. The system state $\underline{x}_k$ is a six-dimensional vector with two dimensions for the position, two dimensions for the velocity, and two dimensions for the acceleration. A near-constant acceleration model is used. The system matrix is given by

$$A_k = \begin{pmatrix} 1 & \Delta & \Delta^2/2 & 0 & 0 & 0 \\ 0 & 1 & \Delta & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta & \Delta^2/2 \\ 0 & 0 & 0 & 0 & 1 & \Delta \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

with the sampling interval $\Delta = 0.25$ s. The process noise covariance matrix is given by

$$C_k^w = \begin{pmatrix} \Delta^5/20 & \Delta^4/8 & \Delta^3/6 & 0 & 0 & 0 \\ \Delta^4/8 & \Delta^3/3 & \Delta^2/2 & 0 & 0 & 0 \\ \Delta^3/6 & \Delta^2/2 & \Delta & 0 & 0 & 0 \\ 0 & 0 & 0 & \Delta^5/20 & \Delta^4/8 & \Delta^3/6 \\ 0 & 0 & 0 & \Delta^4/8 & \Delta^3/3 & \Delta^2/2 \\ 0 & 0 & 0 & \Delta^3/6 & \Delta^2/2 & \Delta \end{pmatrix}.$$

We have a sensor network consisting of six sensor nodes and one fusion node. Two sensors measure the position, two sensors measure the velocity and two sensors measure the acceleration. The measurement noise covariance matrices are given by

$$C_k^{z,i} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \text{ for } i \in 1, \dots, 6.$$

Monte Carlo simulations with 500 independent runs over 100 time steps are performed. Since (A_k, C_k^w) is stable and (A_k, H_k) is detectable the error covariance matrix and the MSE converge to a unique values [43]. Based on the simulation, actual error covariance matrices and MSE values are computed for each algorithm. Monte Carlo simulations are performed for different average communication rates for each of the three algorithms. For the CKF, communication is performed randomly, but with different average rates. Note that only current measurements are communicated. If the measurement $\underline{z}_k^i$ is not sent to the fusion center at time k, the information will not be available at the center at any future time.

The first ODKF algorithm (Algorithm 1) from Section 3 is performed with random communication as well as with data-driven communication. In the latter case, the parameter α is varied to achieve different rates. The second algorithm (Algorithm 2) from Section 4 and the third algorithm (Algorithm 3) from Section 5 are performed with data-driven communication. For Algorithm 2 both parameters α and B are varied, for Algorithm 3 the parameter B is varied. The compared methods are:

Algorithm 1: ODKF algorithm from Section 3 with random and data-driven communication.
Algorithm 2: ODKF algorithm from Section 4 with data-driven communication and parameters α, B.
Algorithm 3: ODKF algorithm from Section 5 with data-driven communication and parameter B.

The simulation results are shown in Figure 1. The MSEs and the traces of the error covariance matrices are depicted relative to the average communication rate in the network. Since for Algorithm 2 different parameter combinations lead to different results, we have only included the results with the smallest error covariance matrices in the plot.

Only for the centralized Kalman filter and for the Algorithm 2 and 3, communication rates lower than 0.5 are given. We can observe that for Algorithm 1 data-driven communication leads to an improved estimate compared to random communication. However, it also leads to a larger trace of the error covariance matrix and thus, to a larger uncertainty reported with the estimate.

We also can observe that for communication rates in range $[0.5, 1]$ the results of Algorithms 2 and 3 with data-driven communication are almost equal. This can be explained by the fact that Algorithm 2 extends Algorithm 3 share a common triggering criterion, and the fusion formulas for both algorithms are equal if each sensor communicates with the center at least every other time step.
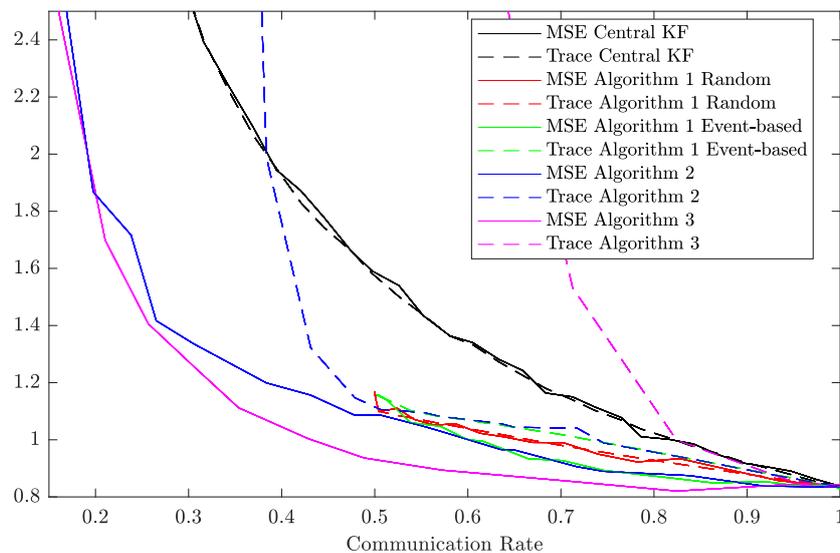

Figure 1. MSEs and traces of the error covariance matrices are plotted relative to the communication rate. Each communication rate corresponds to one Monte Carlo simulation with 500 runs over 100 time steps. MSEs are shown as solid lines, traces are shown as dashed lines.

Figure 1 shows that for each of the algorithms the MSE is always smaller than or equal to the trace of the error covariance matrix. This illustrates that the estimators provide consistent results. The traces are good estimates of the MSEs except for low communication rates in Algorithm 3 and very low communication rates in Algorithm 2. Thus, the trace of the error covariance matrices—the uncertainty reported by the estimators—is not significantly larger than the actual uncertainty in most of the cases. Each of the distributed fusion algorithms performs better in terms of the MSE as compared to the centralized algorithm. This can be explained by the fact that in the distributed network the fused estimates contain the information of all past measurements, while in the centralized network only the current measurements are fused.

7. Conclusions

In this article, the optimally distributed Kalman filter (ODKF) has been extended by data-driven communication strategies in order to bypass the need for full communication that is usually required by the ODKF to compute an estimate. Since the ODKF may provide inconsistent results if data transmissions are omitted, the missing estimates are replaced by predictions from previous time steps and consistent bounds on the error covariance matrix are computed. The first proposed technique allows for communication rates in the range $[0.5, 1]$ while the second and the third algorithm allow for any communication rate in range $[0, 1]$. In a centralized Kalman filter (CKF), where measurements are

directly sent to the central node, missing or lost transmissions to the center node need to be repeated in order to avoid loss of measurement data. In this regard, the proposed extensions of the ODKF can significantly outperform the CKF: The local estimates of each sensor node comprise the entire history of local measurements and hence, suspended transmissions do not lead to a loss of information in the network.

Author Contributions: Katharina Dormann designed and perfomed the simulations and analyses. Benjamin Noack and Uwe D. Hanebeck proposed the bounds for the data-driven communication strategy and supervised all the tasks related to the work. Katharina Dormann and Benjamin Noack prepared the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

We will now show that the resulting estimate $\hat{\underline{x}}_k^{e,d}$ and the error covariance matrix $C_k^{e,d}$ are equal to the estimate $\hat{\underline{x}}_k^{e,c}$ and the error covariance matrix $C_k^{e,c}$ computed by the CKF according to (12) and (13).
Proof: $C_k^{e,d'} = C_k^{e,c}$ holds due to

$$
\begin{aligned}
(C_k^{e,d'})^{-1} &= \sum_{i=1}^{N} (\bar{C}_k^e)^{-1} - \sum_{i=m+1}^{N} \left(H_k^i\right)^T (C_k^{z,i})^{-1} H_k^i \\
&= N\left((\bar{C}_k^e)^{-1} - \frac{1}{N} \sum_{i=m+1}^{N} \left(H_k^i\right)^T (C_k^{z,i})^{-1} H_k^i \right) \\
&= N\left(\left(\bar{C}_k^p\right)^{-1} + \frac{1}{N} \sum_{i=1}^{N} \left(H_k^i\right)^T (C_k^{z,i})^{-1} H_k^i \right. \\
&\quad \left. - \frac{1}{N} \sum_{i=m+1}^{N} \left(H_k^i\right)^T (C_k^{z,i})^{-1} H_k^i \right) \\
&= N\left(\left(\bar{C}_k^p\right)^{-1} + \frac{1}{N} \sum_{i=1}^{m} \left(H_k^i\right)^T (C_k^{z,i})^{-1} H_k^i \right) \\
&= N \left(\bar{C}_k^p\right)^{-1} + \sum_{i=1}^{m} \left(H_k^i\right)^T (C_k^{z,i})^{-1} H_k^i \\
&= \left(C_k^{p,c}\right)^{-1} + \sum_{i=1}^{m} \left(H_k^i\right)^T (C_k^{z,i})^{-1} H_k^i = (C_k^{e,c})^{-1} \ .
\end{aligned}
$$

Accordingly, $\hat{\underline{x}}_k^{e,d'} = \hat{\underline{x}}_k^{e,c}$ holds due to

$$
\begin{aligned}
(C_k^{e,d'})^{-1} \hat{\underline{x}}_k^{e,d'} &= \sum_{i=1}^{m} (\bar{C}_k^e)^{-1} \bar{\underline{x}}_k^{e,i} + \sum_{i=m+1}^{N} \left(\bar{C}_k^p\right)^{-1} \bar{\underline{x}}_k^{p,i} \\
&= \sum_{i=1}^{m} \left(\left(\bar{C}_k^p\right)^{-1} \bar{\underline{x}}_k^{p,i} + \left(H_k^i\right)^T (C_k^{z,i})^{-1} \underline{z}_k^i \right) + \sum_{i=m+1}^{N} \left(\bar{C}_k^p\right)^{-1} \bar{\underline{x}}_k^{p,i} \\
&= \sum_{i=1}^{m} \left(\bar{C}_k^p\right)^{-1} \bar{\underline{x}}_k^{p,i} + \sum_{i=m+1}^{N} \left(\bar{C}_k^p\right)^{-1} \bar{\underline{x}}_k^{p,i} + \sum_{i=1}^{m} \left(H_k^i\right)^T (C_k^{z,i})^{-1} \underline{z}_k^i \\
&= N \left(\bar{C}_k^p\right)^{-1} \frac{1}{N} \sum_{i=1}^{N} \bar{\underline{x}}_k^{p,i} + \sum_{i=1}^{m} \left(H_k^i\right)^T (C_k^{z,i})^{-1} \underline{z}_k^i \\
&= \left(C_k^{p,c}\right)^{-1} \hat{\underline{x}}_k^{p,c} + \sum_{i=1}^{m} \left(H_k^i\right)^T (C_k^{z,i})^{-1} \underline{z}_k^i
\end{aligned}
$$

$$= \left(C_k^{e,c}\right)^{-1} \hat{\underline{x}}_k^{e,c} = \left(C_k^{e,d'}\right)^{-1} \hat{\underline{x}}_k^{e,c} .$$

$\square$

Appendix B

In the following, relationship (20) is proven.

Proof. As a symmetric positive definite matrix, B can be written as a product $B = AA^T$, where A is a lower triangular matrix with positive diagonal entries, using the Cholesky decomposition. A^T is then an upper triangular matrix with positive diagonal entries. As triangular matrices with positive diagonal entries, the matrices A and A^T are invertible. We define $\underline{b} := A^{-1} d_k^i$. We have then

$$d_k^i(d_k^i)^T \leq B \iff B - d_k^i(d_k^i)^T \geq 0 \iff AA^T - d_k^i(d_k^i)^T \geq 0$$
$$\iff A\left(I - A^{-1}d_k^i(d_k^i)^T(A^T)^{-1}\right)A^T \geq 0 \iff I - A^{-1}d_k^i(d_k^i)^T(A^T)^{-1} \geq 0$$
$$\iff I - \underline{b}\underline{b}^T \geq 0 \iff \underline{b}\underline{b}^T \leq I .$$

We also have

$$(d_k^i)^T B^{-1} d_k^i \leq 1 \iff (d_k^i)^T (A^T)^{-1} A^{-1} d_k^i \leq 1 \iff \underline{b}^T \underline{b} \leq 1 \iff 1 - \underline{b}^T \underline{b} \geq 0 .$$

We still have to show that

$$I - \underline{b}\underline{b}^T \geq 0 \iff 1 - \underline{b}^T \underline{b} \geq 0 .$$

First, we will show "$\Rightarrow$".
From $I - \underline{b}\underline{b}^T \geq 0$ we have $\underline{a}^T \underline{a} - \underline{a}^T \underline{b}\underline{b}^T \underline{a} = \underline{a}^T\left(I - \underline{b}\underline{b}^T\right)\underline{a} \geq 0 \ \ \forall \underline{a} \neq \underline{0}$. With $\underline{a} = \underline{b}$ we have $\underline{b}^T \underline{b} - \underline{b}^T \underline{b}\underline{b}^T \underline{b} \geq 0$. Dividing by $\underline{b}^T \underline{b}$ gives $1 - \underline{b}^T \underline{b} \geq 0$.
We will now show "$\Leftarrow$".
From $\underline{b}^T \underline{b} \leq 1$ we have $\underline{a}^T \underline{b} \leq \|\underline{a}\|\|\underline{b}\| \leq \|\underline{a}\|$ and $\underline{b}^T \underline{a} \leq \|\underline{a}\|\|\underline{b}\| \leq \|\underline{a}\| \ \ \forall \underline{a}$. We have then $\underline{a}^T \underline{b}\underline{b}^T \underline{a} \leq \underline{a}^T \underline{a} \ \ \forall \underline{a}$. It follows $\underline{a}^T\left(I - \underline{b}\underline{b}^T\right)\underline{a} = \underline{a}^T \underline{a} - \underline{a}^T \underline{b}\underline{b}^T \underline{a} \geq 0 \ \ \forall \underline{a} \neq \underline{0}$ and thus, $I - \underline{b}\underline{b}^T \geq 0$.
$\square$

Appendix C

We will now show that the resulting estimate is consistent, i.e., relationship (22) holds.

Proof. We have

$$\hat{\underline{x}}_k^{e,d''} = \hat{\underline{x}}_k^{e,d'} - C_k^{e,d'}\left(\bar{C}_k^p\right)^{-1} \sum_{i=m+1}^{N} \tilde{d}_k^i .$$

The expected estimation error is then given by

$$E\left(\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d''}\right)\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d''}\right)^T\right) \tag{A1}$$
$$= E\left(\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d'}\right)\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d'}\right)^T\right)$$
$$- E\left(\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d'}\right)\left(C_k^{e,d'}(\bar{C}_k^p)^{-1} \sum_{i=m+1}^{N} \tilde{d}_k^i\right)^T\right)$$

$$- E\left(\left(C_k^{e,d'} \left(\bar{C}_k^p \right)^{-1} \sum_{i=m+1}^{N} \tilde{d}_k^i \right) \left(\underline{x}_k - \hat{\underline{x}}_k^{e,d'} \right)^T \right)$$

$$+ E\left(\left(C_k^{e,d'} \left(\bar{C}_k^p \right)^{-1} \sum_{i=m+1}^{N} \tilde{d}_k^i \right) \cdot \right.$$
$$\left. \left(C_k^{e,d'} \left(\bar{C}_k^p \right)^{-1} \sum_{i=m+1}^{N} \tilde{d}_k^i \right)^T \right).$$

Due to the orthogonality principle [44] we have

$$E\left(\left(\underline{x}_k - \hat{\underline{x}}_k^{e,d'} \right) \left(C_k^{e,d'} \left(\bar{C}_k^p \right)^{-1} \sum_{i=m+1}^{N} \tilde{d}_k^i \right)^T \right) = 0 \,,$$

and

$$E\left(\left(C_k^{e,d'} \left(\bar{C}_k^p \right)^{-1} \sum_{i=m+1}^{N} \tilde{d}_k^i \right) \left(\underline{x}_k - \hat{\underline{x}}_k^{e,d'} \right)^T \right) = 0 \,.$$

We can now write (A1) as

$$C_k^{e,d'} + C_k^{e,d'} \left(\bar{C}_k^p \right)^{-1} E\left(\sum_{i=m+1}^{N} \sum_{j=m+1}^{N} \tilde{d}_k^i \left(\tilde{d}_k^j \right)^T \right) \cdot$$
$$\left(\bar{C}_k^p \right)^{-1} \left(C_k^{e,d'} \right)^T \,.$$

To complete the proof we still have to show that

$$E\left(\sum_{i=m+1}^{N} \sum_{j=m+1}^{N} \tilde{d}_k^i \left(\tilde{d}_k^j \right)^T \right) \le (N - m - l)^2 \mathbf{B} \,.$$

For $i = j$ and $i, j \in \{m+1, \dots, N\}$ we have

$$\tilde{d}_k^i \left(\tilde{d}_k^j \right)^T \le \mathbf{B} \,,$$

since no communication was performed from sensor i to the center at time k. For $i \ne j$ and $i, j \in \{m+1, \dots, N\}$, we have

$$\mathbf{0} \le \left(\tilde{d}_k^i - \tilde{d}_k^j \right) \left(\tilde{d}_k^i - \tilde{d}_k^j \right)^T$$
$$= \tilde{d}_k^i \left(\tilde{d}_k^i \right)^T - \tilde{d}_k^i \left(\tilde{d}_k^j \right)^T - \tilde{d}_k^j \left(\tilde{d}_k^i \right)^T + \tilde{d}_k^j \left(\tilde{d}_k^j \right)^T \,.$$

We have then

$$\tilde{d}_k^i \left(\tilde{d}_k^j \right)^T + \tilde{d}_k^j \left(\tilde{d}_k^i \right)^T \le \tilde{d}_k^i \left(\tilde{d}_k^i \right)^T + \tilde{d}_k^j \left(\tilde{d}_k^j \right)^T \le 2\mathbf{B} \,.$$

The number of summands different from zero in the sum $\sum_{i=m+1}^{N} \sum_{j=m+1}^{N} \tilde{d}_k^i (\tilde{d}_k^j)^T$ is $(N - m - l)^2$. Thus, we have

$$\sum_{i=m+1}^{N} \sum_{j=m+1}^{N} \tilde{d}_k^i \left(\tilde{d}_k^j \right)^T \le (N - m - l)^2 \mathbf{B} \,.$$

It follows that

$$E\left(\sum_{i=m+1}^{N}\sum_{j=m+1}^{N} d_k^i (d_k^j)^T\right) \leq (N - m - l)^2 \boldsymbol{B}\,.$$

□

References

1. Zhang, L.; Liu, Y.; Deng, P. Odor Recognition in Multiple E-Nose Systems With Cross-Domain Discriminative Subspace Learning. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 1679–1692.

2. Zhang, L.; Wang, X.; Huang, G.B.; Liu, T.; Tan, X. Taste Recognition in E-Tongue Using Local Discriminant Preservation Projection. *IEEE Trans. Cybern.* **2018**, *PP*, 1–14.

3. Zhang, L.; Zuo, W.; Zhang, D. LSDT: Latent Sparse Domain Transfer Learning for Visual Adaptation. *IEEE Trans. Image Process.* **2016**, *25*, 1177–1191.

4. Thrun, S.; Burgard, W.; Fox, D. *Probabilistic Robotics*; Intelligent Robotics and Autonomous Agents; MIT Press: Cambridge, MA, USA, 2005.

5. Werner-Allen, G.; Johnson, J.; Ruiz, M.; Lees, J.; Welsh, M. Monitoring Volcanic Eruptions with a Wireless Sensor Network. In Proceedings of the Second European Workshop on Wireless Sensor Networks (EWSN 2005), Istanbul, Turkey, 31 January–2 February 2005; pp. 108–120.

6. Manolakos, E.S.; Logaras, E.; Paschos, F. Wireless Sensor Network Application for Fire Hazard Detection and Monitoring. In *Sensor Applications, Experimentation, and Logistics*; Komninos, N., Akan, O., Bellavista, P., Cao, J., Dressler, F., Ferrari, D., Gerla, M., Kobayashi, H., Palazzo, S., Sahni, S., et al., Eds.; Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering; Springer: Berlin/Heidelberg, Germany, 2010; doi:10.1007/978-3-642-11870-8_1.

7. Kalman, R.E. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, *82*, 35–45.

8. Noack, B.; Sijs, J.; Reinhardt, M.; Hanebeck, U.D. Treatment of Dependent Information in Multisensor Kalman Filtering and Data Fusion. In *Multisensor Data Fusion: From Algorithms and Architectural Design to Applications*; Fourati, H., Ed.; CRC Press: Boca Raton, FL, USA, 2015; pp. 169–192.

9. Ajgl, J.; Šimandl, M. Design of a Robust Fusion of Probability Densities. In Proceedings of the 2015 American Control Conference (ACC 2015), Chicago, IL, USA, 1–3 July 2015.

10. Bakr, M.A.; Lee, S. Distributed Multisensor Data Fusion under Unknown Correlation and Data Inconsistency. *Sensors* **2017**, *17*, 2472, doi:10.3390/s17112472.

11. Bar-Shalom, Y.; Campo, L. The Effect of the Common Process Noise on the Two-Sensor Fused-Track Covariance. *IEEE Trans. Aerosp. Electron. Syst.* **1986**, *AES-22*, 803–805.

12. Steinbring, J.; Noack, B.; Reinhardt, M.; Hanebeck, U.D. Optimal Sample-Based Fusion for Distributed State Estimation. In Proceedings of the 19th International Conference on Information Fusion (Fusion 2016), Heidelberg, Germany, 5–8 July 2016.

13. Julier, S.J.; Uhlmann, J.K. A Non-divergent Estimation Algorithm in the Presence of Unknown Correlations. In Proceedings of the IEEE American Control Conference (ACC 1997), Albuquerque, NM, USA, 6 June 1997; Volume 4, pp. 2369–2373.

14. Uhlmann, J.K. General data fusion for estimates with unknown cross covariances. In Proceedings of the International Society for Optics and Photonics on Aerospace/Defense Sensing and Controls, Orlando, FL, USA, 14 June 1996; pp. 536–547.

15. Sijs, J. *State Estimation in Networked Systems*; Technical University Eindhoven: Eindhoven, The Netherlands, 2012.

16. Sijs, J.; Lazar, M. State fusion with unknown correlation: Ellipsoidal intersection. *Automatica* **2012**, *48*, 1874–1878.

17. Noack, B.; Sijs, J.; Hanebeck, U.D. Algebraic Analysis of Data Fusion with Ellipsoidal Intersection. In Proceedings of the 2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2016), Baden-Baden, Germany, 19–21 September 2016.

18. Noack, B.; Sijs, J.; Reinhardt, M.; Hanebeck, U.D. Decentralized Data Fusion with Inverse Covariance Intersection. *Automatica* **2017**, *79*, 35–41.

19.　Noack, B.; Sijs, J.; Hanebeck, U.D. Inverse Covariance Intersection: New Insights and Properties. In Proceedings of the 20th International Conference on Information Fusion (Fusion 2017), Xi'an, China, 10–13 July 2017.

20.　Koch, W. On Optimal Distributed Kalman Filtering and Retrodiction at Arbitrary Communication Rates for Maneuvering Targets. In Proceedings of the 2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2008), Seoul, Korea, 20–22 August 2008.

21.　Koch, W. Exact Update Formulae for Distributed Kalman Filtering and Retrodiction at Arbitrary Communication Rates. In Proceedings of the 12th International Conference on Information Fusion (Fusion 2009), Seattle, WA, USA, 6–9 July 2009; pp. 2209–2216.

22.　Govaers, F.; Koch, W. Distributed Kalman Filter Fusion at Arbitrary Instants of Time. In Proceedings of the 13th International Conference on Information Fusion (Fusion 2010), Edinburgh, UK, 26–29 July 2010.

23.　Govaers, F.; Koch, W. On the Globalized Likelihood Function for Exact Track-To-Track Fusion at Arbitrary Instants of Time. In Proceedings of the 14th International Conference on Information Fusion (Fusion 2011), Chicago, IL, USA, 5–8 July 2011.

24.　Reinhardt, M.; Noack, B.; Hanebeck, U.D. The Hypothesizing Distributed Kalman Filter. In Proceedings of the 2012 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2012), Hamburg, Germany, 13–15 September 2012.

25.　Reinhardt, M.; Noack, B.; Hanebeck, U.D. Advances in Hypothesizing Distributed Kalman Filtering. In Proceedings of the 16th International Conference on Information Fusion (Fusion 2013), Istanbul, Turkey, 9–12 July 2013.

26.　Pfaff, F.; Noack, B.; Hanebeck, U.D.; Govaers, F.; Koch, W. Information Form Distributed Kalman Filtering (IDKF) with Explicit Inputs. In Proceedings of the 20th International Conference on Information Fusion (Fusion 2017), Xi'an, China, 10–13 July 2017.

27.　Noack, B.; Pfaff, F.; Hanebeck, U.D. Optimal Kalman Gains for Combined Stochastic and Set-Membership State Estimation. In Proceedings of the 51st IEEE Conference on Decision and Control (CDC 2012), Maui, HI, USA, 10–13 December 2012.

28.　Pfaff, F.; Noack, B.; Hanebeck, U.D. Optimal Distributed Combined Stochastic and Set-Membership State Estimation. In Proceedings of the 20th International Conference on Information Fusion (Fusion 2017), Xi'an, China, 10–13 July 2017.

29.　Battistelli, G.; Chisci, L.; Selvi, D. Distributed Kalman Filtering with Data-Driven Communication. In Proceedings of the 19th International Conference on Information Fusion (Fusion 2016), Heidelberg, Germany, 5–8 July 2016.

30.　Battistelli, G.; Chisci, L.; Selvi, D. A Distributed Kalman Filter with Event-Triggered Communication and Guaranteed Stability. *Automatica* **2018**, *93*, 75–82.

31.　Dormann, K.; Noack, B.; Hanebeck, U.D. Distributed Kalman Filtering With Reduced Transmission Rate. In Proceedings of the 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2017), Daegu, South Korea, 16–18 November 2017.

32.　Mutambara, A.G.O. *Decentralized Estimation and Control for Multisensor Systems*; CRC Press, Inc.: Boca Raton, FL, USA, 1998.

33.　Noack, B.; Lyons, D.; Nagel, M.; Hanebeck, U.D. Nonlinear Information Filtering for Distributed Multisensor Data Fusion. In Proceedings of the 2011 American Control Conference (ACC 2011), San Francisco, CA, USA, 29 June–1 July 2011.

34.　Suh, Y.S.; Nguyen, V.H.; Ro, Y.S. Modified Kalman filter for networked monitoring systems employing a send-on-delta method. *Automatica* **2007**, *43*, 332–338.

35.　Marck, J.W.; Sijs, J. Relevant sampling applied to event-based state-estimation. In Proceedings of the 2010 Fourth International Conference on Sensor Technologies and Applications (SENSORCOMM), Venice, Italy, 18–25 July 2010; pp. 618–624.

36.　Shi, L.; Johansson, K.H.; Qiu, L. Time and event-based sensor scheduling for networks with limited communication resources. *IFAC Proc. Vol.* **2011**, *44*, 13263–13268.

37.　Battistelli, G.; Benavoli, A.; Chisci, L. Data-driven communication for state estimation with sensor networks. *Automatica* **2012**, *48*, 926–935.

38.　Sijs, J.; Noack, B.; Hanebeck, U.D. Event-based State Estimation with Negative Information. In Proceedings of the 16th International Conference on Information Fusion (Fusion 2013), Istanbul, Turkey, 9–12 July 2013.

39. Shi, D.; Chen, T.; Shi, L. An Event-Triggered Approach to State Estimation with Multiple Point- and Set-Valued Measurements. *Automatica* **2014**, *50*, 1641–1648.

40. Sijs, J.; Kester, L.; Noack, B. A Study on Event Triggering Criteria for Estimation. In Proceedings of the 17th International Conference on Information Fusion (Fusion 2014), Salamanca, Spain, 7–10 July 2014.

41. Trimpe, S.; D'Andrea, R. Event-based state estimation with variance-based triggering. *IEEE Trans. Autom. Control* **2014**, *59*, 3266–3281.

42. Han, D.; Mo, Y.; Wu, J.; Weerakkody, S.; Sinopoli, B.; Shi, L. Stochastic event-triggered sensor schedule for remote state estimation. *IEEE Trans. Autom. Control* **2015**, *60*, 2661–2675.

43. Sinopoli, B.; Schenato, L.; Franceschetti, M.; Poolla, K.; Jordan, M.I.; Sastry, S.S. Kalman filtering with intermittent observations. *IEEE Trans. Autom. Control* **2004**, *49*, 1453–1464.

44. Moon, T.K.S.; Wynn, C. *Mathematical Methods and Algorithms for Signal Processing*; Number 621.39: 51 MON; Prentice Hall: Upper Saddle River, NJ, USA, 2000.