COST-EFFECTIVE DIFFUSION KALMAN FILTERING WITH IMPLICIT MEASUREMENT EXCHANGES

Sayed Pouria Talebi, Sithan Kanna*, Yili Xia†, and Danilo P. Mandic**

*Electrical and Electronic Engineering Department, Imperial College London, SW7 2AZ, UK
†The School of Information Science and Engineering, Southeast University, Nanjing 210096, China
E-mails:{s.talebi12, shri.kanagasabapathy08, d.mandic}@imperial.ac.uk, yili_xia@seu.edu.cn

ABSTRACT

A resource effective extension to the class of distributed real-time diffusion Kalman filters is proposed. The proposed scheme removes the need to share measurement variables explicitly, by sharing only the state estimates and state error covariance matrices which implicitly contain the information about the measurements, observations matrices, and noise covariance matrices. The proposed distributed Kalman filter is quiet general, and its performance is comparable to that of existing diffusion based schemes, while having lower communication and computational requirements per-iteration compared to current distributed Kalman filtering algorithms.

***Index Terms*—** Kalman filter, distributed estimation, diffusion strategy, distributed adaptive filtering, sensor networks.

1. INTRODUCTION

The widespread emergence of intelligent sensor networks for applications ranging from robotics to environmental monitoring has been enabled by the availability of inexpensive sensors equipped with communications and computational capabilities. The task of developing fast and robust signal processing algorithms for these sensor networks has therefore been a subject of great interest in the signal processing, control, and machine learning communities [1, 2, 3, 4]. An example of an algorithm that has been studied extensively in the context of multi-agent networks is the distributed Kalman filter. This is attributed to the optimality of the (single-agent) Kalman filter for linear Gaussian systems and the flexibility of the state-space representation to model a large class of real-world problems.

Early work in the field of distributed Kalman filtering focused on decentralizing the Kalman filtering operations using individual agents which communicate with a fusion center [5]. This method is referred to as the centralized Kalman filter, due to the fact that a central fusion center has access to all the information in the network. Since communicating to a single fusion center makes the network vulnerable to even a single point of failure, decentralized solutions, such as distributed Kalman filters where each node is required to share all its information with every other node in the network, were introduced [6, 7]. This method effectively replicates the operation of the centralized Kalman filter at each node, this severely burdens the network with communication traffic overhead and large number of computations [6].

More general distributed Kalman filters were proposed in the consensus literature, where the constraint that the nodes had to communicate with every other node in the network was relaxed [8, 9]. To compensate for the fact that nodes can only access the measurements in their neighborhood, consensus Kalman filters include a consensus step for their state estimates whereby the individual nodes exchange and average their intermediate state estimates with their neighbors several times before the next measurement is obtained [10]. In other words, consensus filters operate at two time-scales, a longer time-scale for the measurement updates and a shorter time-scale for the consensus update.

The class of distributed Kalman filtering algorithms proposed in [11] are based on diffusion adaptive algorithms [12, 13], that enable both the measurement update and information fusion throughout the network to be applied in a single time-scale, and have become known as diffusion Kalman filters. Furthermore, it was shown that diffusion strategies outperform their consensus based counterparts as they provide for a more comprehensive diffusion of data in the network [14].

A fundamental feature in the diffusion strategy is that only the state estimates, together with observation variables, are shared in the network. However, results in fusion theory show that optimal fusion of estimated variables require the sharing of both the state estimates and their covariance matrices [15]. To this end, fusion theory has been applied in conjunction with the diffusion strategy to introduce a distributed Kalman filtering scheme based on covariance intersection [16, 17]. Consequently, the communication requirements for distributed Kalman filters based on covariance intersection are more than that of standard distributed Kalman filters as the state covariance matrix needs to be communicated together with the observation variables and estimates of the state vector.

In this work, we propose an extension of the diffusion Kalman filter where we draw upon the ideas from covariance intersection without imposing excessive communication or computational burden on the network. Specifically, the proposed distributed Kalman filter mitigates the need to share observation variables while only requiring the sharing of the intermediate state estimates and state error covariance matrices. This is achieved by diffusing not only the state vector but also the state covariance matrix through the network. The motivation for this distributed Kalman filter comes from our earlier work in [18] where we conjectured that the diffusion step implicitly shares the measurements in the network and a separate step to incorporate the measurements in the neighborhood may be redundant.

Mathematical notations: Scalars, column vectors, and matrices are denoted by lowercase, bold lowercase, and bold uppercase letters respectively. The transpose operator is denoted by $(\cdot)^{\mathsf{T}}$, whereas the statistical expectation operator is denoted by $\mathbb{E}\{\cdot\}$. Finally, the real domain is denoted by $\mathbb{R}$.

2. PROBLEM FORMULATION

Consider a general network that is represented by an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, with the node set $\mathcal{N} = \{1, 2, \ldots, N\}$, where N

is the number of nodes in the network and the edge set $\mathcal{E}$ denotes the connections between the nodes in the network. The neighborhood of a node i, denoted by $\mathcal{N}_i$, is defined as all the nodes connected to node i including itself. The cardinality of the set $\mathcal{N}_i$ is defined as the number of connections node i has with its neighbors including itself and is denoted by $|\mathcal{N}_i| = N_i$. At each time instant, k, node i is tasked to estimate a state vector $\boldsymbol{x}_k \in \mathbb{R}^M$ which is assumed to be identical throughout the network but observed locally through measurements $\boldsymbol{y}_{i,k} \in \mathbb{R}^L$. The measurements and state are coupled via a state-space model given by

$$\begin{aligned} \boldsymbol{x}_k &= \mathbf{A}_k \boldsymbol{x}_{k-1} + \boldsymbol{\nu}_k \\ \boldsymbol{y}_{i,k} &= \mathbf{H}_{i,k} \boldsymbol{x}_k + \boldsymbol{\omega}_{i,k} \end{aligned} \tag{1}$$

where $\mathbf{A}_k \in \mathbb{R}^{M \times M}$ is the state transition matrix and $\mathbf{H}_{i,k} \in \mathbb{R}^{L \times M}$ is the observation matrix, whereas the process noise $\boldsymbol{\nu}_k \in \mathbb{R}^M$ and observation noise $\boldsymbol{\omega}_{i,k} \in \mathbb{R}^L$ are temporally uncorrelated and spatially independent zero-mean white Gaussian noise processes with a joint covariance matrix given by [1]

$$\mathbb{E}\left\{ \begin{bmatrix} \boldsymbol{\nu}_k \\ \boldsymbol{\omega}_{i,k} \end{bmatrix} \begin{bmatrix} \boldsymbol{\nu}_n^{\mathsf{T}} & \boldsymbol{\omega}_{\ell,n}^{\mathsf{T}} \end{bmatrix} \right\} = \begin{bmatrix} \mathbf{C}_{\boldsymbol{\nu}_k} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_{\boldsymbol{\omega}_{i,k}} \delta_{i,\ell} \end{bmatrix} \delta_{k,n} \tag{2}$$

where $\delta_{k,n}$ denotes the Kronecker delta function, that is

$$\delta_{k,n} = \begin{cases} 1, & k = n, \\ 0, & \text{otherwise}. \end{cases}$$

The optimal solution to this problem comes in the form of the centralized Kalman filter implemented at each node [22], the operations of which are summarized in Algorithm 1, where $\hat{\boldsymbol{x}}_{k|k-1}$ and $\hat{\boldsymbol{x}}_{k|k}$ denote respectively the *a priori* and *a posteriori* estimates of $\boldsymbol{x}_k$.

Algorithm 1. Centralized Kalman Filter [22]

Initialize with:

$$\begin{aligned} \hat{\boldsymbol{x}}_{0|0} &= \mathbb{E}\{\boldsymbol{x}_0\} \\ \mathbf{M}_{0|0} &= \mathbb{E}\left\{(\boldsymbol{x}_0 - \mathbb{E}\{\boldsymbol{x}_0\})(\boldsymbol{x}_0 - \mathbb{E}\{\boldsymbol{x}_0\})^{\mathsf{T}}\right\} \end{aligned}$$

Model update:

$$\begin{aligned} \hat{\boldsymbol{x}}_{k|k-1} &= \mathbf{A}_k \hat{\boldsymbol{x}}_{k-1|k-1} \\ \mathbf{M}_{k|k-1} &= \mathbf{A}_k \mathbf{M}_{k-1|k-1} \mathbf{A}_k^{\mathsf{T}} + \mathbf{C}_{\boldsymbol{\nu}_k} \end{aligned} \tag{3}$$

Measurement update:

$$\mathbf{S}_k = \sum_{\ell=1}^{N} \mathbf{H}_{\ell,k}^{\mathsf{T}} \mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1} \mathbf{H}_{\ell,k} \tag{4a}$$

$$\boldsymbol{q}_k = \sum_{\ell=1}^{N} \mathbf{H}_{\ell,k}^{\mathsf{T}} \mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1} \boldsymbol{y}_{\ell,k} \tag{4b}$$

$$\mathbf{M}_{k|k}^{-1} = \mathbf{M}_{k|k-1}^{-1} + \mathbf{S}_k \tag{4c}$$

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \mathbf{M}_{k|k}\big(\boldsymbol{q}_k - \mathbf{S}_k \hat{\boldsymbol{x}}_{k|k-1}\big) \tag{4d}$$

Although the centralized Kalman filter presents the optimal solution, the calculations of $\mathbf{S}_k$ and $\boldsymbol{q}_k$ in (4a) and (4b) require all nodes to share the parameters $\mathbf{H}_{\ell,k}^{\mathsf{T}} \mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1} \mathbf{H}_{\ell,k}$ and $\mathbf{H}_{\ell,k}^{\mathsf{T}} \mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1} \boldsymbol{y}_{\ell,k}$. Taking into account that in sensor networks, communication is restricted to a local neighborhood of a node, the implementation of centralized Kalman filters imposes a severe communication burden and is therefore not suitable for real-time applications. In the context of the diffusion Kalman filter [11], local approximations of $\mathbf{S}_k$ and $\boldsymbol{q}_k$ from (4a) and (4b) given by

$$\mathbf{S}_{i,k} = \sum_{\ell \in \mathcal{N}_i} \mathbf{H}_{\ell,k}^{\mathsf{T}} \mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1} \mathbf{H}_{\ell,k} \;\;\&\;\; \boldsymbol{q}_{i,k} = \sum_{\ell \in \mathcal{N}_i} \mathbf{H}_{\ell,k}^{\mathsf{T}} \mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1} \boldsymbol{y}_{\ell,k} \tag{5}$$

are used to obtain intermediate estimates of the state vector as

$$\boldsymbol{\psi}_{i,k} = \underbrace{\mathbf{A}_k \hat{\boldsymbol{x}}_{i,k-1|k-1}}_{\hat{\boldsymbol{x}}_{i,k|k-1}} + \mathbf{M}_{i,k|k}\big(\boldsymbol{q}_{i,k} - \mathbf{S}_{i,k} \hat{\boldsymbol{x}}_{i,k|k-1}\big) \tag{6}$$

where $\mathbf{M}_{i,k|k}^{-1} = \big(\mathbf{A}_k \mathbf{M}_{i,k-1|k-1} \mathbf{A}_k^{\mathsf{T}} + \mathbf{C}_{\boldsymbol{\nu}_k}\big)^{-1} + \mathbf{S}_{i,k}$, whereas $\hat{\boldsymbol{x}}_{i,k|k-1}$ and $\hat{\boldsymbol{x}}_{i,k|k}$ denote the *a priori* and *a posteriori* estimates of the state vector at node i at time instant k, with

$$\hat{\boldsymbol{x}}_{i,k|k} = \sum_{\ell \in \mathcal{N}_i} a_{i,\ell} \boldsymbol{\psi}_{\ell,k} \tag{7}$$

while $a_{i,\ell}$ denotes the diffusion coefficients which must satisfy $\sum_{\ell \in \mathcal{N}_i} a_{i,\ell} = 1$, and are chosen by the algorithm designer [13].

3. PROPOSED DISTRIBUTED KALMAN FILTER

From (6), notice that the information regarding measurements, observation matrices, and observation covariance matrices from the neighborhood of node i is implicitly embedded within the intermediate state estimates $\boldsymbol{\psi}_{i,k}$. Therefore, as long as appropriate values of $a_{i,\ell}$ are chosen, it is possible to simultaneously accomplish the information sharing steps in (5) and diffusion step in (7) using a single diffusion step.

Considering the operations of the centralized Kalman filter, upon replacing (4a) and (4b) into (4d), we have

$$\begin{aligned} \hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} &+ \mathbf{M}_{k|k} \sum_{\ell=1}^{N} \mathbf{H}_{\ell,k}^{\mathsf{T}} \mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1} \boldsymbol{y}_{\ell,k} \\ &- \mathbf{M}_{k|k} \sum_{\ell=1}^{N} \mathbf{H}_{\ell,k}^{\mathsf{T}} \mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1} \mathbf{H}_{\ell,k} \hat{\boldsymbol{x}}_{k|k-1} \end{aligned}$$

which after some mathematical manipulations, gives

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \sum_{\ell=1}^{N} \mathbf{M}_{k|k} \mathbf{H}_{\ell,k}^{\mathsf{T}} \mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1} \left(\boldsymbol{y}_{\ell,k} - \mathbf{H}_{\ell,k} \hat{\boldsymbol{x}}_{k|k-1}\right). \tag{8}$$

The *a posteriori* estimate of the state vector given in (8) can be alternatively calculated by the summation

$$\hat{\boldsymbol{x}}_{k|k} = \frac{1}{N} \sum_{\ell=1}^{N} \boldsymbol{\psi}_{\ell,k} \tag{9}$$

where the intermediate state estimate, $\boldsymbol{\psi}_{\ell,k}$, is given by

$$\boldsymbol{\psi}_{\ell,k} = \hat{\boldsymbol{x}}_{k|k-1} + N \mathbf{M}_{k|k} \mathbf{H}_{\ell,k}^{\mathsf{T}} \mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1} \left(\boldsymbol{y}_{\ell,k} - \mathbf{H}_{\ell,k} \hat{\boldsymbol{x}}_{k|k-1}\right). \tag{10}$$

In addition, replacing (4a) into (4c) yields the diffusion step for the state error covariance matrix in the form of

$$\mathbf{M}_{k|k}^{-1} = \mathbf{M}_{k|k-1}^{-1} + \sum_{\ell=1}^{N} \mathbf{H}_{\ell,k}^{\mathsf{T}} \mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1} \mathbf{H}_{\ell,k} = \frac{1}{N} \sum_{\ell=1}^{N} \boldsymbol{\Gamma}_{\ell,k} \tag{11}$$

[1] Although the concepts are introduced in a real-valued framework, for generality and notational simplicity, these concepts can be readily applied for complex/quaternion-valued signals using widely-linear models [19]. The reader is referred to [20, 18, 21] for more information on diffusion Kalman filtering strategies for general complex/quaternion-valued signals.

where each node updates its intermediate state error covariance matrix using

$$\mathbf{\Gamma}_{\ell,k} = \mathbf{M}_{k|k-1}^{-1} + N\mathbf{H}_{\ell,k}^{\mathsf{T}}\mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1}\mathbf{H}_{\ell,k}. \tag{12}$$

Now, the assumption is made that the network is connected, that is there exists a path between any two nodes in the network, which allows $\mathbf{M}_{k|k}$ in the formulation given in (11) and $\hat{\boldsymbol{x}}_{k|k}$ in the formulation given in (9) to be calculated through the diffusion of matrices $\mathbf{\Gamma}_{\ell,k}$ as given in (12) and local estimates $\boldsymbol{\psi}_{\ell,k}$ as given in (10). The operations of such a diffusion Kalman filter are summarized in Algorithm 2.

Algorithm 2. Cost-Effective Diffusion Kalman Filter with Implicit Measurement Updates

---

For nodes $i = \{1, \ldots, N\}$:
Initialize with:

$$\begin{aligned} \hat{\boldsymbol{x}}_{i,0|0} &= \mathbb{E}\{\boldsymbol{x}_0\} \\ \mathbf{M}_{i,0|0} &= \mathbb{E}\left\{(\boldsymbol{x}_0 - \mathbb{E}\{\boldsymbol{x}_0\})(\boldsymbol{x}_0 - \mathbb{E}\{\boldsymbol{x}_0\})^{\mathsf{T}}\right\} \end{aligned}$$

Model update:

$$\begin{aligned} \hat{\boldsymbol{x}}_{i,k|k-1} &= \mathbf{A}_k\hat{\boldsymbol{x}}_{i,k-1|k-1} \\ \mathbf{M}_{i,k|k-1} &= \mathbf{A}_k\mathbf{M}_{i,k-1|k-1}\mathbf{A}_k^{\mathsf{T}} + \mathbf{C}_{\boldsymbol{\nu}_k} \end{aligned}$$

Measurement update:

$$\begin{aligned} \mathbf{\Gamma}_{i,k} &= \mathbf{M}_{i,k|k-1}^{-1} + N_i\mathbf{H}_{i,k}^{\mathsf{T}}\mathbf{C}_{\boldsymbol{\omega}_{i,k}}^{-1}\mathbf{H}_{i,k} \\ \mathbf{M}_{i,k|k}^{-1} &= \frac{1}{N_i}\sum_{\ell\in\mathcal{N}_i}\mathbf{\Gamma}_{\ell,k} \\ \mathbf{G}_{i,k} &= N_i\mathbf{M}_{i,k|k}\mathbf{H}_{i,n}^{\mathsf{T}}\mathbf{C}_{\boldsymbol{\omega}_{i,k}}^{-1} \\ \boldsymbol{\psi}_{i,k} &= \hat{\boldsymbol{x}}_{i,k|k-1} + \mathbf{G}_{i,k}\big(\boldsymbol{y}_{i,k} - \mathbf{H}_{i,k}\hat{\boldsymbol{x}}_{i,k|k-1}\big) \\ \hat{\boldsymbol{x}}_{i,k|k} &= \frac{1}{N_i}\sum_{\ell\in\mathcal{N}_i}\boldsymbol{\psi}_{\ell,k} \end{aligned}$$

State error covariance matrix diffusion. Sharing the intermediate state error covariance matrices $\mathbf{\Gamma}_{\ell,k}$, as opposed to the update terms $\mathbf{H}_{\ell,k}^{\mathsf{T}}\mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1}\mathbf{H}_{\ell,k}$ in conventional diffusion Kalman filters, enables every node to have access (implicitly) not only to the information from its neighbors but also to that of their neighboring nodes.

4. PERFORMANCE ANALYSIS

The difference between the true state vector and the local estimate at node i at a time instant k is given by $\boldsymbol{\epsilon}_{i,k} = \boldsymbol{x}_k - \boldsymbol{\psi}_{i,k}$, which can be expressed as

$$\boldsymbol{\epsilon}_{i,k} = \boldsymbol{x}_k - \hat{\boldsymbol{x}}_{i,k|k-1} - \mathbf{G}_{i,k}\left(\boldsymbol{y}_{i,k} - \mathbf{H}_{i,k}\hat{\boldsymbol{x}}_{i,k|k-1}\right). \tag{13}$$

Upon replacing $\boldsymbol{y}_{i,k} = \mathbf{H}_{i,k}\boldsymbol{x}_k + \boldsymbol{\omega}_{i,k}$ and $\boldsymbol{\epsilon}_{i,k|k-1} = \boldsymbol{x}_k - \hat{\boldsymbol{x}}_{i,k|k-1}$ into (13), we have

$$\boldsymbol{\epsilon}_{i,k} = \left(\mathbf{I} - \mathbf{G}_{i,k}\mathbf{H}_{i,k}\right)\boldsymbol{\epsilon}_{i,k|k-1} - \mathbf{G}_{i,k}\boldsymbol{\omega}_{i,k}. \tag{14}$$

Furthermore, substituting $\boldsymbol{\epsilon}_{i,k|k-1} = \mathbf{A}_k\boldsymbol{\epsilon}_{i,k-1|k-1} + \boldsymbol{\nu}_k$, obtained from the true state evolution in (1) into (14), gives

$$\begin{aligned} \boldsymbol{\epsilon}_{i,k} =& \left(\mathbf{I} - \mathbf{G}_{i,k}\mathbf{H}_{i,k}\right)\mathbf{A}_k\boldsymbol{\epsilon}_{i,k-1|k-1} \\ &+ \left(\mathbf{I} - \mathbf{G}_{i,k}\mathbf{H}_{i,k}\right)\boldsymbol{\nu}_k - \mathbf{G}_{i,k}\boldsymbol{\omega}_{i,k}. \end{aligned} \tag{15}$$

Consider also the difference between the true state vector and its estimate obtained at node i, given by

$$\boldsymbol{\epsilon}_{i,k|k} = \boldsymbol{x}_k - \hat{\boldsymbol{x}}_{i,k|k} = \boldsymbol{x}_k - \frac{1}{N_i}\sum_{\ell\in\mathcal{N}_i}\boldsymbol{\psi}_{\ell,k} = \frac{1}{N_i}\sum_{\ell\in\mathcal{N}_i}\boldsymbol{\epsilon}_{\ell,k}. \tag{16}$$

Now, replacing the intermediate state error evolution in (15) into the diffusion step in (16), gives a recursive expression for the state vector estimation error in the form

$$\begin{aligned} \boldsymbol{\epsilon}_{i,k|k} =& \frac{1}{N_i}\sum_{\ell\in\mathcal{N}_i}\left(\mathbf{I} - \mathbf{G}_{\ell,k}\mathbf{H}_{\ell,k}\right)\mathbf{A}_k\boldsymbol{\epsilon}_{\ell,k-1|k-1} \\ &+ \frac{1}{N_i}\sum_{\ell\in\mathcal{N}_i}\left(\mathbf{I} - \mathbf{G}_{\ell,k}\mathbf{H}_{\ell,k}\right)\boldsymbol{\nu}_k - \frac{1}{N_i}\sum_{\ell\in\mathcal{N}_i}\mathbf{G}_{\ell,k}\boldsymbol{\omega}_{\ell,k}. \end{aligned} \tag{17}$$

From Algorithm 2, we can now substitute the gain values in (17) as

$$\mathbf{G}_{\ell,k}\mathbf{H}_{\ell,k} = N_\ell\mathbf{M}_{\ell,k|k}\mathbf{H}_{\ell,k}^{\mathsf{T}}\mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1}\mathbf{H}_{\ell,k} = \mathbf{M}_{\ell,k|k}\mathbf{\Upsilon}_{\ell,k}$$

where $\mathbf{\Upsilon}_{\ell,k} = N_\ell\mathbf{H}_{\ell,k}^{\mathsf{T}}\mathbf{C}_{\boldsymbol{\omega}_{\ell,k}}^{-1}\mathbf{H}_{\ell,k}$, so that (17) becomes

$$\begin{aligned} \boldsymbol{\epsilon}_{i,k|k} =& \frac{1}{N_i}\sum_{\ell\in\mathcal{N}_i}\left(\mathbf{I} - \mathbf{M}_{\ell,k|k}\mathbf{\Upsilon}_{\ell,k}\right)\mathbf{A}_k\boldsymbol{\epsilon}_{\ell,k-1|k-1} \\ &+ \frac{1}{N_i}\sum_{\ell\in\mathcal{N}_i}\left(\mathbf{I} - \mathbf{M}_{\ell,k|k}\mathbf{\Upsilon}_{\ell,k}\right)\boldsymbol{\nu}_k - \frac{1}{N_i}\sum_{\ell\in\mathcal{N}_i}\mathbf{G}_{\ell,k}\boldsymbol{\omega}_{\ell,k}. \end{aligned} \tag{18}$$

Mean performance. Taking the statistical expectation of (18) and noting that $\boldsymbol{\nu}_k$ and $\boldsymbol{\omega}_{\ell,n}$ are zero-mean results in

$$\mathbb{E}\left\{\boldsymbol{\epsilon}_{i,k|k}\right\} = \frac{1}{N_i}\sum_{\ell\in\mathcal{N}_i}\left(\mathbf{I} - \mathbf{M}_{\ell,k|k}\mathbf{\Upsilon}_{\ell,k}\right)\mathbf{A}_k\mathbb{E}\left\{\boldsymbol{\epsilon}_{\ell,k-1|k-1}\right\}. \tag{19}$$

Therefore, given that all the nodes in the network are initialized with $\hat{\boldsymbol{x}}_{i,0|0} = \mathbb{E}\{\boldsymbol{x}_0\}$, the expression in (19) indicates that the algorithm operates in an unbiased fashion.

Mean square performance. Given the recursion for the state error vector in (18), the state error covariance matrix at node i can be expressed as

$$\begin{aligned} \mathbf{\Sigma}_{i,k} =& \mathbb{E}\left\{\boldsymbol{\epsilon}_{i,k|k}\boldsymbol{\epsilon}_{i,k|k}^{\mathsf{T}}\right\} \\ =& \frac{1}{N_i^2}\sum_{m\in\mathcal{N}_i}\sum_{n\in\mathcal{N}_i}\mathbb{E}\left\{\boldsymbol{\zeta}_{m,k}\boldsymbol{\zeta}_{n,k}^{\mathsf{T}}\right\} \\ &+ \frac{1}{N_i^2}\sum_{m\in\mathcal{N}_i}\sum_{n\in\mathcal{N}_i}\mathbb{E}\left\{\boldsymbol{\xi}_{m,k}\boldsymbol{\xi}_{n,k}^{\mathsf{T}}\right\} \\ &+ \frac{1}{N_i^2}\sum_{m\in\mathcal{N}_i}\sum_{n\in\mathcal{N}_i}\mathbb{E}\left\{\boldsymbol{\chi}_{m,k}\boldsymbol{\chi}_{n,k}^{\mathsf{T}}\right\} \end{aligned} \tag{20}$$

where

$$\begin{aligned} \boldsymbol{\zeta}_{m,k} &= \left(\mathbf{I} - \mathbf{M}_{m,k|k}\mathbf{\Upsilon}_{m,k}\right)\mathbf{A}_k\boldsymbol{\epsilon}_{m,k-1|k-1} \\ \boldsymbol{\xi}_{m,k} &= \left(\mathbf{I} - \mathbf{M}_{m,k|k}\mathbf{\Upsilon}_{m,k}\right)\boldsymbol{\nu}_k \\ \boldsymbol{\chi}_{m,k} &= \mathbf{G}_{m,k}\boldsymbol{\omega}_{m,k}. \end{aligned}$$

Now, we shall make the following standard assumptions, typically applied in Kalman filtering analysis [23]:

- The state evolution function and observation functions for all nodes in the network are time invariant, that is

$$\lim_{k\to\infty}\mathbf{A}_k = \mathbf{A} \;\;\&\;\; \forall\ell\in\mathcal{N}: \lim_{k\to\infty}\mathbf{H}_{\ell,k} = \mathbf{H}_\ell$$

- The state evolution and observation noises are stationary

$$\lim_{k\to\infty} \mathbf{C}_{\nu_k} = \mathbf{C}_{\nu} \;\;\&\;\; \forall \ell \in \mathcal{N}: \lim_{k\to\infty} \mathbf{C}_{\omega_{\ell,k}} = \mathbf{C}_{\omega_\ell}$$

- The matrix pairs $\forall \ell \in \mathcal{N}: \{\mathbf{A}_k, \mathbf{H}_{\ell,k}\}$ are observable and the matrix pair $\{\mathbf{A}_k, \mathbf{C}_{\nu}^{\frac{1}{2}}\}$ is controllable.

Then, it follows that for all nodes in the network, $\boldsymbol{\Upsilon}_{\ell,k}$ and $\mathbf{M}_{\ell,k|k}$ become time invariant. In addition, from Algorithm 2, a time invariant $\mathbf{M}_{\ell,k|k}$ results in the matrix $\mathbf{G}_{\ell,k}$ also becoming time invariant, which can be summarized as

$$\text{If } \forall \ell \in \mathcal{N}: \lim_{k\to\infty} \mathbf{M}_{\ell,k|k} = \mathbf{M}_{\ell} \Rightarrow \forall \ell \in \mathcal{N}: \lim_{k\to\infty} \mathbf{G}_{\ell,k} = \mathbf{G}_{\ell}$$

and therefore $\boldsymbol{\Sigma}_{i,k}$ converges.

Communication requirements. The communication requirements of different distributed Kalman filtering algorithms are compared in Table 1. Observe that the algorithm developed in this work has the lowest communication requirements, achieved by only sharing the intermediate states $\boldsymbol{\psi}_{\ell,k}$ and state error covariance matrix $\boldsymbol{\Gamma}_{\ell,k}$.

Table 1: Communicated variables for different distributed Kalman filtering strategies.

Strategy	Communicated Variables
Local [7, 11]	$\mathbf{H}_{\ell,k},\ \boldsymbol{y}_{\ell,k},\ \mathbf{C}_{\omega_{\ell,k}}^{-1}$
Diffusion [11]	$\mathbf{H}_{\ell,k},\ \boldsymbol{y}_{\ell,k},\ \mathbf{C}_{\omega_{\ell,k}}^{-1},\ \boldsymbol{\psi}_{\ell,k}$
Consensus [8]	$\mathbf{H}_{\ell,k},\ \boldsymbol{y}_{\ell,k},\ \mathbf{C}_{\omega_{\ell,k}}^{-1},\ \boldsymbol{\psi}_{\ell,k}$
Previous work [18]	$\mathbf{H}_{\ell,k},\ \mathbf{C}_{\omega_{\ell,k}}^{-1},\ \boldsymbol{\psi}_{\ell,k}$
Current work	$\boldsymbol{\psi}_{\ell,k},\ \boldsymbol{\Gamma}_{\ell,k}$

5. PERFORMANCE VERIFICATION

In order to demonstrate the performance of the proposed algorithm, we considered a target tracking application in a network of 20 nodes with the topology shown in Fig. 1. The state vector, $\boldsymbol{x}_k = [x_k, y_k, \dot{x}_k, \dot{y}_k]^{\mathsf{T}}$, consists of the positions, x_k, y_k, and velocities, $\dot{x}_k, \dot{y}_k$, in the horizontal and vertical positions, respectively.

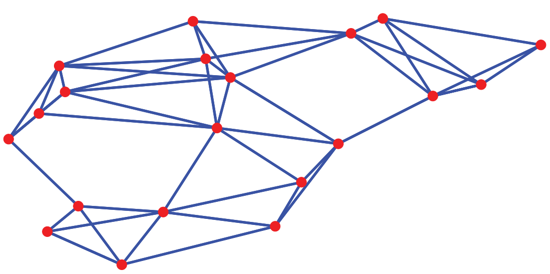

Fig. 1: The network of 20 nodes used in simulations.

The state is assumed to experience an unknown acceleration which is modeled as the process noise $\boldsymbol{\nu}_k = [\ddot{x}_k, \ddot{y}_k]^{\mathsf{T}}$. The state-space equations for this problem then become

$$\boldsymbol{x}_k = \begin{bmatrix} 1 & 0 & \Delta T & 0 \\ 0 & 1 & 0 & \Delta T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \boldsymbol{x}_{k-1} + \begin{bmatrix} \frac{1}{2}(\Delta T)^2 & 0 \\ 0 & \frac{1}{2}(\Delta T)^2 \\ \Delta T & 0 \\ 0 & \Delta T \end{bmatrix} \boldsymbol{\nu}_k$$

$$\boldsymbol{y}_{i,k} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \boldsymbol{x}_k + \boldsymbol{\omega}_{i,k}$$

where only the positions x_k, y_k are observed at each node i. The sampling interval was chosen to be $\Delta T = 1/25\,\mathrm{s}$. Fig. 2 shows that the proposed Kalman filter in Algorithm 2 was able to track the moving object and that the nodes in the network were able to reach a consensus within 2 s.

Next, the performance of the proposed algorithm was benchmarked against existing distributed Kalman filtering algorithms: "Local" Kalman filter (eq. (13) in [11]), "Diffusion" strategy (Algorithm 2 in [11]), and the "Consensus" Kalman filter algorithm (Algorithm 3 in [8]). The performance of the centralised scheme [22], which requires all the nodes to communicate their measurements to a fusion center is included for completeness. Fig. 3 shows the steady-state mean square deviation (MSD) at each node i, defined as

$$\mathrm{MSD}_i = \frac{1}{1500} \sum_{k=1000}^{2500} \|\boldsymbol{x}_k - \hat{\boldsymbol{x}}_{i,k}\|^2.$$

Observe that the proposed algorithm outperforms both the consensus and local schemes and achieves a steady-state mean square deviation (MSD) close to that of the diffusion Kalman filter, while having the lowest communication requirements (see Table 1).

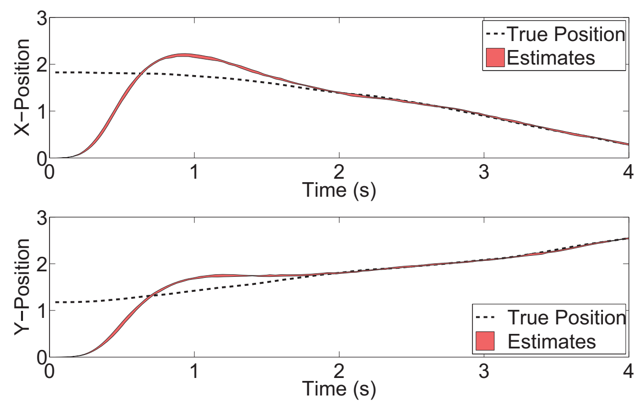

Fig. 2: Tracking performance of the proposed Kalman filter across all the nodes. Estimates of the target position across all 20 nodes lie within the region in red.

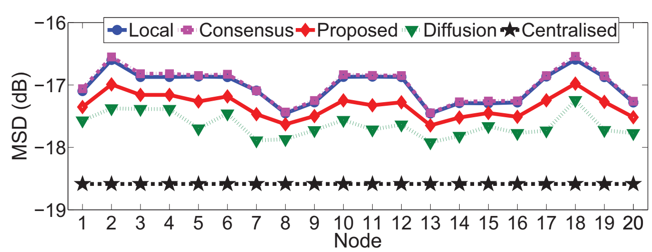

Fig. 3: MSD of the state estimates across all 20 nodes.

6. CONCLUSION

A novel distributed diffusion Kalman filtering algorithm has been developed in order to mitigate the need to share observation variables. The proposed algorithm only requires the diffusion of an intermediate state estimate and estimates of the state error covariance matrix. In this setting, the least amount of communication traffic is required over the network, while maintaining a performance comparable to that of the state of the art distributed Kalman filtering algorithms.

7. REFERENCES

[1] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1847–1864, November 2010.

[2] Y. Xia, D. P. Mandic, and A. H. Sayed, "An adaptive diffusion augmented CLMS algorithm for distributed filtering of noncircular complex signals," *IEEE Signal Processing Letters*, vol. 18, no. 11, pp. 659–662, November 2011.

[3] M. S. Mahmoud and H. M. Khalid, "Distributed Kalman filtering: A bibliographic review," *IET Control Theory Applications*, vol. 7, no. 4, pp. 483–501, March 2013.

[4] O. Hlinka, F. Hlawatsch, and P. M. Djuric, "Distributed particle filtering in agent networks: A survey, classification, and comparison," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 61–81, January 2013.

[5] H. R. Hashemipour, S. Roy, and A. J. Laub, "Decentralized structures for parallel Kalman filtering," *IEEE Transactions on Automatic Control*, vol. 33, no. 1, pp. 88–94, 1988.

[6] J. 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, April 1979.

[7] B. S. Rao and H. F. Durrant-Whyte, "Fully decentralised algorithm for multisensor Kalman filtering," *IEE Proceedings D - Control Theory and Applications*, vol. 138, no. 5, pp. 413–420, September 1991.

[8] R. Olfati-Saber, "Distributed Kalman filtering for sensor networks," *In Proceedings of the 46th IEEE Conference on Decision and Control*, pp. 5492 –5498, December 2007.

[9] R. Olfati-Saber, "Kalman-consensus filter: Optimality, stability, and performance," *In Proceedings of the 48th IEEE Conference on Decision and Control*, pp. 7036–7042, December 2009.

[10] Z. Hidayat, R. Babuska, B. De Schutter, and A. Nunez, "Decentralized Kalman filter comparison for distributed-parameter systems: A case study for a 1D heat conduction process," *In Proceedings of the 16th IEEE Conference on Emerging Technologies & Factory Automation (ETFA)*, pp. 1–8, September 2011.

[11] F. S. Cattivelli and A. H. Sayed, "Diffusion strategies for distributed Kalman filtering and smoothing," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2069 –2084, September 2010.

[12] A. H. Sayed, "Adaptive networks," *Proceedings of the IEEE*, vol. 102, no. 4, pp. 460–497, April 2014.

[13] A. H. Sayed, "Diffusion adaptation over networks," *Academic Press Library in Signal Processing*, vol. 3, pp. 323–454, 2014.

[14] S. Y. Tu and A. H. Sayed, "Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks," *IEEE Transactions on Signal Processing*, vol. 60, no. 12, pp. 6217–6234, December 2012.

[15] S. J. Julier and J. K. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," *In Proceedings of the 1997 American Control Conference*, vol. 4, pp. 2369–2373 vol.4, June 1997.

[16] J. Hu, L. Xie, and C. Zhang, "Diffusion Kalman filtering based on covariance intersection," *IEEE Transactions on Signal Processing*, vol. 60, no. 2, pp. 891–902, February 2012.

[17] Y. Zhang, C. Wang, N. Li, and J. Chambers, "Diffusion Kalman filter based on local estimate exchanges," *In Proceedings of the 2015 IEEE International Conference on Digital Signal Processing (DSP)*, pp. 828–832, July 2015.

[18] S. P. Talebi, S. Kanna, Y. Xia, and D. P. Mandic, "A distributed quaternion Kalman filter with applications to fly-by-wire systems," *In Proceedings of the 2016 IEEE International Conference on Digital Signal Processing (DSP), preprint*, October 2016.

[19] D. P. Mandic and V. S. L. Goh, *Complex valued nonlinear adaptive filters: Noncircularity, widely linear and neural models*, Wiley, Hoboken, NJ, 2009.

[20] S. Kanna, D. H. Dini, Y. Xia, R. S.Y. Hui, and D. P. Mandic, "Distributed widely linear Kalman filtering for frequency estimation in power networks," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 1, no. 1, pp. 45–57, March 2015.

[21] S. P. Talebi, S. Kanna, and D. P. Mandic, "A distributed quaternion Kalman filter with applications to smart grid and target tracking," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 4, pp. 477–488, December 2016.

[22] R. Olfati-Saber, "Distributed Kalman filter with embedded consensus filters," *In Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 8179–8184, December 2005.

[23] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear estimation*, vol. 1, Prentice Hall, Upper Saddle River, NJ, 2000.