Distributed Kalman Filtering With Reduced Transmission Rate

Katharina Dormann, Benjamin Noack, and Uwe D. Hanebeck

***Abstract*— The centralized Kalman filter can be implemented in such a way that the required calculations can be distributed over multiple nodes in a network, each of which processes only the locally acquired sensor data. The main downside of this implementation is that it requires each distributed sensor node to communicate with the fusion center in every time step so as to compute the optimal state estimate. In this paper, two distributed Kalman filtering algorithms are proposed to overcome these limitations. The first algorithm merely requires communication of each local sensor node with the fusion center in every other time step. The second algorithm even allows for a lower communicate rate. Both algorithms apply event-based communication to compute consistent estimates and to reduce the estimation error for a fixed communication rate. Simulations demonstrate that both algorithms perform better in terms of the mean squared estimation error than the centralized Kalman filter.**

I. INTRODUCTION

A well-known algorithm for data fusion in sensor networks is the Kalman filter [1]. We consider *centralized* and *distributed* sensor networks. In a centralized sensor network the local sensors send their measurements to the central node in the network. The central node fuses the measurements using the Kalman filter algorithm. The result is an estimate of the state that has been measured by the local nodes. In contrast, in a distributed sensor network each local sensor node uses its measurements to compute a state estimate with the help of a local Kalman filter. The central node then fuses the local estimates to obtain a superior estimate.

One advantage of distributed sensor networks compared to centralized sensor networks is the reduction of communication load in the network. Since the local estimates contain the information from all past measurements, the estimates can be sent to the central node after any number of time steps, without losing the information from the past measurements. Another advantage of distributed networks is their robustness. The local estimates can be used as a backup in case the central node fails.

However, the task of fusing local estimates is challenging due to *correlations* between local estimates [2], [3]. Naive fusion of correlated estimates leads to an underestimation of the actual estimation error by the computed error covariance matrix. The estimator is then called *inconsistent*. One possibility to solve this problem is to keep track of the correlations between the local estimates. If the correlations are known, a *consistent* estimate can be computed using *Bar-Shalom-Campo* fusion [4]. In the case of unknown correlations, the fusion algorithms *Covariance Intersection* [5], [6] and its further development *Inverse Covariance Intersection* [7], [8] can be applied to compute consistent estimates. However, these algorithms return error covariance matrices that *overestimate* the actual estimation error.

Koch and Govaers have proposed a distributed fusion algorithm that does not require to keep track of the correlations and provides an error covariance matrix which exactly describes the actual estimation error [9]–[12]. This is realized by a distributed implementation of the centralized Kalman filter. The local estimates are neither optimal nor unbiased, nor does the error covariance matrix describe the actual estimation error. But fusing them at the fusion center results in an estimate that equals the fusion result of a centralized Kalman filter. Also, some relaxations can be implemented [13], [14] that rely on a hypothesis about information provided by the entire network.

A drawback of this algorithm is that it only provides fusion formulas for those time steps, at which the estimates of all sensors in the network are available at the center. In this paper, we will present two generalizations of this algorithm. These novel algorithms also provide fusion formulas for the case that not all estimates are available at the center at a particular time step. Thus, the newly developed algorithms allow for asynchronous communication in the network. With the help of these novel distributed fusion algorithms, it is possible to reduce the communication load in the network while still providing good estimates at the fusion center.

The first algorithm returns an error covariance matrix which exactly describes the actual estimation error. However, the algorithm requires that each sensor communicates with the center node at least every other time step. The second algorithm allows for arbitrary communication rates. It uses an event-based communication strategy and schedules the data to be sent to the fusion center according to its relevance. It also returns a consistent estimate.

The paper is structured as follows. In Section II the centralized and the distributed Kalman filtering algorithms are described, and the problem is formulated. In Section III, we describe the first new distributed algorithm which allows for omitted estimates at the fusion center. In Section IV, we describe the second new distributed algorithm which allows for omitted estimates at the fusion center over multiple time steps. Finally, in Section V, we show the results of an experimental evaluation of the algorithms.

*This work was supported by the German Research Foundation (DFG) under grant NO 1133/1-1.

Katharina Dormann, Benjamin Noack, and Uwe D. Hanebeck are with the Intelligent Sensor-Actuator-Systems Laboratory (ISAS), Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology (KIT), Germany. uhenw@student.kit.edu, benjamin.noack@ieee.org, uwe.hanebeck@ieee.org

II. CENTRALIZED AND DISTRIBUTED KALMAN FILTERING

We consider a sensor network with N local sensor nodes and one central unit, which has no sensing capabilities. We consider a discrete-time linear dynamic system. The true state of the system at time k is denoted by $\underline{x}_k$. The system evolves according to

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

where $\boldsymbol{A}_k$ is a matrix describing the system model and $\underline{w}_k$ is the system noise, which is assumed to be Gaussian noise with zero mean, $\underline{w}_k \sim \mathcal{N}(\underline{0}, \boldsymbol{C}_k^w)$. At each time step k, each sensor i produces a measurement $\underline{\boldsymbol{z}}_k^i$ according to

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

where $\boldsymbol{H}_k^i$ is a matrix describing the measurement model and $\underline{\boldsymbol{v}}_k^i$ the measurement noise, which is assumed to be Gaussian noise with zero mean, $\underline{\boldsymbol{v}}_k \sim \mathcal{N}(\underline{0}, \boldsymbol{C}_k^{z,i})$.

In a centralized sensor network, the central unit fuses the measurements from the local nodes according to

$$(\boldsymbol{C}_k^{e,c})^{-1} \hat{\underline{x}}_k^{e,c} = (\boldsymbol{C}_k^{p,c})^{-1} \hat{\underline{x}}_k^{p,c} + \sum_{i=1}^{N} \left(\boldsymbol{H}_k^i\right)^T (\boldsymbol{C}_k^{z,i})^{-1} \underline{\boldsymbol{z}}_k^i \ ,$$
$$(\boldsymbol{C}_k^{e,c})^{-1} = (\boldsymbol{C}_k^{p,c})^{-1} + \sum_{i=1}^{N} \left(\boldsymbol{H}_k^i\right)^T \left(\boldsymbol{C}_k^{z,i}\right)^{-1} \boldsymbol{H}_k^i \, .$$

These equations correspond to the information form [15], [16] of the filtering step of the standard Kalman filter, where the noise of different local sensors is assumed to be uncorrelated. $\hat{\underline{x}}_k^{e,c}$ and $\boldsymbol{C}_k^{e,c}$ denote the state estimate and the corresponding error covariance matrix after the fusion step. $\hat{\underline{x}}_k^{p,c}$ and $\boldsymbol{C}_k^{p,c}$ denote the state estimate and the corresponding error covariance matrix after the prediction step. After each fusion, a prediction is performed at the center by

$$\left(\boldsymbol{C}_{k+1}^{p,c}\right)^{-1} \hat{\underline{x}}_{k+1}^{p,c} = \left(\boldsymbol{C}_{k+1}^{p,c}\right)^{-1} \boldsymbol{A}_k \hat{\underline{x}}_k^{e,c} \ ,$$
$$\left(\boldsymbol{C}_{k+1}^{p,c}\right)^{-1} = \left(\boldsymbol{A}_k \boldsymbol{C}_k^{e,c} \boldsymbol{A}_k^T + \boldsymbol{C}_k^w\right)^{-1} \ .$$

Since these equations correspond to the standard Kalman filter, the centralized Kalman filter is unbiased and optimal in the minimum mean squared error sense. Also, the computed error covariance matrix is equal to the actual estimation error, i.e.,

$$\boldsymbol{C}_k^{e,c} = E\left(\left(\hat{\underline{x}}_k^{e,c} - \underline{\boldsymbol{x}}_k\right) \left(\hat{\underline{x}}_k^{e,c} - \underline{\boldsymbol{x}}_k\right)^T \right) \ . \tag{1}$$

In [9]–[12], a distributed Kalman filter algorithm has been proposed, which is also unbiased, optimal in the minimum mean squared error sense, and for which (1) holds. This is achieved by defining a local filtering scheme such that the fusion result is equal to the fusion result of a centralized Kalman filter. We will describe the algorithm in the following.

The local sensor nodes perform a *modified* version of the Kalman filtering algorithm. They work with so called *globalized* local states estimates [1] and error covariance matrices. Let $(\hat{\underline{x}}_0^{e,i}, \boldsymbol{C}_0^{e,i})$ be the initial state estimates and error covariance matrices for all sensors $i \in \{1, \ldots, N\}$. They are sent to the central node and fused according to

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

These are the initial globalized estimates and error covariance matrices, which are sent back to the local nodes. The globalized error covariance matrix is equal for each sensor and thus, is not denoted with the sensor index i. This also applies to all future time steps. The local prediction step is given by

$$\bar{\underline{x}}_{k+1}^{p,i} = \boldsymbol{A}_k \bar{\underline{x}}_k^{e,i} \ , \tag{2}$$
$$\bar{\boldsymbol{C}}_{k+1}^{p} = \boldsymbol{A}_k \bar{\boldsymbol{C}}_k^e \boldsymbol{A}_k^T + N \boldsymbol{C}_k^w \ .$$

The local filtering step is given by

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

The central node receives the globalized estimates $(\bar{\underline{x}}_k^{e,i}, \bar{\boldsymbol{C}}_k^e)$ from each sensor and fuses them according to

$$\left(\boldsymbol{C}_k^{e,d}\right)^{-1} = \sum_{i=1}^{N} \left(\bar{\boldsymbol{C}}_k^e\right)^{-1} \ , \tag{3}$$

$$\left(\boldsymbol{C}_k^{e,d}\right)^{-1} \hat{\underline{x}}_k^{e,d} = \sum_{i=1}^{N} \left(\bar{\boldsymbol{C}}_k^e\right)^{-1} \bar{\underline{x}}_k^{e,i} \, . \tag{4}$$

$\hat{\underline{x}}_k^{e,d}$ denotes the state estimate after the fusion step in the distributed sensor network, and $\boldsymbol{C}_k^{e,d}$ denotes the corresponding error covariance matrix.

From (3) and (4), we can easily derive that

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

The same equations apply to the predicted estimates and error covariance matrices. It can be shown that

$$\hat{\underline{x}}_k^{e,d} = \hat{\underline{x}}_k^{e,c} \, , \tag{5}$$
$$\boldsymbol{C}_k^{e,d} = \boldsymbol{C}_k^{e,c} \, , \tag{6}$$

where we assume that $\hat{\underline{x}}_k^{e,c}$ and $\boldsymbol{C}_k^{e,c}$ have been computed in a centralized sensor network where each sensor node communicates with the central unit at every time step. Note that communication in past time steps does not influence $\hat{\underline{x}}_k^{e,d}$ and $\boldsymbol{C}_k^{e,d}$, i.e., the equalities hold independently of the past communication pattern in the distributed network. It follows from (5) and (6) that

$$\boldsymbol{C}_k^{e,d} = E\left(\left(\hat{\underline{x}}_k^{e,d} - \underline{\boldsymbol{x}}_k\right) \left(\hat{\underline{x}}_k^{e,d} - \underline{\boldsymbol{x}}_k\right)^T \right) \tag{7}$$

[1] Although, strictly speaking, the local parameters do not represent estimates of the states, we denote them as local estimates.

is equal to (1). Consider now the case that only m out of N sensors send their estimates to the fusion center at time k. Equations (3) and (4) then become

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

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

The resulting estimate $\hat{\underline{x}}_k^{e,d}$ and error covariance matrix $\boldsymbol{C}_k^{e,d}$ are *not* equal to the estimate $\hat{\underline{x}}_k^{e,c}$ and the error covariance matrix $\boldsymbol{C}_k^{e,c}$ computed according to

$$(\boldsymbol{C}_k^{e,c})^{-1} \hat{\underline{x}}_k^{e,c} = (\boldsymbol{C}_k^{p,c})^{-1} \hat{\underline{x}}_k^{p,c} + \sum_{i=1}^{m} (\boldsymbol{H}_k^i)^T (\boldsymbol{C}_k^{z,i})^{-1} \underline{\boldsymbol{z}}_k^i \ , \quad (8)$$

$$(\boldsymbol{C}_k^{e,c})^{-1} = (\boldsymbol{C}_k^{p,c})^{-1} + \sum_{i=1}^{m} \left(\boldsymbol{H}_k^i\right)^T \left(\boldsymbol{C}_k^{z,i}\right)^{-1} \boldsymbol{H}_k^i \quad (9)$$

since in the centralized sensor network the fusion result depends also on the communication pattern in past time steps. The equality between the centralized and distributed fusion result would only hold in case that exactly m measurements had been fused in every past time step in the distributed network. As a consequence, when applying the presented distributed Kalman filtering algorithm although some estimates are not available at the center, we will, in general, not get an unbiased and optimal estimate, and consistency according to (7) can be violated.

In Section III, we present a new distributed fusion algorithm which allows for the absence of estimates at the fusion center and still provides a fusion result which is equal to the fusion result of the centralized Kalman filter. With this algorithm, we are able to reduce by half the communication rate. Section IV introduces a second algorithm that can even reach a lower communication rate by applying a bound on the non-transmitted information.

III. DISTRIBUTED KALMAN FILTER WITH OMITTED ESTIMATES

We consider again the distributed Kalman filtering algorithm described in Section II. We consider the case that at time k only sensor nodes $1, \ldots, m$ communicate with the fusion center, but sensor nodes $m+1, \ldots, N$ do not. However, we assume that at time $k-1$ sensors $m+1, \ldots, N$ had communicated with the fusion center. Thus, we can assume that the predicted estimates $\bar{\underline{x}}_k^{p,m+1}, \ldots, \bar{\underline{x}}_k^{p,N}$ can be computed using (2) and are available at time k. We now replace the fusion equations (3) and (4) by the generalized fusion equations

$$\left(\boldsymbol{C}_k^{e,d'}\right)^{-1} = \sum_{i=1}^{N} \left(\bar{\boldsymbol{C}}_k^{e}\right)^{-1} - \sum_{i=m+1}^{N} (\boldsymbol{H}_k^i)^T (\boldsymbol{C}_k^{z,i})^{-1} \boldsymbol{H}_k^i \ , \quad (10)$$

$$\left(\boldsymbol{C}_k^{e,d'}\right)^{-1} \hat{\underline{x}}_k^{e,d'} = \sum_{i=1}^{m} \left(\bar{\boldsymbol{C}}_k^{e}\right)^{-1} \bar{\underline{x}}_k^{e,i} + \sum_{i=m+1}^{N} (\bar{\boldsymbol{C}}_k^{p})^{-1} \bar{\underline{x}}_k^{p,i} \ . \quad (11)$$

We will now show that $\hat{\underline{x}}_k^{e,d'} = \hat{\underline{x}}_k^{e,c}$ and $\boldsymbol{C}_k^{e,d'} = \boldsymbol{C}_k^{e,c}$, where we assume that $\hat{\underline{x}}_k^{e,c}$ and $\boldsymbol{C}_k^{e,c}$ have been computed by (8) and (9) in a centralized sensor network, where each sensor node communicates with the central unit at every time step until time $k-1$ (included), but only the first m nodes communicate with the central unit a time k.

Proof: $\boldsymbol{C}_k^{e,d'} = \boldsymbol{C}_k^{e,c}$ holds due to

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

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

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

□

We have shown the equality of the fusion result of the newly developed distributed Kalman filtering algorithm to the fusion result of the centralized Kalman filter. Thus, unbiasedness, optimality, and (7) are inherited from the centralized Kalman filter.

We have generalized the original distributed Kalman filtering 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 communicated at time $k-1$, i.e., each sensor has to communicate with the center at least every other time step. Thus, the communication rate can be lowered 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 an event-based communication strategy and thus, scheduling the data according to the information contained. Valuable results have already been achieved using event-based communication in distributed sensor networks [17]–[25]. The idea is that each local sensor evaluates if the distance between the predicted estimate $\underline{\bar{x}}_k^{p,i}$ and the filtered estimate $\underline{\bar{x}}_k^{e,i}$ is large, and thus, if the measurement $\underline{\boldsymbol{z}}_k^i$ 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.

However, we have to take into consideration that due to the globalization $\underline{\bar{x}}_k^{p,i}$ and $\underline{\bar{x}}_k^{e,i}$ are *not unbiased* estimates of the true state. Experiments have shown 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 even *diverges*. Therefore, to evaluate the relevance of the measurement $\underline{z}_k^i$, we consider the difference between the standard Kalman filter estimates, which is equal to the weighted difference between the measurement and the prediction converted to the measurement space, i.e.,

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

where $\boldsymbol{K}_k^i$ denotes the Kalman gain. To obtain the standard Kalman filer estimates, the standard Kalman filtering algorithm has to run in parallel to the globalized version of the Kalman filter at each sensor node. The following event-based communication strategy is applied.

$$\text{If } \|\underline{\hat{x}}_k^{p,i} - \underline{\hat{x}}_k^{e,i}\| \leq \alpha \tag{12}$$

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 $\underline{\hat{x}}_k^{p,i}$, and not to the fused estimate $\underline{\hat{x}}_k^{p,d}$. Still, experiments (see Section V) will show that by using the event-based communication strategy instead of random communication, for a fixed communication rate an improvement of the MSE of the fused estimate $\underline{\hat{x}}_k^{e,d'}$ can be achieved.

As mentioned before, communication rates lower than 0.5 cannot be achieved with the newly developed distributed fusion algorithm. In Section IV, we will present a generalization of the algorithm, which allows for any communication pattern in the network and thus, for any communication rate, and still provides consistent fusion results.

IV. DISTRIBUTED KALMAN FILTER WITH OMITTED ESTIMATES OVER MULTIPLE TIME STEPS

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. By $\underline{\bar{x}}_k^{pp,i}$, we denote the estimate which has been computed by applying equation (2) not only once but any number of times, until the next communication with the center node occurs. In case that prediction has been performed only once, we have $\underline{\bar{x}}_k^{pp,i} = \underline{\bar{x}}_k^{p,i}$. Fusion equation (11) is generalized to

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

The new estimate $\underline{\hat{x}}_k^{e,d''}$ can be expressed in terms of the estimate $\underline{\hat{x}}_k^{e,d'}$ from (11) as follows:

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

We define

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

The expected estimation error is then given by

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

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 (12). We ensure that in case of communication the matrix $d_k^i\left(d_k^i\right)^T$ is bounded, by using the communication strategy

$$\begin{aligned}
&\text{If } \|\underline{\hat{x}}_k^{pp,i} - \underline{\hat{x}}_k^{e,i}\| \leq \alpha \text{ and } d_k^i\left(d_k^i\right)^T \leq \boldsymbol{B} \\
&\qquad\text{do not send estimate to the fusion center} \\
&\text{else} \\
&\qquad\text{send estimate to the fusion center,}
\end{aligned} \tag{13}$$

where $\underline{\hat{x}}_k^{pp,i}$ denotes the estimate which has been computed by applying the standard Kalman filtering prediction step multiple times and $\boldsymbol{B}$ denotes any given symmetric positive definite square-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.

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 *diverges*. As a consequence, the matrix $d_k^i\left(d_k^i\right)^T$ also diverges. In contrast to the previous fusion algorithm, using the standard Kalman filtering estimates $\underline{\hat{x}}_k^{pp,i}$, $\underline{\hat{x}}_k^{p,i}$ 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 [13], [14]. In each prediction and filtering step, each local node computes a new debiasing matrix. The 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

$$\Delta_k^{p,i} = \boldsymbol{A}_k\Delta_k^{e,i}\boldsymbol{A}_k^{-1}\,. \tag{14}$$

$\Delta_k^{pp,i}$ is computed by applying equation (14) 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 [13], [14], i.e.,

$$\begin{aligned}
E\left(\left(\Delta_k^{e,i}\right)^{-1}\underline{\bar{x}}_k^{e,i}\right) &= E\left(\underline{\boldsymbol{x}}_k\right), \\
E\left(\left(\Delta_k^{p,i}\right)^{-1}\underline{\bar{x}}_k^{p,i}\right) &= E\left(\underline{\boldsymbol{x}}_k\right)\,.
\end{aligned}$$

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

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

We define now

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

and then have

$$\begin{aligned}
E\left(\underline{\tilde{x}}_k^{pp,i} - \underline{\bar{x}}_k^{p,i}\right) &= E\left(\Delta_k^{p,i}(\Delta_k^{pp,i})^{-1}\underline{\bar{x}}_k^{pp,i} - \underline{\bar{x}}_k^{p,i}\right) \\
&= \Delta_k^{p,i}E\left((\Delta_k^{pp,i})^{-1}\underline{\bar{x}}_k^{pp,i} - \left(\Delta_k^{p,i}\right)^{-1}\underline{\bar{x}}_k^{p,i}\right) \\
&= \Delta_k^{p,i}E\left(\underline{\boldsymbol{x}}_{k+n} - \underline{\boldsymbol{x}}_k\right)\,.
\end{aligned}$$

Thus, in general the difference $\underline{\tilde{x}}_k^{pp,i} - \underline{\bar{x}}_k^{p,i}$ does not diverge. We define

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

and replace in (13) the second inequality by

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

We can now define the new fusion equations as follows.

$$\begin{aligned}
\boldsymbol{C}_k^{e,d''} &= \boldsymbol{C}_k^{e,d'} \\
&\quad + (N-m-l)^2\boldsymbol{C}_k^{e,d'}\left(\bar{\boldsymbol{C}}_k^p\right)^{-1}\boldsymbol{B}\left(\bar{\boldsymbol{C}}_k^p\right)^{-1}\left(\boldsymbol{C}_k^{e,d'}\right)^T \\
\underline{\hat{x}}_k^{e,d''} &= \boldsymbol{C}_k^{e,d'}\left(\sum_{i=1}^{m}(\bar{\boldsymbol{C}}_k^e)^{-1}\underline{\bar{x}}_k^{e,i} + \sum_{i=m+1}^{N}(\bar{\boldsymbol{C}}_k^p)^{-1}\underline{\tilde{x}}_k^{pp,i}\right),
\end{aligned}$$

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 $\underline{\tilde{x}}_k^{pp,i} - \underline{\bar{x}}_k^{p,i} = 0$. Note that the fusion formulas are equal to (10) and (11) for $N = m + l$, i.e., for the case that each sensor sends its estimate to the center at least every other time step.

We will now show that the resulting estimate is *consistent*, i.e.,

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

Proof: We have

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

The expected estimation error is then given by

$$\begin{aligned}
&E\left(\left(\underline{\boldsymbol{x}}_k - \underline{\hat{x}}_k^{e,d''}\right)\left(\underline{\boldsymbol{x}}_k - \underline{\hat{x}}_k^{e,d''}\right)^T\right) \\
&= E\left(\left(\underline{\boldsymbol{x}}_k - \underline{\hat{x}}_k^{e,d'}\right)\left(\underline{\boldsymbol{x}}_k - \underline{\hat{x}}_k^{e,d'}\right)^T\right) \\
&\quad - E\left(\left(\underline{\boldsymbol{x}}_k - \underline{\hat{x}}_k^{e,d'}\right)\left(\boldsymbol{C}_k^{e,d'}(\bar{\boldsymbol{C}}_k^p)^{-1}\sum_{i=m+1}^{N}\tilde{d}_k^i\right)^T\right) \\
&\quad - E\left(\left(\boldsymbol{C}_k^{e,d'}(\bar{\boldsymbol{C}}_k^p)^{-1}\sum_{i=m+1}^{N}\tilde{d}_k^i\right)\left(\underline{\boldsymbol{x}}_k - \underline{\hat{x}}_k^{e,d'}\right)^T\right) \\
&\quad + E\left(\left(\boldsymbol{C}_k^{e,d'}(\bar{\boldsymbol{C}}_k^p)^{-1}\sum_{i=m+1}^{N}\tilde{d}_k^i\right)\cdot\right. \\
&\qquad\left.\left(\boldsymbol{C}_k^{e,d'}\left(\bar{\boldsymbol{C}}_k^p\right)^{-1}\sum_{i=m+1}^{N}\tilde{d}_k^i\right)^T\right)\,.
\end{aligned} \tag{15}$$

Due to the orthogonality principle [26] we have

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

and

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

We can now write (15) as

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

To complete the proof we still have to show that

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

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

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

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

$$\begin{aligned} \boldsymbol{0} &\leq \left(\tilde{d}_k^i - \tilde{d}_k^j\right) \left(\tilde{d}_k^i - \tilde{d}_k^j\right)^T \\ &= \tilde{d}_k^i \left(\tilde{d}_k^i\right)^T - \tilde{d}_k^i \left(\tilde{d}_k^j\right)^T - \tilde{d}_k^j \left(\tilde{d}_k^i\right)^T + \tilde{d}_k^j \left(\tilde{d}_k^j\right)^T . \end{aligned}$$

We have then

$$\tilde{d}_k^i \left(\tilde{d}_k^j\right)^T + \tilde{d}_k^j \left(\tilde{d}_k^i\right)^T \leq \tilde{d}_k^i \left(\tilde{d}_k^i\right)^T + \tilde{d}_k^j \left(\tilde{d}_k^j\right)^T \leq 2\boldsymbol{B} .$$

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

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

It follows that

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

□

V. EXPERIMENTAL EVALUATION

We apply the centralized Kalman filter as well as the two newly developed distributed algorithms to a single-target tracking problem. The system state $\underline{\boldsymbol{x}}_k$ 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

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

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

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

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

$$\boldsymbol{C}_k^{z,i} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \text{ for } i \in 1, \ldots, 6 .$$

Monte Carlo simulations with 500 independent runs over 100 time steps are performed. Since $(\boldsymbol{A}_k, \boldsymbol{C}_k^w)$ is stable and $(\boldsymbol{A}_k, \boldsymbol{H}_k)$ is detectable the error covariance matrix and the MSE converge to a unique values [27]. These values are considered the error covariance matrix and the MSE of the particular algorithm.

Monte Carlo simulations are performed for different average communication rates for each of the three algorithms. For the centralized Kalman filter communication is performed randomly, but with different average rates. Note that only current measurements are communicated. If the measurement $\underline{\boldsymbol{z}}_k^i$ 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 newly developed algorithm (Algorithm 1) is performed with random communication as well as with event-based communication. In the latter case, the parameter α is varied to achieve different rates. The second algorithm (Algorithm 2) is performed with event-based communication, where both parameters α and $\boldsymbol{B}$ are varied.

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 graphic.

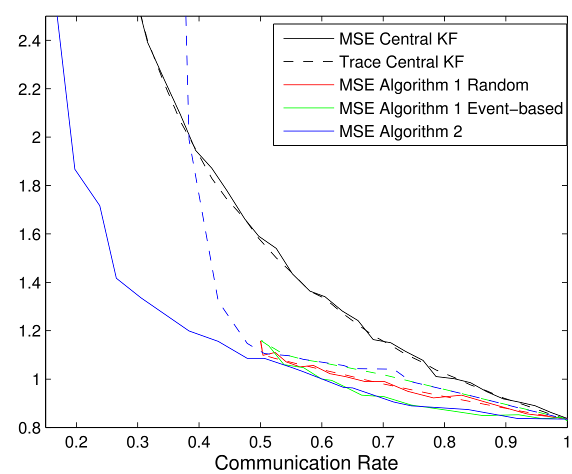

Fig. 1: MSEs and traces of the error covariance matrices are plotted relative to the communication rate. Each communication rate corresponds to one Monte Carlo simulation with 500 runs over 100 time steps. MSEs are shown as solid lines, traces are shown as dashed lines.

Only for the centralized Kalman filter and for Algorithm 2, communication rates lower than 0.5 are given. We can observe that for Algorithm 1 event-based 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 of the estimate.

We also can observe that for communication rates in range $[0.5, 1]$ the results of Algorithms 1 and 2 with event-based communication are almost equal. This can be explained by the fact that Algorithm 2 extends Algorithm 1 by an additional 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 shows that for each of the algorithms the MSE is always smaller than or equal to the trace of the error covariance matrix. This follows from the consistency of the estimators. The traces are good estimates of the MSEs except for very low communication rates in Algorithm 2. Thus, the uncertainty of the estimates is not overestimated too much by the trace of the error covariance matrices.

Each of the distributed fusion algorithms performs better in terms of small MSEs compared to the centralized algorithm (except for full rate communication). 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.

VI. Conclusions

In this paper, we have presented two novel distributed fusion algorithms which do not require to keep track of the correlations between the local estimates, but still provide consistent estimates. The algorithms are generalizations of another distributed fusion algorithm that requires full rate communication to compute current estimates. The novel algorithms overcome this drawback. The first algorithm allows for communication rates in the range $[0.5, 1]$ while the second algorithm allows for any communication rate in range $[0, 1]$. Both algorithms apply event-based communication to improve the fusion result. The consistency of the algorithms and the superiority over the centralized Kalman filter could be confirmed by simulations.

References

[1] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[2] B. Noack, J. Sijs, M. Reinhardt, and U. D. Hanebeck, "Treatment of Dependent Information in Multisensor Kalman Filtering and Data Fusion," in *Multisensor Data Fusion: From Algorithms and Architectural Design to Applications*, H. Fourati, Ed. CRC Press, Aug. 2015, pp. 169–192.

[3] J. Steinbring, B. Noack, M. Reinhardt, and U. D. Hanebeck, "Optimal Sample-Based Fusion for Distributed State Estimation," in *Proceedings of the 19th International Conference on Information Fusion (Fusion 2016)*, Heidelberg, Germany, Jul. 2016.

[4] Y. Bar-Shalom and L. Campo, "The effect of the common process noise on the two-sensor fused-track covariance," *IEEE Transactions on Aerospace and Electronic Systems*, no. 6, pp. 803–805, 1986.

[5] S. J. Julier and J. K. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *In Proceedings of the American Control Conference*. Citeseer, 1997.

[6] J. K. Uhlmann, "General data fusion for estimates with unknown cross covariances," in *Aerospace/Defense Sensing and Controls*. International Society for Optics and Photonics, 1996, pp. 536–547.

[7] B. Noack, J. Sijs, M. Reinhardt, and U. D. Hanebeck, "Decentralized data fusion with inverse covariance intersection," *Automatica*, vol. 79, pp. 35–41, May 2017.

[8] B. Noack, J. Sijs, and U. D. Hanebeck, "Inverse covariance intersection: New insights and properties," in *Proceedings of the 20th International Conference on Information Fusion (Fusion 2017)*, Xi'an, China, Jul. 2017.

[9] W. Koch, "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, Republic of Korea, Aug. 2008.

[10] W. Koch, "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, Washington, USA, Jul. 2009, pp. 2209–2216.

[11] F. Govaers and W. Koch, "Distributed kalman filter fusion at arbitrary instants of time," in *Proceedings of the 13th International Conference on Information Fusion (Fusion 2010)*, Edinburgh, United Kingdom, Jul. 2010.

[12] F. Govaers and W. Koch, "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, Illinois, USA, Jul. 2011.

[13] M. Reinhardt, B. Noack, and U. D. Hanebeck, "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, Sep. 2012.

[14] M. Reinhardt, B. Noack, and U. D. Hanebeck, "Advances in Hypothesizing Distributed Kalman Filtering," in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, Jul. 2013.

[15] A. G. O. Mutambara, *Decentralized Estimation and Control for Multisensor Systems*. Boca Raton, Florida, USA: CRC Press, Inc., 1998.

[16] B. Noack, D. Lyons, M. Nagel, and U. D. Hanebeck, "Nonlinear Information Filtering for Distributed Multisensor Data Fusion," in *Proceedings of the 2011 American Control Conference (ACC 2011)*, San Francisco, California, USA, Jun. 2011.

[17] Y. S. Suh, V. H. Nguyen, and Y. S. Ro, "Modified kalman filter for networked monitoring systems employing a send-on-delta method," *Automatica*, vol. 43, no. 2, pp. 332–338, 2007.

[18] J. W. Marck and J. Sijs, "Relevant sampling applied to event-based state-estimation," in *2010 Fourth International Conference on Sensor Technologies and Applications (SENSORCOMM)*, 2010, pp. 618–624.

[19] L. Shi, K. H. Johansson, and L. Qiu, "Time and event-based sensor scheduling for networks with limited communication resources," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 13 263–13 268, 2011.

[20] G. Battistelli, A. Benavoli, and L. Chisci, "Data-driven communication for state estimation with sensor networks," *Automatica*, vol. 48, no. 5, pp. 926–935, 2012.

[21] J. Sijs, B. Noack, and U. D. Hanebeck, "Event-based state estimation with negative information," in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, 2013.

[22] D. Shi, T. Chen, and L. Shi, "An event-triggered approach to state estimation with multiple point-and set-valued measurements," *Automatica*, vol. 50, no. 6, pp. 1641–1648, 2014.

[23] J. Sijs, L. Kester, and B. Noack, "A study on event triggering criteria for estimation," in *Proceedings of the 17th International Conference on Information Fusion (Fusion 2014)*, Salamanca, Spain, Jul. 2014.

[24] S. Trimpe and R. D'Andrea, "Event-based state estimation with variance-based triggering," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3266–3281, 2014.

[25] D. Han, Y. Mo, J. Wu, S. Weerakkody, B. Sinopoli, and L. Shi, "Stochastic event-triggered sensor schedule for remote state estimation," *IEEE Transactions on Automatic Control*, vol. 60, no. 10, pp. 2661–2675, 2015.

[26] T. K. S. Moon and C. Wynn, *Mathematical methods and algorithms for signal processing*. Prentice Hall, 2000, no. 621.39: 51 MON.

[27] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, "Kalman filtering with intermittent observations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1453–1464, 2004.