Automatic Exploitation of Independencies for Covariance Bounding in Fully Decentralized Estimation

Benjamin Noack, Marcus Baum, and Uwe D. Hanebeck *

* *Intelligent Sensor-Actuator-Systems Laboratory (ISAS),*
Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT),
Karlsruhe, Germany;
(e-mail: noack@kit.edu, marcus.baum@kit.edu, uwe.hanebeck@ieee.org)

Abstract: Especially in the field of sensor networks and multi-robot systems, fully decentralized estimation techniques are of particular interest. As the required elimination of the complex dependencies between estimates generally yields inconsistent results, several approaches, e.g., covariance intersection, maintain consistency by providing conservative estimates. Unfortunately, these estimates are often too conservative and therefore, much less informative than a corresponding centralized approach. In this paper, we provide a concept that conservatively decorrelates the estimates while bounding the unknown correlations as closely as possible. For this purpose, known independent quantities, such as measurement noise, are explicitly identified and exploited. Based on tight covariance bounds, the new approach allows for an intuitive and systematic derivation of appropriate tailor-made filter equations and does not require heuristics. Its performance is demonstrated in a comparative study within a typical SLAM scenario.

1. INTRODUCTION

This paper is concerned with the problem of deriving state estimators for large-scale systems. Two basic scenarios can be identified, where efficient decentralized estimation algorithms are of prime importance. The first situation is concerned with the localization of sensor nodes in a large sensor network by relative range measurements between the nodes. The sensor network can comprise thousands of nodes, which leads to a very large state vector. The second scenario is the localization of a single or several robots that simultaneously perform mapping of an unknown environment. The number of landmarks even in a relatively structured office environment can go into the thousands, so that also in this robotic Simultaneous Localization and Mapping (SLAM) problem, the state vector can become extremely large.

Standard filtering methods cannot cope with problems of this size as the representation of the estimate does not scale linearly with the number of states. Thus, approximations are required that lead to a smaller storage and processing complexity.

The paper is structured as follows. Section 2 defines the state estimation problem solved in this paper. Section 3 then gives an overview of related state-of-the-art solutions. The basic principle of fully decentralized estimation is introduced in Section 4. Based on covariance bounds given in Section 4.1, a general recipe for deriving practical decentralized filters is given in Section 4.2. The automatic exploitation of independent noise terms is then shown in Section 4.3. Against the background of the SLAM problem, the derived results are exemplified by means of static and dynamic scenarios in Section 5.

Contributions of the paper

The paper provides an intuitive recipe for systematically deriving suboptimal but consistent filtering algorithms for large-scale systems that reduce both storage and processing complexity. With this approach, neither knowledge of complex nonlinear filters nor a deeper insight into the updating structure is required. By simply using the standard Kalman filter equations and the concept of covariance bounds, appropriate (implicitly nonlinear) filtering equations are derived automatically.

2. PROBLEM FORMULATION

The abstract problem we face is the estimation of a large state vector $\underline{x}_k$ that consists of certain substates according to

$$\underline{x}_k = \left[(\underline{x}_k^{(1)})^T, \ldots, (\underline{x}_k^{(N)})^T \right]^T \qquad (1)$$

from successive relative measurements between these substates. W.l.o.g. we assume that at a time step k always two substates $\underline{x}_k^{(i)}$ and $\underline{x}_k^{(j)}$ for $i, j \in \{1, \ldots, N\}$ of $\underline{x}_k$ are affected by a measurement $\hat{\underline{y}}_k$ that is a realization of the random vector $\underline{y}_k$. In the case of the considered sensor network example, substates are the locations of the individual nodes related by range measurements. For the robotic SLAM example, substates are given by the observer location and the landmark positions. The measurements

performed by the robot either relate substates corresponding to two landmarks or the substates corresponding to observer and a single landmark.

The standard solution for solving this state estimation problem is to consider the entire state vector $\underline{\boldsymbol{x}}_k$ and update the estimate of this state vector based on the measurements by means of a suitable filter. In the case of linear measurement equations, the Kalman filter is the best linear unbiased estimator and even the best estimator when the measurement noise is Gaussian. For the remainder of this paper, we assume that the considered measurement equations are either linear or an appropriate form of linearization has already been performed.

Centralized state estimation by means of a Kalman filter stores and maintains the entire covariance matrix (Smith et al. (1986)). This includes the block covariance matrices corresponding to the substates on its diagonal and the cross-covariance matrices relating the substates. The big advantage of storing and maintaining the cross-information between substates is that updating a single substate can lead to an update of many other substates, so that information is propagated through the entire state. However, this comes at the price of a high storage and processing complexity as the size of the covariance matrix is quadratic in the number of states.

In this paper, we desire to design a fully decentralized filter for solving the considered large-scale state estimation problems. For that purpose, the full state vector is split up into "natural" substates. Updating is restricted to the two substates affected by the current measurement. Only the covariance matrices corresponding to the substates are stored and maintained. All the cross-covariance matrices are discarded.

The substates in decentralized filtering could correspond to the basic substates described above or contain several of these basic substates for which it is desired to maintain a full covariance matrix. In the considered sensor network example, larger substates correspond to groups of nodes that form a natural cluster.

From all the available approaches for estimating the desired state, this form of decentralized filtering is the least costly solution in terms of both storage and processing complexity. On the other hand, by discarding cross-information between substates, the filter loses its ability to propagate measurement information through the state vector, i.e., to perform implicit updating (Castellanos et al. (1997)).

The major challenge that arises from discarding the cross-covariance matrices between substates, however, is the fact that this information is required even for the case of updating two substates by means of a relative measurement as will be shown in Section 4. Hence, the problem we solve in this paper is to reconstruct the desired cross-covariance matrices on the fly in order to facilitate the update process. As it is impossible to reconstruct the matrices exactly, appropriate upper bounds will be derived in Subsection 4.1.

3. RELATED WORK

In order to decentralize full covariance estimation approaches, the correlation structure between the partial states cannot be updated in its entirety anymore and therefore cannot be maintained. The consistency of the state estimates can then only be preserved, if the covariance matrix is bounded conservatively. The covariance intersection (Julier and Uhlmann (1997); Uhlmann et al. (1997)) or covariance bounds (Hanebeck et al. (2001); Hanebeck and Horn (2001b)) algorithms provide axis-aligned covariance matrices that bound the true covariance matrix. As it has been shown in (Julier and Uhlmann (2007); Hanebeck and Horn (2001a)), this approach generally results into too convervative estimates. Significantly more informative and less conservative estimates can be attained with the split covariance intersection technique, as proposed in (Julier and Uhlmann (2001, 2007)), or additional restrictions on the joint covariance matrix, as derived in (Hanebeck and Horn (2001a); Reece and Roberts (2005)). Independent noise is herein explicitly exploited and splitted off from the covariance matrix to be bounded, so that tighter bounds are possible. Unfortunately, these algorithms generally require an explicit identification and exploitation of the uncorrelated terms and are therefore rather complicated and less intuitive.

In this paper, we show that independent noise can automatically be exploited and that intuitive decentralized estimation formulas can be derived.

4. DECENTRALIZED ESTIMATION

In the following, we confine ourselves to those partial states i and j of (1), between which a measurement takes place. The two affected fusion nodes i and j *both* use a measurement equation according to

$$\hat{\underline{y}}_k = \left[\mathbf{H}_k^{(i)} \ \mathbf{H}_k^{(j)} \right] \begin{bmatrix} \underline{\boldsymbol{x}}_k^{(i)} \\ \underline{\boldsymbol{x}}_k^{(j)} \end{bmatrix} + \underline{\boldsymbol{v}}_k$$

that relates the two substates i and j with the given measurement $\hat{\underline{y}}_k$. Collecting the two substates $\underline{\boldsymbol{x}}_k^{(i)}$ and $\underline{\boldsymbol{x}}_k^{(j)}$ in a vector $\underline{\boldsymbol{x}}_k$ gives

$$\underline{\boldsymbol{x}}_k = \begin{bmatrix} \underline{\boldsymbol{x}}_k^{(i)} \\ \underline{\boldsymbol{x}}_k^{(j)} \end{bmatrix} \ .$$

This allows the application of the standard Kalman filter formulas

$$\hat{\underline{x}}_k^e = \hat{\underline{x}}_k^p + \mathbf{C}_k^p \mathbf{H}_k^T \left(\mathbf{C}_k^v + \mathbf{H}_k \mathbf{C}_k^p \mathbf{H}_k^T \right)^{-1} \left(\hat{\underline{y}}_k - \mathbf{H}_k \hat{\underline{x}}_k^p \right) \ ,$$

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

with the updated and the predicted state

$$\hat{\underline{x}}_k^e = \begin{bmatrix} \hat{\underline{x}}_k^{(e,i)} \\ \hat{\underline{x}}_k^{(e,j)} \end{bmatrix} , \ \hat{\underline{x}}_k^p = \begin{bmatrix} \hat{\underline{x}}_k^{(p,i)} \\ \hat{\underline{x}}_k^{(p,j)} \end{bmatrix} \ ,$$

respectively,

$$\mathbf{H}_k = \left[\mathbf{H}_k^{(i)} \ \mathbf{H}_k^{(j)} \right] \ ,$$

and

$$\mathbf{C}_k^e = \begin{bmatrix} \mathbf{C}_k^{(e,i,i)} & \mathbf{C}_k^{(e,i,j)} \\ \mathbf{C}_k^{(e,j,i)} & \mathbf{C}_k^{(e,j,j)} \end{bmatrix} , \ \mathbf{C}_k^p = \begin{bmatrix} \mathbf{C}_k^{(p,i,i)} & \mathbf{C}_k^{(p,i,j)} \\ \mathbf{C}_k^{(p,j,i)} & \mathbf{C}_k^{(p,j,j)} \end{bmatrix} \ .$$

The posterior expected values are given by

$$
\begin{bmatrix} \hat{\underline{x}}_k^{(e,i)} \\ \hat{\underline{x}}_k^{(e,j)} \end{bmatrix} = \begin{bmatrix} \hat{\underline{x}}_k^{(p,i)} \\ \hat{\underline{x}}_k^{(p,j)} \end{bmatrix} + \begin{bmatrix} \mathbf{C}_k^{(p,i,i)} & \mathbf{C}_k^{(p,i,j)} \\ \mathbf{C}_k^{(p,j,i)} & \mathbf{C}_k^{(p,j,j)} \end{bmatrix} \begin{bmatrix} \left(\mathbf{H}_k^{(i)}\right)^T \\ \left(\mathbf{H}_k^{(j)}\right)^T \end{bmatrix}
$$

$$
\left(\mathbf{C}_k^v + \begin{bmatrix} \mathbf{H}_k^{(i)} & \mathbf{H}_k^{(j)} \end{bmatrix} \begin{bmatrix} \mathbf{C}_k^{(p,i,i)} & \mathbf{C}_k^{(p,i,j)} \\ \mathbf{C}_k^{(p,j,i)} & \mathbf{C}_k^{(p,j,j)} \end{bmatrix} \begin{bmatrix} \left(\mathbf{H}_k^{(i)}\right)^T \\ \left(\mathbf{H}_k^{(j)}\right)^T \end{bmatrix} \right)^{-1}
$$

$$
\left(\hat{\underline{y}}_k - \begin{bmatrix} \mathbf{H}_k^{(i)} & \mathbf{H}_k^{(j)} \end{bmatrix} \begin{bmatrix} \hat{\underline{x}}_k^{(p,i)} \\ \hat{\underline{x}}_k^{(p,j)} \end{bmatrix} \right) \ ,
$$

while the posterior joint covariance matrix is

$$
\begin{bmatrix} \mathbf{C}_k^{(e,i,i)} & \mathbf{C}_k^{(e,i,j)} \\ \mathbf{C}_k^{(e,j,i)} & \mathbf{C}_k^{(e,j,j)} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_k^{(p,i,i)} & \mathbf{C}_k^{(p,i,j)} \\ \mathbf{C}_k^{(p,j,i)} & \mathbf{C}_k^{(p,j,j)} \end{bmatrix}
$$

$$
- \begin{bmatrix} \mathbf{C}_k^{(p,i,i)} & \mathbf{C}_k^{(p,i,j)} \\ \mathbf{C}_k^{(p,j,i)} & \mathbf{C}_k^{(p,j,j)} \end{bmatrix} \begin{bmatrix} \left(\mathbf{H}_k^{(i)}\right)^T \\ \left(\mathbf{H}_k^{(j)}\right)^T \end{bmatrix}
$$

$$
\left(\mathbf{C}_k^v + \begin{bmatrix} \mathbf{H}_k^{(i)} & \mathbf{H}_k^{(j)} \end{bmatrix} \begin{bmatrix} \mathbf{C}_k^{(p,i,i)} & \mathbf{C}_k^{(p,i,j)} \\ \mathbf{C}_k^{(p,j,i)} & \mathbf{C}_k^{(p,j,j)} \end{bmatrix} \begin{bmatrix} \left(\mathbf{H}_k^{(i)}\right)^T \\ \left(\mathbf{H}_k^{(j)}\right)^T \end{bmatrix} \right)^{-1}
$$

$$
\begin{bmatrix} \mathbf{H}_k^{(i)} & \mathbf{H}_k^{(j)} \end{bmatrix} \begin{bmatrix} \mathbf{C}_k^{(p,i,i)} & \mathbf{C}_k^{(p,i,j)} \\ \mathbf{C}_k^{(p,j,i)} & \mathbf{C}_k^{(p,j,j)} \end{bmatrix} \ .
$$

Obviously, the correlation information $\mathbf{C}_k^{(p,i,j)}$ is required for performing updates as well as the filtered estimates become correlated after the update, i.e., $\mathbf{C}_k^{(e,i,j)} \neq 0$, even if there has been no prior correlation. Discarding and ignoring the correlations provides by no means a decentralized fusion methodology and leads to inconsistent estimates. Therefore, we derive an algorithm that conservatively decorrelates the estimates.

4.1 Decorrelation of Estimates

The basic idea for reconstructing the correlation information in $\mathbf{C}_k^p$ is to replace $\mathbf{C}_k^p$ with a "larger" covariance matrix $\tilde{\mathbf{C}}$, i.e., $\tilde{\mathbf{C}} \geq \mathbf{C}_k^p$. In doing so, it can be ensured that no inconsistent estimation results are obtained, because the used covariance matrix is more conservative than the true one. For deriving possible covariance bounds $\tilde{\mathbf{C}}$, we exploit the fact that the union of the sigma bounds of all possible joint covariances is an axis-aligned convex set, see Figure 1.

Theorem 4.1. *Given a positive definite symmetric matrix* $\tilde{\mathbf{C}}$ *with*

$$
\tilde{\mathbf{C}} = \begin{bmatrix} \mathbf{C}_{xx} & \mathbf{C}_{xy} \\ \mathbf{C}_{yx} & \mathbf{C}_{yy} \end{bmatrix} \ ,
$$

a "larger" matrix $\mathbf{C} \geq \tilde{\mathbf{C}}$ *is given by*

$$
\mathbf{C} = \begin{bmatrix} \dfrac{1}{0.5 - \kappa} \mathbf{C}_{xx} & \mathbf{0} \\ \mathbf{0} & \dfrac{1}{0.5 + \kappa} \mathbf{C}_{yy} \end{bmatrix} \ ,
$$

with $\kappa \in (-0.5, 0.5)$.

PROOF. Follows from

$$
\begin{bmatrix} \underline{x}^T & \underline{y}^T \end{bmatrix} \left(\mathbf{C} - \tilde{\mathbf{C}} \right) \begin{bmatrix} \underline{x} \\ \underline{y} \end{bmatrix} \geq 0 \text{ for all } \begin{bmatrix} \underline{x} \\ \underline{y} \end{bmatrix} \ .
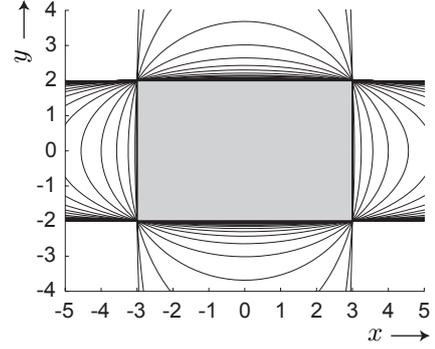$$

Fig. 1. Family of covariance bounds for given marginal distributions, see Hanebeck et al. (2001).

$\square$

The above theorem yields a family of possible covariance bounds, namely one covariance bound for each $\kappa \in (-0.5, 0.5)$ (see Figure 1). The particular value for κ is chosen in order to minimize a certain criterium, like the degree of uncertainty of the fusion result (see the next Section). Note that covariance bounds also exist for unknown but constrained cross-correlations, see Hanebeck et al. (2001) and Reece and Roberts (2005).

4.2 Recipe for Designing Decentralized Estimators

Based on the concept of covariance bounds derived in the previous subsection, a recipe for intuitively and systematically deriving suboptimal but consistent filtering algorithms for large-scale systems is given in the following. The estimator design consists of four steps:

(1) Select the system and measurement equation. These equations should only incorporate relevant substates. In this manner, only correlation information, which is required to perform the measurement update / time update, has to be reconstructed.
(2) Write down the associated Kalman filter equations by assuming all required information to be available.
(3) Identify the missing information, i.e., correlations. Unknown correlations may occur between the state vector or parts of the state vector and also the noise terms.
(4) Replace the missing information with a bounding covariance matrix according to Theorem 4.1. If further constraints on the correlations are known, even tighter bounds as described in Hanebeck et al. (2001) can be used. The filter equations now depend on a scalar parameter κ.

The above filter design recipe automatically yields a nonlinear filter that works as follows:

(1) Perform the measurement update / time update dependent on the scalar parameter κ.
(2) Determine an optimal κ_{opt} according to a suitable criterion, e.g., the determinant or trace of the updated covariance matrix.
(3) Use the optimal κ_{opt} to compute the final estimate.

The estimator for unknown correlations is only slightly more complex than the original estimator. It is only required to additionally choose the scalar parameter κ.

For practical estimators, this is usually quite easy and can be done by employing a proper optimization algorithm. Even though the correlations are unknown, the obtained estimator is guaranteed to be consistent.

4.3 Exploiting Independent Measurement Noise

In the following, the filter equations for the decentralized approach based on the covariance bounds are derived. For this purpose, we follow the recipe given in the previous Section 4.2.

The first three steps of the recipe have already been done in the beginning of Section 4. The missing information in the filter step are the cross-covariance matrices of the predicted state $\mathbf{C}_k^{(p,i,j)}$ and $\mathbf{C}_k^{(p,j,i)}$ in Section 4. Hence, according to Step (4), $\mathbf{C}_k^p$ is replaced by its covariance bound according to Theorem 4.1, i.e.,

$$\mathbf{C}_k^p \le \begin{bmatrix} \dfrac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} & \mathbf{0} \\ \mathbf{0} & \dfrac{\mathbf{C}_k^{(p,j,j)}}{0.5 + \kappa_k} \end{bmatrix} ,$$

In analogy to Section 4, applying the Kalman Filter equations yields the estimates

$$\begin{bmatrix} \hat{\underline{x}}_k^{(e,i)} \\ \hat{\underline{x}}_k^{(e,j)} \end{bmatrix} = \begin{bmatrix} \hat{\underline{x}}_k^{(p,i)} \\ \hat{\underline{x}}_k^{(p,j)} \end{bmatrix} + \begin{bmatrix} \dfrac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} & \mathbf{0} \\ \mathbf{0} & \dfrac{\mathbf{C}_k^{(p,j,j)}}{0.5 + \kappa_k} \end{bmatrix} \begin{bmatrix} \left(\mathbf{H}_k^{(i)}\right)^T \\ \left(\mathbf{H}_k^{(j)}\right)^T \end{bmatrix}$$

$$\left(\mathbf{C}_k^v + \begin{bmatrix} \mathbf{H}_k^{(i)} & \mathbf{H}_k^{(j)} \end{bmatrix} \begin{bmatrix} \dfrac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} & \mathbf{0} \\ \mathbf{0} & \dfrac{\mathbf{C}_k^{(p,j,j)}}{0.5 + \kappa_k} \end{bmatrix} \begin{bmatrix} \left(\mathbf{H}_k^{(i)}\right)^T \\ \left(\mathbf{H}_k^{(j)}\right)^T \end{bmatrix} \right)^{-1}$$

$$\left(\hat{y}_k - \begin{bmatrix} \mathbf{H}_k^{(i)} & \mathbf{H}_k^{(j)} \end{bmatrix} \begin{bmatrix} \hat{\underline{x}}_k^{(p,i)} \\ \hat{\underline{x}}_k^{(p,j)} \end{bmatrix} \right) ,$$

and covariance matrix

$$\begin{bmatrix} \mathbf{C}_k^{(e,i,i)} & \mathbf{C}_k^{(e,i,j)} \\ \mathbf{C}_k^{(e,j,i)} & \mathbf{C}_k^{(e,j,j)} \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} & \mathbf{0} \\ \mathbf{0} & \dfrac{\mathbf{C}_k^{(p,j,j)}}{0.5 + \kappa_k} \end{bmatrix}$$

$$- \begin{bmatrix} \dfrac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} & \mathbf{0} \\ \mathbf{0} & \dfrac{\mathbf{C}_k^{(p,j,j)}}{0.5 + \kappa_k} \end{bmatrix} \begin{bmatrix} \left(\mathbf{H}_k^{(i)}\right)^T \\ \left(\mathbf{H}_k^{(j)}\right)^T \end{bmatrix}$$

$$\left(\mathbf{C}_k^v + \begin{bmatrix} \mathbf{H}_k^{(i)} & \mathbf{H}_k^{(j)} \end{bmatrix} \begin{bmatrix} \dfrac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} & \mathbf{0} \\ \mathbf{0} & \dfrac{\mathbf{C}_k^{(p,j,j)}}{0.5 + \kappa_k} \end{bmatrix} \begin{bmatrix} \left(\mathbf{H}_k^{(i)}\right)^T \\ \left(\mathbf{H}_k^{(j)}\right)^T \end{bmatrix} \right)^{-1}$$

$$\begin{bmatrix} \mathbf{H}_k^{(i)} & \mathbf{H}_k^{(j)} \end{bmatrix} \begin{bmatrix} \dfrac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} & \mathbf{0} \\ \mathbf{0} & \dfrac{\mathbf{C}_k^{(p,j,j)}}{0.5 + \kappa_k} \end{bmatrix} .$$

At this point it is important to note that the independence of the measurement noise was explicitly used in the measurement update. Further simplifications then lead to the updated estimates of the desired partial states

$$\hat{\underline{x}}_k^{(e,i)} = \hat{\underline{x}}_k^{(p,i)} + \frac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} \left(\mathbf{H}_k^{(i)}\right)^T$$

$$\left(\mathbf{C}_k^v + \mathbf{H}_k^{(i)} \frac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} \left(\mathbf{H}_k^{(i)}\right)^T + \mathbf{H}_k^{(j)} \frac{\mathbf{C}_k^{(p,j,j)}}{0.5 + \kappa_k} \left(\mathbf{H}_k^{(j)}\right)^T \right)^{-1}$$

$$\left(\hat{\underline{y}}_k - \mathbf{H}_k^{(i)} \hat{\underline{x}}_k^{(p,i)} - \mathbf{H}_k^{(j)} \hat{\underline{x}}_k^{(p,j)} \right) ,$$

with covariance matrix

$$\mathbf{C}_k^{(e,i,i)} = \frac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} - \frac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} \left(\mathbf{H}_k^{(i)}\right)^T$$

$$\left(\mathbf{C}_k^v + \mathbf{H}_k^{(i)} \frac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} \left(\mathbf{H}_k^{(i)}\right)^T + \mathbf{H}_k^{(j)} \frac{\mathbf{C}_k^{(p,j,j)}}{0.5 + \kappa_k} \left(\mathbf{H}_k^{(j)}\right)^T \right)^{-1}$$

$$\mathbf{H}_k^{(i)} \frac{\mathbf{C}_k^{(p,i,i)}}{0.5 - \kappa_k} .$$

Actually, these updated estimates depend on the scalar parameter κ_k, which specifies a particular covariance bound. A suitable κ_k can for instance be chosen in such a way that the determinant or trace of $\mathbf{C}_k^{(e,i,i)}$ is as small as possible, i.e., the most certain estimate. In order to avoid a numerical calculation of κ_k, approximate solutions, as derived in (Niehsen (2002); Fränken and Hüpper (2005)) can be employed.

Similar expressions are obtained for the dual update step for $\hat{\underline{x}}_k^{(e,j)}$ and $\mathbf{C}_k^{(e,j,j)}$.

5. SIMULATIONS

5.1 1D Example

In order to exemplify the derived concept, we employ the illustrative example from Hanebeck and Horn (2001a), where a robot estimates its position between two parallel walls by means of distance measurements. During the first 100 time steps, only measurements to the lower wall are available. In the 100 successive time steps, both walls are measured. The true one-dimensional positions are 10 m for the robot and the walls are located at 0 m and 20 m. The prior estimates are $\hat{\underline{x}}_k^R = 11$ m for the robot and $\hat{x}_k^1 = 1$ m and $\hat{x}_k^2 = 19.9$ m for the lower and upper wall, respectively. The corresponding variances are $C_k^R = 4\,\mathrm{m}^2$, $C_k^1 = 1\,\mathrm{m}^2$, and $C_k^R = 0.01\,\mathrm{m}^2$. The measurements are affected by a zero-mean Gaussian noise $\boldsymbol{v}_k$ with variance $0.09\,\mathrm{m}^2$. The positions are then related to the measurements through the models

$$\hat{\underline{y}}_k^1 = [1 \ -1 \ 0] \begin{bmatrix} x_k^R \\ x_k^1 \\ x_k^2 \end{bmatrix} + \boldsymbol{v}_k \ \text{and} \ \hat{\underline{y}}_k^2 = [1 \ 0 \ -1] \begin{bmatrix} x_k^R \\ x_k^1 \\ x_k^2 \end{bmatrix} + \boldsymbol{v}_k .$$

The results of the full covariance formulation are shown in Fig. 2, which represent the optimal solution to the estimation problem. Neglecting the correlation results into the inconsistent estimates shown in Fig. 3. In contrast, Fig. (4) presents very conservative estimates. For these estimates, the covariance intersection algorithm has been applied. As shown in Fig. (5), the estimates can become significantly more informative and smoother, when the
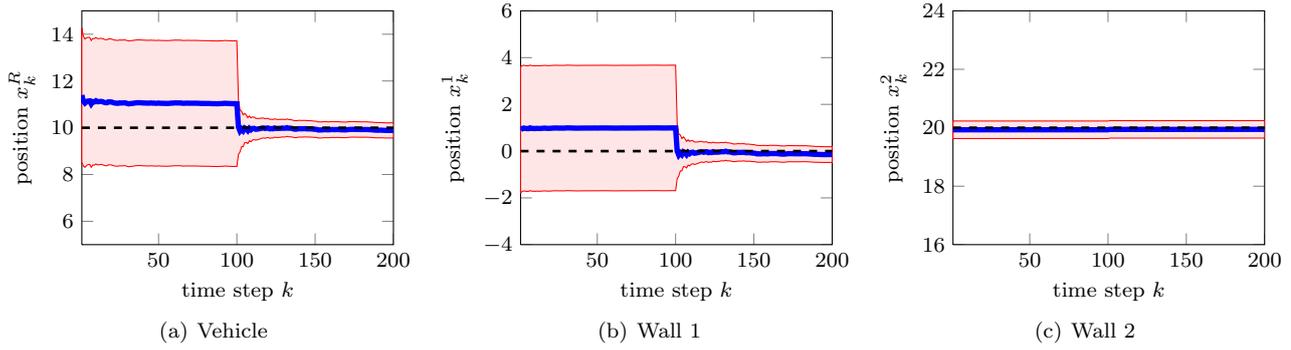
(a) Vehicle   (b) Wall 1   (c) Wall 2

Fig. 2. The state estimates of the positions are depicted in blue and their 3σ-bounds are drawn red. During the first 100 steps, only wall 1 is measured by the robot. All correlations are maintained, which yields the optimal state estimates.
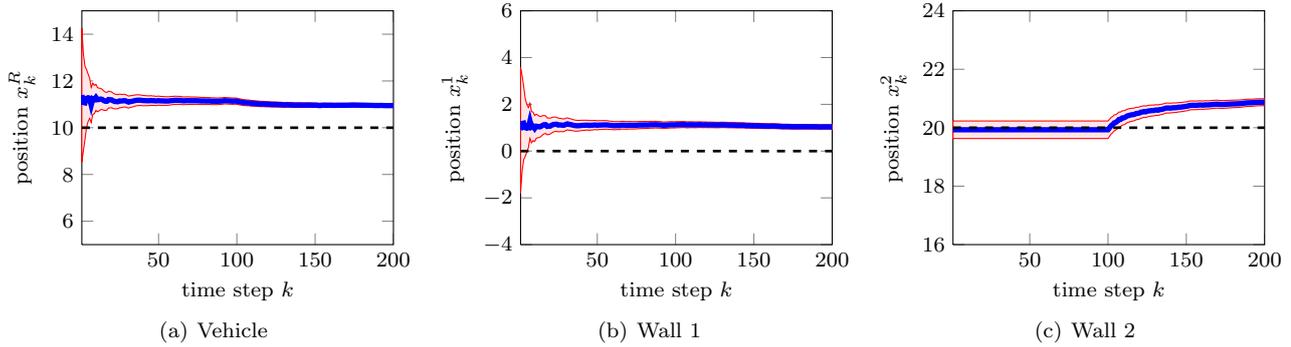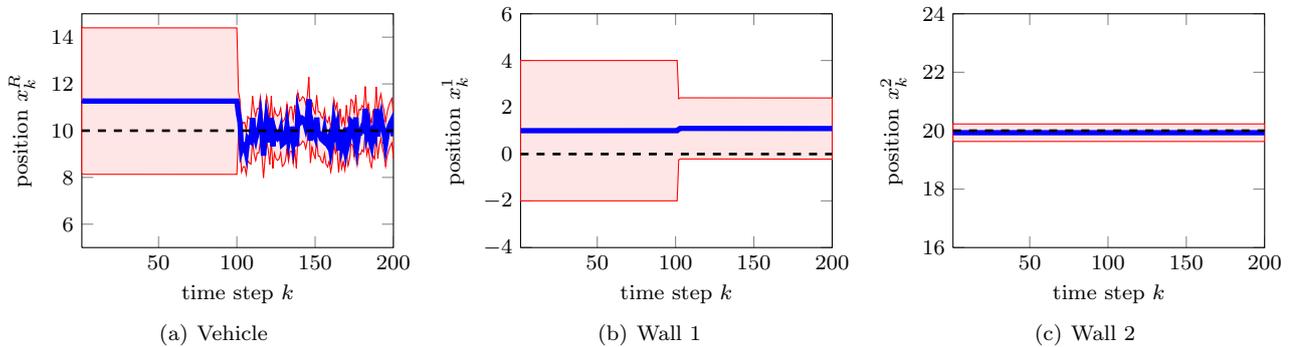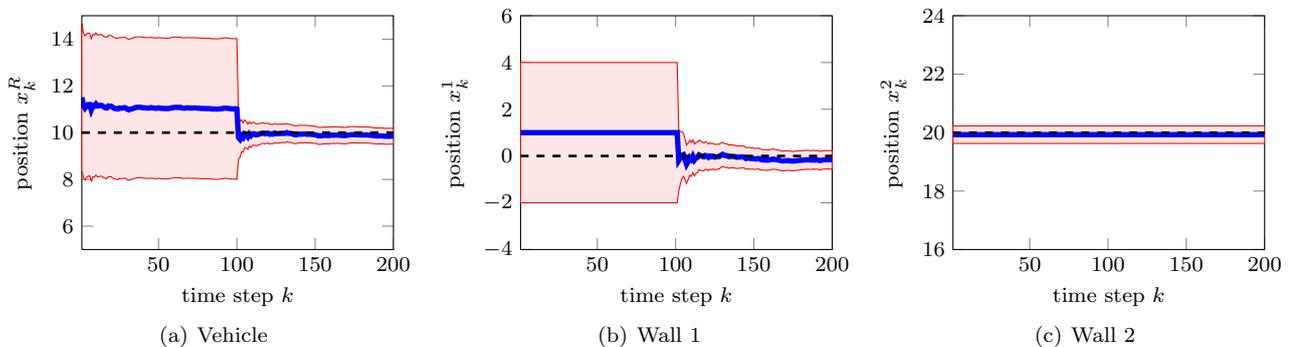

(a) Vehicle   (b) Wall 1   (c) Wall 2

Fig. 3. The state estimates of the positions are depicted in blue and their 3σ-bounds are drawn red. During the first 100 steps, only wall 1 is measured by the robot. Correlations are omitted and the estimates become inconsistent.

(a) Vehicle   (b) Wall 1   (c) Wall 2

Fig. 4. The state estimates of the positions are depicted in blue and their 3σ-bounds are drawn red. During the first 100 steps, only wall 1 is measured by the robot. Correlations are omitted, but bounded conservatively by covariance intersection.

(a) Vehicle   (b) Wall 1   (c) Wall 2

Fig. 5. The state estimates of the positions are depicted in blue and their 3σ-bounds are drawn red. During the first 100 steps, only wall 1 is measured by the robot. Correlations are omitted, but bounded conservatively. The parameter κ_k is chosen for each partial update seperately. This estimation algorithm yields results close to the full covariance formulation.
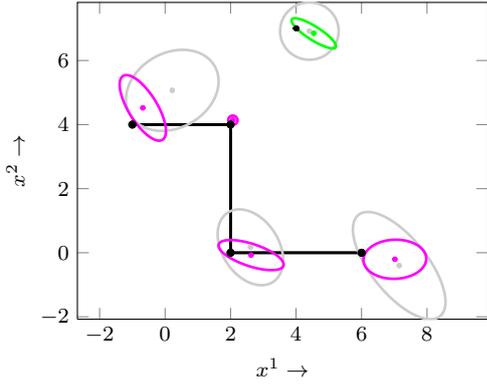
Fig. 6. Simulation results for Full SLAM: The prior estimates are plotted in grey. The estimation result for the walls are plotted in purple. The vehicle's position estimate is green.
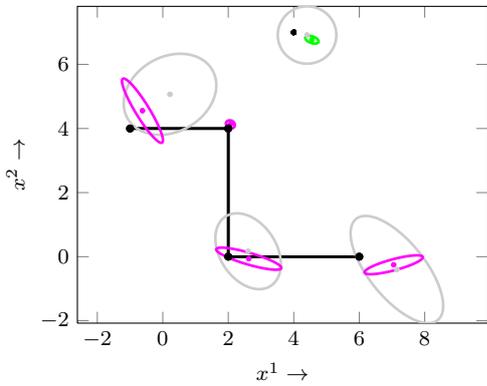

Fig. 7. Neglecting correlations. Especially the state of the vehicle is dramatically underestimated.

Fig. 8. Standard covariance intersection.

Fig. 9. Presented Approach, where the independent measurement noise is exploited.

proposed filter is applied. The results now come close to the optimal estimates, which is due to the automatic exploitation of the independent measurement noise.

5.2 2D Example: Relative Range Measurements

In this subsection, relative range measurements between a robot and randomly selected landmarks are performed in order to estimate the positions of the robot and of the landmarks simultaneously. Since the Euclidean distance is employed, which yields a nonlinear measurement equation, we confine ourselves to an idealized version of linearization: The measurement mapping is linearized around the true values, which are typically unknown to the estimator. In doing so, we avoid the introduction of additional effects that would result from linearization around estimated values and can therefore analyze the approaches more closely. For the purpose of illustration, first, a simulation with a non-moving stationary robot and only a few landmarks is considered. Subsequently, a realistic scenario with a large number of landmarks and a moving mobile robot is treated.

Stationary Robot   The landmarks in this scenario represent the corners of walls, whose true positions are depicted as black lines in Fig. 6-9. The northeastern black dot represents the true position of the robot. The prior knowledge for each position estimate is shown as light gray circles. For this simulation, 100 measurements are performed.
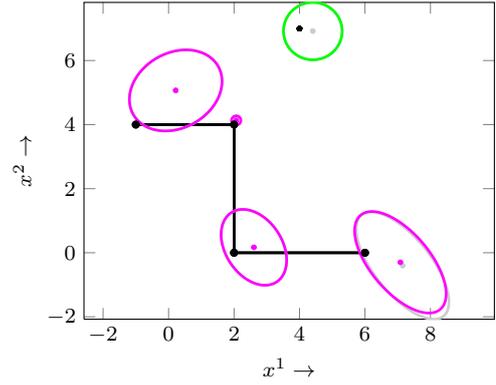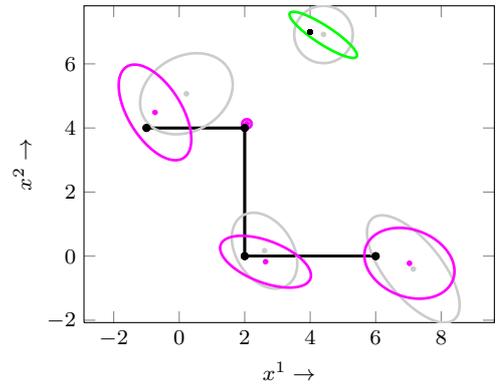
Fig. 6 presents the optimal estimation results, where all correlations are maintained. In Fig. 7, the correlations are omitted, so that inconsistent estimates are obtained and especially the robot's position is severely underestimated. Covariance intersection guarantees consistent estimates, but the updated estimates can barely be distinguished from the prior estimates. The estimates exploiting the independent measurement noise, depicted in Fig. 9, are significantly better and approach the optimal results.

Mobile Robot   In a second simulation, we consider a mobile robot that moves along the black dashed trajectory in Fig. 10. The motion model is linearized at each time instant and an extended Kalman filter prediction is employed. The mobile robot uses the same measurement model as in the above example. The sensors have a maximum range of 10 m. At each time instant, the distance to every landmark within this range is measured and its position estimate is updated. 100 Monte Carle runs have been performed for each filter algorithm the corresponding root mean squared errors (rmse) are depicted in Fig. 11. The rmse has been calcaluted for the entire state vector comprising the robot's state as well as each landmark position. The full covariance Kalman filter provides the best performance, whereas ignoring cross-correlations finally yields the worst results. The proposed approach shows constantly better results that the general CI approach.
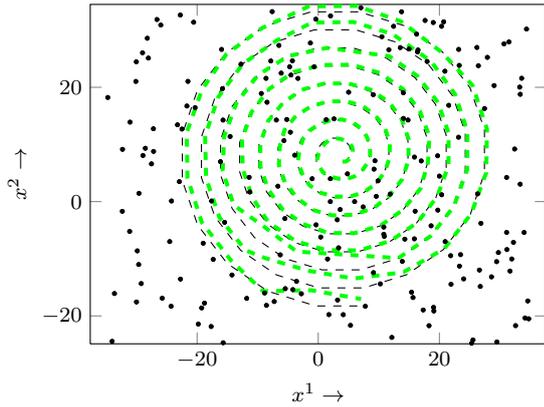
Fig. 10. Trajectory of the mobile robot and 200 uncertain landmarks. An example of an estimated trajectory is plotted in green.
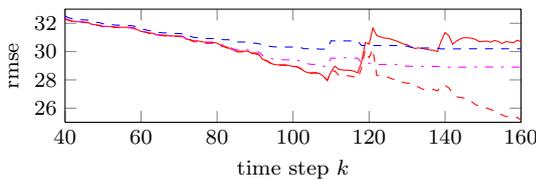

Fig. 11. Root mean squared errors of full covariance Kalman filter (dashed, red), decentralized Kalman filter (solid, red) with ignored cross-correlations, general CI (dashed, blue), and the proposed approach (dash-dotted, magenta).

6. CONCLUSIONS

This paper presents a new approach for designing suboptimal filters for large-scale estimation problems based on covariance matrix bounds. Instead of applying complicated filtering concepts as in Hanebeck and Horn (2001a), Julier and Uhlmann (2007), by using the new approach, a designer just

(1) selects the desired system and measurement equation,
(2) writes down the associated filter equations by assuming all required information to be available,
(3) identifies the missing information,
(4) and conservatively reconstructs this information by means of covariance bounds.

Dependency structures are optimally exploited in order to avoid overly conservative bounds, where known correlations and unknown correlations are distinguished. This is performed in an intuitive, uncomplicated, and systematic way.

The focus of this paper is on fully decentralized filter structures that split the state vector into certain substates and only maintain the covariance matrices of these substates. In addition, independence of the measurement noise is assumed and exploited, which leads to a dramatic increase in performance. Of course, the approach can be used for treating more general filter structures and more complicated dependence / independence structures.

It is important to note that no heuristics or special insights into the system operation are required for deriving appropriate filter equations. Only a single parameter selecting a specific member from the family of tightest covariance bounds needs to be optimized at every update step.

REFERENCES

Castellanos, J.A., Tardós, J.D., and Schmidt, G. (1997). Building a Global Map of the Environment of a Mobile Robot: The Importance of Correlations. In *IEEE International Conference on Robotics and Automation.*

Fränken, D. and Hüpper, A. (2005). Improved Fast Covariance Intersection for Distributed Data Fusion. In *Proceedings of the 8th International Conference on Information Fusion (Fusion 2005).* Philadelphia, Pennsylvania, USA.

Hanebeck, U.D., Briechle, K., and Horn, J. (2001). A Tight Bound for the Joint Covariance of Two Random Vectors with Unknown but Constrained Cross–Correlation. In *Proceedings of the 2001 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2001)*, 85–90. Baden–Baden, Germany.

Hanebeck, U.D. and Horn, J. (2001a). An Efficient Method for Simultaneous Map Building and Localization. In *Proceedings of SPIE, Vol. 4385, AeroSense Symposium.* Orlando, Florida.

Hanebeck, U.D. and Horn, J. (2001b). New Estimators for Mixed Stochastic and Set Theoretic Uncertainty Models: The General Case. In *Proceedings of the 2001 American Control Conference (ACC 2001).* Arlington, Virginia.

Julier, S. and Uhlmann, J.K. (1997). A Nondivergent Estimation Algorithm in the Presence of Unknown Correlations. In *Proceedings of the American Control Conference (ACC'1997).*

Julier, S.J. and Uhlmann, J.K. (2007). Using Covariance Intersection for SLAM. volume 55, 3 – 20. Simultaneous Localisation and Map Building.

Julier, S. and Uhlmann, J. (2001). Simultaneous Localisation and Map Building using Split Covariance Intersection. volume 3, 1257 –1262 vol.3.

Niehsen, W. (2002). Information Fusion based on Fast Covariance Intersection Filtering. In *Proceedings of the 5th International Conference on Information Fusion (Fusion 2002).* Annapolis, Maryland, USA.

Reece, S. and Roberts, S. (2005). Robust, Low-Bandwidth, Multi-Vehicle Mapping. In *Proceedings of the 8th International Conference on Information Fusion (Fusion 2005).* Philadelphia, Pennsylvania, USA.

Smith, R., Self, M., and Cheeseman, P. (1986). Estimating Uncertain Spatial Relationships in Robotics. In *Proceedings of the Proceedings of the Second Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-86)*, 267–288. Elsevier Science, New York, NY.

Uhlmann, J.K., Julier, S.J., and Csorba, M. (1997). Nondivergent Simultaneous Map Building and Localization using Covariance Intersection. volume 3087, 2–11. SPIE.