Distributed Estimation using Square Root Decompositions of Dependent Information

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*—Sensor networks allow robust and precise estimation by fusing estimates from several distributed sensor nodes. Because of the often limited communication resources, a trade-off between the amount of information communicated and the quality of the fusion result has to be made. On the one hand, obtaining the optimal fusion result often needs an infeasible amount of additional information, but on the other hand, conservative methods usually lead to more pessimistic results in comparison. This paper proposes a square root decomposition of the incorporated noise terms to reconstruct the cross-covariance matrices between sensor nodes. To save communication bandwidth, a residual is defined that allows bounding of the cross-covariance matrix with a reduced number of noise terms. The consistency of the proposed method is demonstrated by two simulation examples featuring a linear and a nonlinear setup and is compared with other state-of-the-art fusion methods.**

I. Introduction

Distributed estimation is an increasingly important task in many modern applications, such as environmental monitoring or indoor-navigation, due to the growing number of smart mobile devices in every day life [1]. Since the beginning of distributed estimation in the 1970s, a vast number of algorithms have been developed to address different aspects of distributed estimation [2]. For optimal processing of multiple sensor measurements, centralized approaches such as the information filter [3]–[5] or the optimally distributed Kalman filter [6]–[8] can be used. Yet, they lack scalability, robustness, and modularity compared to methods that do not necessarily need a central unit for calculation. Distributed estimation is, therefore, a better way to handle limited bandwidth, communication rate, and energy, which are always concerns in sensor networks. On the downside, these algorithms need to address the correlations that arise due to common prior information and process noise as well as double counting [9]–[12].

Since neglecting the correlations [13] between sensor nodes may lead to inconsistency of the fused estimate, there are several different approaches to address the unknown dependencies between state estimates. Usually, this is done by finding an upper bound for the cross-covariances between sensor nodes, which leads to a suboptimal result compared to using the actual cross-covariances. One of the most common algorithms is covariance intersection [14]–[16] which makes no assumption of the correlations and always yields a consistent result. Since covariance intersection often results in a rather pessimistic result, several other suboptimal approaches have been formulated, such as ellipsoidal intersection [17], inverse covariance intersection [18], [19], methods for improved parameterization of covariance intersection [20]–[22] or methods to further tighten the bound [23]. Other authors formulated a fusion rule using optimization-based approaches [24], [25].

Since approximating the correlations to formulate the fusion rule usually leads to suboptimal fusion results, many other authors tried to reconstruct the cross-covariances between sensor nodes in order to find the optimal fusion result. In [26], the common past invariant Ensemble KF (CPI-EnKF) is used to reconstruct the correlations between a number of cooperatively localizing vehicles. This approach was further developed by [27], who used a number of random samples to calculate the correlations between sensor nodes. While these approaches only lead to approximated cross-covariances, the authors in [28] used deterministic samples to reconstruct the cross-covariance matrices optimally, which can also be seen as a square root decomposition of the incorporated noise terms. This sample-based fusion was further discussed in [29] to limit the number of samples included in the sample set under the assumption that old noise terms asymptotically approach zero over time. Since this assumption is not always valid, the fusion result may become inconsistent over time.

In [30], a generalized covariance intersection method was proposed, that uses a square root decomposition of the noise terms. Further, older noise terms are collected in a residual term and bounded to always obtain a consistent fusion result. By combining this square root decomposition with the recursive calculation of cross-covariances provided by the sample-based fusion, we aim to formulate a flexible and more intuitive method to keep track of the cross-correlations between sensor nodes in a distributed fashion without violating the consistency of the fusion result. In this paper, we propose a square root decomposition of noise terms. We will define a moving horizon square root decomposition matrix to keep track of the cross-covariance matrices in a distributed fashion that is calculated recursively. Further, we will propose a method to keep a residual term that will collect every noise term excluded from the square root decomposition matrix. This residual will be bounded, thus resulting in a consistent fusion result. We

will show that the proposed method can be used to find a compromise between a tight fusion result and the available bandwidth. Further, we will discuss how this method can be adapted to account for nonlinearities in the system.

The remainder of this paper is structured as follows. Section II provides an introduction to the fusion problem and the underlying cross-correlations between sensor nodes. Afterwards, a fusion method based on square root decomposition of dependent information is proposed and further developed using only a limited number of entries and a residual which is used to bound the remaining cross-covariances. Section IV shows the simulation results using the proposed method. The results are discussed in Section V and the advantages compared to the sample-based fusion are highlighted. Finally, the paper is concluded in Section VI.

II. Problem Formulation

We consider a discrete-time linear time-variant stochastic dynamic system with state matrix $\mathbf{A}_k$, state vector $\underline{\boldsymbol{x}}_k$ of state dimension N, input matrix $\mathbf{B}_k$, input vector $\underline{u}_k$, zero-mean white Gaussian system noise $\underline{\boldsymbol{w}}_k$ with covariance matrix $\mathbf{Q}_k$

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

The system is observed by a network of L sensor nodes, where every individual node i receives measurements using the observation model $\mathbf{C}_k^{(i)}$ according to

$$\underline{\boldsymbol{y}}_k^{(i)} = \mathbf{C}_k^{(i)} \underline{\boldsymbol{x}}_{\boldsymbol{k}} + \underline{\boldsymbol{v}}_k^{(i)} \text{ with } \underline{\boldsymbol{v}}_k^{(i)} \sim \mathcal{N}\big(\underline{0}, \mathbf{R}_k^{(i)}\big) , \quad (2)$$

which are affected by white Gaussian noise $\underline{\boldsymbol{v}}_k^{(i)}$ with covariance matrix $\mathbf{R}_k^{(i)}$. Further, every sensor node uses a Kalman filter or one of its derivatives [31] to minimize the error in the state estimate $\underline{\hat{x}}^{(i)}$.

To increase precision and robustness, the results of the individual sensor nodes are fused. This can be done using the Bar-Shalom/Campo formulas [32] for the multi-sensor case [33]. During the fusion step, the joint state vector

$$\underline{\hat{m}}_{k|k} = \left[\left(\underline{\hat{x}}_{k|k}^{(1)}\right)^\top \quad , \ldots , \quad \left(\underline{\hat{x}}_{k|k}^{(L)}\right)^\top \right]^\top$$

and the joint covariance matrix

$$\mathbf{J}_{k|k} = \begin{bmatrix} \mathbf{P}_{k|k}^{(1)} & \mathbf{P}_{k|k}^{(1,2)} & \cdots & \mathbf{P}_{k|k}^{(1,L)} \\ \mathbf{P}_{k|k}^{(2,1)} & \mathbf{P}_{k|k}^{(2)} & \cdots & \mathbf{P}_{k|k}^{(2,L)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{k|k}^{(L,1)} & \mathbf{P}_{k|k}^{(L,2)} & \cdots & \mathbf{P}_{k|k}^{(L)} \end{bmatrix} \quad (3)$$

are used to calculate the fused state estimate and covariance matrix

$$\underline{\hat{x}}_{k|k} = \mathbf{P}_{k|k} \mathbf{H}^\top \left(\mathbf{J}_{k|k}\right)^{-1} \underline{\hat{m}}_{k|k} ,$$

$$\mathbf{P}_{k|k} = \left(\mathbf{H}^\top \left(\mathbf{J}_{k|k}\right)^{-1} \mathbf{H} \right)^{-1} ,$$

where the matrix $\mathbf{H} = [\mathbf{I}, \ldots, \mathbf{I}]^\mathrm{T}$ describes how the local states vectors of the sensor nodes map into the fused state vector. The block entries $\mathbf{P}_{k|k}^{(1)}, \ldots, \mathbf{P}_{k|k}^{(L)}$ on the main diagonal refer to the covariance matrices of the local Kalman filters and are therefore known. The entries on the off-diagonals, on the other hand, represent the cross-covariances between the sensor nodes and are usually unknown. When all processing steps are known, the cross-covariances between the sensor nodes can be calculated recursively [9]. During the time update, the cross-covariances between the sensor nodes are updated by the prediction and the process noise is included, yielding the recursive formula

$$\begin{aligned} \mathbf{P}_{k|k-1}^{(i,j)} &= E[(\underline{\hat{x}}_{k-1|k-1}^{(i)} - \underline{\boldsymbol{x}}_k)(\underline{\hat{x}}_{k-1|k-1}^{(j)} - \underline{\boldsymbol{x}}_k)^\mathrm{T}] \\ &= \mathbf{A}_k \mathbf{P}_{k-1|k-1} (\mathbf{A}_k)^\mathrm{T} + \mathbf{Q}_k , \end{aligned} \quad (4)$$

where $\mathbf{P}_{k-1|k-1}$ at the initial time step $k = 1$ is the common prior information $\mathbf{P}_0$ between the sensor nodes. During the measurement update, the cross-covariances are updated by

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

with $\mathbf{L}_k^{(i)} = \mathbf{I} - \mathbf{K}_k^{(i)} \mathbf{C}_k^{(i)}$. This recursive calculation of the cross-covariances can also be written explicitly

$$\mathbf{P}_{k|k}^{(i,j)} = \mathbf{T}_{0,k}^{(i)} \mathbf{P}_0 (\mathbf{T}_{0,k}^{(j)})^\mathrm{T} + \sum_{\tau=1}^{k} \mathbf{T}_{\tau,k}^{(i)} \mathbf{Q}_\tau (\mathbf{T}_{\tau,k}^{(j)})^\mathrm{T} . \quad (6)$$

as a sum of independent noise terms $\mathbf{Q}_\tau$ included at time step τ and the common prior information $\mathbf{P}_0$ [30]. The matrix $\mathbf{T}_{\tau,k}$ denotes the individual matrix transformation that arise from the Kalman filter steps. Thus, we have now recapitulated the recursive and the explicit form of the cross-covariance calculation. It is evident from the these equations, that keeping track of all processing steps and noise terms requires information about every Kalman filter step and the incorporated noise terms and is thus infeasible in large sensor networks. To still be able to optimally reconstruct the cross-covariances between sensor nodes, a distributed calculation is necessary.

III. Square root Decomposition of Noise Terms

In this section, we will first derive the square root decomposition of noise terms to calculate the cross-covariances between two nodes i and j. The recursive computation of the cross-covariances will then be limited to a fixed number of noise terms in order to reduce communication requirements. The remaining noise terms will be collected in a residual term that will be bounded to keep the fusion result consistent.

A. Optimal Square Root Decomposition

To be able to distribute the calculation of the cross-covariances between sensor nodes, a useful decomposition that allows reconstruction during the fusion step has to be found.

The cross-covariance from (6) can be rewritten as a square root decomposition

$$\begin{aligned}\mathbf{P}_{k|k}^{(i,j)} &= \mathbf{T}_0^{(i)}\sqrt{\mathbf{P}_0}(\sqrt{\mathbf{P}_0})^{\mathrm{T}}(\mathbf{T}_0^{(j)})^{\mathrm{T}} \\ &\quad + \sum_{\tau=1}^{k}\mathbf{T}_\tau^{(i)}\sqrt{\mathbf{Q}_\tau}(\sqrt{\mathbf{Q}_\tau})^{\mathrm{T}}(\mathbf{T}_\tau^{(j)})^{\mathrm{T}} \\ &= \sum_{\tau=0}^{k}\mathbf{\Sigma}_\tau^{(i)}(\mathbf{\Sigma}_\tau^{(j)})^{\mathrm{T}}\,.\end{aligned}$$

By decomposing the cross-covariances into the square root form, a suitable decomposition is found that allows the reconstruction of the cross-covariances. These square root decomposition terms will be kept in the matrix

$$\mathcal{S}_k^{(i)} = \left[\mathbf{\Sigma}_0^{(i)}, \mathbf{\Sigma}_1^{(i)}, \ldots, \mathbf{\Sigma}_k^{(i)}\right],$$

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

$$\mathcal{S}_0^{(i)} = \mathbf{\Sigma}_0^{(i)}\,,$$

where $\mathbf{\Sigma}_0^{(i)}$ denotes for the square root decomposition $\sqrt{\mathbf{P}_0}$ of the common prior information between all sensor nodes. Afterwards, the matrix is linearly transformed by the prediction step and the new noise term $\mathbf{\Sigma}_k^{(i)} = \sqrt{\mathbf{Q}_k}$ is included (see equation (4)). The matrix is further updated by a transformation $\mathbf{L}_k^{(i)} = \mathbf{I} - \mathbf{K}_k^{(i)}\mathbf{C}_k^{(i)}$ for the measurement step (see equation (5)), yielding

$$\mathcal{S}_k^{(i)} = \mathbf{L}_k^{(i)}\left[\mathbf{A}_k^{(i)}\mathcal{S}_{k-1}^{(i)}\,,\ \mathbf{\Sigma}_k^{(i)}\right]. \tag{7}$$

The size of this matrix will therefore grow linearly with time. The cross-covariances can then be reconstructed according to

$$\mathbf{P}_k^{(i,j)} = \sum_{\tau=0}^{k}\mathbf{\Sigma}_m^{(i)}(\mathbf{\Sigma}_m^{(j)})^{\mathrm{T}} = \mathcal{S}_k^{(i)}(\mathcal{S}_k^{(j)})^{\mathrm{T}}.$$

Because of the limited bandwidth that is usually present in sensor networks, the matrix $\mathcal{S}_k^{(i)}$ needs to be limited in the number of entries.

B. Limiting the Number of Square Root Decompositions

To address the constrained bandwidth, only a limited number of noise terms can be kept in the matrix $\mathcal{S}_k^{(i)}$. We will start by dividing the matrix $\mathcal{S}_k^{(i)}$ into two parts

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

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

$$\mathcal{S}_{k,\mathcal{T}}^{(i)} = \left[\mathbf{\Sigma}_{k-\mathcal{T}+1}^{(i)}, \mathbf{\Sigma}_{k-\mathcal{T}+2}^{(i)}, \ldots, \mathbf{\Sigma}_k^{(i)}\right],$$

which includes only noise terms up to a limited time horizon $\mathcal{T}$. The noise terms in $\mathcal{S}_{k,\Omega}$ are discarded and only the noise terms in $\mathcal{S}_{k,\mathcal{T}}$ are kept and used to reconstruct the cross-covariance matrix. Since the cross-covariances are calculated by summing over a number of positive definite noise terms, the exclusion of noise terms results in a cross-covariance term smaller than the actual cross-covariance matrix. As a result, the fused covariance is underestimated as well and can become inconsistent.

To prevent this from happening, an additional residual term $\mathbf{\Omega}_k$ is kept, that will be recursively calculated and includes all noise terms $\mathcal{S}_{k,\Omega}^{(i)}$ that are excluded from the matrix $\mathcal{S}_{k,\mathcal{T}}^{(i)}$. To recursively calculate these parameters similar to Section III-A, the initialization is expanded by also initializing the residual, yielding

$$\mathbf{\Omega}_0^{(i)} = \mathbf{0}\,.$$

Afterwards we can update the square root matrix and concatenate it with the newest entries till we reach the time horizon $\mathcal{T}$. Once this time horizon is reached, we need to exclude the oldest noise term from the square root decomposition matrix $\mathcal{S}_{k,\mathcal{T}}$ and add it to the residual

$$\begin{aligned}\mathbf{\Omega}_k^{(i)} &= \mathbf{\Omega}_{k-1}^{(i)} + \mathbf{\Sigma}_{k-\mathcal{T}}^{(i)}(\mathbf{\Sigma}_{k-\mathcal{T}}^{(i)})^{\mathrm{T}} \\ &= \mathcal{S}_{k,\Omega}^{(i)}\big(\mathcal{S}_{k,\Omega}^{(i)}\big)^{\mathrm{T}}.\end{aligned} \tag{8}$$

The exclusion of noise terms from the matrix can be seen as a shifting operation, which can be described mathematically as multiplying the matrix $\mathcal{S}_{k-1,\mathcal{T}}^{(i)}$ with a shift matrix $\mathbf{U}$

$$\mathcal{S}_{k-1,\mathcal{T}}^{(i)} := \mathcal{S}_{k-1,\mathcal{T}}^{(i)}\mathbf{U}\,,$$

where the matrix $\mathbf{U}$ to shift all entries N positions to the left is defined as

$$\mathbf{U} = \begin{bmatrix}\mathbf{0}_{N\times D'} \\ \mathbf{I}_{D'\times D'}\end{bmatrix},$$

with $D' = N(\mathcal{T}-1)$. This results in a cancellation of the first noise term in the matrix. Afterwards, a new term can be concatenated (see equation (7))

$$\mathcal{S}_{k,\mathcal{T}}^{(i)} = \mathbf{L}_k^{(i)}\left[\mathbf{A}_k^{(i)}\mathcal{S}_{k-1,\mathcal{T}}^{(i)}\,,\ \mathbf{\Sigma}_k^{(i)}\right].$$

Now a fusion rule needs to be formulated to incorporate the cross-covariances $\mathbf{P}_{k,\mathcal{T}}$ reconstructed with the matrix $\mathcal{S}_{k,\mathcal{T}}$ and bound the remaining cross-covariances using the residual $\mathbf{\Omega}_k$. For this, we will first consider the optimal joint covariance matrix

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

Since we excluded some of the correlations, this matrix can only be partially recovered. The joint covariance matrix can thus also be written as

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

with the cross-covariances divided into one part that is reconstructed $\mathbf{P}_{k,\mathcal{T}}^{(i,j)}$ and another part $\mathbf{P}_{k,\Omega}^{(i,j)}$ that is unknown. We can decompose the joint covariance matrix further, yielding

$$\mathbf{J}_k = \begin{bmatrix} \mathbf{P}_k^{(i)} & \mathbf{P}_{k,\mathcal{T}}^{(i,j)} \\ \mathbf{P}_{k,\mathcal{T}}^{(j,i)} & \mathbf{P}_k^{(j)} \end{bmatrix} - \begin{bmatrix} \mathbf{P}_{k,\Omega}^{(i)} & 0 \\ 0 & \mathbf{P}_{k,\Omega}^{(j)} \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{P}_{k,\Omega}^{(i)} & \mathbf{P}_{k,\Omega}^{(i,j)} \\ \mathbf{P}_{k,\Omega}^{(j,i)} & \mathbf{P}_{k,\Omega}^{(j)} \end{bmatrix}}_{\mathbf{J}_{k,\Omega}} .$$

We recall equation (8) where we can find that

$$\mathbf{P}_{k,\Omega}^{(i)} = \mathcal{S}_{k,\Omega}^{(i)} \big(\mathcal{S}_{k,\Omega}^{(i)}\big)^{\mathrm{T}} = \mathbf{\Omega}_k^{(i)} .$$

We stored the residual $\mathbf{\Omega}_k$ before and thus know the entries on the block main diagonal of $\mathbf{J}_{k,\Omega}$. Since the cross-covariances on the off-diagonals can not be reconstructed anymore, we aim to find a bound for the cross-covariances [11] according to

$$\begin{bmatrix} \frac{1}{\omega_i}\mathbf{\Omega}_k^{(i)} & 0 \\ 0 & \frac{1}{\omega_j}\mathbf{\Omega}_k^{(j)} \end{bmatrix} \geq \begin{bmatrix} \mathbf{\Omega}_k^{(i)} & \mathbf{P}_{k,\Omega}^{(i,j)} \\ \mathbf{P}_{k,\Omega}^{(j,i)} & \mathbf{\Omega}_k^{(j)} \end{bmatrix} = \mathbf{J}_{k,\Omega} .$$

To find a bound, we need to make sure that it is a valid cross-covariance matrix. The matrix $\mathbf{J}_{k,\Omega}$ can be written as

$$\mathbf{J}_{k,\Omega} = \begin{bmatrix} \mathcal{S}_{k,\Omega}^{(i)} \\ \mathcal{S}_{k,\Omega}^{(j)} \end{bmatrix} \begin{bmatrix} \mathcal{S}_{k,\Omega}^{(i)} \\ \mathcal{S}_{k,\Omega}^{(j)} \end{bmatrix}^{\mathrm{T}} .$$

Since $\mathbf{A}(\mathbf{A})^{\mathrm{T}} \geq 0$ and therefore a semidefinit symmetrical matrix, we know that $\mathbf{J}_{k,\Omega}$ is a valid cross-covariance matrix and thus can be bounded. Therefore, we can now formulate the bounded joint covariance matrix as

$$\widetilde{\mathbf{J}}_k = \begin{bmatrix} \mathbf{P}_k^{(i)} - \mathbf{\Omega}_k^{(i)} & \mathbf{P}_k^{(i,j)} \\ \mathbf{P}_k^{(j,i)} & \mathbf{P}_k^{(j)} - \mathbf{\Omega}_k^{(j)} \end{bmatrix} + \begin{bmatrix} \frac{1}{\omega_i}\mathbf{\Omega}_k^{(i)} & 0 \\ 0 & \frac{1}{\omega_j}\mathbf{\Omega}_k^{(j)} \end{bmatrix} ,$$

where $\widetilde{\mathbf{J}}_k > \mathbf{J}_k$. This joint covariance matrix can also be formulated for an arbitrary number of L sensor nodes similar to equation (3) according to

$$\widetilde{\mathbf{J}}_k = \begin{bmatrix} \mathbf{P}_k^{(1)} - \mathbf{\Omega}_k^{(1)} & \mathbf{P}_{k,\mathcal{T}}^{(1,2)} & \dots & \mathbf{P}_{k,\mathcal{T}}^{(1,L)} \\ \mathbf{P}_{k,\mathcal{T}}^{(2,1)} & \mathbf{P}_k^{(2)} - \mathbf{\Omega}_k^{(2)} & \dots & \mathbf{P}_{k,\mathcal{T}}^{(2,L)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{k,\mathcal{T}}^{(L,1)} & \mathbf{P}_{k,\mathcal{T}}^{(L,2)} & \dots & \mathbf{P}_k^{(L)} - \mathbf{\Omega}_k^{(L)} \end{bmatrix}$$

$$+ \begin{bmatrix} \frac{1}{\omega_1}\mathbf{\Omega}_k^{(i)} & 0 & \dots & 0 \\ 0 & \frac{1}{\omega_2}\mathbf{\Omega}_k^{(2)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{\omega_L}\mathbf{\Omega}_k^{(L)} \end{bmatrix} .$$

The weighting factors ω can be found by minimizing the trace or determinant of the joint covariance matrix. Alternatively, an approximate closed-form solution [30] such as the one proposed by [34] can be used, where the trace ϵ is used to calculate the weighting factor ω of sensor node n in respect to the traces of all L sensor nodes in the system according to

$$\omega_n = \frac{D_n}{D_0} = \frac{\prod_{i=1, i\neq n}^{L} \epsilon_i}{\sum_{i=1}^{L} \prod_{i=j, j\neq i}^{L} \epsilon_j} = \frac{1/\epsilon_n}{\sum_{i=1}^{L} 1/\epsilon_i} .$$

The developed algorithm is able to keep track of the cross-covariances between sensor nodes that are caused by common prior information and common process noise in a distributed fashion and without violation of consistency.

C. Extension to Nonlinear Problems

The previous sections described how to obtain the square root decomposition of noise terms in a linear system. Nonetheless, many applications require the utilization of nonlinear filters to cope with nonlinearities in system and measurement equations. For this, we assume that sensor node i uses a nonlinear system model

$$\underline{x}_k = \underline{f}_k(\underline{\boldsymbol{x}}_k) + \underline{\boldsymbol{w}}_k$$

and a nonlinear measurement function

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

where the additive system noise $\underline{\boldsymbol{w}}_k$ and measurement noise $\underline{\boldsymbol{v}}_k$ are defined as in equations (1) and (2). In order to keep track of the cross-covariances, we need to find the transformation which alters the covariance according to equation (7). This linear transformation can be found for the Kalman filter and all of its derivatives [35], such as the EKF [31], the UKF [36] or any other regression Kalman filter that uses a set of samples to approximate the transformation [35].

IV. Evaluation

To evaluate the proposed square root decomposition based fusion method (SqDF), two evaluation examples featuring a linear system and a system with a nonlinear measurement model are simulated.

A. Linear Evaluation Example

The following section will show the performance of the proposed method with and without bounding and different time horizons $\mathcal{T}$. The results will be compared with other fusion methods, in particular, covariance intersection (CI), naïve fusion, which ignores the correlations between the sensor nodes and inverse covariance intersection (ICI). Further, all methods are compared to the proposed method with an infinite time horizon, which corresponds to the optimal fusion result (see Section III-A). For the evaluation, a discrete-time time-invariant linear stochastic system model is used

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

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

The system is observed by two sensor nodes A and B using a linear measurement model

$$\underline{y}_k^{(i)} = \mathbf{C}^{(i)}\underline{\boldsymbol{x}}_{\boldsymbol{k}} + \underline{\boldsymbol{v}}_k^{(i)} \text{ with } \underline{\boldsymbol{v}}_k^{(i)} \sim \mathcal{N}\big(\underline{0}, \mathbf{R}_k^{(i)}\big) .$$

Every measurement is corrupted by additive-white Gaussian noise $\underline{\boldsymbol{v}}_k^{(i)}$ with covariance matrix $\mathbf{R}_k^{(i)} = \mathbf{R}_k^{(j)} = 50$ and the

individual measurement matrices of every sensor node are set to

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

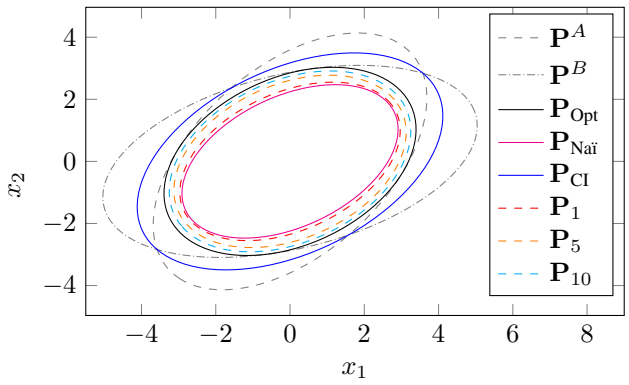
The model is simulated for 100 time steps and the fusion step is performed every 10^{th} time step. The initial conditions for both sensor nodes are the same with $\hat{\underline{x}}_0 = \underline{0}$ and $\mathbf{P}_0 = 5\,\mathbf{I}$. After each fusion step, the local state estimate and covariance matrix are reinitialized by the fusion results. Figure 1 shows the error ellipses after the first fusion step. The matrices $\mathbf{P}^A$ and $\mathbf{P}^B$ denote the covariances of the local Kalman filters. The matrix $\mathbf{P}_{\text{Opt}}$ is calculated using the proposed square root decomposition matrix with an infinite time horizon. The matrices smaller than this optimal solution underestimate the error and therefore lead to inconsistent results. On the other hand, the matrices that are bigger than the optimal solution will not underestimate the error, but they are more pessimistic than the optimal covariance matrix since they overestimate the error. As expected, naïve fusion results in the smallest fused covariance matrix $\mathbf{P}_{\text{Naï}}$ since the correlation between the sensor nodes is completely dismissed. On the other hand, CI is the most pessimistic fusion method since it makes the worst case assumption that both sensor nodes may be fully correlated.

Figure 1(a) shows the fusion results using the proposed square root decomposition fusion without bounding for time horizons $\mathcal{T} = 1$, $\mathcal{T} = 5$ and $\mathcal{T} = 10$. The fusion result for $\mathcal{T} = 10$ includes all process noise terms incorporated during the Kalman filter steps but excludes the common process noise between the sensor nodes before that. It can be shown that the acquired fusion result without bounding always yields a result between the one from naïve fusion and the optimal fusion result. This is expected since the correlation can be calculated as a sum of positive definite matrices and excluding some of the terms leads to a smaller result and therefore to an underestimated cross-covariance. Still, having some information about the correlation is better than having no information.

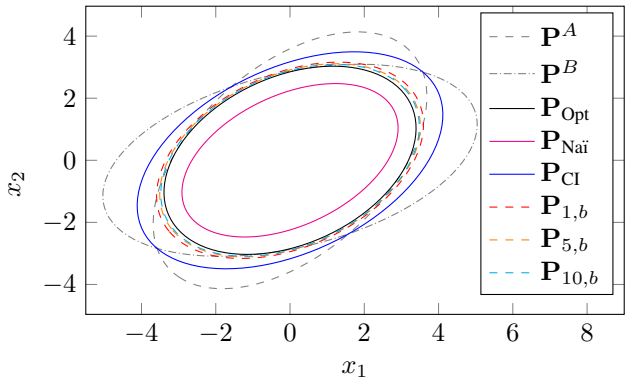
Figure 1(b) shows the fusion results with bounding for the same time horizons. Here it can be seen that the fusion results are always consistent and between the optimal result and CI. The more information about the correlation is available the less bounding has to be done and the closer the result is to the optimal fusion result. But also with just a time horizon of $\mathcal{T} = 1$, the fusion result is still much closer to the optimal result and therefore does not overestimate the covariance matrix as much as CI.

In Figure 2(a) the mean square error over 1000 Monte Carlo runs is depicted. Here again, the optimal fusion result obtained with the proposed method including all noise terms is the method that performs the best and naïve fusion is the method that performs the worst.

The average normalized estimation error squared (ANEES) is shown in Figure 2(b). The ANEES is a metric for the relationship between the estimated error and the actually achieved error. Therefore, it is a measure of how consistent



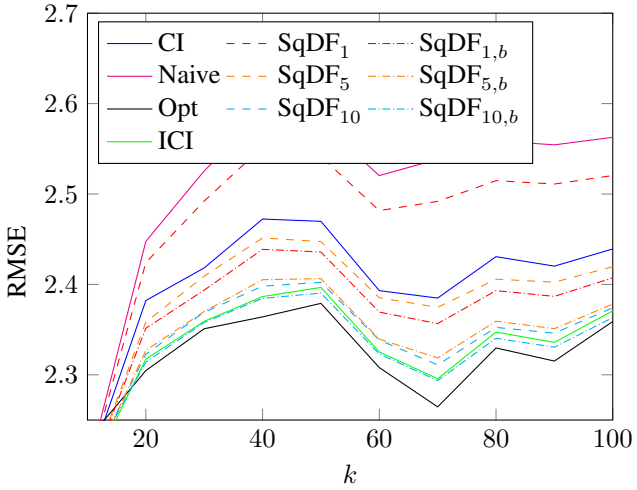
(a) Comparison of several fusion results with the square-root decomposition without bounding.



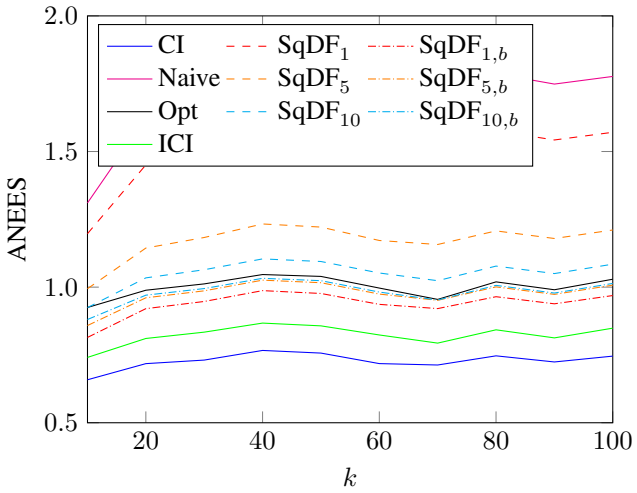
(b) Comparison of several fusion results with the square-root decomposition with bounding.

Fig. 1: Comparison of the error ellipses of covariance intersection ($\mathbf{P}_{\text{CI}}$), naïve fusion $\mathbf{P}_{\text{Naï}}$, optimal fusion using an infinite time horizon ($\mathbf{P}_{\text{Opt}}$), square root decomposition-based fusion $\mathbf{P}_{\mathcal{T}}$ with time horizon $\mathcal{T}$ with and without bounding.

an estimator is. A value below one means that the error is overestimated and, respectively, an ANEES above one means that the error is underestimated and the estimator will become inconsistent. The optimal fusion result obtained by the sample-based fusion is very close to one. CI is the method with the lowest ANEES meaning it is the most pessimistic method, and naïve fusion is the method with the highest ANEES, meaning it is the most optimistic and therefore becomes inconsistent. In addition, the plots show, that the proposed method is achieving better results the more time steps are included in the square root decomposition matrix. The bounded version always performs better than the unbounded version of the proposed method. Yet, it yields less conservative results than CI. The results also show, that the ICI method yields almost equally good results as the square root decomposition fusion with the most noise terms included, which is an interesting find.



(a) Root Mean Square Error (RMSE) from 1000 test runs.



(b) Average normalized estimation error squared (ANEES) from 1000 test runs.

Fig. 2: Comparison of covariance intersection (CI), naïve fusion, optimal fusion using an infinite time horizon (Opt), inverse covariance intersection (ICI), square root decomposition-based fusion with time horizon $\mathcal{T}$ without bounding (SqDF$_{\mathcal{T}}$) and with bounding (SqDF$_{\mathcal{T},b}$) for the linear example.

B. Nonlinear Evaluation Example

To show that the proposed method is also able to handle nonlinearities of the system, this section shows a system with a nonlinear measurement model. For the system model, a constant velocity model

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

for the movement on a two-dimensional plane with the system matrix and process noise matrix

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

$$\mathbf{Q} = 0.01 \begin{bmatrix} \frac{1}{3}\Delta T & 0 & \frac{1}{2}\Delta T & 0 \\ 0 & \frac{1}{3}\Delta T & 0 & \frac{1}{2}\Delta T \\ \frac{1}{2}\Delta T & 0 & \Delta T & 0 \\ 0 & \frac{1}{2}\Delta T & 0 & \Delta T \end{bmatrix}$$

is chosen. Furthermore, the target is observed by four sensor nodes that are positioned at the corners of a 20×20 square at positions $P^{(i)} = [P_x^{(i)}, P_y^{(i)}]^{\mathrm{T}}$ according to

$$P^{(1)} = \begin{bmatrix} 10 \\ 10 \end{bmatrix}, P^{(2)} = \begin{bmatrix} -10 \\ 10 \end{bmatrix}, P^{(3)} = \begin{bmatrix} -10 \\ -10 \end{bmatrix}, P^{(4)} = \begin{bmatrix} 10 \\ -10 \end{bmatrix}$$

that measure the distance r and the angle θ towards the moving target by the measurement function

$$\underline{y}_k^{(i)} = \underline{h}_k^{(i)}(\underline{\boldsymbol{x}}_k) + \underline{\boldsymbol{v}}_k^{(i)}$$
$$\begin{bmatrix} r^{(i)} \\ \theta^{(i)} \end{bmatrix} = \begin{bmatrix} \sqrt{\left((x_x)_k - P_x^{(i)}\right)^2 + \left((x_y)_k - P_y^{(i)}\right)^2} \\ \text{atan2}\left(x_{y,k} - P_y^{(i)}, x_{x,k} - P_x^{(i)}\right) \end{bmatrix} + \underline{\boldsymbol{v}}_k^{(i)},$$

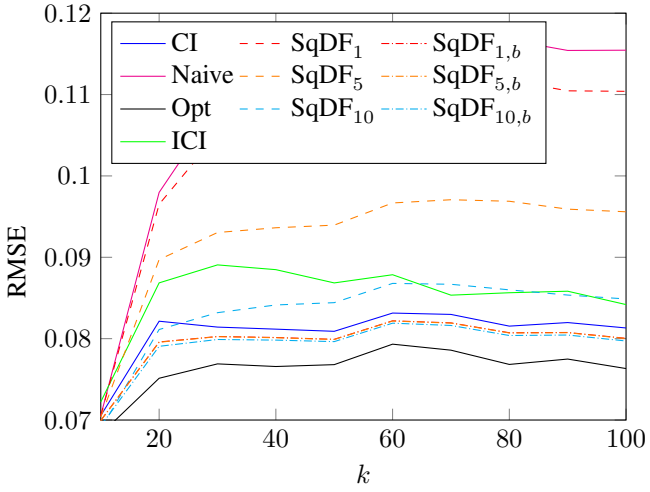
with measurement noise

$$\mathbf{R}^{(i)} = \begin{bmatrix} 0.05 & 0 \\ 0 & 0.5\frac{\pi}{180} \end{bmatrix}$$

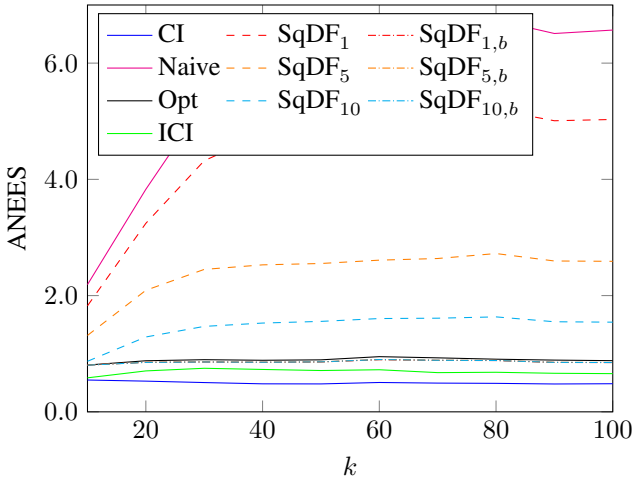
and initial settings $\underline{\hat{x}}_0 = \underline{0}$ and $\mathbf{P}_0 = 0.1\mathbf{I}$. Every sensor node uses an UKF [36] and updates the square root decomposition matrix based on the assumptions stated in Section III-C. Figure 3 shows the RMSE and ANEES for 1000 Monte Carlo runs. It is obvious that neglecting the cross-correlations between sensor nodes causes large errors. It also shows that using the proposed method without bounding leads to inconsistency and therefore to high errors. The proposed method with the bounding technique shows good and consistent results and outperforms CI and ICI.

V. Results and Discussion

The proposed square root decomposition based fusion method is able to keep track of cross-covariances caused by both common process noise and common prior information. It can be adapted to varying bandwidth requirements by reducing the number of square root decompositions. Since the correlation information is still contained in the residual, the remaining cross-covariance matrix can be bounded and therefore a consistent fusion result can be obtained. The square root decomposition is very similar to the sample-based fusion proposed in [28], where the square root decomposition of filtering steps and noise terms is used to create a set of deterministic samples. This sample set has the disadvantage that it can not be altered, e.g., by further filtering at the fusion center, after it has been created. The square root decomposition proposed in this work is still flexible, allows



(a) Root Mean Square Error (RMSE) from 1000 test runs.



(b) Average normalized estimation error squared (ANEES) from 1000 test runs.

Fig. 3: Comparison of covariance intersection (CI), naïve fusion, optimal fusion using an infinite time horizon (Opt), inverse covariance intersection (ICI), square root decomposition-based fusion with time horizon $\mathcal{T}$ without bounding ($\text{SqDF}_{\mathcal{T}}$) and with bounding ($\text{SqDF}_{\mathcal{T},b}$) for the nonlinear example.

reducing the square root matrix even after it has been created and also allows further processing. Thus, it can keep track of correlations after the fusion already has taken place. Therefore it can potentially be used for sensor networks with unknown topologies and fusion that is hierarchical or decentralized.

VI. Conclusion

This paper proposed a novel recursive fusion technique using a moving horizon matrix of square root noise decompositions. To keep the fusion result consistent, a residual to account for discarded noise terms is stored and used for bounding. This technique offers a flexible trade-off between communication requirements and fusion result quality. Further research will investigate how this technique can be used in networks with hierarchical and decentralized topologies.

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] 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.

[3] A. G. Mutambara, *Decentralized Estimation and Control for Multisensor Systems*. CRC Press, 1998.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] J. Sijs and M. Lazar, "State-fusion with Unknown Correlation: Ellipsoidal Intersection," *Automatica*, vol. 48, no. 8, pp. 1874–1878, Aug. 2012.

[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] M. Reinhardt, B. Noack, and U. D. Hanebeck, "Closed-form Optimization of Covariance Intersection for Low-dimensional Matrices," in *Proceedings of the 15th International Conference on Information Fusion (Fusion 2012)*, Singapore, Jul. 2012.

[21] U. Orguner, "Approximate Analytical Solutions for the Weight Optimization Problems of CI and ICI," in *Proceedings of the IEEE ISIF Workshop on Sensor Data Fusion: Trends, Solutions, Applications (SDF 2017)*, Bonn, Germany, Oct. 2017.

[22] J. Ajgl, M. Šimandl, M. Reinhardt, B. Noack, and U. D. Hanebeck, "Covariance Intersection in State Estimation of Dynamical Systems," in *Proceedings of the 17th International Conference on Information Fusion (Fusion 2014)*, Salamanca, Spain, Jul. 2014.

[23] J. Ajgl and M. Šimandl, "On linear estimation fusion under unknown correlations of estimator errors," in *Proceedings of the 19th IFAC World Congress (IFAC 2014)*, Cape Town, South Africa, Aug. 2014.

[24] B. Chen, G. Hu, D. W. C. Ho, and L. Yu, "A new approach to linear/nonlinear distributed fusion estimation problem," *CoRR*, vol. abs/1708.08583, 2017.

[25] M. Zarei-Jalalabadi, S. M. B. Malaek, and S. S. Kia, "A Track-to-Track Fusion Method for Tracks With Unknown Correlations," *IEEE Control Systems Letter*, vol. 2, no. 2, pp. 189–194, Apr. 2018.

[26] 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.

[27] M. Reinhardt, B. Noack, and U. Hanebeck, "Reconstruction of Joint Covariance Matrices in Networked Linear Systems," in *2014 48th Annual Conference on Information Sciences and Systems, CISS 2014*, Mar. 2014.

[28] 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.

[29] 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.

[30] 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.

[31] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches / Dan Simon*. Hoboken, N.J.: Wiley [Chichester] : John Wiley [distributor], 2006.

[32] 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*, vol. 22, no. 6, pp. 803–805, Nov. 1986.

[33] V. Shin, Y. Lee, and T.-S. Choi, "Generalized Millman's Formula and its Application for Estimation Problems," *Signal Processing*, vol. 86, no. 2, Feb. 2006.

[34] 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.

[35] T. Lefebvre, H. Bruyninckx, and J. de Schutter, "Kalman Filters for Nonlinear Systems: A Comparison of Performance," *International Journal of Control*, vol. 77, no. 7, pp. 639–653, May 2004.

[36] S. J. Julier and J. K. Uhlmann, "Unscented Filtering and Nonlinear Estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.