Author:	Hu, Jinwen; Xie, Lihua; Zhang, Cishen
Title:	Diffusion Kalman filtering based on covariance intersection
Editor:	Sergio Bittanti, Angelo Cenedese and Sandro Zampieri
Conference name:	18th International Federation of Automatic Control (IFAC) World Congress
Conference location:	Milan, Italy
Conference dates:	28 August - 02 September 2011
Series title and volume:	18, no. 1
Place published:	Italy
Publisher:	International Federation of Automatic Contro
Year:	2011
URL:	http://hdl.handle.net/1959.3/194068

Preprints of the 18th IFAC World Congress
Milano (Italy) August 28 - September 2, 2011

Diffusion Kalman Filtering Based on Covariance Intersection

Jinwen Hu, Lihua Xie, Cishen Zhang

School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: {HU0002EN, ELHXIE, ECSZhang}@ntu.edu.sg).

Abstract: This paper is concerned with distributed Kalman filtering for linear time-varying systems over multi-agent sensor networks. We propose a diffusion Kalman filtering algorithm based on a covariance intersection method, where local estimates are fused by incorporating the covariance information of local Kalman filters. Our algorithm leads to a stable estimate for each agent regardless of whether the system is uniformly observable with respect to local measurements as long as the system is uniformly observable under global sensor measurements and the communication is sufficiently fast compared to the sampling period. Simulation results validate the effectiveness of the proposed distributed Kalman filtering algorithm.

1. INTRODUCTION

Distributed fusion and estimation is an important topic in sensor networks and has been studied for applications such as environmental monitoring, surveillance, target tracking, etc; see, for example, [Li et al., 2002] and the references therein. In these applications, sensors installed on static or mobile agents can obtain measurements in a parallel manner. The communication and computation capabilities of each agent enables acquiring information from neighbors and processing it individually to get an estimate of a concerned physical quantity. The distributed processing greatly alleviates the computation load as taken by the fusion center in the centralized fusion and estimation. The distributed fusion and estimation problem has been considered in many applications [Li et al., 2002, Chang et al., 1997, Mori et al., 2002].

Recently, the distributed Kalman filtering received great attentions. In Hashemipour et al. [1988], a decentralized hierarchical structure was proposed, in which the estimates of local Kalman filters are calculated individually by each agent and then combined by a fusion center. A distributed Kalman filtering method using weighted averaging was proposed in Alriksson and Rantzer [2006], which requires the global information of the state covariances. Consensus-based distributed Kalman filters were proposed in Spanos et al. [2005], Olfati-Saber [2005, 2007], where local measurements are exchanged among neighbors in order to get the same estimate for each agent in the steady state and such estimate is exactly the global optimal estimate by a centralized estimation. However, multiple consensus iterations are usually needed to reach the steady state for the next Kalman update. Cattivelli and Sayed [2010] proposed a distributed Kalman filtering (DKF) algorithm, in which each agent exchanges their measurements as well as the pre-estimate of each agent. Measurements of neighboring agents are integrated to perform the local Kalman filtering and get a pre-estimate of the state, after which the pre-estimates of neighboring agents are fused locally by a convex combination to refine the pre-estimates. Different from the consensus approaches, this new strategy does not rely on multiple communication steps to reach a steady state for all agents before the Kalman update, which improves the efficiency of incorporating new measurement information. This diffusion strategy is of more practical use when dealing with dynamic state vectors, where new measurements must be processed in a timely manner instead of waiting for a consensus to be achieved. In Cattivelli and Sayed [2009], the DKF algorithm was further developed with adaptive weights by optimizing a locally defined cost function. However, this adaptive rule cannot guarantee the estimation stability for each agent. Moreover, the estimation covariance information is not taken into consideration in the choice of combination weights in Cattivelli and Sayed [2010] and Cattivelli and Sayed [2009], which we believe can play an important role in improving the estimation performance.

Most of the existing distributed fusion methods involving the covariance information are based on the assumption that correlations in the estimates are known to each neighboring agent [Li et al., 2002, Chang et al., 1997, Mori et al., 2002]. However, in a decentralized network, each agent only has information about its local topology and neighbors' estimates, and the cross-correlation of the estimates between each two agents is usually unknown. In Julier and Uhlmann [1997, 2001], the Covariance Intersection (CI) algorithm was proposed for fusion of multiple consistent estimates with unknown correlations, which uses a convex combination of the estimates and chooses the combination weights by minimizing the trace or determinant of an upper bound of the error covariance matrix. Julier and Uhlmann [1997], Uhlmann et al. [1999], Arambel et al. [2002], Wang and Li [2009] applied the CI algorithm in the state estimation together with Kalman filter, where the estimates of individual Kalman filters are fused locally by the CI algorithm. However, the stability of the estimation may not be guaranteed by the pure CI algorithm when the local observability is lost.

In this paper, we propose a diffusion Kalman filtering algorithm based on covariance intersection (CI-DKF). Our algorithm allows each agent to obtain a stable estimate (i.e. with bounded error covariance) by sharing information only with its neighbors. Different from the DKF algorithm proposed in [Cattivelli and Sayed, 2010], our estimates are fused by the CI algorithm which incorporates the error covariance information. The CI-DKF algorithm is mainly applied in the case where local observability is lost for some agents. A consensus-based information gathering scheme is designed to rebuild the local observability within a finite time duration. We prove that the covariance of the estimation error of each agent is bounded if the uniform observability condition is satisfied globally.

This paper is organized as follows. We first give the system model and recall an existing DKF algorithm in Section 2. In Section 3, a choice rule of adaptive weights is designed based on the CI algorithm. Then, we propose the CI-DKF algorithm for the cases where local observability is lost in Section 4. The effectiveness of the CI-DKF algorithm is testified by simulation in Section 5.

2. BACKGROUND

2.1 System Description

We consider a set of N agents with limited communication range and spatially distributed over a surveillance region. Agent k takes measurement $y_{k,i}$ of a common environment state x_i independently at time i with individual observation matrix $H_{k,i}$. The state-space model for each agent k is of the form:

$$\begin{aligned} x_{i+1} &= F_i x_i + w_i \\ y_{k,i} &= H_{k,i} x_i + v_{k,i} \end{aligned} \tag{1}$$

where w_i is the process noise and $v_{k,i}$ the measurement noise of agent k at time i. Hence, if the measurement information is processed in a centralized manner, we apply the augmented forms:

$$y_i = \begin{bmatrix} y_{1,i} \\ \vdots \\ y_{N,i} \end{bmatrix}, \quad H_i = \begin{bmatrix} H_{1,i} \\ \vdots \\ H_{N,i} \end{bmatrix}, \quad v_i = \begin{bmatrix} v_{1,i} \\ \vdots \\ v_{N,i}. \end{bmatrix}$$

w_i and $v_{k,i}$ are assumed to be zero-mean, uncorrelated and white with

$$\mathrm{E} \begin{bmatrix} w_i \\ v_i \end{bmatrix} \begin{bmatrix} w_j \\ v_j \end{bmatrix}^T = \begin{bmatrix} Q_i & 0 \\ 0 & R_i \end{bmatrix} \delta_{ij}, \quad \mathrm{E} v_{k,i} v_{l,j}^T = R_{k,i} \delta_{kl} \delta_{ij}$$

where δ_{ij} is the Kronecker delta. Q_i and $R_{k,i}$ are assumed to be positive definite and upper bounded. The matrices F_i and $H_{k,i}$ are also assumed to be bounded.

The topology of the network of all agents can be modeled by an undirected graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$. $\mathcal{V} = 1, 2, \ldots, N$ is the vertex set and $\mathcal{E} \subset \{\{k,l\} | k,l \in \mathcal{V}\}$ is the edge set, where each edge $\{k,l\}$ is an unordered pair of distinct agents. The graph or the network is connected if for any two vertices k and l there exists a sequence of edges (a path) $\{k,h_1\}, \{h_1,h_2\}, \ldots, \{h_{n-1},h_n\}, \{h_n,l\}$ in $\mathcal{E}$. The graph can be time-varying with $\mathcal{G}_i = (\mathcal{E}_i, \mathcal{V})$, where $\mathcal{E}_i$ is the edge set at time i. Let $\mathcal{N}_{k,i} = \{l \in \mathcal{V} | k,l \in \mathcal{E}_i\}$ denote the set of neighbors of agent k at time i. An agent is assumed to be a neighbor of itself, i.e. $k \in \mathcal{N}_{k,i}$. The degree (number of neighbors) of agent k at time i is denoted as $d_{k,i} = |\mathcal{N}_{k,i}|$.

Let the set $\{k_{m_i}\}$ $(m_i = 1, \ldots, d_{k,i})$ denote the indices of the neighbors of agent k at time i. Then, we can define the local observation matrix for each k:

$$H_{k,i}^{\mathrm{loc}} \triangleq \mathrm{col}\left\{H_{k_1,i}, H_{k_2,i}, \ldots, H_{k_{d_{k,i}},i}\right\}.$$

2.2 Diffusion Kalman Filtering

The DKF algorithm proposed in Cattivelli and Sayed [2010] is shown by Algorithm 1. The objective of the DKF implementations is for every agent k in the network to compute an estimate of the unknown state x_i, while sharing data only with its neighbors $\{l \in \mathcal{N}_{k,i}\}$. We denote the estimate of x_i obtained by node k and based on local observations up to time j as $\hat{x}_{k,i|j}$. At every time instant i, agents communicate to their neighbors the quantities $H_{k,i}^T R_{k,i}^{-1} H_{k,i}$ and $H_{k,i}^T R_{k,i}^{-1} y_{k,i}$ for the incremental update and the intermediate estimate $\psi_{k,i}$ for the diffusion update. The diffusion update in Algorithm 1 requires the introduction of a $pN \times pN$ diffusion matrix $\mathcal{C}_i$ subject to:

$$\mathcal{C}_i \mathbf{1} = \mathbf{1} \tag{2}$$

where $\mathbf{1}$ is an $Np \times 1$ column vector with unit entries. $\mathcal{C}_i$ can be partitioned into $p \times p$ blocks with each block $C_{k,l,i}\,(k,\, l = 1, 2, \ldots, p)$ being an $N \times N$ matrix as a weight to combine the estimate $\psi_{l,i}$.

Algorithm 1. (Diffusion Kalman filtering)

Start with $x_{k,0|-1} = 0$, $P_{k,0|-1} = \Pi_0$ and $i = 0$ for all k, given a pre-defined diffusion matrix $\mathcal{C} = \{C_{k,l}\}_{Np \times Np}$:

Step 1: Incremental Update:

$$S_{k,i} = \sum_{l \in \mathcal{N}_{k,i}} H_{l,i}^T R_{l,i}^{-1} H_{l,i}$$

$$q_{k,i} = \sum_{l \in \mathcal{N}_{k,i}} H_{l,i}^T R_{l,i}^{-1} y_{l,i}$$

$$P_{k,i|i}^{-1} = P_{k,i|i-1}^{-1} + S_{k,i}$$

$$\psi_{k,i} = \hat{x}_{k,i|i-1} + P_{k,i|i}\left[q_{k,i} - S_{k,i}\hat{x}_{k,i|i-1}\right]$$

Step 2: Diffusion Update:

$$\hat{x}_{k,i|i} = \sum_{l \in \mathcal{N}_{k,i}} C_{k,l} \psi_{l,i}$$

$$\hat{x}_{k,i+1|i} = F_i \hat{x}_{k,i|i}$$

$$P_{k,i+1|i} = F_i P_{k,i|i} F_i^T + Q_i$$

$$i \leftarrow i + 1.$$

3. CI-BASED ADAPTIVE DIFFUSION MATRIX

The diffusion matrix $\mathcal{C}_i$ plays an important role in the diffusion update. A fixed $\mathcal{C}_i$ is used in Cattivelli and Sayed [2010], and an adaptive $\mathcal{C}_i$ is further developed in Cattivelli and Sayed [2009] where agent k adaptively chooses the combination weights $C_{k,l,i}$ by minimizing $\mathrm{E}\left\|x_i - \sum_{l \in \mathcal{N}_k} C_{k,l,i}\psi_{l,i}\right\|^2$. Since $\mathrm{E}\left[x_i^T \psi_{l,i}\right]$ and $\mathrm{E}\left[\psi_{l,i}^T \psi_{l,i}\right]$ are not known, they are replaced by the instantaneous approximations $\hat{x}_{k,i-1|i}^T \psi_{l,i-1}$ and $\psi_{l,i-1}^T \psi_{l,i-1}$ respectively in Cattivelli and Sayed [2009]. The accuracy of such approximation highly relies on the accuracy of the estimates $\hat{x}_{k,i-1|i}^T$ and $\psi_{l,i-1}^T \psi_{l,i-1}$ and can not guarantee the stability of the local Kalman filter of agent k especially when the local detectability is lost, as will be shown by the simulations in Section 5. Therefore, we need to seek another effective way of choosing the adaptive weights that can improve the estimation of each agent.

The diffusion matrix influences the performance of the whole network through assigning different weights to different local Kalman filters at each iteration. Heuristically speaking, a larger weight should be assigned to the local Kalman filter with better estimation performance which can be evaluated by, for example, its estimation error covariance matrix. Based on this idea, we introduce the CI algorithm to incorporate the information of error covariance for choosing the diffusion matrix.

Suppose agent k is going to fuse the estimates $\phi_{l,i}$ from its neighbors with corresponding covariance matrices $P_{l,i}$. The fusion based on the CI algorithm of Julier and Uhlmann [1997] is given by

$$\hat{x}_{k,i} = \Lambda_{k,i} \sum_{l \in \mathcal{N}_{k,i}} \beta_{k,l,i} P_{l,i}^{-1} \phi_{l,i}$$
$$\Lambda_{k,i} = \left(\sum_{l \in \mathcal{N}_{k,i}} \beta_{k,l,i} P_{l,i}^{-1} \right)^{-1} \tag{3}$$

where $\beta_{k,l,i}$ with $0 \leqslant \beta_{k,l,i} \leqslant 1$ and $\sum_{l \in \mathcal{N}_{k,i}} \beta_{k,l,i} = 1$ are chosen such that the trace or determinant of $\Lambda_{k,i}$ is minimized. Then, the weighting matrix for $\phi_{l,i}$ is

$$C_{k,l,i} = \beta_{k,l,i} \Lambda_{k,i} P_{l,i}^{-1}. \tag{4}$$

It should be noted that $P_{l,i}$ does not have to be the true error covariance matrix of $\phi_{l,i}$, and its influence on the estimation stability will be discussed later. Due to the high computation load for choosing the optimal $\beta_{k,l,i}$, several fast CI algorithms that produce suboptimal solutions have been proposed in terms of trace or determinant minimization [Niehsen, 2002, Franken and Hupper, 2005, Wang and Li, 2009]. For the sake of computational simplicity, we use the simplified algorithm proposed in Niehsen [2002] which gives

$$\beta_{k,l,i} = \frac{1/tr\,(P_{l,i})}{\sum_{m \in \mathcal{N}_{k,i}} 1/tr\,(P_{m,i})}.$$

A even more simplified form is 0-1 weighting:

$$g_{k,i} = arg \min_{l \in \mathcal{N}_{k,i}} tr\left(P_{l,i|i}\right)$$
$$\beta_{k,l,i} = \begin{cases} 1, & if\ l = g_{k,i}; \\ 0, & otherwise. \end{cases}$$

which implies

$$C_{k,l,i} = \begin{cases} I, & if\ l = g_{k,i}; \\ 0, & otherwise. \end{cases} \tag{5}$$

4. CI-BASED DIFFUSION KALMAN FILTERING ALGORITHM

Our aim is to seek methods such that each agent can obtain a stable estimate (i.e., its error covariance is bounded) even when the estimation based only on local measurements may be unstable. Before designing the algorithm, we recall the uniform observability of linear time-varying systems [Anderson and Moore, 1981]. Consider the system with time-varying global measurement matrices F_i, H_i and let the observability gramian be given by

$$\overline{W}_{i+\delta,i} = \sum_{t=i}^{i+\delta} \Phi_{t,i}^T H_t^T H_t \Phi_{t,i} \tag{6}$$

for some integer $\delta > 0$, where $\Phi_{i,i} = I$ and

$$\Phi_{t,i} = F_{t-1} \cdots F_i$$

for $t > i$. The matrices F_i and H_i are said to satisfy the uniform observability condition, if there are real numbers $\underline{\eta}$, $\overline{\eta} > 0$ and an integer $\delta > 0$, such that

$$\underline{\eta} I \leqslant \overline{W}_{i+\delta,i} \leqslant \overline{\eta} I. \tag{7}$$

In the same way, we can define the uniform observability for each local system. Consider the local system of agent k with time-varying matrix F_i and $H_{k,i}^{\text{loc}}$, and let the observability gramian be given by

$$W_{i+\delta_k,i}^k = \sum_{t=i}^{i+\delta_k} \Phi_{t,i}^T (H_{k,t}^{\text{loc}})^T H_{k,t}^{\text{loc}} \Phi_{t,i} \tag{8}$$

for some integer $\delta_k > 0$. The matrices F_i and $H_{k,i}^{\text{loc}}$ are said to satisfy the uniform observability condition, if there are real numbers $\underline{\eta}_k$, $\overline{\eta}_k > 0$ and an integer $\delta_k > 0$, such that

$$\underline{\eta}_k I \leqslant W_{i+\delta_k,i}^k \leqslant \overline{\eta}_k I. \tag{9}$$

In a network, (9) may not hold for all agents and we use Ω to denote the set of agents for which (9) holds, i.e., $k \in \Omega$ if F_i and $H_{k,i}^{\text{loc}}$ satisfy the uniform observability condition, and $k \notin \Omega$ otherwise. In the sequel, we will discuss the problem under two scenarios, partial local observability ($\Omega \neq \varnothing$ and each agent k knows whether $k \in \Omega$ or not) and no local observability ($\Omega = \varnothing$).

4.1 Partial Local Observability

Since the estimate of agent $k \in \Omega$ is already stable due to the uniform observability condition satisfied by F_i and $H_{k,i}^{\text{loc}}$, our job is to let each agent $l \notin \Omega$ obtain a stable estimate by fusing the estimates of other agents. Hence, we propose the CI-based diffusion Kalman filtering (CI-DKF) algorithm for partial local observability (Algorithm 2) for the two different sets of agents in the network. Algorithm 2 below requires at every instant i, each agent communicate to its neighbors the quantities $H_{k,i}^T R_{k,i}^{-1} H_{k,i}$, $H_{k,i}^T R_{k,i}^{-1} y_{k,i}$ and $P_{k,i|i-1}$ by one message. Besides the difference in the diffusion updating scheme, Algorithm 2 does not require each agent to communicate an intermediate estimate as Algorithm 1 does, which needs an extra message.

Algorithm 2. (CI-based diffusion Kalman filtering for partial local observability)

Start with $x_{k,0|-1} = 0$, $P_{k,0|-1} = \Pi_0$ and $i = 0$ for all k:
  Step 1: Diffusion Update:
    Calculate $\Lambda_{k,i}$ as defined by (3) and the diffusion matrix $\mathcal{C}_i$ by (4);
    For $k \in \Omega$, if $P_{k,i|i-1} > \Lambda_{k,i}$,
      $\hat{x}_{k,i|i-1} \leftarrow \sum_{l \in \mathcal{N}_{k,i}} C_{k,l,i} \hat{x}_{l,i|i-1}$
      $P_{k,i|i-1} \leftarrow \Lambda_{k,i}$
    For $k \notin \Omega$,
      $\hat{x}_{k,i|i-1} \leftarrow \sum_{l \in \mathcal{N}_{k,i}} C_{k,l,i} \hat{x}_{l,i|i-1}$
      $P_{k,i|i-1} \leftarrow \Lambda_{k,i}$
  Step 2: Standard Kalman Filter Update:
    $S_{k,i} = \sum_{l \in \mathcal{N}_{k,i}} H_{l,i}^T R_{l,i}^{-1} H_{l,i}$
    $q_{k,i} = \sum_{l \in \mathcal{N}_{k,i}} H_{l,i}^T R_{l,i}^{-1} y_{l,i}$

$$P_{k,i|i}^{-1} = P_{k,i|i-1}^{-1} + S_{k,i}$$
$$\hat{x}_{k,i|i} = \hat{x}_{k,i|i-1} + P_{k,i|i}\left[q_{k,i} - S_{k,i}\hat{x}_{k,i|i-1}\right]$$
$$\hat{x}_{k,i+1|i} = F_i\hat{x}_{k,i|i}$$
$$P_{k,i+1|i} = F_iP_{k,i|i}F_i^T + Q_i$$
$$i \leftarrow i+1.$$

Theorem 1. With $P_{l,0|-1} = \Pi_0 \geqslant \mathrm{E}\left[\tilde{x}_{l,0|-1}\tilde{x}_{l,0|-1}^T\right]$ for all agents $l = 1, \cdots, N$, if there exists an agent $k \in \Omega$ and the network is connected all the time, then $P_{l,i|i-1}$ is bounded and $P_{l,i|i-1} \geqslant \mathrm{E}\left[\tilde{x}_{k,i|i-1}\tilde{x}_{k,i|i-1}^T\right]$ for $l = 1, \cdots, N$ through all iterations of Algorithm 2 .

Proof. The proof is omitted due to space limitation.

4.2 No Local Observability

For the case of no local observability, one choice to make Algorithm 2 applicable is to rebuild the local observability for some agents. The key to rebuilding the observability is to get enough measurement information of the state. As we know, the optimal estimate that can be obtained for the whole network is the one computed in a centralized manner by gathering the measurement information of all agents. In the Kalman filter update step, the measurement information of agent k is contained in $S_{k,i}$ and $q_{k,i}$, and the corresponding information matrix and information vector for the optimal estimate are respectively:

$$\bar{S}_i = \sum_{k=1}^{N} H_{k,i}^T R_{k,i}^{-1} H_{k,i} = H_i^T R_i^{-1} H_i$$
$$\bar{q}_i = \sum_{k=1}^{N} H_{k,i}^T R_{k,i}^{-1} y_{k,i} = H_i^T R_i^{-1} y_i \qquad (10)$$

Due to limited communication and storage capability, each agent may not gather all of the information required for the optimal estimate within finite time, but it is still possible for each agent to obtain a stable estimate by gathering enough information.

In recent years, many consensus approaches have been proposed for distributed sensor fusion [Olfati-Saber, 2005, Xiao et al., 2005, Olfati-Saber, 2007, Ren et al., 2007]. By these approaches, $S_{k,i}$ and $q_{k,i}$ for each agent k approach $\bar{S}_i/N$ and $\bar{q}_{k,i}/N$ respectively in an asymptotic manner as the number of communication cycles goes to infinity. The finite-time consensus for continuous variables is shown to be achievable in Wang and Xiao [2010]. However, there has been no effective method to achieve the finite-time consensus for discrete variables. Olfati-Saber [2005] first applied a consensus protocol in the distributed Kalman filtering where each agent implements the consensus protocol to gather measurement information between two successive Kalman filter updates as shown in Fig. 1. However, by this approach, $S_{k,i}$ and $q_{k,i}$ may not have achieved consensus, i.e. converge to the values of $\bar{S}_i$ and $\bar{q}_{k,i}$ before the Kalman filter update within a finite period of time, and the error caused by treating $S_{k,i}$ and $q_{k,i}$ as $\bar{S}_i$ and $\bar{q}_{k,i}$ respectively may destroy the estimation stability.

For computational simplicity, we adopt the distributed consensus protocol proposed in Xiao et al. [2005] for information exchange between two successive Kalman filter updates (Fig. 1), but the received information is processed

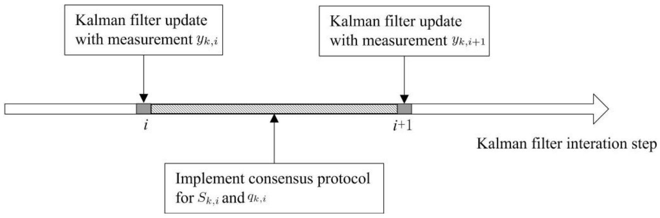


Fig. 1. Implementation of consensus protocol embedded in the Kalman filter for agent k.

in a different way to guarantee the estimation stability. In Xiao et al. [2005], each agent aims to let $S_{k,i}$ and $q_{k,i}$ reach a consensus by the following protocol:

$$S_{k,i}(j+1) = \left(1 - \frac{d_{k,i}(j+1)-1}{N}\right) S_{k,i}(j) + \frac{1}{N}\sum_{\substack{l\in\mathcal{N}_{k,i}(j+1)\\ l\neq k}} S_{l,i}(j)$$
$$q_{k,i}(j+1) = \left(1 - \frac{d_{k,i}(j+1)-1}{N}\right) q_{k,i}(j) + \frac{1}{N}\sum_{\substack{l\in\mathcal{N}_{k,i}(j+1)\\ l\neq k}} q_{l,i}(j) \qquad (11)$$

where i refers to the i-th sampling interval and j the j-th communication cycle. Then, we can get

$$S_{k,i}(j) = \sum_{l=1}^{N} \rho_{k,l,i}(j)\, S_{k,i}(0)$$
$$q_{k,i}(j) = \sum_{l=1}^{N} \rho_{k,l,i}(j)\, q_{k,i}(0) \qquad (12)$$

where $\rho_{k,l,i}$ are a set of weights satisfying

$$0 \leqslant \rho_{k,l,i}(j) \leqslant 1, \quad \rho_{k,l,i}(j) = \rho_{l,k,i}(j), \quad \sum_{l=1}^{N}\rho_{k,l,i}(j) = 1.$$

It is straightforward to get the following conclusion.

Lemma 2. If the network is connected through all j, then there exists a time $0 < t_o \leqslant N-1$ such that for each agent k and $j \geqslant t_o$,

$$\rho_{k,l,i}(j) \geqslant (\frac{1}{N})^{t_o} \geqslant (\frac{1}{N})^{N-1}$$

Summarizing the results above, we design the information gathering scheme (Algorithm 3), where t_o is a given time length for communication. Then, we can define a set Υ_i, where $k \in \Upsilon_i$, if $\min_l \rho_{k,l,t}(t_o) > 0$, and $k \notin \Upsilon_i$ otherwise. With Algorithm 3 embedded, the CI-DKF algorithm for the case of no local observability is shown in Algorithm 4.

Algorithm 3. (Information gathering scheme)

Start with $S_{k,i}(0) = H_{k,i}^T R_{k,i}^{-1} H_{k,i}$, $q_{k,i}(0) = H_{k,i}^T R_{k,i}^{-1} y_{k,i}$, $\rho_{k,k,i}(0) = 1$, $\rho_{k,l,i}(0) = 0$ for $l \neq k$, and $j = 1$ for all k:
  Repeat the following steps until $j > t_o$:
    Calculate $S_{k,i}(j)$ and $q_{k,i}(j)$ by (11);
    Calculate $\rho_{k,l,i}(j)$ $(l = 1, \ldots N)$ by (**??**);
    $j \leftarrow j+1$
  end
  $$S_{k,i} \leftarrow \frac{1}{\max_l \rho_{k,l,i}(t_o)} S_{k,i}(t_o)$$

$$q_{k,i} \leftarrow \frac{1}{\max_{l} \rho_{k,l,i}(j)} q_{k,i}(t_o).$$

Algorithm 4. (CI-based diffusion Kalman filtering for no local observability)

Start with $x_{k,0|-1} = 0$, $P_{k,0|-1} = \Pi_0$ and $i = 0$ for all k:
Step 1: Information Gathering:
Take measurement and implement Algorithm 3 to get $S_{k,i}$, $q_{k,i}$ and $\rho_{k,l,i}$ $(l = 1, \ldots N)$;
Step 2: Diffusion Update:
Calculate $\Lambda_{k,i}$ as defined by (3) and the diffusion matrix $\mathcal{C}_i$ by (4);
For $k \in \Upsilon_i$, if $P_{k,i|i-1} > \Lambda_{k,i}$
$\hat{x}_{k,i|i-1} \leftarrow \sum_{l \in \mathcal{N}_{k,i}} C_{k,l,i} \hat{x}_{l,i|i-1}$
$P_{k,i|i-1} \leftarrow \Lambda_{k,i}$
For $k \notin \Upsilon_i$,
$\hat{x}_{k,i|i-1} \leftarrow \sum_{l \in \mathcal{N}_{k,i}} C_{k,l,i} \hat{x}_{l,i|i-1}$
$P_{k,i|i-1} \leftarrow \Lambda_{k,i}$
Step 3: Standard Kalman Filter Update:
$P^{-1}_{k,i|i} = P^{-1}_{k,i|i-1} + S_{k,i}$
$\hat{x}_{k,i|i} = \hat{x}_{k,i|i-1} + P_{k,i|i} \left[q_{k,i} - S_{k,i} \hat{x}_{k,i|i-1} \right]$
$\hat{x}_{k,i+1|i} = F_i \hat{x}_{k,i|i}$
$P_{k,i+1|i} = F_i P_{k,i|i} F_i^T + Q_i$
$i \leftarrow i + 1.$

Lemma 3. With $P_{k,0|-1} = \Pi_0 \geqslant \mathrm{E}\left[\tilde{x}_{k,0|-1}\tilde{x}^T_{k,0|-1}\right]$ for all agents $k = 1, \cdots, N$, $P_{k,i|i-1} \geqslant \mathrm{E}\left[\tilde{x}_{k,i|i-1}\tilde{x}^T_{k,i|i-1}\right]$ holds through all iterations of Algorithm 4.

Theorem 4. With $P_{l,0|-1} = \Pi_0 \geqslant \mathrm{E}\left[\tilde{x}_{l,0|-1}\tilde{x}^T_{l,0|-1}\right]$ for all agents $l = 1, \cdots, N$, if there exists an agent $k \in \Upsilon_i$ through all iterations of Algorithm 4 and the network is connected all the time, then $P_{l,i|i-1}$ is bounded for all agents if F_i and H_i satisfy the uniform observability condition.

Proof. The proof is omitted due to space limitation.

5. SIMULATION

5.1 Simulation Environment

A time-invariant system is considered for the ease of simulation, though the proposed algorithm is not restricted to it. A stationary sensor network is estimating the dynamic energy intensity of two stationary sources, the positions of which are known. We use two exponential functions to denote the energy intensity of two sources spreading over a surveillance region and the system model is given by:

$$F = \begin{bmatrix} 1 & 0.005 \\ 0 & 1 \end{bmatrix}, \quad G = I, \quad H_k = \begin{bmatrix} e^{-\lambda(s_k - \mu_1)^2} \\ e^{-\lambda(s_k - \mu_2)^2} \end{bmatrix},$$
$$Q = 5I, \quad R_k = 20$$

where s_k is the position of agent k , and μ_1 and μ_2 are the positions of the two sources. In the following simulations, we will set different values for attenuation factor λ and communication range r_c to get different energy intensity field and network topology respectively. The threshold for the identity classification on observability is set as $W_{\text{th}} = 0.01I$, and $k \in \Omega$ if $W^k_{i+p,i} \geqslant W_{\text{th}}$. Similarly, we also set $\rho_{\text{th}} = 0.001$ and $k \in \Upsilon$ if $\min_{l} \rho_{k,l,t}(t_o) > 0 \geqslant \rho_{\text{th}}$.

We use the mean-square deviation (MSD), the same performance index adopted in Cattivelli and Sayed [2010], to evaluate the algorithm performance. The MSD for agent k at time i is defined as

$$\text{MSD}_{k,i} \triangleq \mathrm{E}\left[\tilde{x}_{k,i|i-1}\tilde{x}^T_{k,i|i-1}\right].$$

Then, the MSD of the whole network is calculated as

$$\text{MSD}_i \triangleq \frac{1}{N}\sum_{k=1}^{N} \text{MSD}_{k,i}.$$

We compare MSD_i of different algorithms. The results are averaged over 100 independent experiments.

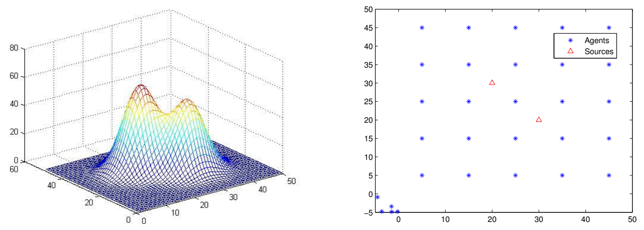
We first implement simulations in the case of partial local observability and compare the performance of four different algorithms: DKF algorithm (Algorithm 1) with adaptive weights, CI-DKF algorithm for partial local observability (Algorithm 2) with weight choice rule (4) and (5), and no diffusion algorithm (i.e. each agent only implements the standard Kalman filter update). In this case, $N = 25$ agents are uniformly deployed over a $50 \times 50\text{m}^2$ square region and we let each one of them be able to fully observe the state by setting $\lambda = 0.02$ and $r_c = 15$m (Fig. 2). Besides, five more agents are added randomly within $[-5, 0] \times [-5, 0]$ which can not observe the state since they are too far away from the sources. The positions of the two sources are fixed at $\mu_1 = [20\ 30]^T$ and $\mu_2 = [30\ 20]^T$.

Then we test the performance of the CI-DKF algorithm for no local observability (Algorithm 4) with weight choice rule (4), which is compared to the no diffusion algorithm with information gathering scheme. In this simulation, we place the sources at $\mu_1 = [0\ 50]^T$ and $\mu_2 = [50\ 0]^T$ respectively and choose $N = 25$, $\lambda = 0.02$ and $r_c = 15$m such that no agent can observe the whole state (Fig. 3). Two different values of the communication time length $t_o = 3$ and $t_o = 6$ are simulated.

5.2 Simulation Results

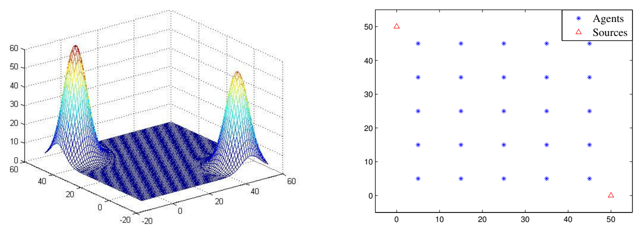
In the case of partial local observability, MSD by the DKF algorithm and the no fusion algorithm both diverge, while MSD by the CI-DKF algorithm is still bounded and very low as shown in Fig. 4a. Besides, the comparison of diffusion algorithms with no fusion algorithm illustrates that diffusion can improve the network average performance. Another point to note is that the CI-DKF algorithm with rule (4) and (5) have nearly the same performance. This suggests us to use rule (5) in real applications due to its simplified computation.

Fig. 4b shows that the no diffusion algorithm with the information gathering scheme cannot achieve stable estimation when $t_o = 3$, while the CI-DKF algorithm can. When $t_o = 6$, the local observability is rebuilt for all agents and both of the CI-DKF algorithm and the no diffusion algorithm can achieve stable estimation. Besides, the MSD by the CI-DKF algorithm is smaller than that by the no diffusion algorithm in these cases.



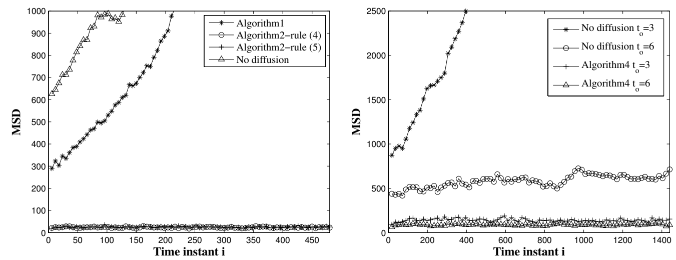
(a) Energy intensity field. (b) Source and agent positions.

Fig. 2. Simulation setup of Scenario I.



(a) Energy intensity field. (b) Source and agent positions.

Fig. 3. Simulation setup of Scenario II.



(a) Scenario I. (b) Scenario II.

Fig. 4. The mean-square deviation.

6. CONCLUSIONS

In this paper, we proposed a CI-based diffusion Kalman filtering (CI-DKF) algorithm by incorporating the covariance information. The CI-DKF algorithm can be applied in the case of lacking local observability. A consensus-based information gathering scheme is embedded when no single agent can observe the state. Simulation shows that the CI-DKF algorithm has better performance than that by the original DKF and those by local Kalman filters.

REFERENCES

P. Alriksson and A. Rantzer. Distributed Kalman filtering using weighted averaging. In *Proceedings of the International Symposium on Mathematical Theory of Networks and Systems*, 2006.

B. Anderson and J. Moore. Detectability and stabilizability of time-varying discrete-time linear systems. *SIAM Journal on Control and Optimization*, 19(1):20–32, 1981.

P. Arambel, C. Rago, and R. Mehra. Covariance intersection algorithm for distributed spacecraft state estimation. In *Proceedings of the American Control Conference*, 2002.

F. Cattivelli and A. Sayed. Diffusion strategies for distributed Kalman filtering and smoothing. *IEEE Transactions on Automatic Control*, 55(9):2069–2084, 2010.

F. Cattivelli and A. Sayed. Diffusion distributed Kalman filtering with adaptive weights. In *Proceedings of the Asilomar Conference on Signals, Systems and Computers*, 2009.

K. Chang, R. Saha, and Y. Bar-Shalom. On optimal track-to-track fusion. *IEEE Transactions on Aerospace and Electronic Systems*, 33(4):1271–1276, 1997.

D. Franken and A. Hupper. Improved fast covariance intersection for distributed data fusion. In *Proceedings of the International Conference on Information Fusion*, 2005.

H. Hashemipour, S. Roy, and A. Laub. Decentralized structures for parallel Kalman filtering. *IEEE Transactions on Automatic Control*, 33(1):88–94, 1988.

S. Julier and J. Uhlmann. A non-divergent estimation algorithm in the presence of unknowncorrelations. In *Proceedings of the American Control Conference*, 1997.

S. Julier and J. Uhlmann. General decentralized data fusion with covariance intersection (CI). *Handbook of multisensor data fusion*, 2001.

X. Li, K. Zhang, J. Zhao, and Y. Zhu. Optimal linear estimation fusion. v. relationships. In *Proceedings of the International Conference on Information Fusion*, 2002.

S. Mori, W. Barker, C. Chong, and K. Chang. Track association and track fusion with nondeterministic target dynamics. *IEEE Transactions on Aerospace and Electronic Systems*, 38(2):659–668, 2002.

W. Niehsen. Information fusion based on fast covariance intersection filtering. In *Proceedings of the International Conference on Information Fusion*, 2002.

R. Olfati-Saber. Distributed Kalman filter with embedded consensus filters. In *Proceedings of the IEEE Conference on Decision and Control*, 2005.

R. Olfati-Saber. Distributed Kalman filtering for sensor networks. In *Proceedings of the IEEE Conference on Decision and Control*, 2007.

W. Ren, R. Beard, and E. Atkins. Information consensus in multivehicle cooperative control. *IEEE Control Systems Magazine*, 27(2):71–82, 2007.

D. Spanos, R. Olfati-Saber, and R. Murray. Approximate distributed Kalman filtering in sensor networks with quantifiable performance. In *Proceedings of the International Symposium on Information Processing in Sensor Networks*, 2005.

J. Uhlmann, S. Julier, B. Kamgar-Parsi, M. Lanzagorta, and H. Shyu. NASA Mars rover: A testbed for evaluating applications of covariance intersection. In *Proceedings of the Society of Photo-Optical Instrumentation Engineers Conference Series*, 1999.

L. Wang and F. Xiao. Finite-time consensus problems for networks of dynamic agents. *IEEE Transactions on Automatic Control*, 55(4):950–955, 2010.

Y. Wang and X. Li. A fast and fault-tolerant convex combination fusion algorithm under unknown cross-correlation. In *Proceedings of the International Conference on Information Fusion*, 2009.

L. Xiao, S. Boyd, and S. Lall. A scheme for robust distributed sensor fusion based on average consensus. In *Proceedings of the international symposium on information processing in sensor networks*, 2005.