Information Form Distributed Kalman Filtering (IDKF) with Explicit Inputs

Florian Pfaff, Benjamin Noack, Uwe D. Hanebeck,
Intelligent Sensor-Actuator-Systems Laboratory (ISAS)
Institute for Anthropomatics and Robotics
Karlsruhe Institute of Technology (KIT), Germany
{florian.pfaff, benjamin.noack, uwe.hanebeck}@kit.edu

Felix Govaers, and Wolfgang Koch
Sensor Data and Information Fusion
Fraunhofer FKIE, Wachtberg, Germany
{felix.govaers, wolfgang.koch}@fkie.fraunhofer.de

***Abstract*—With the ubiquity of information distributed in networks, performing recursive Bayesian estimation using distributed calculations is becoming more and more important. There are a wide variety of algorithms catering to different applications and requiring different degrees of knowledge about the other nodes involved. One recently developed algorithm is the distributed Kalman filter (DKF), which assumes that all knowledge about the measurements, except the measurements themselves, are known to all nodes. If this condition is met, the DKF allows deriving the optimal estimate if all information is combined in one node at an arbitrary time step. In this paper, we present an information form of the distributed Kalman filter (IDKF) that allows the use of explicit system inputs at the individual nodes while still yielding the same results as a centralized Kalman filter.**

***Index Terms*—Distributed Kalman filter, recursive Bayesian estimation, information filter, track-to-track fusion**

I. Introduction

With the advent of cheap integrated circuits and sensors, the ability to observe phenomena from multiple perspectives has become viable and affordable. However, to reduce costs and keep sensors small, only very limited energy resources are integrated into a single sensor node. Thus, it is essential to conserve energy at the sensor nodes. In sensor networks, the most expensive operation is communication and hence, communication should be minimized when jointly observing one phenomenon using multiple sensor nodes.

For linear models and uncorrelated measurements, the Kalman filter is the Linear Minimum Mean Squared Error (LMMSE) estimator [1, Ch. 13]. If the sensors perform measurements in a time-synchronized manner and communication is fast enough (or if the nodes wait until all measurements of a time step have been received before performing the next prediction step), it is possible in theory that one or all nodes use a centralized Kalman filter and obtain the optimal result. This is because if any node has access to all measurements in every time step, the distributed nature of the measurements obtained can be completely hidden from the filter and the node can act as if it was equipped with multiple sensors. In practice, however, transmitting all measurements is only feasible if communication is very cheap, which is usually not the case for sensor networks [2].

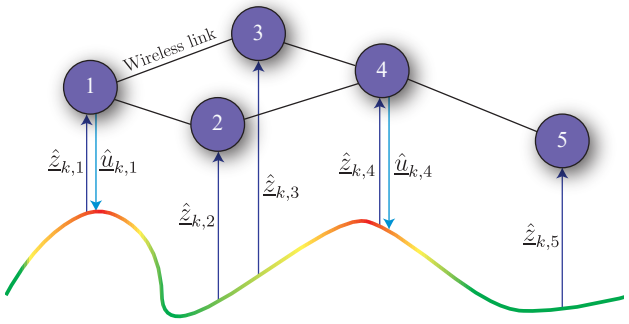

Figure 1. A phenomenon observed by a sensor network. Each node performs measurements at equidistant points in time. The vector $\hat{\underline{u}}_{k,s}$ stands for the input of the node s at time step k while the vector $\underline{z}_{k,s}$ denotes the measurement obtained by the node s at time step k . The nodes only influence the system when deemed necessary.

To save communication, all nodes can act independently and use a local Kalman filter to obtain estimates that are only locally optimal. Then, the question is how the individually obtained posterior estimates can be combined. This is not only a challenge computationally but also semantically as the two posterior densities to be combined both include prior information. Due to the influence of the prior information, the standard Kalman filter formulae should not be used to fuse such estimates even in the absence of prediction steps. A second challenge is that the information that the individual nodes obtain using measurements become correlated during the prediction steps.

The distributed Kalman filter (DKF) [3] can address both of these challenges, given all nodes use the same system model and have full knowledge about the measurement models and the points in time at which measurements are observed. The DKF has been explained in a different form using the information filter form of the Kalman filter in [4]. We propose a similar filter that we call the information form distributed Kalman filter (IDKF) that also allows the use of explicit inputs. We show how explicit inputs can be incorporated in the distributed calculation, given the system model that a centralized Kalman filter would use for the respective scenario. Based on this, we describe how inputs may influence the system so that the scenario is still suited to optimal distributed estimation. We

show that as long as the inputs influence the state linearly and the uncertainties involved do not depend on the inputs, the inputs may be arbitrary and no knowledge about whether a node influences the system via an input is necessary. For example, in the sensor network shown in Fig. 1, the nodes can decide if they wish to influence the system based on the current measurement, their current battery status, their accumulated local data, or (if data from the other nodes are available) based on the globally optimal estimate. An application of such a sensor network could be, e.g., to control the composition of a fluid when the nodes can add chemicals at their respective locations. In addition to the prediction formulae that support explicit inputs, we give different ways to initialize the filter and provide the corresponding semantics.

The paper is structured as follows. Following this introduction, we provide an overview of related work in the second section. In the third section, basics about stochastic models, Kalman filtering, and the information filter are laid out. In the fourth section, we provide the formulae of the IDKF and prove that the prediction and update equations given can be used to obtain the same result that a centralized Kalman filter would obtain. Some properties of the IDKF are provided in the fifth section. In the sixth and last section, we provide a conclusion and an outlook.

II. Related Work

To provide an estimate based on the information of multiple nodes, several approaches have been proposed [5], [6], [7]. There are simple approaches such as using a convex combination of the two estimates with weightings according to the covariances of the estimates. A more sophisticated approach was proposed by Bar-Shalom and Campo [8], which exploits the cross-covariances to fuse estimates. The approach yields a fusion result that corresponds rather to a maximum likelihood estimate [9] than to an LMMSE estimate given all measurements acquired. Furthermore, a wide variety of other approaches exist, such as tracklet fusion algorithms [10], [11], [12] that aim to strip the estimates of the doubly integrated prior information. Another strategy is to keep track of the history [13], [14] and to fuse augmented state representations.

The different approaches come with different prerequisites. While simple combination approaches do not require any further knowledge, the Bar-Shalom–Campo formulae require knowledge about the cross-covariance. Recently, the distributed Kalman filter (DKF) [3] has been proposed, which allows deriving the same result as a centralized Kalman filter. When using the DKF, each node requires a considerable amount of knowledge about the other nodes. Not only do all nodes have to use the same system model, each node also requires full knowledge about the measurement models of all nodes and knowledge about the times at which measurements are performed by the individual nodes.

The insights of the DKF have been developed over the course of multiple publications [3], [4], [15], [16], [17]. The approach is special as it the first to achieve the optimal fusion result for dynamic systems with nonzero process noise without requiring regular communication or transmission of a vector that grows with time. For a sensor network with N nodes, each node only needs to update and keep track of data of the size of the state vector and the covariance matrix. When the data of all nodes are obtained by a node at an arbitrary point in time, the optimal estimate can be retrieved. The DKF does, in general, not support data-dependent extensions such as the EKF. However, this weakness can be alleviated by establishing a hypothesis [18] on the globalized parameters. Deviations from the hypothesis can be counteracted with the aid of specific debiasing techniques [19], [20]. In a similar way, communication uncertainties can be treated [21].

The information form of the DKF presented in [4] is similar to the globalized likelihood version in [3] but the derivation in information filter form allows for a different perspective on the DKF. By regarding the DKF in information form, it can be seen that, in fact, the DKF can be interpreted as an adapted version of a centralized Kalman filter. For the DKF, the equations of the centralized Kalman filter are split up among the individual nodes in a way that the sum of the estimates always yields the optimal estimate as obtained by the centralized Kalman filter.

III. Basics of Kalman Filtering and the Information Filter

In the course of this paper, we show that the formulae of the IDKF yield the same results as a centralized Kalman filter with explicit inputs. However, to benefit from the LMMSE property of the Kalman filter, the usual requirements to use the Kalman filter as an LMMSE estimator have to be met. For the standard Kalman filter to be the LMMSE estimator, we require linear system and measurement models and the noise terms need to be uncorrelated [1, Ch. 13]. To introduce our notation directly in the proper context, we briefly give the formulae for the system evolution and for the generation of measurements as well as the prediction and update step of the Kalman filter in the first subsection of this section. In the second subsection, we introduce the information form of the Kalman filter including a way to incorporate multiple measurements in the same time step.

A. System and Measurement Models and the Kalman Filter

The system evolves according to the system model

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

with the random vector describing the state $\boldsymbol{x}_k$, the system evolution matrix $\mathbf{A}_k$, the system input matrix $\mathbf{B}_k$, the input $\underline{\hat{u}}_k$, and a random vector describing an additive system noise term $\underline{\boldsymbol{w}}_k$. The Kalman filter preserves the LMMSE property by propagating the current mean $\underline{\hat{x}}_k^p$ to the next time step using the system input according to

$$\underline{\hat{x}}_{k+1}^p = \mathbf{A}_k \underline{\hat{x}}_k^e + \mathbf{B}_k \underline{\hat{u}}_k \ . \tag{1}$$

The predicted covariance $\mathbf{C}_{k+1}^p$ is then calculated according to

$$\mathbf{C}_{k+1}^p = \mathbf{A}_k \mathbf{C}_k^e \mathbf{A}_k^T + \mathbf{C}_k^w \ ,$$

depending on the covariance matrix $\mathbf{C}_k^e$ of the posterior at time step k and the covariance matrix $\mathbf{C}_k^w$ of the system noise term $\underline{\boldsymbol{w}}_k$. Note that for the DKF, all nodes have to use the same $\mathbf{A}_k$ and $\mathbf{C}_k^w$ and thus, they must not depend on the current estimate. In distributed scenarios, the models are often assumed to be time-invariant.

The measurement equation is given by

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

with the measurement matrix $\mathbf{H}_k$ and the measurement noise term $\underline{\boldsymbol{v}}_k$. For the update step, the Kalman gain

$$\mathbf{K}_k = \mathbf{C}_k^p \mathbf{H}_k^T (\mathbf{C}_k^v + \mathbf{H}_k \mathbf{C}_k^p \mathbf{H}_k)^{-1}$$

depending on the noise covariance $\mathbf{C}_k^v$ is calculated in each time step. Using this gain and the actual measurement $\underline{\hat{z}}_k$, we obtain the updated mean and covariance according to

$$\begin{aligned} \underline{\hat{x}}_k^e &= \underline{\hat{x}}_k^p + \mathbf{K}_k (\underline{\hat{z}}_k - \mathbf{H}_k \underline{\hat{x}}_k) \ , \\ \mathbf{C}_k^e &= \mathbf{C}_k^p - \mathbf{K}_k \mathbf{H}_k \mathbf{C}_k^p \ . \end{aligned}$$

B. Information Form of the Kalman Filter

In the information form of the Kalman filter [22] (also called information filter), we keep track of an information vector $\underline{\hat{y}}_k$ and an information matrix $\mathbf{Y}_k$. They relate to the original state space mean and covariance according to

$$\underline{\hat{y}}_k = \mathbf{C}_k^{-1} \underline{\hat{x}}_k \quad \text{and} \quad \mathbf{Y}_k = \mathbf{C}_k^{-1} \ .$$

As the covariance matrix is positive definite, the information matrix is invertible and we can thus perform the transformation back to state space by solving the above equations for $\underline{\hat{x}}$ and $\mathbf{C}_k$, respectively.

In the update step of the information filter, a vector $\hat{\underline{j}}_k$ containing the new information of the measurement $\underline{\hat{z}}_k$ is calculated and then fused with the current information about the state via a summation. Thus, the update equations for the information vector are

$$\underline{\hat{y}}_k^e = \underline{\hat{y}}_k^p + \hat{\underline{j}}_k \ , \qquad \hat{\underline{j}}_k = \mathbf{H}_k^T (\mathbf{C}_k^v)^{-1} \underline{\hat{z}}_k \ .$$

In the update equations for the information matrix, a matrix-valued term $\mathbf{J}_k$ to be added to the current information matrix is determined. This leads to the update equations

$$\mathbf{Y}_k^e = \mathbf{Y}_k^p + \mathbf{J}_k \ , \qquad \mathbf{J}_k = \mathbf{H}_k^T (\mathbf{C}_k^v)^{-1} \mathbf{H}_k$$

for the covariance.

If we have N measurements indexed by s, the update step can be performed analogously for all measurements $\underline{\hat{z}}_{k,s}$, measurement noise covariance matrices $\mathbf{C}_{k,s}^v$, and measurement matrices $\mathbf{H}_{k,s}$. This leads to the formulae

$$\underline{\hat{y}}_k^e = \underline{\hat{y}}_k^p + \sum_{s=1}^{N} \hat{\underline{j}}_{k,s} \ , \qquad \hat{\underline{j}}_{k,s} = \mathbf{H}_{k,s}^T (\mathbf{C}_{k,s}^v)^{-1} \underline{\hat{z}}_{k,s} \tag{2}$$

in total for the update of the information vector and

$$\mathbf{Y}_k^e = \mathbf{Y}_k^p + \sum_{s=1}^{N} \mathbf{J}_{k,s} \ , \quad \mathbf{J}_{k,s} = \mathbf{H}_{k,s}^T (\mathbf{C}_{k,s}^v)^{-1} \mathbf{H}_{k,s} \tag{3}$$

for the information matrix.

In the context of distributed estimation, each of the N measurements is generated by a different sensor node. Thus, the measurement covariance matrices $\mathbf{C}_{k,s}^v$ and measurement matrices $\mathbf{H}_{k,s}$ describe the measurement models of the individual sensor nodes. A centralized Kalman filter with all information at its disposal could directly use the formulae (2) and (3). However, to employ these formulae without modification, all measurements would have to be transferred to the node running the centralized Kalman filter.

While much less information needs to be sent for the DKF, all nodes need to know the measurement matrices and measurement covariance matrices of all sensors. Thus, the measurement matrices must not be dependent on the current estimate, such as when using linearization as in the extended Kalman filter.

To perform the prediction step in the information form, the easiest way is to transform the information vector and matrix to state space, perform the prediction as usual, and then transform the result back to information space. This can be performed using

$$\begin{aligned} \underline{\hat{y}}_{k+1}^p &= \mathbf{Y}_{k+1}^p \big(\mathbf{A}_k (\mathbf{Y}_k^e)^{-1} \underline{\hat{y}}_k^e + \mathbf{B}_k \underline{\hat{u}}_k \big) \ , \\ \mathbf{Y}_{k+1}^p &= (\mathbf{A}_k (\mathbf{Y}_k^e)^{-1} \mathbf{A}_k^T + \mathbf{C}_k^w)^{-1} \ . \end{aligned}$$

IV. Information Form of the Distributed Kalman Filter

The key feature of an energy-efficient distributed filter is that the nodes need to be able to accumulate information about the measurements in a way that does not require communication. In the following, we denote the information accumulated by the node s by $\underline{\hat{y}}_{k,s}$. In the IDKF, we ensure that the globally optimal information vector $\underline{\hat{y}}_{k,\text{glob}}$ as obtained by a centralized Kalman filter in information form can always be computed via the sum

$$\underline{\hat{y}}_{k,\text{glob}} = \sum_{s=1}^{N} \underline{\hat{y}}_{k,s} \ . \tag{4}$$

In the following, we provide formulae to include measurements in the information vector $\underline{\hat{y}}_{k,s}$ and perform prediction steps considering explicit inputs while maintaining this property. We provide a proof by induction, starting with the initialization in Sec. IV-A as the base case. For the induction step, we show that this property is preserved throughout update and prediction steps in Sec. IV-B and Sec. IV-C, respectively.

A. Initialization of the IDKF

As a Bayesian estimator, the Kalman filter is initialized using a prior distribution. In [3], it is suggested to initialize the filters in the individual nodes using the first measurement of the first sensor, which can be seen as taking an empirical Bayes [23, Ch. 4.1] approach. Since the formulae in [3] are given in state space, it is necessary that the state is fully observable from the first measurement as the measurement matrix is not invertible otherwise. The initialization of the DKF in information form [4] does not require this as the information vectors and matrices are initialized using zero vectors and zero matrices. Within the

framework of Bayesian estimation, this strategy corresponds to the use of an uninformative prior [23, Ch. 4.1].

In the following, we assume that precisely one prior distribution is given. For the filter results of the IDKF to be equal to those obtained using a centralized Kalman filter, the prior used by the IDKF must be the same that would be used in the centralized Kalman filter. The parameters required by the Kalman filter are the mean $\hat{\underline{x}}_0^p$ and the covariance $\mathbf{C}_0^p$ of the prior distribution. The initial information vector $\hat{\underline{y}}_{0,\text{glob}}^p$ and information matrix $\mathbf{Y}_{0,\text{glob}}^p$ of the centralized Kalman filter that is globally optimal can then be calculated according to

$$\hat{\underline{y}}_{0,\text{glob}}^p = (\mathbf{C}_0^p)^{-1}\hat{\underline{x}}_0^p \quad \text{and} \quad \mathbf{Y}_{0,\text{glob}}^p = (\mathbf{C}_0^p)^{-1} \ .$$

Now, the information vector is to be distributed among all nodes in such a way that (4) holds. While there are multiple possibilities to distribute the information, there are two ways that are intuitive. The first is to distribute the information evenly, meaning the information vector

$$\hat{\underline{y}}_{0,s}^p = \frac{1}{N}\hat{\underline{y}}_{0,\text{glob}}^p$$

is used for all of the N nodes. This subdivision trivially ensures that (4) holds. We do not intend the vector $\hat{\underline{y}}_{k,s}$ to have any special semantics aside that the sum of all yields the globally optimal information vector. We do not assign a compatible information matrix to this information vector and all nodes instead directly keep track of $\mathbf{Y}_{k,\text{glob}}$ and initialize their covariance matrices using $\mathbf{Y}_{0,\text{glob}}^p$. It is important to note that an individual $\hat{\underline{y}}_{0,s}^p$ combined with $\mathbf{Y}_{0,\text{glob}}^p$ does not represent the prior distribution. Only by combining the sum of all $\hat{\underline{y}}_{0,s}^p$ with the information matrix $\mathbf{Y}_{0,\text{glob}}^p$, we can retrieve the prior distribution.

The second approach to distribute the information vector $\hat{\underline{y}}_{0,\text{glob}}^p$ is to select one specific node with index m that keeps track of the prior distribution. Then, we can set

$$\hat{\underline{y}}_{0,m}^p = \hat{\underline{y}}_{0,\text{glob}}^p \ ,$$

while all other information vectors are initialized as zero vectors. Again, all nodes keep track of $\mathbf{Y}_{k,\text{glob}}$. Semantically, the node m has the valid prior distribution described by the information vector $\hat{\underline{y}}_{0,m}^p$ and the information matrix $\mathbf{Y}_{0,\text{glob}}^p$ at its disposal, while all other nodes have no information at the time of the initialization. In all $\hat{\underline{y}}_{k,s}^p$ with $s \neq m$, only information derived from the measurements is accumulated, whereas $\hat{\underline{y}}_{0,m}^p$ is used to keep track of both the prior information and the measurements obtained by the node with index m. While both approaches allow an arbitrary node to derive the optimal estimate when the information of all nodes is summed up, the latter approach especially makes sense if the node m is particularly important, e.g., when it is the only node that requires the globally optimal estimate.

B. Update Step of the IDKF

The update step can be performed in an efficient manner by simply adding the information contained in the new measurement of the respective sensor, as shown in the following theorem.

Theorem 1. *Using the update equations*

$$\hat{\underline{y}}_{k,s}^e = \hat{\underline{y}}_{k,s}^p + \hat{\underline{j}}_{k,s} \ , \qquad \hat{\underline{j}}_{k,s} = \mathbf{H}_{k,s}^T(\mathbf{C}_{k,s}^v)^{-1}\hat{\underline{z}}_{k,s} \ , \tag{5}$$

it can be ensured that (4) holds after the update step. Meaning, the sum of the new vectors $\hat{\underline{y}}_{k,s}^e$ is equivalent to the result $\hat{\underline{y}}_{k,glob}^e$ of the centralized Kalman filter, given that (4) holds for the $\hat{\underline{y}}_{k,s}^p$.

Proof: According to (2), the centralized Kalman filter having $\hat{\underline{y}}_{k,\text{glob}}^p$ at its disposal can perform its update step via

$$\hat{\underline{y}}_{k,\text{glob}}^e = \hat{\underline{y}}_{k,\text{glob}}^p + \sum_{s=1}^{N} \hat{\underline{j}}_{k,s} \ .$$

Using that (4) holds for the $\hat{\underline{y}}_{k,s}^p$, we rewrite this as

$$\begin{aligned}\hat{\underline{y}}_{k,\text{glob}}^e &= \sum_{s=1}^{N} \hat{\underline{y}}_{k,s}^p + \sum_{s=1}^{N} \hat{\underline{j}}_{k,s} \\ &= \sum_{s=1}^{N} \left(\hat{\underline{y}}_{k,s}^p + \hat{\underline{j}}_{k,s}\right) \\ &= \sum_{s=1}^{N} \hat{\underline{y}}_{k,s}^e \ ,\end{aligned}$$

which concludes our proof. ∎

To be able to perform future prediction steps and provide the state space estimate when the resuls are fused, we also have to update $\mathbf{Y}_{k,\text{glob}}$ that each node keeps track of. According to our assumptions, every node has full knowledge about all $\mathbf{H}_{k,s}$ and $\mathbf{C}_{k,s}^v$. Thus, each node can derive $\mathbf{Y}_{k,\text{glob}}^e$ from $\mathbf{Y}_{k,\text{glob}}^p$ just like in a centralized Kalman filter according to

$$\mathbf{Y}_{k,\text{glob}}^e = \mathbf{Y}_{k,\text{glob}}^p + \sum_{s=1}^{N} \mathbf{J}_{k,s} \text{ with } \mathbf{J}_{k,s} = \mathbf{H}_{k,s}^T(\mathbf{C}_{k,s}^v)^{-1}\mathbf{H}_{k,s} \ .$$

C. Prediction Step of the IDKF

In the versions of the DKF proposed in [3] and [4], inputs are only supported implicitly. Instead of using the prediction step as given in (1), the prediction equation

$$\hat{\underline{x}}_{k+1}^p = \mathbf{A}_k\hat{\underline{x}}_k^e$$

is employed. As our matrix $\mathbf{A}_k$ must not be data-dependent, information about the inputs must be contained in the vector $\hat{\underline{x}}_k^e$, which potentially leads to larger vectors. In practice, the computational effort of the Kalman filter scales approximately cubically in the size of the state vector $\hat{\underline{x}}$. Thus, implicitly modeling the input may result in a higher computational effort.

For example, let us assume we observe a phenomenon described by a state vector with four components and we have 100 nodes using an input vector with four components each. Then, directly encoding all inputs in a model for the centralized Kalman filter (which is required by the DKF) would result in a state vector with 404 entries. For the IDKF, we describe

how inputs can be integrated explicitly in the prediction step, without a need to encode the input in the state vector.

As the IDKF is to yield the same results as a centralized Kalman filter, we use the prediction step of the regular Kalman filter (1) as the basis for our considerations. We now assume that the (global) input vector $\underline{\hat{u}}_{k,\text{glob}}$ can be written as a linear combination of the inputs at the individual nodes. In this case, the prediction step of the centralized Kalman filter

$$\underline{\hat{x}}^p_{k+1,\text{glob}} = \mathbf{A}_k \underline{\hat{x}}^e_{k,\text{glob}} + \mathbf{B}_{k,\text{glob}} \underline{\hat{u}}_{k,\text{glob}}$$

can be rewritten as a prediction step depending on all individual input vectors $\underline{\hat{u}}_{k,s}$ of the N nodes according to

$$\underline{\hat{x}}^p_{k+1,\text{glob}} = \mathbf{A}_k \underline{\hat{x}}^e_{k,\text{glob}} + \sum_{s=1}^{N} \mathbf{B}_{k,s} \underline{\hat{u}}_{k,s} \ . \tag{6}$$

Since each node generally only knows its own input vector, only the vector $\underline{\hat{u}}_{k,s}$ may be used for the prediction step of the node s.

As an essential part of our prediction step, we require the matrices $\mathbf{A}_k$ and $\mathbf{C}^w_k$ such that the evolution of the covariance according to

$$\mathbf{C}^p_{k+1,\text{glob}} = \mathbf{A}_k \mathbf{C}^e_{k,\text{glob}} \mathbf{A}^T_k + \mathbf{C}^w_k$$

properly reflects the evolution of the uncertainty, independent of the inputs at the individual sensors. As the matrices $\mathbf{A}_k$ and $\mathbf{C}^w_k$ would also be required for a centralized Kalman filter, this is no requirement stemming directly from the use of the IDKF. An important difference in a centralized Kalman filter is that $\mathbf{A}_k$ and $\mathbf{C}^w_k$ could be easily made dependent on the actual inputs or the current estimate.

It should be noted that we do not add any additional requirements to those of the regular DKF without explicit modeling of the inputs. First, the DKF requires that the system matrices $\mathbf{A}_k$ and the system noise covariance matrices $\mathbf{C}^w_k$ are known by all nodes, which leads to the requirement that they must not change depending on the input (or else, information about the change would have to be sent to all nodes). Second, the assumption that it must be possible to write the combined input $\underline{\hat{u}}_{k,\text{glob}}$ as a linear combination of the inputs at the individual sensors also applies when modeling the inputs implicitly. If an input is encoded in the state vector, then the input may only affect the system linearly according to the system matrix $\mathbf{A}_k$. Nonlinear combinations would only be possible if $\mathbf{A}_k$ depended on the actual inputs, which is not allowed when using the DKF.

Lemma 2. *Using*

$$\underline{\hat{y}}^p_{k+1,s} = \mathbf{Y}^p_{k+1,\text{glob}} \big(\mathbf{A}_k (\mathbf{Y}^e_{k,\text{glob}})^{-1} \underline{\hat{y}}^e_{k,s} + \mathbf{B}_{k,s} \underline{\hat{u}}_{k,s} \big) \ ,$$

we can ensure that property (4) holds after the prediction step. This means that (4) holds for the $\underline{\hat{y}}^p_{k+1,s}$, given that the condition is fulfilled for the $\underline{\hat{y}}^e_{k,s}$. Since each node only requires its own input $\underline{\hat{u}}_{k,s}$, it is possible to use the formula in a distributed manner.

Proof: First, we need to be able to calculate $\mathbf{Y}^p_{k+1,\text{glob}}$. As described in Sec. IV-B, we have $\mathbf{Y}^e_{k,\text{glob}}$ at our disposal from the previous update step. As required by the assumptions of the IDKF, all matrices needed to update the covariance, namely $\mathbf{A}_k$ and $\mathbf{C}^w_k$, are available at all nodes. Thus, each node can calculate $\mathbf{Y}^p_{k+1,\text{glob}}$ from $\mathbf{Y}^e_{k,\text{glob}}$ according to

$$\mathbf{Y}^p_{k+1,\text{glob}} = \big(\mathbf{A}_k (\mathbf{Y}^e_{k,\text{glob}})^{-1} \mathbf{A}^T_k + \mathbf{C}^w_{k,} \big)^{-1} \ .$$

As the main part of the proof, we simply perform algebraic reformulations and use our base case. We can write

$$\begin{aligned}
\underline{\hat{y}}^p_{k+1,\text{glob}} &= \mathbf{Y}^p_{k+1,\text{glob}} \, \underline{\hat{x}}^p_{k+1,\text{glob}} \\
&= \mathbf{Y}^p_{k+1,\text{glob}} \left(\mathbf{A}_k \underline{\hat{x}}^e_{k,\text{glob}} + \sum_{s=1}^{N} \mathbf{B}_{k,s} \underline{\hat{u}}_{k,s} \right) \\
&= \mathbf{Y}^p_{k+1,\text{glob}} \left(\mathbf{A}_k (\mathbf{Y}^e_{k,\text{glob}})^{-1} \underline{\hat{y}}^e_{k,\text{glob}} + \sum_{s=1}^{N} \mathbf{B}_{k,s} \underline{\hat{u}}_{k,s} \right)
\end{aligned}$$

and use the base case to obtain

$$\begin{aligned}
\underline{\hat{y}}^p_{k+1,\text{glob}} &= \mathbf{Y}^p_{k+1,\text{glob}} \left(\mathbf{A}_k (\mathbf{Y}^e_{k,\text{glob}})^{-1} \sum_{s=1}^{N} \underline{\hat{y}}^e_{k,s} + \sum_{s=1}^{N} \mathbf{B}_{k,s} \underline{\hat{u}}_{k,s} \right) \\
&= \mathbf{Y}^p_{k+1,\text{glob}} \left(\sum_{s=1}^{N} \mathbf{A}_k (\mathbf{Y}^e_{k,\text{glob}})^{-1} \underline{\hat{y}}^e_{k,s} + \sum_{s=1}^{N} \mathbf{B}_{k,s} \underline{\hat{u}}_{k,s} \right) \\
&= \sum_{s=1}^{N} \left(\mathbf{Y}^p_{k+1,\text{glob}} \big(\mathbf{A}_k (\mathbf{Y}^e_{k,\text{glob}})^{-1} \underline{\hat{y}}^e_{k,s} + \mathbf{B}_{k,s} \underline{\hat{u}}_{k,s} \big) \right) \\
&= \sum_{s=1}^{N} \underline{\hat{y}}^p_{k+1,s} \ .
\end{aligned}$$

Thus, we have proven that the property (4) holds for the resulting $\underline{\hat{y}}^p_{k+1,s}$. ■

V. Properties of the IDKF

Now that we have shown how both the update and prediction steps can be performed while maintaining the property (4), both parts of the induction step have been proven. Thus, we have proven that when performing the initialization as well as prediction and update steps according to the IDKF formulae, the optimal information vector as calculated by a centralized Kalman filter $\underline{\hat{y}}_{k,\text{glob}}$ can be obtained using the sum of the individual information vectors $\underline{\hat{y}}_{k,s}$ of all sensors, as illustrated in Table I. Thus, even if no communication has occurred before, the globally optimal estimate $\underline{\hat{x}}_{k,\text{glob}}$ can be derived at an arbitrary time step when a node has received all $\underline{\hat{y}}_{k,s}$. The essential formulae to use the IDKF in practice are summed up in Table II.

An important insight that is facilitated by the novel prediction formulae explicitly respecting the system inputs is that it must be possible to write the total effect of all inputs as a linear combination of the effects of the inputs at the individual nodes. Furthermore, how they are combined must not depend on the actual inputs. However, as long as the system covariance matrix $\mathbf{C}^w_k$ is not influenced by the input of a node, the node is free

Table I

TABLE SHOWING THE EVOLUTION OF THE INDIVIDUAL $\hat{\underline{y}}_{k,s}$ AT ALL SENSOR NODES IF EACH NODE KEEPS TRACK OF A FRACTION OF THE PRIOR. THE SUM ALWAYS YIELDS THE GLOBALLY OPTIMAL INFORMATION VECTOR THAT CAN BE USED TO DERIVE THE ESTIMATE THAT A CENTRALIZED KALMAN FILTER WOULD PROVIDE.

Step	Initialization	Filter	Prediction	Filter	Prediction	Filter	$\cdots$
Node 1	$\frac{1}{N}\hat{\underline{y}}^p_{0,\text{glob}}$	$\hat{\underline{y}}^e_{0,1}$	$\hat{\underline{y}}^p_{1,1}$	$\hat{\underline{y}}^e_{1,1}$	$\hat{\underline{y}}^p_{2,1}$	$\hat{\underline{y}}^e_{2,1}$	$\cdots$
Node 2	$\frac{1}{N}\hat{\underline{y}}^p_{0,\text{glob}}$	$\hat{\underline{y}}^e_{0,2}$	$\hat{\underline{y}}^p_{1,2}$	$\hat{\underline{y}}^e_{1,2}$	$\hat{\underline{y}}^p_{2,2}$	$\hat{\underline{y}}^e_{2,2}$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$
Node N	$\frac{1}{N}\hat{\underline{y}}^p_{0,\text{glob}}$	$\hat{\underline{y}}^e_{0,N}$	$\hat{\underline{y}}^p_{1,N}$	$\hat{\underline{y}}^e_{1,N}$	$\hat{\underline{y}}^p_{2,N}$	$\hat{\underline{y}}^e_{2,N}$	$\cdots$
Σ	$\hat{\underline{y}}^p_{0,\text{glob}}$	$\hat{\underline{y}}^e_{0,\text{glob}}$	$\hat{\underline{y}}^p_{1,\text{glob}}$	$\hat{\underline{y}}^e_{1,\text{glob}}$	$\hat{\underline{y}}^p_{2,\text{glob}}$	$\hat{\underline{y}}^e_{2,\text{glob}}$	$\cdots$

Table II

TABLE SUMMARIZING THE FORMULAE FOR THE ESSENTIAL OPERATIONS OF THE IDKF WHEN EACH NODE KEEPS TRACK OF A FRACTION OF THE PRIOR.

	Information vector	Information matrix
Initialization	$\hat{\underline{y}}^p_{0,s} = \frac{1}{n}(\mathbf{C}^p_0)^{-1}\hat{\underline{x}}^p_0$	$\mathbf{Y}^p_{0,\text{glob}} = (\mathbf{C}^p_0)^{-1}$
Filter step	$\hat{\underline{y}}^e_{k,s} = \hat{\underline{y}}^p_{k,s} + \mathbf{H}^T_{k,s}(\mathbf{C}^v_{k,s})^{-1}\hat{\underline{z}}_{k,s}$	$\mathbf{Y}^e_{k,\text{glob}} = \mathbf{Y}^p_{k,\text{glob}} + \sum_{s=1}^N \mathbf{H}^T_{k,s}(\mathbf{C}^v_{k,s})^{-1}\mathbf{H}_{k,s}$
Prediction step	$\hat{\underline{y}}^p_{k+1,s} = \mathbf{Y}^p_{k+1,\text{glob}}\big(\mathbf{A}_k(\mathbf{Y}^e_{k,\text{glob}})^{-1}\hat{\underline{y}}^e_{k,s} + \mathbf{B}_{k,s}\hat{\underline{u}}_{k,s}\big)$	$\mathbf{Y}^p_{k+1,\text{glob}} = \big(\mathbf{A}_k(\mathbf{Y}^e_{k,\text{glob}})^{-1}\mathbf{A}^T_k + \mathbf{C}^w_{k,}\big)^{-1}$
Deriving optimal estimate	$\hat{\underline{x}}_k = \mathbf{Y}^{-1}_{k,\text{glob}}\sum_{s=1}^N \hat{\underline{y}}^e_{k,s}$	$\mathbf{C}_{k,\text{glob}} = \mathbf{Y}^{-1}_{k,\text{glob}}$

to use arbitrary inputs without voiding the optimality of the distributed estimation. Thus, even when independent controllers are running on the individual nodes, it is possible to obtain the optimal estimate if all knowledge for the application of the IDKF is known by all nodes.

Moreover, there are a variety of other properties of the DKF that are evident from the IDKF formulation. First, the IDKF formulae cannot be used to directly obtain locally optimal estimates. However, if we have local system models that can be used for the individual nodes (note that the IDKF only requires the model a centralized Kalman filter would use), then it is easy to obtain the locally optimal result by simply letting a second locally optimal Kalman filter run beside the IDKF. Using compatible system models, the local Kalman filters can be reinitialized using the globally optimal information whenever it can be derived by the respective node. When using a local controller, the output of the local Kalman filter could even be used to determine an appropriate input without voiding the optimality of the estimation result obtained via the IDKF formulae.

Using the equivalence of the results of the IDKF to those of a centralized Kalman filter, we can state a second useful property. The IDKF inherits all properties of the Kalman filter and is thus the LMMSE estimator when the noise terms are uncorrelated and the MMSE estimator when the state and the noise are jointly Gaussian distributed [1, Ch. 13].

Another interesting property is how the combination of the information represented by the $\hat{\underline{y}}_{k,s}$ can be performed in state space. As the optimal mean in state space can be obtained via

$$\hat{\underline{x}}_{k,\text{glob}} = (\mathbf{Y}_{k,\text{glob}})^{-1}\hat{\underline{y}}_{k,\text{glob}} = \sum_{s=1}^{N}(\mathbf{Y}_{k,\text{glob}})^{-1}\hat{\underline{y}}_{k,s} \ ,$$

we can use the state space vector

$$\hat{\underline{x}}_{k,s} = (\mathbf{Y}_{k,\text{glob}})^{-1}\hat{\underline{y}}_{k,s}$$

to represent the information that is contained in the information vector $\hat{\underline{y}}_{k,s}$ of the respective node in state space. It is also possible to transmit this vector instead of $\hat{\underline{y}}_{k,s}$. As

$$\hat{\underline{x}}_{k,\text{glob}} = \sum_{s=1}^{N}(\mathbf{Y}_{k,\text{glob}})^{-1}\hat{\underline{y}}_{k,s} = \sum_{s=1}^{N}\hat{\underline{x}}_{k,s}$$

holds, the vectors must also be combined using a sum in state space, which may seem a bit counterintuitive.

The fourth interesting property is that no designated fusion center is required. Any node that obtains all $\hat{\underline{y}}_{k,s}$ can produce the information vector representing the globally optimal estimate. Furthermore, if the information is transferred along multiple hops and no prediction and update steps are performed before the end of the transmissions, individual nodes can accumulate information using a summation before sending it to the next node. Obviously, it must be ensured that no information vector $\hat{\underline{y}}_{k,s}$ is added twice in the sum. This needs to be enforced, e.g., by also sending the IDs of all sensors whose information is included in the transmitted information vector. As can be seen in the following example, this property can be used to significantly reduce the communication overhead.

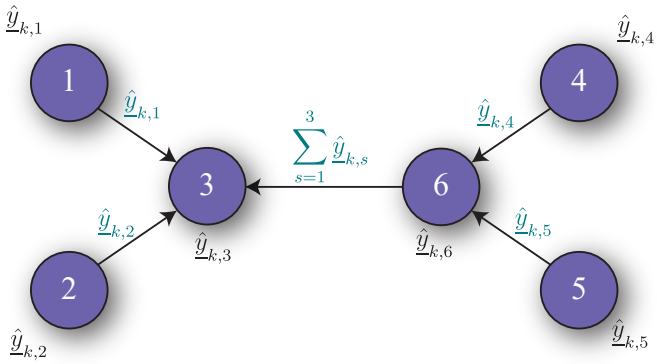

Figure 2. Sensor network in the example. The IDs of the nodes are shown in white. The information that each node keeps track of is marked in black at the node. To derive the globally optimal estimate at node 3, only the five vectors indicated in blue at the edges need to be transferred.

Example 3. In our example, we have a sensor network with six nodes as shown in Fig. 2. Let us assume no update and prediction steps are performed during the following communication events. First, the nodes with the IDs 1 and 2 send their $\hat{\underline{y}}_{k,s}$ to the node 3. Likewise, the nodes 4 and 5 send their information to the node with ID 6. Up to now, a total of four vectors have been transferred. Afterward, if the node 3 sends the sum of its own information and the information obtained from the nodes 1 and 2 to the node 6, then the node 6 is able to derive the globally optimal result. Alternatively, the node 6 could send its own information and the information of the nodes 4 and 5 to the node 3. Then, the node 3 is able to derive the globally optimal estimate. Thus, for $N = 6$ sensor nodes, only $N - 1 = 5$ vectors need to be transferred to obtain the globally optimal result at node 3 or 6. By transferring 5 vectors more, the optimal result can be distributed in the entire network (less communication is required if multicasts or broadcasts can be sent). If the network topology is known, accumulation of the information can be used in real-world applications, e.g., by ensuring that the node 3 delays the transmission of its information until the information of the nodes 1 and 2 have been received. Regardless of how the information vectors are transferred, there is never a need to transfer any information matrices as all nodes keep track of the same matrix $\mathbf{Y}_{k,\mathrm{glob}}$.

For a known topology and fully reliable links, the IDKF can be combined with minimum spanning trees to allow a single node to derive the optimal estimate at the lowest costs possible. To achieve this minimal cost, each node has to know from which nodes it will receive data and to which node the data needs to be transferred. As the optimal estimate can be distributed in the whole network at the same cost, all nodes can obtain the optimal estimate at twice the cost required to attain the optimal estimate at one node.

It should be noted that while this approach yields the cheapest way to obtain the optimal estimate, it is generally not the fastest if the time is measured, e.g., in hops. Furthermore, if the links are unreliable, the transmission rules may cease to work and the accumulated data may not reach the desired node. When dealing with unreliable links or if only simple transmission rules can be used by the individual nodes, other approaches such as distributed measurement fusion [24] may be more suitable to the scenario at hand.

VI. Conclusion

In this paper, we have provided a variant of the DKF in information form called the IDKF that allows modeling inputs explicitly. Explicit inputs can improve the comprehensibility of the models and lower computational efforts. Semantically, the IDKF works like a centralized Kalman filter, just with summands distributed among the sensor nodes. The optimal result as obtained by a centralized Kalman filter can be attained using a simple summation of the terms distributed among the nodes. Using induction, we have shown that this property can be maintained when using the formulae provided to perform update steps and prediction steps with explicit inputs. By taking a close look at explicit and implicit inputs in the context of the DKF, we have derived properties that need to be fulfilled for the inputs to be compatible with the DKF framework. Finally, we have presented further interesting properties of the DKF that are naturally obtained using the IDKF formulation.

An important area of future research is to investigate how the idea of the distributed Kalman filter can be applied, at least to a certain degree, to nonlinear filtering and data-dependent algorithms [25]. Another area of future work is to analyze how the DKF can be used to save communication in networks with broadcast functionality or unreliable links. Future work will entail to take a closer look at generalizations, such as in the field of combined stochastic and set-membership state estimation [26], [27]. First work to integrate set-membership uncertainties in a distributed filter that minimizes the stochastic uncertainty is given in [28]. In future work, we plan to work on improving criteria that respect both stochastic and set-membership uncertainties (as has been regarded in [29] for centralized applications) in a distributed filter.

Acknowledgment

The IGF project 18798 N of the research association Forschungs-Gesellschaft Verfahrens-Technik e.V. (GVT) was supported via the AiF in a program to promote the Industrial Community Research and Development (IGF) by the Federal Ministry for Economic Affairs and Energy on the basis of a resolution of the German Bundestag. This work was also supported by the German Research Foundation (DFG) under grant NO 1133/1-1.

References

[1] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 1993.

[2] D. Ciuonzo, G. Papa, G. Romano, P. S. Rossi, and P. Willett, "One-Bit Decentralized Detection with a Rao Test for Multisensor Fusion," *IEEE Signal Processing Letters*, vol. 20, no. 9, pp. 861–864, 2013.

[3] F. Govaers and W. Koch, "An Exact Solution to Track-to-Track-Fusion at Arbitrary Communication Rates," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 2718–2729, 2012.

[4] C.-Y. Chong and S. Mori, "Optimal Fusion for Non-Zero Process Noise," in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*. IEEE, 2013, pp. 365–371.

[5] S. Mori, W. H. Barker, C.-Y. Chong, and K.-C. Chang, "Track Association and Track Fusion with Nondeterministic Target Dynamics," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 2, pp. 659–668, 2002.

[6] D. L. Hall, C.-Y. Chong, J. Llinas, and M. E. Liggins II, Eds., *Distributed Data Fusion for Network-Centric Operations*, ser. The Electrical Engineering and Applied Signal Processing Series. CRC Press, 2013.

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

[8] 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.

[9] K.-C. Chang, R. K. Saha, and Y. Bar-Shalom, "On Optimal Track-to-Track Fusion," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 4, pp. 1271–1276, 1997.

[10] O. E. Drummond, "Tracklets and a Hybrid Fusion With Process Noise," in *Proceedings of SPIE, Signal and Data Processing of Small Targets*, vol. 3163, 1997.

[11] X. Tian and Y. Bar-Shalom, "Exact Algorithms for Four Track-to-Track Fusion Configurations: All you Wanted to Know but Were Afraid to Ask," in *Proceedings of the 12th International Conference on Information Fusion (Fusion 2009)*, Seattle, Washington, USA, Jul. 2009.

[12] C.-Y. Chong, S. Mori, F. Govaers, and W. Koch, "Comparison of Tracklet Fusion and Distributed Kalman Filter for Track Fusion," in *Proceedings of the 17th International Conference on Information Fusion (Fusion 2014)*, Salamanca, Spain, Jul. 2014.

[13] T. Bailey, M. Bryson, H. Mu, J. Vial, L. McCalman, and H. Durrant-Whyte, "Decentralised Cooperative Localisation for Heterogeneous Teams of Mobile Robots," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA 2011)*, Shanghai, China, May 2011.

[14] W. Koch and F. Govaers, "On Accumulated State Densities with Applications to Out-of-Sequence Measurement Processing," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 4, pp. 2766–2778, Oct. 2011.

[15] W. Koch, "On Optimal Distributed Kalman Filtering and Retrodiction at Arbitrary Communication Rates for Maneuvering Targets," in *2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Aug 2008, pp. 457–462.

[16] ——, "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)*, Jul. 2009, pp. 2209–2216.

[17] 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.

[18] 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.

[19] ——, "Advances in Hypothesizing Distributed Kalman Filtering," in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, Jul. 2013.

[20] F. Govaers, A. Charlish, and W. Koch, "Covariance Debiasing for the Distributed Kalman Filter," in *2013 16th International Conference on Information Fusion (FUSION 2013)*. IEEE, 2013, pp. 61–67.

[21] M. Reinhardt, B. Noack, S. Kulkarni, and U. D. Hanebeck, "Distributed Kalman Filtering in the Presence of Packet Delays and Losses," in *Proceedings of the 17th International Conference on Information Fusion (Fusion 2014)*, Salamanca, Spain, Jul. 2014.

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

[23] E. L. Lehmann and G. Casella, *Theory of Point Estimation*, 2nd ed. Springer Science & Business Media, 2006.

[24] C. Y. Chong, K. C. Chang, and S. Mori, "Comparison of Optimal Distributed Estimation and Consensus Filtering," in *19th International Conference on Information Fusion (Fusion 2016)*, Jul. 2016, pp. 1034–1041.

[25] C.-Y. Chong and S. Mori, "Graphical Models for Nonlinear Distributed Estimation," in *Proceedings of the 7th International Conference on Information Fusion (Fusion 2004)*, Stockholm, Sweden, Jun. 2004.

[26] B. Noack, V. Klumpp, and U. D. Hanebeck, "State Estimation with Sets of Densities Considering Stochastic and Systematic Errors," in *Proceedings of the 12th International Conference on Information Fusion (Fusion 2009)*, Seattle, Washington, USA, Jul. 2009.

[27] B. Noack, F. Pfaff, and U. D. Hanebeck, "Combined Stochastic and Set-Membership Information Filtering in Multisensor Systems," in *Proceedings of the 15th International Conference on Information Fusion (Fusion 2012)*, Singapore, Jul. 2012.

[28] F. Pfaff, B. Noack, and U. D. Hanebeck, "Optimal Distributed Combined Stochastic and Set-Membership State Estimation," in *Proceedings of the 20th International Conference on Information Fusion (Fusion 2017)*, Xi'an, China, Jul. 2017.

[29] B. Noack, F. Pfaff, and U. D. Hanebeck, "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, Hawaii, USA, Dec. 2012.