

Fully Decentralized Estimation Using Square-Root Decompositions

Susanne Radtke, Benjamin Noack, and Uwe D. Hanebeck

Intelligent Sensor-Actuator-Systems Laboratory (ISAS)

Institute for Anthropomatics and Robotics

Karlsruhe Institute of Technology (KIT), Germany

susanne.radtke@kit.edu, benjamin.noack@ieee.org, uwe.hanebeck@ieee.org

Abstract—Networks consisting of several spatially distributed sensor nodes are useful in many applications. While distributed processing of information can be more robust and flexible than centralized filtering, it requires careful consideration of dependencies between local state estimates. This paper proposes an algorithm to keep track of dependencies in decentralized systems where no dedicated fusion center is present. Specifically, it addresses double counting of measurement information due to intermediate fusion results as well as correlations due to common process noise and common prior information. To limit the necessary amount of data, this paper introduces a method to bound correlations partially, leading to a more conservative fusion result while reducing the necessary amount of data. Simulation studies compare the performance and convergence rate of the proposed algorithm to other state-of-the-art methods.

Index Terms—Decentralized estimation, data fusion, sensor networks.

I. INTRODUCTION

Sensor networks consist of several spatially distributed sensor nodes that can cooperatively perform a variety of different tasks [1], e.g., tracking a moving target using a network of cameras. While centralized processing of measurements can be done optimally, network topology and communication bandwidth often forbid processing measurements in a central processing unit, since nodes are only able to communicate with their closest neighbors. Distributed estimation allows the processing of measurements in a local processing unit. This local information is then communicated and fused with information from neighboring sensor nodes. It has been shown that distributed processing of sensor data can be more robust, flexible, and scalable [2], but it introduces dependencies that need to be addressed carefully to ensure consistent fusion results.

Within the past forty years, many algorithms [3] have been proposed to address the problems arising in distributed estimation. This includes using the information form of the Kalman filter [4]–[6] or formulating an optimally distributed Kalman filter [7]–[9]. Since these algorithms require to continuously communicate between sensor nodes, other approaches propose to use local Kalman filters and fuse their respective state estimates. Several publications address the correlations due to common process noise and common prior information [10]–[13]. When neglecting dependencies [14], fused estimates tend to become inconsistent as the uncertainty is underestimated. Covariance intersection [15]–[17] aims to find a conservative

fusion rule to always ensure consistent results. As these are often too conservative, other approaches try to find closer bounds, e.g., inverse covariance intersection [18], [19]. Specifically for different network topologies, other algorithms such as the channel filter [2], the information graph-approach [20] or the information matrix fusion [21], [22] were proposed.

Another class of algorithms aims to converge to a global estimate by iteratively exchanging information between neighboring nodes. Prominent representatives include consensus on measurements [23], consensus on information [24], [25], or hybrid approaches [26], [27]. Consensus methods can be regarded as a suboptimal fusion rule [28] where the averaging of the information does not represent the actual information in the network and does also not consider redundant information systematically. For simpler network topologies, several approaches trying to reconstruct the cross-covariance matrix between state estimates using ensembles, e.g., the common past invariant Ensemble KF (CPI-EnKF) [29], or using samples [30]–[32] have been proposed. Furthermore, a reconstruction of cross-covariance matrices using square-root decompositions was proposed by [33], [34]. The reconstruction of cross-covariances has advantageous properties as it allows for the design of consistent fusion methods that are generally more accurate and do not over- or underestimate the uncertainty. Yet, it requires the communication of additional information between sensor nodes leading to a trade-off between optimality and network capacity.

The square-root decomposition as initially proposed in [34] considers fusion in network topologies with only one dedicated fusion center. In this paper, we apply the decompositions to decentralized estimation tasks, where each node may sporadically serve as a fusion center. Nodes can exchange their estimates and fuse their local estimates with the received information. For this purpose, each node must keep track of correlations during its local processing steps. Not only common process needs to be encoded in the square-root decompositions, but also double counting of information poses a problem in decentralized network topologies and needs to be tracked. Due to the storage requirements and communication load associated with the square-root decompositions, the nodes can reach a compromise between fusion quality and resource demands by introducing partial bounds on the correlations.

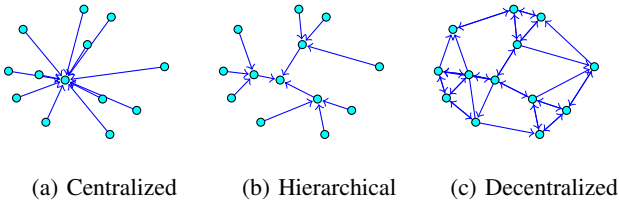


Fig. 1: Different network topologies.

The paper is structured as follows. We first revisit the problem of fusing several state estimates in a central fusion node and introduce the square-root decomposition method to reconstruct the cross-covariance matrix between estimates Sec. II. Decentralized network topologies in the absence of a dedicated fusion center are studied in Sec. III. The evaluation in Sec. IV studies different scenarios and also provides a comparison with consensus methods. Sec. VI concludes the paper.

II. PROBLEM FORMULATION

We consider a discrete-time linear time-variant stochastic dynamic system with time index k and state transition matrix \mathbf{A}_k , state vector \mathbf{x}_k of state dimension N , and zero-mean white Gaussian system noise \mathbf{w}_k with noise dimension $W = N$ and covariance matrix \mathbf{Q}_k , i.e.,

$$\mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{w}_k, \text{ with } \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k). \quad (1)$$

The system is observed by a network of N_s sensor nodes, where each individual node i receives measurements using the observation model \mathbf{C}^i and covariance \mathbf{R}_k^i according to

$$\mathbf{z}_k^i = \mathbf{C}^i \mathbf{x}_k + \mathbf{v}_k^i, \text{ with } \mathbf{v}_k^i \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k^i). \quad (2)$$

We assume that each node i computes a state estimate $\hat{\mathbf{x}}_{k|k}^i$ with error covariance matrix $\mathbf{P}_{k|k}^i$.

A. Fusion of Estimates

Without loss of generality, we confine ourselves to the fusion of two estimates as multiple estimates can be fused sequentially. In the following discussions, we also omit the time index k for the sake of clarity. The fusion of two state estimates $\hat{\mathbf{x}}^i$ and $\hat{\mathbf{x}}^j$ can take place at an arbitrary time step k and is a linear combination with the fusion gains \mathbf{F}^i and \mathbf{F}^j . Depending on the chosen fusion algorithm, the gains can be determined according to the Bar-Shalom/Campo formulas but can also be fixed weighting matrices. The fused estimate becomes

$$\hat{\mathbf{x}}^f = \mathbf{F}^i \hat{\mathbf{x}}^i + \mathbf{F}^j \hat{\mathbf{x}}^j, \quad (3)$$

with $\mathbf{F}^i + \mathbf{F}^j = \mathbf{I}$ and the corresponding error covariance matrix

$$\mathbf{P}^f = [\mathbf{F}^i \quad \mathbf{F}^j] \mathbf{J} [\mathbf{F}^i \quad \mathbf{F}^j]^T. \quad (4)$$

The joint error covariance matrix is

$$\mathbf{J} = \begin{bmatrix} \mathbf{P}^i & \mathbf{P}^{i,j} \\ \mathbf{P}^{j,i} & \mathbf{P}^j \end{bmatrix},$$

where $\mathbf{P}^{i,j} = (\mathbf{P}^{j,i})^T$ denote the cross-covariances and characterize the dependencies between the state estimates.

State estimates can be correlated due to common process noise and common prior information but also due to sharing of measurements that may also be a result of intermediate fusion results and loops within the sensor network. Only proper treatment of these correlations allows correct and consistent fusion results.

B. Correlations due to Common Process Noise and Common Prior Information

In systems with a central fusion node, see Figure 1(a), state estimates are correlated due to common process noise and common prior information. When all processing steps are known, the cross-covariances between state estimates can be calculated recursively [10]. During the time update, the process noise is incorporated and the cross-covariance matrix is updated leading to the recursive formula

$$\begin{aligned} \mathbf{P}_{k|k-1}^{i,j} &= E[(\hat{\mathbf{x}}_{k|k-1}^i - \mathbf{x}_k)(\hat{\mathbf{x}}_{k|k-1}^j - \mathbf{x}_k)^T] \\ &= \mathbf{A}_k \mathbf{P}_{k-1|k-1} \mathbf{A}_k^T + \mathbf{Q}_k, \end{aligned} \quad (5)$$

where $\mathbf{P}_{k-1|k-1}^{i,j}$ for time step $k = 1$ is the common prior covariance $\mathbf{P}_{0|0}$. During the measurement update, the cross-covariance is updated with the gain $\mathbf{L}_k^i = \mathbf{I} - \mathbf{K}_k^i \mathbf{C}_k^i$ by

$$\begin{aligned} \mathbf{P}_{k|k}^{i,j} &= E[(\hat{\mathbf{x}}_{k|k}^i - \mathbf{x}_k)(\hat{\mathbf{x}}_{k|k}^j - \mathbf{x}_k)^T] \\ &= \mathbf{L}_k^i \mathbf{P}_{k|k-1}^{i,j} (\mathbf{L}_k^j)^T, \end{aligned} \quad (6)$$

where \mathbf{K}_k^i is the Kalman gain used at node i . This recursive formulation can also be rewritten explicitly as a sum of dependent covariances

$$\mathbf{P}_{k|k}^{i,j} = \mathbf{T}_{0,k}^i \mathbf{P}_{0|0} (\mathbf{T}_{0,k}^j)^T + \sum_{\tau=1}^k \mathbf{T}_{\tau,k}^i \mathbf{Q}_\tau (\mathbf{T}_{\tau,k}^j)^T, \quad (7)$$

where at every time step τ we include the new process noise \mathbf{Q}_τ . The matrix $\mathbf{T}_{\tau,k}$ denote the individual matrix transformations that are result of the local Kalman filters (see equations (5) and (6)). In large sensor networks, keeping track of these correlations can be cumbersome and often infeasible as it requires full communication of all processing steps. Therefore, the methods in [33], [34] propose the use of square-root decompositions to keep track of correlated dependent noise terms.

C. Square-Root Decomposition of Common Process Noise

The recursive formula of (7) can be reformulated as a square-root decomposition

$$\begin{aligned} \mathbf{P}_{k|k}^{i,j} &= \mathbf{T}_{0,k}^i \sqrt{\mathbf{P}_{0|0}} (\sqrt{\mathbf{P}_{0|0}})^T (\mathbf{T}_{0,k}^j)^T \\ &+ \sum_{\tau=1}^k \mathbf{T}_{\tau,k}^i \sqrt{\mathbf{Q}_\tau} (\sqrt{\mathbf{Q}_\tau})^T (\mathbf{T}_{\tau,k}^j)^T \\ &= \sum_{\tau=0}^k \mathbf{\Sigma}_{\tau,Q}^i (\mathbf{\Sigma}_{\tau,Q}^j)^T. \end{aligned}$$

This square-root decomposition is stored in the matrix

$$\mathbf{S}_{k,Q}^i = \left[\Sigma_{0,Q}^i, \Sigma_{1,Q}^i, \dots, \Sigma_{k,Q}^i \right]$$

and includes all noise terms until the current time step k and has the dimension $M = N \times D = N \times (N + (k - 1) \times W)$. The calculation of this matrix can be done recursively. At time step $k = 0$, it is initialized with

$$\mathbf{S}_{0,Q}^i = \Sigma_{0,Q}^i = \sqrt{\mathbf{P}_0},$$

and the matrix is then linearly transformed by the time update and a new noise term $\Sigma_{k,Q}^i = \sqrt{\mathbf{Q}_k}$ is included. Further, the matrix is then updated using the gain matrix of the Kalman filter update $\mathbf{L}_k^i = \mathbf{I} - \mathbf{K}_k^i \mathbf{C}_k^i$ which leads to

$$\mathbf{S}_{k,Q}^i = \mathbf{L}_k^i \left[\mathbf{A}_k^i \mathbf{S}_{k-1,Q}^i, \Sigma_{k,Q}^i \right].$$

When the fusion step is reached, the cross-covariance matrix between node i and node j can be reconstructed as

$$\mathbf{P}_{k,Q}^{i,j} = \sum_{m=0}^k \Sigma_{m,Q}^i (\Sigma_{m,Q}^j)^T = \mathbf{S}_{k,Q}^i (\mathbf{S}_{k,Q}^j)^T. \quad (8)$$

By including a new process noise term at every time update, the square-root decomposition matrix $\mathbf{S}_{k,Q}^i$ will continue to grow linearly in size. Since communication bandwidth is limited in sensor networks, we need to find a trade-off between the optimal decomposition of dependent information and the communication capacity.

D. Limiting the Number of Square-Root Decomposition Terms for Process Noise and Common Prior Information

In order to keep the number of entries in the square-root decomposition matrix constant, the square-root matrix will be decomposed [34] into two parts

$$\mathbf{S}_k^i = \left[\mathbf{S}_{k,\mathcal{T}}^i, \mathbf{S}_{k,\Omega}^i \right],$$

where $\mathbf{S}_{k,\mathcal{T}}^i$ is a moving horizon square-root decomposition matrix

$$\mathbf{S}_{k,\mathcal{T}}^i = \left[\Sigma_{k-\mathcal{T}+1}^i, \Sigma_{k-\mathcal{T}+2}^i, \dots, \Sigma_k^i \right] \quad (9)$$

that will only include the dependent noise terms up to a user-defined time horizon \mathcal{T} . The remaining noise terms will be removed from the square-root matrix and summarized in a residual $\mathbf{S}_{k,\Omega}^i$. This residual has to be bounded in order to obtain a consistent fusion result. To formulate the fusion rule, we use the optimal joint covariance matrix

$$\mathbf{J}_k = \begin{bmatrix} \mathbf{P}_k^i & \mathbf{P}_{k,\mathcal{T}}^{i,j} \\ \mathbf{P}_{k,\mathcal{T}}^{j,i} & \mathbf{P}_k^j \end{bmatrix}.$$

We can now decompose this matrix into a part $\mathbf{P}_{k,\mathcal{T}}^{i,j}$ that we can reconstruct and a part $\mathbf{P}_{k,\Omega}^{i,j}$ that is correlated but whose exact correlation we cannot reconstruct anymore, i.e.,

$$\mathbf{J}_k = \begin{bmatrix} \mathbf{P}_k^i & \mathbf{P}_{k,\mathcal{T}}^{i,j} + \mathbf{P}_{k,\Omega}^{i,j} \\ \mathbf{P}_{k,\mathcal{T}}^{j,i} + \mathbf{P}_{k,\Omega}^{j,i} & \mathbf{P}_k^j \end{bmatrix}.$$

This residual can be calculated recursively and includes all dependent noise terms not included in the square-root matrix $\mathbf{S}_{k,\mathcal{T}}^i$. With the residual, we obtain

$$\mathbf{S}_{k,\Omega}^i (\mathbf{S}_{k,\Omega}^i)^T = \Omega_{k,\Omega}^i. \quad (10)$$

We now aim to find a bound according to

$$\begin{bmatrix} \frac{1}{\omega} \Omega_{k,\Omega}^i & 0 \\ 0 & \frac{1}{1-\omega} \Omega_{k,\Omega}^j \end{bmatrix} \geq \begin{bmatrix} \Omega_{k,\Omega}^i & \mathbf{P}_{k,\Omega}^{i,j} \\ \mathbf{P}_{k,\Omega}^{j,i} & \Omega_{k,\Omega}^j \end{bmatrix}.$$

Finally, we can now formulate the new suboptimal joint covariance matrix

$$\tilde{\mathbf{J}}_k = \begin{bmatrix} \mathbf{P}_k^i - \Omega_{k,\Omega}^i & \mathbf{P}_{k,\mathcal{T}}^{i,j} \\ \mathbf{P}_{k,\mathcal{T}}^{j,i} & \mathbf{P}_k^j - \Omega_{k,\Omega}^j \end{bmatrix} + \begin{bmatrix} \frac{1}{\omega} \Omega_{k,\Omega}^i & 0 \\ 0 & \frac{1}{1-\omega} \Omega_{k,\Omega}^j \end{bmatrix}, \quad (11)$$

which we will use for the fusion step according to formulas (3) and (4). The weighting factors ω can be found by minimizing the trace or determinant of the fused covariance (4). Alternatively, an approximate solution such as the one proposed by [33], [35] can be used.

III. CONTRIBUTION

The square-root decomposition enables the nodes to encode correlated process noise in a distributed fashion. The central node in Fig. 1(a) does not need to keep track of the correlations, processing steps, or number of nodes as all the required information is provided by the nodes themselves. Modifications to the square-root decomposition are necessary when nodes are organized in hierarchical network topologies as shown in Fig. 1(b), where intermediate fusion nodes exist. Each fusion step alters the correlation structure among the nodes, which has to be encoded properly and is discussed in Sec. III-A. The decentralized network architecture depicted in Fig. 1(c) exhibits cycles that lead to double counting of information. Sec. III-B discusses how additional data structures can be introduced to cover correlations caused by double counting.

A. Hierarchical Fusion

In a hierarchical fusion architecture, nodes may fuse estimates and pass them to the upper layer for a subsequent fusion step. Such intermediate fusion nodes hence have to take into account correlations for the fusion but simultaneously have to compute an updated square-root decomposition for the subsequent fusion steps. Each node i can fuse its estimate with an estimate received from node j by using the fusion formulas (3) and (4). The required cross-covariance matrices $\mathbf{P}^{i,j} = (\mathbf{P}^{j,i})^T$ are obtained by the square-root decomposition, i.e., by using (8).

For the subsequent fusion layer, the square-root decomposition needs to encode the correlation structure of the fusion result $\hat{\mathbf{x}}^f$. The cross-covariance matrix for this fusion result $\hat{\mathbf{x}}^f$ and the estimate $\hat{\mathbf{x}}^l$ of a third node l yields

$$\begin{aligned} \mathbf{P}^{f,l} &= E[(\hat{\mathbf{x}}^f - \mathbf{x})(\hat{\mathbf{x}}^l - \mathbf{x})^T] \\ &= E[(\mathbf{F}^i \hat{\mathbf{x}}^i + \mathbf{F}^j \hat{\mathbf{x}}^j - \mathbf{x})(\hat{\mathbf{x}}^l - \mathbf{x})^T] \\ &= \mathbf{F}^i \mathbf{P}^{i,l} + \mathbf{F}^j \mathbf{P}^{j,l}. \end{aligned}$$

The dependencies $\mathbf{P}^{i,l}$ and $\mathbf{P}^{j,l}$ are given by the corresponding square-root decompositions, i.e.,

$$\mathbf{P}^{i,l} = \mathbf{S}_Q^i (\mathbf{S}_Q^l)^\top \text{ and } \mathbf{P}^{j,l} = \mathbf{S}_Q^j (\mathbf{S}_Q^l)^\top .$$

Hence, the fused square-root decomposition for the $\mathbf{P}^{f,l}$ has the form

$$\mathbf{S}_Q^f = \mathbf{F}^i \mathbf{S}_Q^i + \mathbf{F}^j \mathbf{S}_Q^j , \quad (12)$$

which gives $\mathbf{P}^{f,l} = \mathbf{S}_Q^f (\mathbf{S}_Q^l)^\top$ for any l .

For a finite horizon τ , \mathbf{S}^f only partially covers the correlations, and the fusion node also has to update the residual term (10). According to the chosen weight ω in (11), the residual becomes

$$\begin{aligned} \Omega_Q^f &= \frac{1}{\omega} \mathbf{F}^i \Omega_Q^i (\mathbf{F}^i)^\top + \frac{1}{1-\omega} \mathbf{F}^j \Omega_Q^j (\mathbf{F}^j)^\top \\ &\geq \mathbf{F}^i \Omega_Q^i (\mathbf{F}^i)^\top + \mathbf{F}^i \Omega_Q^{i,j} (\mathbf{F}^j)^\top \\ &\quad + \mathbf{F}^j \Omega_Q^{j,i} (\mathbf{F}^i)^\top + \mathbf{F}^j \Omega_Q^j (\mathbf{F}^j)^\top , \end{aligned} \quad (13)$$

which is a bound since any information about $\Omega_Q^{i,j}$ has been discarded.

B. Double Counting

Double counting occurs when two nodes i and j fuse their estimates for a second time. In other words, the information sent out by node i circles back this node over possibly multiple hops and processing steps. Not only common process noise then leads to correlations, but also measurements incorporated in the estimates reappear at the nodes and introduce additional correlations. In the latter case, two estimates are to be fused that share the same information. For this reason, each node i keeps track of an additional list of measurement noise terms

$$\mathbf{S}_{k,R^i}^i = \left[\Sigma_{0,R^i}^i, \Sigma_{1,R^i}^i, \dots, \Sigma_{k,R^i}^i \right] , \quad (14)$$

to account for double counting of measurements. It is initialized with

$$\mathbf{S}_{0,R^i}^i = \Sigma_{0,R^i}^i = \mathbf{K}_0^i \sqrt{\mathbf{R}_0^i} ,$$

where \mathbf{R}_0^i is the measurement covariance matrix of the first measurement (2) acquired by node i . The matrix \mathbf{K}_0^i is the Kalman gain used in this measurement update. The matrix \mathbf{S}_{k,R^i}^i is recursively updated according to

$$\mathbf{S}_{k,R^i}^i = \mathbf{L}_k^i \left[\mathbf{A}_k^i \mathbf{S}_{k-1,R^i}^i , \Sigma_{k,R^i}^i \right] \quad (15)$$

with

$$\Sigma_{k,R^i}^i = \mathbf{K}_k^i \sqrt{\mathbf{R}_k^i} .$$

When two sensor nodes exchange estimates for fusion, they also pass on all the square-root matrices. These matrices need to be kept separate of each other in order to trace back possible sources of double counting. Node i that receives an estimate from node j then also keeps and manages the set \mathbf{S}_{k,R^j}^i , which is the corresponding set (15) from node j . The own and the

received square-root matrices are updated similarly to (12) and (13) by

$$\begin{aligned} \mathbf{S}_{R^i}^f &= \mathbf{F}^i \mathbf{S}_{R^i}^i + \mathbf{F}^j \mathbf{S}_{R^j}^j , \\ \mathbf{S}_{R^j}^f &= \mathbf{F}^i \mathbf{S}_{R^i}^i + \mathbf{F}^j \mathbf{S}_{R^j}^j . \end{aligned}$$

Bookkeeping of the received \mathbf{S}_{k,R^j}^i resembles (15). However, it differs in that it is filled with zeros during further processing according to

$$\mathbf{S}_{k,R^j}^i = \mathbf{L}_k^i \left[\mathbf{A}_k^i \mathbf{S}_{k-1,R^j}^i , \mathbf{0} \right] \quad (16)$$

as the measurement noise affecting node j is uncorrelated with the estimates at node i for the following time steps.

The square-root matrix \mathbf{S}_{k,R^i}^i can be used in a later fusion step to retrieve the cross-covariances stemming from the previous fusion step by

$$\mathbf{P}_{k,R}^{i,j} = \mathbf{S}_{k,R^i}^i (\mathbf{S}_{k,R^i}^j)^\top + \mathbf{S}_{k,R^j}^j (\mathbf{S}_{k,R^i}^i)^\top , \quad (17)$$

where \mathbf{S}_{k,R^i}^j is the common information with node i that has been tracked in node j . More precisely, \mathbf{S}_{k,R^i}^j is the corresponding set to (16) that was generated by node j when it received information from i . The reconstructed cross-covariance matrix (17) has to be combined with $\mathbf{P}_{k,Q}^{i,j}$ representing the common process noise, which finally results in the full cross-covariance matrix

$$\mathbf{P}_k^{i,j} = \mathbf{P}_{k,Q}^{i,j} + \mathbf{P}_{k,R}^{i,j} .$$

The amount of data that need to be stored and updated by each node grows linearly over time. Especially in networks with many sensor nodes, conservative bounding techniques can allow the nodes to surpass this burden.

a) *Limiting the Number of Square-Root Decomposition Terms for Measurement Noise:* Following the concept in Sec. II-D, we limit the number of processing steps encoded in the square-root decompositions to a fixed horizon \mathcal{T} and summarize the remainder in a residual term Ω_R^i . The matrix (14) becomes

$$\mathbf{S}_R^i = \left[\Sigma_{R,k-\mathcal{T}+1}^i, \Sigma_{R,k-\mathcal{T}+2}^i, \dots, \Sigma_{R,k}^i, \Omega_R^i \right] ,$$

which has a constant number of entries. When two estimates are fused, a bound on the residual matrix as in (13) has to be computed by

$$\Omega_R^f = \frac{1}{\omega} \mathbf{F}^i \Omega_R^i (\mathbf{F}^i)^\top + \frac{1}{1-\omega} \mathbf{F}^j \Omega_R^j (\mathbf{F}^j)^\top .$$

This bound also has to be combined with the residual bound (13) for the process noise.

b) *Keeping Track of Uncorrelated Measurements:* The treatment of correlated measurement information and double counting can be simplified by computing a more general bound on the measurement covariance. This approach circumvent the explicit bookkeeping (14) of the information shared through the fusion of estimates.

The local covariance matrix of sensor node i is rewritten as

$$\mathbf{P}^i = \mathbf{P}_{Q,\mathcal{T}} + \mathbf{P}_{Q,\Omega} + \mathbf{P}_R ,$$

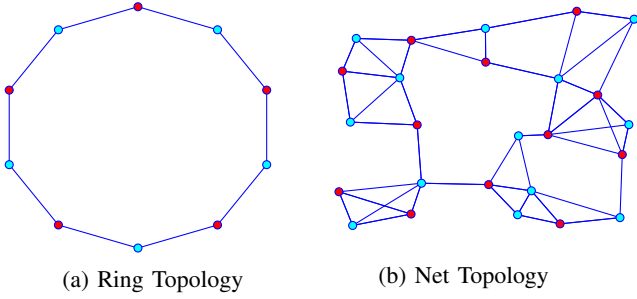


Fig. 2: Network topologies, magenta nodes are using one, blue nodes the other measurement model.

where $\mathbf{P}_{Q,\mathcal{T}}$ accounts for the reconstructable cross-covariance matrix using (9), $\mathbf{P}_{Q,\Omega}$ accounts for the residual (10), and \mathbf{P}_R represents possibly correlated measurement noise. We further separate this into

$$\mathbf{P}_R = \mathbf{P}_R^+ + \mathbf{P}_R^-,$$

where \mathbf{P}_R^+ denotes correlated measurement noise and \mathbf{P}_R^- uncorrelated measurement noise. We can safely assume that measurements which have been obtained between fusion steps and thus have not been shared are uncorrelated. Therefore, only the part accounting for information that has been shared with other sensor nodes before is correlated and needs to be bounded. The uncorrelated measurement noise residual $\mathbf{P}_{k,R}^-$ can be calculated recursively

$$\mathbf{P}_{k,R}^- = \mathbf{L}\mathbf{A}\mathbf{P}_{k-1,R}^- \mathbf{A}^T \mathbf{L}^T + \mathbf{K}_k^i \mathbf{R}_1^i (\mathbf{K}_k^i)^T.$$

To ensure the correctness of this assumption, $\mathbf{P}_{k,R}^-$ will be reset to the zero matrix when the fusion step has been executed or the information has been shared with other sensor nodes. The correlated measurement residual is calculated by

$$\mathbf{\Omega}_R^i = \mathbf{P}^i - \mathbf{\Sigma}_Q^i (\mathbf{\Sigma}_Q^i)^T - \mathbf{\Omega}_Q^i - \mathbf{P}_R^-.$$

The bounded part of the joint covariance matrix becomes

$$\mathbf{\Omega}_k^i = \mathbf{\Omega}_{k,Q}^i + \mathbf{\Omega}_{k,R}^i.$$

The rest of the fusion step is analogous to (11).

IV. EVALUATION

The following section will feature three distinct examples to highlight the performance of the proposed algorithm under different conditions. First, we will discuss an example using only two sensor nodes which constantly exchange information and, therefore, leading to highly correlated estimates. Second, we will discuss the convergence rate of the proposed algorithm and compare it with standard consensus algorithms. Last, a tracking example using 25 heterogeneous sensor nodes in a sparse network but with synchronized fusion steps is analyzed.

A. Two Sensor Nodes

We consider two sensor nodes A and B , which observe the discrete-time time-invariant linear stochastic system in (1) with the parameters

$$\mathbf{A} = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Both sensor nodes draw observations using the linear measurement model (2), where every measurement is corrupted by additive-white Gaussian noise v_k^i with covariance matrix $\mathbf{R}^A = \mathbf{R}^B = 50$ and measurement matrices

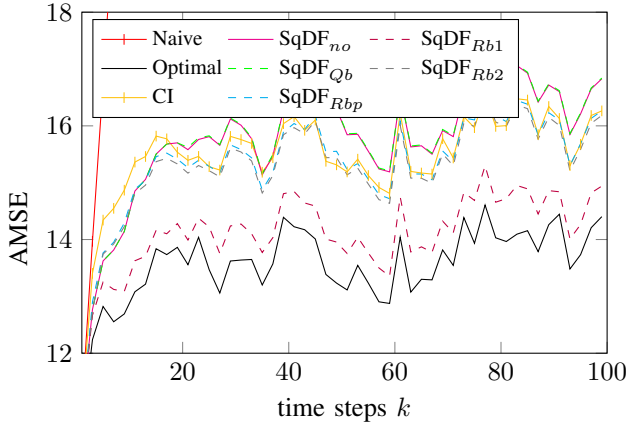
$$\mathbf{C}^A = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad \mathbf{C}^B = \begin{bmatrix} 0 & 1 \end{bmatrix}.$$

The data exchange between the two nodes is performed as follows

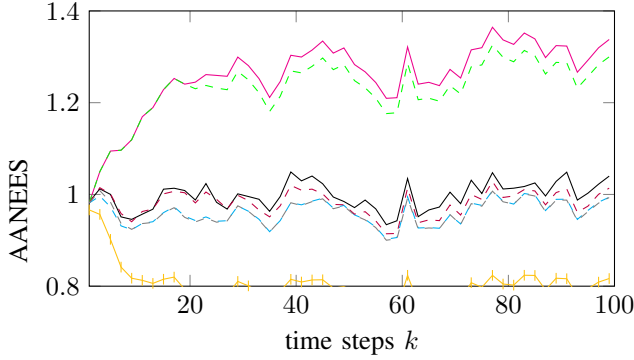
- 1) both sensor nodes execute a local filter update,
- 2) node A sends its local information to node B ,
- 3) node B fuses information according to the selected fusion method and reinitializes its local state and covariance matrix with new fused information,
- 4) both sensor nodes execute a local filter update,
- 5) node B sends its local information to node A ,
- 6) node A fuses information according to the selected fusion method and reinitializes its local state and covariance matrix with new fused information,
- 7) repeat from beginning.

Fig. 3(a) shows the averaged mean squared error (AMSE) of both sensor nodes for 1000 Monte Carlo Runs (MCR). The results are compared with the optimal central fusion result, which shows the lowest MSE as expected. The MSE of the naive fusion, which neglects the correlations between state estimates, immediately diverges. The proposed square-root decomposition (SqRD) is shown in several configurations. The time horizon for the square-root matrix is $\mathcal{T} = 5$. The square-root decomposition without bounding (SqRD_{no}) shows a relatively high MSE as it does not account for older process noise or any correlation due to measurement noise. Bounding of process noise (SqRD_{Qb}) performs a bit better in comparison as it does bound the process noise but also does not account for possibly correlated measurements. Covariance intersection performs better than SqRD_{no} and SqRD_{Qb}, but its performance is limited as it cannot account for uncorrelated parts. Using the proposed algorithm with partial bounding of measurement noise (SqRD_{Rbp}) shows better performance than covariance intersection, as it can find a tighter bound.

Figure 3(b) shows the averaged normalized estimation error squared (ANEES) over both sensor nodes. The ANEES is a measure to determine whether the actual uncertainty matches the expected uncertainty [36]. An ANEES below one indicates a conservative fusion estimate while an ANEES above one indicates an underestimation of the actual uncertainty. Naive fusion diverges again very fast and is therefore not included in the plot, and covariance intersection is overly conservative. Both methods without bounding (SqRD_{no} and SqRD_{Qb}) are inconsistent as it would be expected. The algorithm with



(a) Averaged MSE of both sensor nodes.



(b) Averaged ANEES of both sensor nodes.

Fig. 3: Comparison of the fusion results of different algorithms for 1000 Monte Carlo Runs.

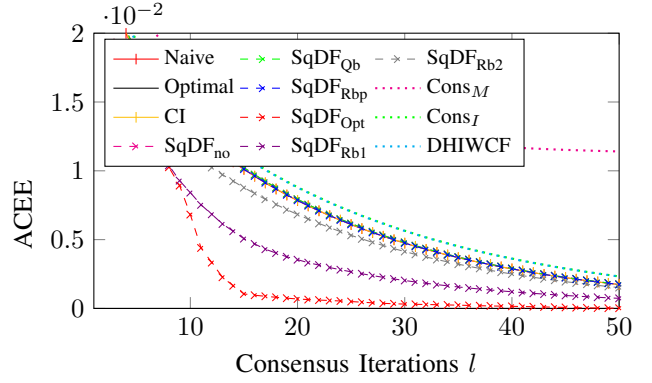
partial bounding is close to one, meaning that the actual MSE of the fused results matches the covariance matrix.

B. Consensus between States

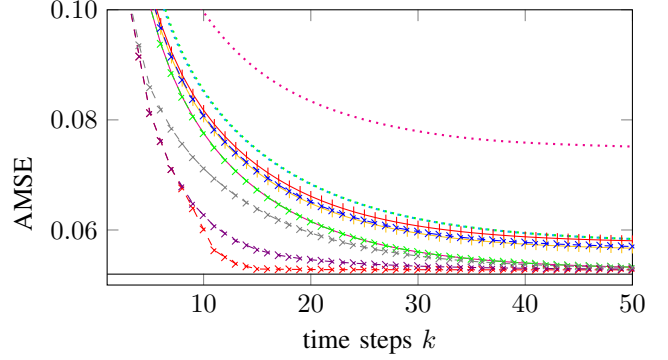
In the following example, we discuss how fast the proposed algorithm converges towards a global consensus. We define the averaged consensus estimate error (ACEE) that indicates the degree of consensus among estimates from all nodes in the network

$$\text{ACEE} = \frac{1}{N_s} \sum_{i=1}^{N_s} (\hat{x}^i - \bar{x}), \quad \bar{x} = \frac{1}{N_s} \sum_{i=1}^{N_s} \hat{x}^i.$$

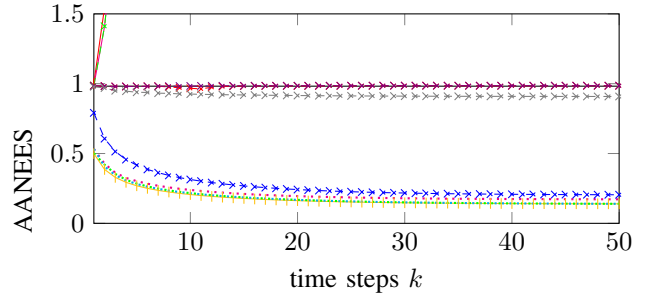
We consider a network of ten sensor nodes with ring topology (see Figure 2(a)). The system description is similar to the one in Example 1 and the sensor nodes alternate between the measurement model of node *A* and node *B*, which can also be seen in the figure. The sensor nodes first perform ten filtering steps independently and then communicate their local information towards their neighbors multiple times. The fusion algorithms are also compared with consensus algorithms, specifically consensus on measurements [23] (Cons_M), consensus on information [25] (Cons_I) and a hybrid consensus method called DHIWCF [27]. We would like to point out



(a) Convergence of state estimates to Consensus



(b) Averaged MSE over all Sensor Nodes



(c) Averaged ANEES over all Sensor Nodes

Fig. 4: Convergence of state estimates towards a common consensus and mean squared error for ring topology with 200 MCR.

that many consensus algorithms have been proposed in recent years and that the utilized algorithms may not be best tailored to the considered problem. Figure 4(a) shows the convergence rate of the state estimates. Covariance intersection, naive fusion, and consensus on information show very similar convergence rates. Consensus on measurements on the other hand, converges much slower. The hybrid consensus algorithm DHIWCF lies between consensus on measurements and consensus on information. Furthermore, we see that the square-root decomposition of the measurement noise clearly improves the convergence rate. Keeping track of all measurements (SqRD_{Opt}) leads to the fastest convergence, followed

by the square-root decomposition with a time horizon $\mathcal{T} = 3$ (SqRD_{Rb1}) and using a time horizon $\mathcal{T} = 1$ (SqRD_{Rb2}). Therefore showing, that even a small time horizon for the measurement noise might make a huge difference. The time horizon of the square root matrix keeping track of the process noise is $\mathcal{T} = 11$. Therefore, process noise and common prior information are tracked completely. Looking at the averaged mean squared error (AMSE) in Figure 4(b) we see again, that the optimal track keeping of correlations achieves the lowest AMSE fastest and almost approaches the result of the centralized optimal fusion result. The square-root decomposition with a smaller time horizon SqRD_{Rb1} and SqRD_{Rb2} also performs well. Lastly, in Figure 4(c), it can be seen that the ANEES over all sensor nodes in the network is close to the optimal fusion result for SqRD_{Rb1}, SqRD_{Opt}, and SqRD_{Rb2}. All square-root decomposition based algorithms, that bound the measurement partially or fully are very close to the performance of covariance intersection and, therefore, overly conservative.

C. Large-Scale Sparse Network

In our last example, we consider a simple tracking example featuring 25 sensor nodes in a sparse network as depicted in Figure 2(b). Nodes always receive information from the three closest sensor nodes. The movement of the tracked object is described by

$$\underline{\mathbf{x}}_{k+1} = \mathbf{A}\underline{\mathbf{x}}_k + \underline{\mathbf{w}}_k \text{ with } \underline{\mathbf{w}}_k \sim \mathcal{N}(\underline{\mathbf{0}}, \mathbf{Q}),$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \Delta T & 0 \\ 0 & 1 & 0 & \Delta T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{Q} = 0.1 \begin{bmatrix} \frac{1}{3}\Delta T & 0 & \frac{1}{2}\Delta T & 0 \\ 0 & \frac{1}{3}\Delta T & 0 & \frac{1}{2}\Delta T \\ \frac{1}{2}\Delta T & 0 & \Delta T & 0 \\ 0 & \frac{1}{2}\Delta T & 0 & \Delta T \end{bmatrix}.$$

Referring again to Figure 2(b), the blue nodes observe the bearing towards a moving target and the red nodes the range. Their observation is described by a nonlinear measurement function

$$\underline{y}_k^i = h^i(\underline{\mathbf{x}}_k) + \underline{\mathbf{v}}_k,$$

where nodes alternate between measuring the bearing or the range towards a moving target

$$h^i(\underline{\mathbf{x}}_k) = \begin{cases} \text{atan2}(x_{y,k} - P_y^i, x_{x,k} - P_x^i) & \text{if } i \text{ is odd,} \\ \sqrt{(x_{x,k} - P_x^i)^2 + (x_{y,k} - P_y^i)^2} & \text{if } i \text{ is even,} \end{cases}$$

with measurement noise

$$\mathbf{R}^i = \left(\frac{\pi}{180}\right)^2 \text{ if } i \text{ is odd, or } \mathbf{R}^i = 0.01 \text{ m}^2 \text{ if } i \text{ is even}$$

at the sensor nodes position $P^i = [P_x^i, P_y^i]^T$. The nodes are placed at random on a 10 m × 10 m field. They perform a synchronized fusion step every 5th time step. Since the most

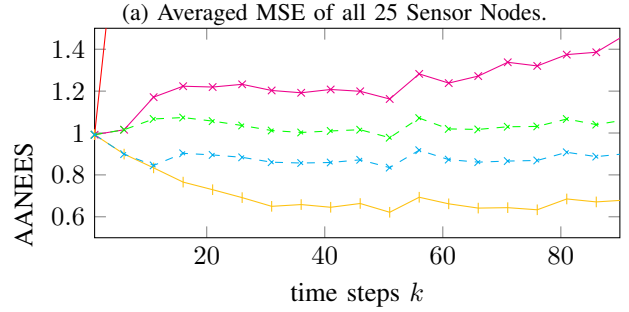
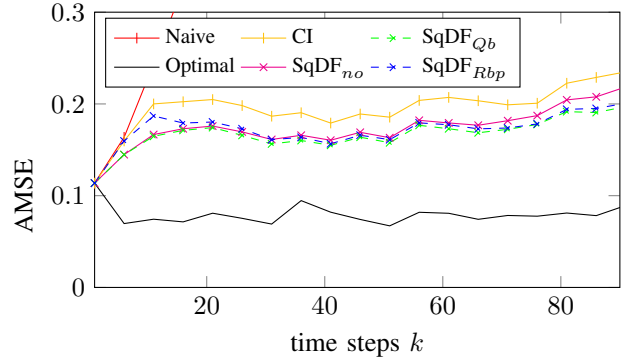


Fig. 5: Comparison of the fusion results of different algorithms for 100 Monte Carlo Runs.

recent five measurements are hence uncorrelated, a square-root decomposition of the measurement noise is not needed as only older measurements are correlated and their influence is increasingly becoming weaker. Therefore, we will utilize the additional information about uncorrelated measurements for the fusion. The time horizon for keeping track of process noise is $\mathcal{T} = 5$. The results of the partial bounding SqRD_{Rbp} and the square-root decomposition without accounting for correlated measurements SqRD_{Qb} have the lowest MSE. As expected, the partial bounding SqRD_{Rbp} is more conservative than SqRD_{Qb} as indicated by the ANEES. We also observe that SqRD_{Qb} is even consistent, i.e., the ANEES is close to one, which can be due to correlations that cancel each other out because of symmetries within the considered setup.

V. RESULTS AND DISCUSSION

The results of the second example show that the convergence rate is improved when cross-covariances can be reconstructed accurately. Yet, the fusion can lead to numerical issues when sensor nodes are highly correlated since the joint covariance matrix cannot be inverted properly. While the additional square-root decomposition of the measurement noise is beneficial, it leads to additional communication that grows with the number of sensor nodes. It might be possible to discard parts of these square-roots, when they travelled too far from their source. Therefore, correlations would only be tracked within a certain influence circle of a node, which might improve scalability of the algorithm. The choice of the time

horizon determining the number of encoded dependent noise terms highly depends on the application and needs thorough consideration.

VI. CONCLUSION

This paper demonstrated the utilization of square-root decompositions to keep track of dependent information in decentralized sensor networks. The experiments confirm that the proposed algorithm produces more accurate fusion results and converges faster to a global consensus. Further, the algorithm can be tailored to many applications as it is able to find a trade off between accuracy and communication requirements.

ACKNOWLEDGMENT

This work was supported by the German Research Foundation (DFG) under grant HA 3789/14-1.

REFERENCES

- [1] 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.
- [2] S. Grime and H. F. Durrant-Whyte, "Data Fusion in Decentralized Sensor Networks," *Control Engineering Practice*, vol. 2, no. 5, pp. 849–863, Oct. 1994.
- [3] C.-Y. Chong, "Forty Years of Distributed Estimation: A Review of Noteworthy Developments," in *Proceedings of the IEEE ISIF Workshop on Sensor Data Fusion: Trends, Solutions, Applications (SDF 2017)*, Bonn, Germany, Oct. 2017.
- [4] A. G. Mutambara, *Decentralized Estimation and Control for Multisensor Systems*. CRC Press, 1998.
- [5] A. G. O. Mutambara and H. F. Durrant-Whyte, "Nonlinear Information Space: A Practical Basis for Decentralization," in *Proceedings of SPIE's International Symposium on Photonics for Industrial Applications, Sensor Fusion VII*, P. S. Schenker, Ed., vol. 2355, no. 1. SPIE, 1994, pp. 97–105.
- [6] 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.
- [7] J. L. Speyer, "Computation and Transmission Requirements for a Decentralized Linear-Quadratic-Gaussian Control Problem," *IEEE Transactions on Automatic Control*, vol. 24, no. 2, pp. 266–269, Apr. 1979.
- [8] 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, Jul. 2012.
- [9] 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.
- [10] Y. Bar-Shalom and L. Campo, "On the Track-to-Track Correlation Problem," *IEEE Transactions on Automatic Control*, vol. 26, no. 2, pp. 571–572, Apr. 1981.
- [11] C.-Y. Chong, K.-C. Chang, and S. Mori, "Distributed Tracking in Distributed Sensor Networks," in *Proceedings of the 1986 American Control Conference (ACC 1986)*, Seattle, Washington, USA, 1986.
- [12] 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.
- [13] Y. Bar-Shalom and L. Campo, "The Effect of the Common Process Noise on the Two-Sensor Fused-Track Covariance," in *IEEE Transactions on aerospace and electronic systems*, vol. Vol. AES-22, No. 6, 1986.
- [14] C.-Y. Chong and S. Mori, "Convex Combination and Covariance Intersection Algorithms in Distributed Fusion," in *Proceedings of the 4th International Conference on Information Fusion (Fusion 2001)*, Montréal, Québec, Canada, Aug. 2001.
- [15] S. J. Julier and J. K. Uhlmann, "A Non-divergent Estimation Algorithm in the Presence of Unknown Correlations," in *Proceedings of the IEEE American Control Conference (ACC 1997)*, vol. 4, Albuquerque, New Mexico, USA, Jun. 1997, pp. 2369–2373.
- [16] L. Chen, P. O. Arambel, and R. K. Mehra, "Fusion under Unknown Correlation: Covariance Intersection Revisited," *IEEE Transactions on Automatic Control*, vol. 47, no. 11, pp. 1879–1882, Nov. 2002.
- [17] M. Reinhardt, B. Noack, P. O. Arambel, and U. D. Hanebeck, "Minimum Covariance Bounds for the Fusion under Unknown Correlations," *IEEE Signal Processing Letters*, vol. 22, no. 9, pp. 1210–1214, Sep. 2015.
- [18] 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.
- [19] 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.
- [20] C.-Y. Chong, K.-C. Chang, and S. Mori, *Distributed Data Fusion for Network-Centric Operations*, ser. The Electrical Engineering and Applied Signal Processing Series. CRC Press, 2013, ch. Fundamentals of Distributed Estimation, pp. 95–124.
- [21] C.-Y. Chong, "Hierarchical estimation," in *MIT/ONR Workshop on C3 Systems*, Monterey, California, USA, 1979.
- [22] X. Tian, Y. bar shalom, T. Yuan, E. Blasch, K. Pham, and G. Chen, "A generalized information matrix fusion based heterogeneous track-to-track fusion algorithm," *Proceedings of SPIE - The International Society for Optical Engineering*, 05 2011.
- [23] R. Olfati-Saber, "Distributed Kalman Filter with Embedded Consensus Filters," in *Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC 2005)*, Sevilla, Spain, Dec. 2005, pp. 8179–8184.
- [24] G. Battistelli and L. Chisci, "Kullback-leibler average, consensus on probability densities, and distributed state estimation with guaranteed stability," *Automatica*, vol. 50, no. 3, p. 707–718, Mar. 2014.
- [25] R. Olfati-Saber, "Distributed Kalman Filtering for Sensor Networks," in *Proceedings of the 46th IEEE Conference on Decision and Control (CDC 2007)*, New Orleans, Louisiana, USA, Dec. 2007, pp. 5492–5498.
- [26] G. Battistelli, L. Chisci, G. Mugnai, A. Farina, and A. Graziano, "Consensus-Based Linear and Nonlinear Filtering," *IEEE Transactions on Automatic Control*, vol. 60, no. 5, pp. 1410–1415, May 2015.
- [27] J. Liu, Y. Liu, K. Dong, Z. Ding, and Y. He, "A Novel Distributed State Estimation Algorithm with Consensus Strategy," *Sensors*, vol. 19, 2019.
- [28] C.-Y. Chong, K.-C. Chang, and S. Mori, "Comparison of Optimal Distributed Estimation and Consensus Filtering," 2016.
- [29] J. Čurn, D. Marinescu, N. O'Hara, and V. Cahill, "Data Incest in Cooperative Localisation with the Common Past-Invariant Ensemble Kalman Filter," in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, Jul. 2013.
- [30] M. Reinhardt, B. Noack, and U. D. Hanebeck, "Reconstruction of Joint Covariance Matrices in Networked Linear Systems," in *Proceedings of the 48th Annual Conference on Information Sciences and Systems (CISS 2014)*, Princeton, New Jersey, USA, Mar. 2014.
- [31] 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.
- [32] S. Radtke, B. Noack, U. D. Hanebeck, and O. Straka, "Reconstruction of Cross-Correlations with Constant Number of Deterministic Samples," in *Proceedings of the 21st International Conference on Information Fusion (Fusion 2018)*, Cambridge, United Kingdom, Jul. 2018.
- [33] M. Reinhardt, S. Kulkarni, and U. D. Hanebeck, "Generalized Covariance Intersection based on Noise Decomposition," in *Proceedings of the 2014 IEEE International Conference on Multisensor Fusion and Information Integration (MFI 2014)*, Beijing, China, Sep. 2014.
- [34] S. Radtke, B. Noack, and U. D. Hanebeck, "Distributed Estimation using Square Root Decompositions of Dependent Information," in *Proceedings of the 22nd International Conference on Information Fusion (Fusion 2019)*, Ottawa, Canada, Jul. 2019.
- [35] W. Niehsen, "Information Fusion based on Fast Covariance Intersection Filtering," in *Proceedings of the 5th International Conference on Information Fusion (Fusion 2002)*, Annapolis, Maryland, USA, Jul. 2002.
- [36] X. R. Li, Z. Zhao, and V. P. Jilkov, "Practical Measures and Test for Credibility of an Estimator," in *Proceedings of Workshop on Estimation, Tracking, and Fusion - A Tribute to Yaakov Bar-Shalom*, 2001.