

RESEARCH

Open Access

Distributed tracking with consensus on noisy time-varying graphs with incomplete data

Yongxiang Ruan^{1*}, Sudharman K Jayaweera¹ and Richard Scott Erwin²

Abstract

In this paper, we formulate a problem of distributed tracking with consensus on a time-varying graph with incomplete data and noisy communication links. We develop a framework to handle a time-varying network topology in which not every node has local observations to generate own local tracking estimates (incomplete data). A distributed tracking-with-consensus algorithm that is suitable for such a noisy, time-varying graph is proposed. We establish the graph conditions so that distributed consensus can be achieved in the presence of noisy communication links when the effective network graph is time-varying. The steady-state performance of the proposed distributed tracking with consensus algorithm is also analyzed and compared with that of the distributed local Kalman filtering with the centralized fusion and centralized Kalman filter. Simulation results and performance analysis of the proposed algorithm are given, showing that the proposed distributed tracking with consensus algorithm performs almost the same as the distributed local Kalman filtering with centralized fusion on noisy time-varying graphs with incomplete data, while the proposed algorithm has the additional advantages of robustness and scalability.

1 Introduction

Multisensor tracking problems have attracted the attention of many researchers in robotics, systems, and control theory over the past three decades [1-4]. Target tracking problems are of great importance in surveillance, security, and information systems for monitoring the behavior of agents using sensor networks, such as tracking pallets in warehouses, vehicles on roadways, or firefighters in burning buildings [5,6]. With the introduction of the concept of consensus, distributed tracking, and coordination without any fusion center have also received considerable attention in recent years [7,8].

Distributed consensus estimation in sensor networks has been studied with both fixed as well as time-varying communication topologies, taking into account issues such as link failure, packet losses and quantization or additive channel noise [8-20]. Olfati-Saber and Murray [9] considered the average consensus for first-order integrator networks with fixed and switching topologies. Kingston and Beard [10] extended the results of [9] to

the discrete-time models and relaxed the graph condition to instantaneously balanced, connected-over-time networks. Xiao and Boyd [11] considered discrete-time distributed averaging consensus over fixed and undirected graphs. They designed the weighted adjacency matrix to optimize the convergence rate by semidefinite programming. Huang and Manton [17] considered the discrete-time average consensus with fixed topologies and measurement noises. They introduced decreasing step size in the protocol to attenuate the noises. Li and Zhang [18-20] considered the continuous-time average consensus with time-varying topologies and communication noises, where time-varying consensus gains are adopted. They gave a necessary and sufficient condition for mean square average consensus with fixed graph topologies and sufficient conditions for mean square average consensus and almost sure consensus with time-varying graph topologies.

On the other hand, the distributed consensus tracking over networks with and without noiseless communication links among nodes have also been considered [21-26]. Recent work in [21,22] considered the distributed consensus tracking over a fixed graph with noiseless communication among nodes. A distributed Kalman filter with embedded consensus filters was proposed in

* Correspondence: yruan@unm.edu

¹Communications and Information Sciences Lab (CISL), Department of Electrical and Computer Engineering, University of New Mexico, Albuquerque, NM, USA

Full list of author information is available at the end of the article

[21] and further extended to heterogeneous and non-linear sensing models in [22]. Distributed Kalman filtering using one-step weighted averaging was considered in [23]. Each node desires an estimate of the observed system and communicates its local estimate with neighbors in the network. Then, new estimate is formed as a weighted average of the neighboring estimates, where the weights are optimized to yield a small estimation error covariance. In [24], the authors presented a distributed Kalman filter to estimate the state of a sparsely connected, large-scale, dynamical system. The complex large-scale systems are decomposed spatially into low-order overlapping subsystems. A fusion algorithm using bipartite fusion graphs and local average consensus algorithms is applied to fuse observations for those overlapping subsystems. A tracking control problem for multiagent consensus with an active leader and variable interconnection topology was considered in [25], where the state of the considered leader keeps changing and may not be measured. A neighbor-based local controller together with a neighbor-based state-estimation rule is given for each agent to track such a leader. In [26], the authors proposed a greedy stepsize sequence design to guarantee the convergence of distributed estimation consensus over a network with noisy links.

Distributed tracking with consensus, addressed in this paper and previous work [27,28], refers to the problem that a group of nodes need to achieve an agreement over the state of a dynamical system by exchanging tracking estimates over a network. For instance, space-object tracking with a satellite surveillance network that consists of fixed nodes that are connected together, and mobile nodes that could only have active links with other nodes within their communication radius could benefit from such distributed tracking with consensus, due to the fact that individual sensor nodes may not have enough observations of sufficient quality [29]. Thus, different sensor nodes may arrive at different local estimates regarding the same space object of interest [29]. Information exchange among nodes may improve the quality of local estimates, and consensus estimates may help avoid conflicting and inefficient distributed decisions. Other applications of this problem include flocking and formation control, real-time monitoring, target tracking, and global positioning system (GPS) [29,30]. In [28], the performance analysis of distributed tracking with consensus on noisy time-varying graphs was addressed, and later, the algorithm of distributed tracking with consensus with incomplete data was proposed without theoretical proof in [27].

The contributions of this work are as follows: (1) We formulate the problem of distributed tracking with consensus on a time-varying graph with incomplete data and noisy communication links. (2) We develop an

algorithm by combining distributed Kalman filtering with consensus updates to handle a time-varying network in which not every node has local observations to generate own local tracking estimates (incomplete data). (3) We establish the graph conditions so that the distributed consensus can be achieved when the graph topology is time-varying and with noisy communication links. (4) We analyze the steady-state performance of the distributed tracking with consensus on both fixed and time-varying graphs and compare with that of the distributed local Kalman filtering with centralized fusion and centralized Kalman filter.

Following notation will be used in this paper. At time j , an undirected graph is denoted by $G(j) = (V, E(j))$ for $j \geq 0$, where $V = \{1, 2, \dots, N\}$ is the node set and $E(j) \subseteq V \times V$ denotes the edge set. A random graph in which the existence of an edge between a pair of vertices in the set $V = \{1, 2, \dots, N\}$ is determined randomly and independent of other edges with probability $p \in (0, 1]$ is denoted by $G(N, p)$. The neighborhood of node n at time j is denoted by $\Omega_n(j) = \{l \in V | \{l, n\} \in E(j)\}$. Node n is said to have degree $d_n(j) = |\Omega_n(j)|$, where $|\cdot|$ refers to the cardinality of a set. Let the degree matrix be the diagonal matrix $D(j) = \text{diag}(d_1(j), \dots, d_N(j))$, where $\text{diag}(d_1, \dots, d_N)$ represents a diagonal matrix with (d_1, \dots, d_N) on its main diagonal. The adjacency matrix of the graph $G(j)$ is denoted by $A(j) = [A_{ln}(j)]$, where $A_{ln}(j) = 1$, if $\{l, n\} \in E(j)$, and $A_{ln}(j) = 0$ otherwise. The graph Laplacian matrix is denoted by $L(j) = D(j) - A(j)$. The Laplacian is a positive semidefinite matrix so that its eigenvalues can always be ordered as $0 = \lambda_1(L) \leq \lambda_2(L) \leq \dots \leq \lambda_N(L)$. The smallest eigenvalue $\lambda_1(L)$ is always equal to zero with $\mathbf{1}$ being the corresponding eigenvector where $\mathbf{1}$ is the vector of all ones of suitable length. For a connected graph, the second eigenvalue $\lambda_2(L) > 0$ and is termed the algebraic connectivity or the Fiedler value of the network [15,31]. Let $p(l, n)$ be the probability that links $\{l, n\}$ of the graph exists (for random graphs considered in this paper, we will always assume that $p(l, n) = p$ for $\forall l, n \in V$). For a directed graph, let (l, n) denote the directed edge from node l to node n . The direct sum of an $N \times N$ matrix B and an $M \times M$ matrix C will be an $(N + M) \times (N + M)$ matrix, denoted by $B \oplus C$, whereas the Kronecker product of an $N \times N$ matrix B and an $M \times M$ matrix C will be an $NM \times NM$ matrix, denoted by $B \otimes C$. From the properties of Kronecker product and eigenvalues [32], for an $N \times N$ matrix B and an $M \times M$ matrix C , if λ_n and ω_m are two eigenvalues of matrices B and C , respectively, for $1 \leq n \leq N$ and $1 \leq m \leq M$, then $\lambda_n \omega_m$ is an eigenvalue of $B \otimes C$. We denote the N -dimensional Euclidean space by \mathbb{R}^N . The $N \times N$ identity matrix is denoted by I_N , while $\mathbf{1}_N$ denotes the column vectors of all ones and \mathbf{e}_m denotes the column vectors with the m th element as one and the rest as zeros. The

operator $\|\cdot\|$ applied to a vector denotes the standard Euclidean 2-norm, while applied to matrices denotes the induced 2-norm, which is equivalent to the matrix spectral radius for symmetric matrices.

The remainder of this paper is organized as follows: Section 2 introduces our assumed system and network model and the proposed distributed tracking with consensus algorithm. In Section 3, conditions for achieving distributed consensus are discussed and the rate of convergence is quantified. The steady-state performance of the proposed distributed tracking with consensus algorithm is also analyzed in Section 3. Section 4 provides detailed simulation results and performance comparison of the proposed distributed tracking with consensus algorithm and that of distributed local Kalman filtering with centralized fusion and centralized Kalman filter. Finally, conclusions are given in Section 5.

2 Problem formulation and the proposed distributed tracking with consensus algorithm

A System model

Consider an N -node sensor network with a *connectivity graph* $G(j) = (V, E(j))$ at time j . Assume that the graph $G(j)$ is undirected, but time varying due to nodes moving out of communication ranges of each other or needing to cease transmissions to save battery power. The objective is to perform distributed tracking of a target and exchange tracking estimates over noisy communication links and try to reach consensus over the network. The tracking updates are performed at k instances, where k denotes the tracking time step ($k = 0, 1, \dots$). Consensus updates are performed between every two tracking updates, where $0 \leq j < J$ denotes the consensus iteration number, and J is the number of consensus iterations per tracking update (assumed to be fixed). The dynamics of the target evolves according to

$$\mathbf{x}(k+1) = \mathbf{F}\mathbf{x}(k) + \mathbf{w}(k); \quad \mathbf{x}(0) \sim \mathcal{N}(\bar{\mathbf{x}}(0), P_0). \quad (1)$$

The sensing model of the n th sensor is

$$\mathbf{y}_n(k) = \mathbf{H}_n \mathbf{x}(k) + \mathbf{v}_n(k), \quad \mathbf{y}_n \in \mathbb{R}^l. \quad (2)$$

Note that the observation matrices \mathbf{H}_n 's can be different for each node. Both $\mathbf{w}(k)$ and $\mathbf{v}_n(k)$ are assumed to be zero-mean white Gaussian noise (WGN) and $\mathbf{x}(0) \in \mathbb{R}^M$ is the initial state of the target. The second-order statistics of the process and measurement noise are given by

$$\mathbb{E}[\mathbf{w}(k)\mathbf{w}(k')^T] = \mathbf{Q}\delta_{kk'}, \quad \mathbb{E}[\mathbf{v}_n(k)\mathbf{v}_{n'}(k')^T] = \mathbf{R}_n\delta_{kk'}\delta_{nn'},$$

where $\delta_{kk'} = 1$ if $k = k'$ and $\delta_{kk'} = 0$, otherwise. Note that the above system model is linear, while the system model assumed in [29] is highly non-linear making it

difficult to analyze to obtain theoretical performance characterization.

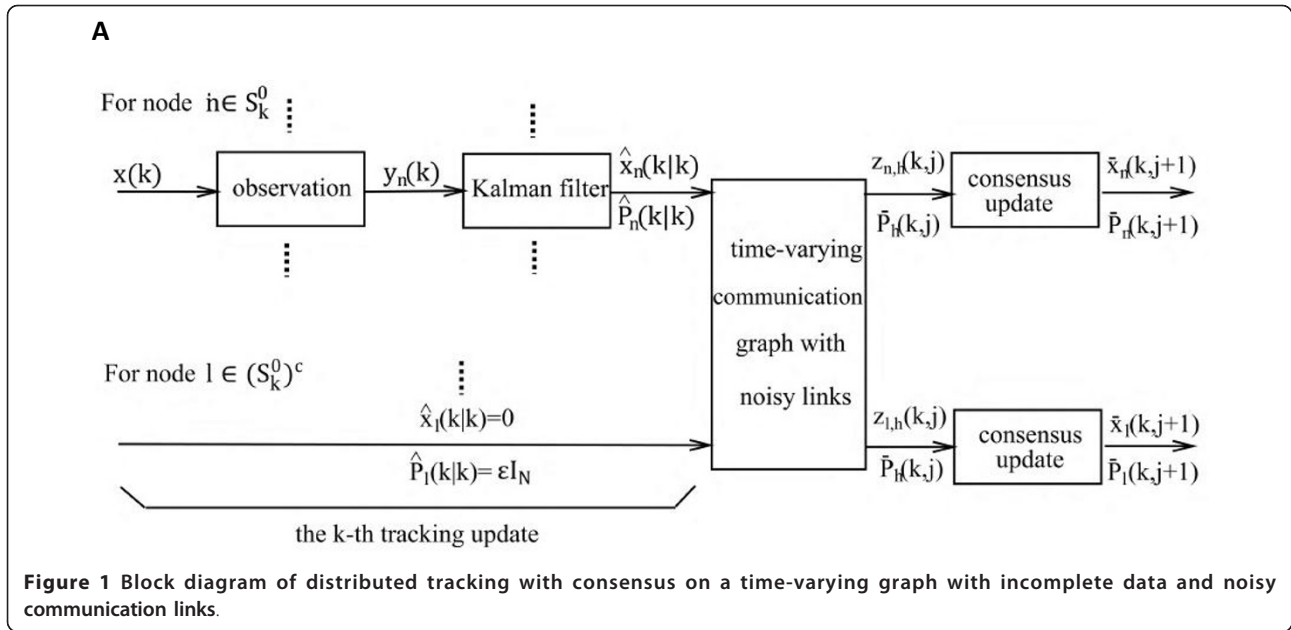
Figure 1 shows the system model of distributed tracking with consensus on a time-varying graph with incomplete data and noisy communication links. Let $\bar{\mathbf{x}}_n(k, j)$ denote the node n 's updated tracking estimate at the j th consensus iteration that follows the k th tracking update step with $\bar{\mathbf{x}}_n(k, 0) = \hat{\mathbf{x}}_n(k|k)$, where $\hat{\mathbf{x}}_n(k|k)$ is the n th node's filtered tracking estimate in the k th tracking update. The received data at node n from node l , for $n \neq l$, at iteration j can be written as

$$\mathbf{z}_{n,l}(k, j) = \bar{\mathbf{x}}_l(k, j) + \phi_{n,l}(j), \quad \text{for } 0 \leq j < J, \quad (3)$$

where $\phi_{n,l}(j)$ denotes the receiver noise at the node n in receiving the estimate of node l at iteration j and $\mathbf{z}_{n,n}(k, j) = \bar{\mathbf{x}}_n(k, j)$. Assume that $\mathbb{E}[\phi_{n,l}(j)] = \mathbf{0}_M$, $\mathbb{E}[\phi_{n,l}(j)\phi_{n,l}^T(j)] = \sigma_{n,l}^2 \mathbf{I}_M$ with $\sup_{n,l,j} \mathbb{E}[\|\phi_{n,l}(j)\|^2] = u < \infty$.

As depicted in Figure 1, at the end of the k th tracking update, each node n which has an observation of the target will have a filtered estimate $\hat{\mathbf{x}}_n(k|k)$ with associated covariance matrix $\hat{\mathbf{P}}_n(k|k)$. In order to improve the tracking estimate accuracy, it will exchange this filtered estimate with its neighbors over noisy communication links and try to reach consensus over the network. Note that, the goal here is to obtain a consensus tracking estimate over the local estimates at each tracking time step k , and thus, the consensus problem is essentially a problem of *consensus in estimation*.

Due to time-varying topology of the network, at any given tracking time step k , not all nodes may observe the target. Thus, these nodes will not have local tracking estimates, which is denoted as *incomplete data*. In previous consensus literature [8-20], all node estimates are taken into account in forming consensus estimates. However, the same method may not be extended to incomplete data case, since the nodes that mostly do not have observation ($\mathbf{y}_n(k) = \mathbf{v}_n(k)$) will exchange their predicted filtered estimates with others. Those predicted tracking estimates are considered as valid estimates and are taken into account to form consensus estimates, which results in inaccurate estimates and worsens the sensor network performance. By considering incomplete data here, the nodes do not have data will not communicate their invalid tracking estimates (by setting $\hat{\mathbf{x}}_n(k|k) = \mathbf{0}$ and $\hat{\mathbf{P}}_n(k|k) = \epsilon \mathbf{I}_M$ for some $\epsilon > 0$ instead). By introducing active node set and effective network graph, each node will notice which node has data in current consensus iteration. Only the estimates from active nodes are considered into forming consensus estimates. The estimates from non-active nodes will not be considered until it forms its updated estimate by fusing the filtered estimates from neighboring active nodes.



Since the non-active nodes join the consensus process without invalid tracking estimates, faster consensus process could be achieved while the network performance is still maintained.

In the space object tracking problem treated in [29], each node observes the target and locally processes its data in *data sampling period*. After forming local estimates, each node will share its information among neighboring nodes in *information sharing period*. Here, the information sharing rate is much larger compared with the data sampling rate so that each data sample node may exchange their local estimates many times in between, which may conceivably lead to better consensus estimates. The distributed tracking with consensus problem as formulated above may have other applications beyond the space object tracking problem, such as in multitarget tracking with a group of autonomous robots [33], battlefield life signs detection by Unmanned Aerial Vehicles (UAVs) [34], package tracking in warehouse by sensor networks [35], etc.

B Network model

We define the *active node set* S_k^j in a time-varying graph $G(j)$ as the set of nodes that have updated local estimates to be shared with others in the j th consensus iteration after the k th tracking update [29]. Define *effective network graph* of a network $G(j) = (V(j), E(j))$ with active node set S_k^j as $\tilde{G}(j) = (V(j), \tilde{E}(j))$, where $\tilde{E}(j) = E(j) \cap \left(\bigcup_{n \in S_k^j} \Upsilon_n^{\text{out}}(j) \right)$, where $\Upsilon_n^{\text{out}}(j) = \{(n, l) | (n, l) \in E(j)\}$ denotes the set of directed edges with initial vertex as n at iteration j . Note that,

the effective network graph $\tilde{G}(j)$ is a directed graph, which is obtained by removing the outgoing edges of the nodes that do not have data in $G(j)$. For a static graph $G(j) = G(V, E)$, $\tilde{E}(j)$ can be written as $\tilde{E}(j) = \tilde{E}(j-1) \cup_{l \in S_k^{j-1}} (\bigcup_{n \in \Omega_l} \Upsilon_n^{\text{out}})$, where $\tilde{E}(0) = E \cap \left(\bigcup_{n \in S_k^0} \Upsilon_n^{\text{out}} \right)$. Note that, the nodes that do not observe the target will not have updated local estimates to share at the beginning of consensus update process (at $j = 0$). However, as information exchange among nodes progresses, some of these nodes may be able to form their own updated local estimates at the consensus iteration j for $j > 0$. Therefore, the active node set S_k^j is time-varying and $S_k^j = S_k^{j-1} \cup_{l \in S_k^{j-1}} \Omega_l(j-1)$, where S_k^0 is the set of nodes that have observations of the target in the k th tracking update step as in Figure 1. Figure 2 shows the relation between the connectivity graph $G(j)$ and the effective network graph $\tilde{G}(j)$ for a graph of 6 nodes with active node set $S_k^j = \{1, 2, 4, 6\}$, where solid circles denote active nodes.

Let $I_{S_k^j}$ denote an $N \times N$ matrix generated from the active node set S_k^j as follows:

$$[I_{S_k^j}]_{nn'} = \begin{cases} 1 & \text{if } n = n' \text{ and } n' \in S_k^j \\ 0 & \text{else} \end{cases}$$

Note that, $I_{S_k^j}$ is a diagonal matrix with n 'th diagonal element equal to zero for $n' \in (S_k^j)^c$, where $(\cdot)^c$ denotes the set complement. By combining the connectivity graph $G(j)$ with the active node set S_k^j , we obtain the

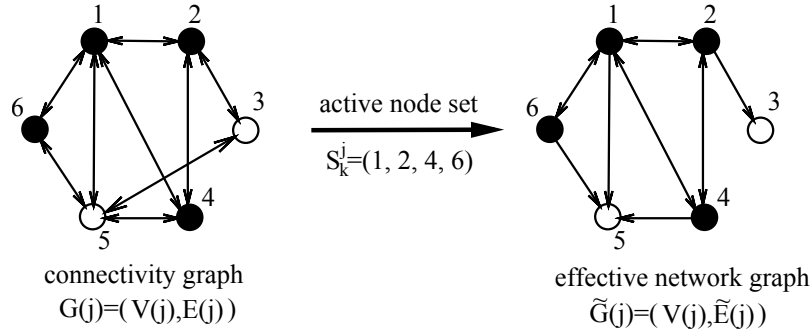


Figure 2 Connectivity graph and effective network graph.

effective network graph $\tilde{G}(j)$. Thus, the adjacency matrix of the effective network graph is given by $A(j) = A(j)I_{S_k^j}$. The corresponding degree matrix $D(j)$ can then be obtained from $A(j)$, and the Laplacian matrix is $L(j) = D(j) - A(j)$ by definition.

As an example, consider the same network model in Figure 2. The matrix $I_{S_k^j} = \text{diag}(1, 1, 0, 1, 0, 1)$. The Laplacian matrices of the connectivity graph and effective network graph are as follows:

$$L(j) = \begin{bmatrix} 4 & -1 & 0 & -1 & -1 & -1 \\ -1 & 3 & -1 & -1 & 0 & 0 \\ 0 & -1 & 2 & 0 & -1 & 0 \\ -1 & -1 & 0 & 3 & -1 & 0 \\ -1 & 0 & -1 & -1 & 4 & -1 \\ -1 & 0 & 0 & 0 & -1 & 2 \end{bmatrix} \quad \text{and} \quad L(j) = \begin{bmatrix} 3 & -1 & 0 & -1 & 0 & -1 \\ -1 & 2 & 0 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 0 & 2 & 0 & 0 \\ -1 & 0 & 0 & -1 & 3 & -1 \\ -1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

C Distributed tracking with consensus algorithm

In this section, we propose a distributed tracking and consensus algorithm for the above distributed tracking problem over a time-varying graph with incomplete data and noisy communication links. This algorithm is based on the architecture that was first proposed in [29] in the special context of consensus tracking in a satellite sensor network for situational awareness.

Figure 3 shows the timing diagram of tracking and consensus updates process in the proposed distributed tracking with consensus algorithm. As in Figure 3, at tracking time step k , node n is assumed to have completed its consensus iterations corresponding to time $k-1$. If the output of this consensus update following the $(k-1)$ th tracking update step is $\tilde{x}_n(k-1, J)$ with the associated covariance matrix $\tilde{P}_n(k-1, J)$, then node n sets $\tilde{x}_n(k-1|k-1) = \tilde{x}_n(k-1, J)$ and $\tilde{P}_n(k-1|k-1) = \tilde{P}_n(k-1, J)$. Next, at the k th tracking update step, each node n where $n \in S_k^j$, passes its observation $y_n(k)$ through its local Kalman filter as follows [1]:

$$\hat{x}_n(k|k-1) = F\tilde{x}_n(k-1|k-1),$$

$$\hat{P}_n(k|k-1) = F\tilde{P}_n(k-1|k-1)F^T + Q,$$

$$K_n(k) = \hat{P}_n(k|k-1)H_n^T(H_n\hat{P}_n(k|k-1)H_n^T + R_n)^{-1}, \quad (4)$$

$$\hat{x}_n(k|k) = \hat{x}_n(k|k-1) + K_n(k)(y_n(k) - H_n\hat{x}_n(k|k-1)),$$

$$\hat{P}_n(k|k) = (I - K_n(k)H_n)\hat{P}_n(k|k-1),$$

where $\tilde{x}_n(k-1|k-1) = \tilde{x}_n(k-1, J)$ with $\tilde{x}_n(0, J) = \tilde{x}(0)$ and $\tilde{P}_n(k-1|k-1) = \tilde{P}_n(k-1, J)$ with $\tilde{P}_n(0, J) = P_0$. Let $\tilde{x}(k-1, j) = [\tilde{x}_1(k-1, j)^T, \tilde{x}_2(k-1, j)^T, \dots, \tilde{x}_N(k-1, j)^T]^T$.

B

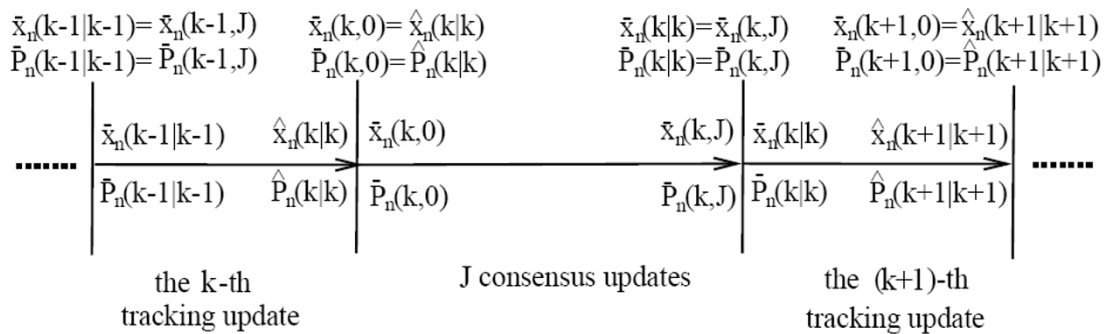


Figure 3 Timing diagram of tracking and consensus updates in the proposed algorithm for distributed tracking with consensus.

Denote $\bar{P}(k-1, j)$ as the covariance matrix corresponding to $\bar{\mathbf{X}}(k-1, j)$. The $\bar{P}_n(k-1, j)$ in (4) can be obtained by extracting the n th $M \times M$ main diagonal block of $\bar{P}(k-1, j)$.

Node n uses its filtered estimate $\hat{\mathbf{x}}_n(k|k)$ obtained by the above tracking update step as the initial estimate for consensus update exchanges by setting $\bar{\mathbf{x}}_n(k, 0) = \hat{\mathbf{x}}_n(k|k)$ with initial covariance matrix $\bar{P}(k, 0) = \hat{P}_1(k|k) \oplus \hat{P}_2(k|k) \oplus \cdots \oplus \hat{P}_N(k|k)$,^a where \oplus denotes the direct sum. On the other hand, for nodes $n \in (S_k^j)^c$, we may arbitrarily set $\hat{\mathbf{x}}_n(k|k) = \mathbf{0}$ and $\hat{P}_n(k|k) = I_M$ for some $\epsilon > 0$.

During the $(j+1)$ th consensus update, each node n forms a linear estimate of the following form as its consensus estimate:

$$\bar{\mathbf{x}}_n(k, j+1) = \bar{\mathbf{x}}_n(k, j) + \gamma_n(j) \sum_{l=1}^N \mathbf{A}_{n,l}(j) (\bar{\mathbf{z}}_{n,l}(k, j) - \bar{\mathbf{z}}_{n,n}(k, j)), \quad (5)$$

where $\gamma_n(j)$ is the n th node's weight coefficient at iteration j and $0 \leq j < J$. We set $\gamma_n(j) = \gamma(j)$ for $n \in S_k^j$ and $\gamma_n(j) = \frac{1}{\sum_{l=1}^N \mathbf{A}_{n,l}(j)}$ for $n \in S_k^j$ and $\sum_{l=1}^N \mathbf{A}_{n,l}(j) \neq 0$. For node n that does not have local tracking estimate, we assume that it will generate its estimate by averaging the tracking estimates from its neighbors.^b

By defining $\bar{\mathbf{X}}(k, j) = [\bar{\mathbf{x}}_1(k, j)^T \bar{\mathbf{x}}_2(k, j)^T, \dots, \bar{\mathbf{x}}_N(k, j)^T]^T$, the consensus update dynamics can be written in vector form as follows:

$$\bar{\mathbf{X}}(k, j+1) = \bar{\mathbf{X}}(k, j) - [(\Gamma(j)L(j)) \otimes I_M] \bar{\mathbf{X}}(k, j) - (\Gamma(j) \otimes I_M) \bar{\Phi}(j), \quad (6)$$

where

$$\Gamma(j) = \text{diag}(\gamma_1(j), \dots, \gamma_N(j)), \quad \bar{\Phi}(j) = [\phi_1(j)^T \cdots \phi_N(j)^T]^T$$

and $\phi_n(j) = -\sum_{l \in \Omega_n(j)} \phi_{n,l}(j)$. Note that, from (3), $\mathbb{E}[\bar{\Phi}(j)] = \mathbf{0}$ and $\sup_j \mathbb{E}[\|\bar{\Phi}(j)\|^2] = \eta \leq N(N-1)u < \infty$.

Let us define $\bar{\mathbf{e}}(k, j)$ to be the error vector at the j th consensus iteration after the k th tracking update: $\bar{\mathbf{e}}(k, j) \triangleq \bar{\mathbf{X}}(k, j) - (\mathbf{1} \otimes I_M) \mathbf{x}(k)$. From (6), it follows that

$$\bar{\mathbf{e}}(k, j+1) = (\mathbf{A}(j) \otimes I_M) \bar{\mathbf{e}}(k, j) - (\Gamma(j) \otimes I_M) \bar{\Phi}(j) + ((\mathbf{A}(j) \otimes I_M) - I) (\mathbf{1} \otimes I_M) \mathbf{x}(k), \quad (7)$$

where $\mathbf{A}(j) = I_N - \Gamma(j)L(j)$. Note that, this coefficient matrix $\mathbf{A}(j)$ is slightly different from the one proposed in [29]. In [29], $\mathbf{A}(j) = \bar{\mathbf{I}}(j) - \gamma(j)\bar{L}(j)$, where $\bar{\mathbf{I}}(j)$ and $\bar{L}(j)$ are the modified identity and Laplacian matrices. The required modification, however, does not lend itself to a convenient relation between the original matrices and the modified ones that can be used in mathematical derivations.

Note that, if the filtered estimate $\hat{\mathbf{x}}_n(k|k)$ at the end of the measurement update stage is an unbiased estimate,

then $\bar{\mathbf{x}}_n(k, 0)$ is also unbiased for all $n \in S_k^j$. From (5), since $\bar{\mathbf{x}}_n(k, j+1) = \frac{1}{\sum_{l=1}^N \mathbf{A}_{n,l}(j)} \sum_{l=1}^N \mathbf{A}_{n,l}(j) (\bar{\mathbf{x}}_l(k, j) + \phi_{n,l}(j))$ for $n \in (S_k^j)^c$, then $\bar{\mathbf{x}}_n(k, j+1)$ is also unbiased for $n \in (S_k^j)^c$ if $\bar{\mathbf{x}}_l(k, j)$ is unbiased for $l \in S_k^j$. From (7), it can be shown that the unbiasedness in consensus estimate $\bar{\mathbf{X}}(k, j)$ can be maintained if matrix $\mathbf{A}(j)$ satisfies the condition $((\mathbf{A}(j) \otimes I_M) - I) (\mathbf{1} \otimes I_M) = \mathbf{0}$, which is equivalent to requiring $((\mathbf{A}(j) - I_N) \mathbf{1}) \otimes I_M = \mathbf{0}$. It follows that the unbiasedness in consensus estimate $\bar{\mathbf{X}}(k, j)$ requires 0 to be an eigenvalue of the Laplacian matrix $L(j)$ with the associated eigenvector $\mathbf{1}$.^c Define the covariance matrix corresponding to $\bar{\mathbf{X}}(k, j)$ as $\bar{P}(k, j) = \mathbb{E}[\bar{\mathbf{e}}(k, j)\bar{\mathbf{e}}(k, j)^T]$. From (7) and unbiasedness condition, it can be easily seen that

$$\begin{aligned} \bar{P}(k, j+1) &= (\mathbf{A}(j) \otimes I_M) \bar{P}(k, j) (\mathbf{A}(j) \otimes I_M)^T \\ &+ \mathbb{E} \left\{ (\Gamma(j) \otimes I_M) \bar{\Phi}(j) \bar{\Phi}(j)^T (\Gamma(j) \otimes I_M) \right\}. \end{aligned} \quad (8)$$

As shown in Figure 3, after J consensus iterations, each node n will feed $\bar{\mathbf{x}}_n(k, J)$ back to their local Kalman filters by setting $\bar{\mathbf{x}}_n(k|k) = \bar{\mathbf{x}}_n(k, J)$ and $\bar{P}_n(k|k) = \bar{P}_n(k, J)$ before starting next tracking update at time $k+1$. Recall that here $\bar{P}_n(k, J)$ is the n th $M \times M$ main diagonal block of $\bar{P}(k, J)$. Algorithm 1 shows a summary of the steps in the proposed distributed tracking with consensus algorithm.

3 Performance analysis

A Conditions for achieving consensus

In this section, we analyze the convergence of the proposed distributed tracking with consensus algorithm and the convergence rate. Note that, the proofs of lemmas and theorems in this section are different from those in [16] due to vector state and incomplete data, which results in two stages of consensus process: obtaining complete data from incomplete data and reaching consensus on complete data. In the scenarios we consider, we assume that the information exchange rate during the consensus update process is much higher

Algorithm 1 Distributed tracking with consensus algorithm

Initialize: $\mathbf{x}(0)$, F , H_m , Q , R_n

while new data exist **do**

Kalman filtering in tracking process:

$$\hat{\mathbf{x}}_n(k|k-1) = F\hat{\mathbf{x}}_n(k-1|k-1)$$

$$\hat{P}_n(k|k-1) = F\hat{P}_n(k-1|k-1)F^T + Q$$

$$K_n(k) = \hat{P}_n(k|k-1)H_n^T (H_n\hat{P}_n(k|k-1)H_n^T + R_n)^{-1}$$

$$\hat{\mathbf{x}}_n(k|k) = \hat{\mathbf{x}}_n(k|k-1) + K_n(k) (\mathbf{y}_n(k) - H_n\hat{\mathbf{x}}_n(k|k-1))$$

$$\hat{P}_n(k|k) = (I - K_n(k)H_n) \hat{P}_n(k|k-1)$$

update the initial state of consensus process:

```

 $\bar{\mathbf{x}}_n(k, 0) = \hat{\mathbf{x}}_n(k|k)$ 
 $\bar{P}(k, 0) = \hat{P}_1(k|k) \oplus \hat{P}_2(k|k) \oplus \dots \oplus \hat{P}_N(k|k)$ 
 $j \leftarrow 0$ 
while  $j \leq J - 1$  do
 $\bar{\mathbf{x}}_n(k, j + 1) = \bar{\mathbf{x}}_n(k, j) + \gamma_n(j) \sum_{l=1}^N \mathbf{A}_{n,l}(j) (\mathbf{z}_{n,l}(k, j) - \mathbf{z}_{n,n}(k, j))$ 
 $j \leftarrow j + 1$ 
end while
 $\bar{\mathbf{x}}_n(k|k) = \bar{\mathbf{x}}_n(k, J)$ 
 $\bar{P}_n(k|k) = \bar{P}_n(k, J)$ 
 $k \leftarrow k + 1$ 
end while

```

compared with the data sampling rate for the tracking updates. Hence, we can assume that $J \gg 1$,^d guaranteeing enough time for information to be exchanged over the network so that consensus can be reached if the weight $\{\gamma(j)\}$ is chosen properly. As mentioned above, for a fixed k and $J \gg 1$, the consensus update process after the k th tracking update can be considered as a consensus in estimation problem. Thus, to simplify notation, in the following, we omit the tracking time step index k in $\bar{\mathbf{X}}(k, j)$.

We start by defining the consensus subspace \mathcal{C} as

$$\mathcal{C} = \{\mathbf{X} \in \mathbb{R}^{NM} | \mathbf{X} = \mathbf{1}_N \otimes \mathbf{a}, \mathbf{a} \in \mathbb{R}^M\}.$$

If the consensus algorithm (6) converges to the consensus subspace \mathcal{C} , each node estimate $\bar{\mathbf{x}}_n(j)$ will converge to the same value $\bar{\mathbf{x}}_n(j) = \mathbf{a}$ for $1 \leq n \leq N$, $\mathbf{a} \in \mathbb{R}^M$ and consensus is reached over the network. It is well known from the stochastic approximation literature [36] that, in order to ensure asymptotic convergence to consensus subspace, the weight coefficient $\gamma(j)$ must satisfy the *persistence condition* as follows

$$\gamma(j) > 0, \quad \sum_{j=0}^{\infty} \gamma(j) = \infty, \quad \sum_{j=0}^{\infty} \gamma(j)^2 < \infty. \quad (9)$$

We recall the following result on distance properties in \mathbb{R}^{NM} :

Lemma 1: Suppose that $\mathbf{X} \in \mathbb{R}^{NM}$ and consider the orthogonal decomposition $\mathbf{X} = \mathbf{X}_C + \mathbf{X}_{C^\perp}$. Then, the Euclidean distance $\rho(\mathbf{X}, \mathcal{C}) = \|\mathbf{X}_{C^\perp}\|$.

In the following, we prove that the consensus algorithm given in (6) converges almost surely (a.s.). This is achieved in two steps: First, Lemma 3 proves that the state vector sequence $\{\bar{\mathbf{X}}(j)\}_{j \geq 0}$ converges a.s. to the consensus subspace \mathcal{C} . Theorem 1 then completes the proof by showing that the sequence of component-wise averages $\{\bar{\mathbf{X}}_{\text{avg}}(j)\}_{j \geq 0}$ converges a.s. to a finite random variable Θ , where $\bar{\mathbf{X}}_{\text{avg}}(j) = \frac{1}{N}(\mathbf{1}^T \otimes I_M)\bar{\mathbf{X}}(j)$. The proof of Theorem 1 will require a basic result on convergence of Markov processes from [36], which is restated as

Lemma 2 in our context. Before stating the lemma, however, we need to introduce the notation of [36].

Let $\{\bar{\mathbf{X}}(j)\}_{j \geq 0}$ be a Markov process in \mathbb{R}^{NM} . Define the generating operator \mathcal{L} corresponding to $\{\bar{\mathbf{X}}(j)\}_{j \geq 0}$ as

$$\mathcal{L}V(j, \bar{\mathbf{X}}) = \mathbb{E}\{V(j+1, \bar{\mathbf{X}}(j+1)) | \bar{\mathbf{X}}(j) = \bar{\mathbf{X}}\} - V(j, \bar{\mathbf{X}}),$$

for functions $V(j, \bar{\mathbf{X}}), j \geq 0, \bar{\mathbf{X}} \in \mathbb{R}^{NM}$, provided the conditional expectation exists. If $D_{\mathcal{L}}$ is the domain of \mathcal{L} , then we say that $V(j, \bar{\mathbf{X}}) \in D_{\mathcal{L}}$ in a domain \mathcal{C} , if $\mathcal{L}V(j, \bar{\mathbf{X}})$ is finite for all $(j, \bar{\mathbf{X}}) \in \mathcal{C}$.

For $G \subset \mathbb{R}^{NM}$, the ϵ -neighborhood of G and its complement are defined as,

$$U_\epsilon(G) = \left\{ \mathbf{X} | \inf_{\mathbf{Y} \in G} \rho(\mathbf{X}, \mathbf{Y}) < \epsilon \right\}, \quad V_\epsilon(G) = \mathbb{R}^{NM} \setminus U_\epsilon(G). \quad (10)$$

With these notations, we may now state the desired lemma on the convergence of Markov processes:

Lemma 2 (Convergence of Markov Processes): Let $\{\bar{\mathbf{X}}(j)\}_{j \geq 0}$ be a Markov process with generating operator \mathcal{L} . Let there exist a non-negative function $V(j, \bar{\mathbf{X}}) \in D_{\mathcal{L}}$ in the domain $G \subset \mathbb{R}^{NM}$ for $j \geq 0$ and $\bar{\mathbf{X}} \in \mathbb{R}^{NM}$. Assume that

$$\inf_{j \geq 0, \bar{\mathbf{X}} \in V_\epsilon(G)} V(j, \bar{\mathbf{X}}) > 0, \quad \forall \epsilon > 0, \quad \text{and} \quad V(j, \bar{\mathbf{X}}) = 0, \quad \bar{\mathbf{X}} \in G,$$

$$\limsup_{\bar{\mathbf{X}} \rightarrow G} \sup_{j \geq 0} V(j, \bar{\mathbf{X}}) = 0, \quad \text{and} \quad \mathcal{L}V(j, \bar{\mathbf{X}}) \leq g(j)(1 + V(j, \bar{\mathbf{X}})) - \gamma(j)\varphi(j, \bar{\mathbf{X}}),$$

where $\varphi(j, \bar{\mathbf{X}}), \bar{\mathbf{X}} \in \mathbb{R}^{NM}$ is a non-negative function such that

$$\inf_{j, \bar{\mathbf{X}} \in V_\epsilon(G)} \varphi(j, \bar{\mathbf{X}}) > 0, \quad \forall \epsilon > 0; \gamma(j) > 0, \quad \sum_{j \geq 0} \gamma(j) = \infty;$$

$$\text{and} \quad g(j) > 0, \quad \sum_{j \geq 0} g(j) = \infty.$$

Then, the Markov process $\{\bar{\mathbf{X}}(j)\}_{j \geq 0}$ with an arbitrary initial distribution

converges almost surely (a.s.) to G as $j \rightarrow \infty$:

$$\mathbb{P}\left(\lim_{j \rightarrow \infty} \rho(\bar{\mathbf{X}}(j), G) = 0\right) = 1.$$

Proof Proof is a vector generalization of that in [16], and is omitted.

Lemma 2 guarantees a.s. convergence of a general Markov process with an arbitrary initial distribution under the assumption of the existence of a Lyapunov function $V(j, \bar{\mathbf{X}})$. In fact, the state vector sequence $\{\bar{\mathbf{X}}(j)\}_{j \geq 0}$ given in (6) is a Markov process, since $\mathbb{P}[\bar{\mathbf{X}}(j) | \bar{\mathbf{X}}(j-1), \dots, \bar{\mathbf{X}}(0)] = \mathbb{P}[\bar{\mathbf{X}}(j) | \bar{\mathbf{X}}(j-1)]$. In the next lemma, we prove that the state estimate sequence $\{\bar{\mathbf{X}}(j)\}_{j \geq 0}$ given in (6) converges a.s. to the consensus subspace \mathcal{C} by showing that the consensus algorithm

over an undirected effective network graph satisfies the Lyapunov function assumptions of Lemma 2.

Lemma 3 (a.s. convergence of the proposed algorithm to the consensus subspace): Consider the consensus algorithm in (6) with initial state $\bar{\mathbf{X}}(0) \in \mathbb{R}^{NM}$. The weight coefficients satisfy the persistence condition in (9). Assume that the undirected connectivity graph Laplacian $L(j)$ is independent of communication noise $\varphi_{n,l}(j)$ for $1 \leq n, l \leq N$. If $L(j) = \bar{L} + \tilde{L}(j)$ with mean $\bar{L} = \mathbb{E}[L(j)]$ is such that $\lambda_2(\bar{L}) > 0$ and $p(l, n) > 0$ for $\{l, n\} \in E(j)$, then

$$\mathbb{P} \left[\lim_{j \rightarrow \infty} \rho(\bar{\mathbf{X}}(j), \mathcal{C}) = 0 \right] = 1.$$

Proof See Appendix A.

Lemma 3 shows that the state estimate sequence $\{\bar{\mathbf{X}}(j)\}_{j \geq 0}$ given in (6) converges a.s. to the consensus subspace \mathcal{C} . The key to the proof is to show that the directed effective network graph will become an undirected graph after all nodes have local estimates and the consensus algorithm over this undirected effective network graph satisfies the condition required in Lemma 2. In the following theorem, we state our main result and complete the convergence proof for the proposed distributed tracking with consensus algorithm by showing that the sequence of component-wise averages $\{\bar{\mathbf{X}}_{\text{avg}}(j)\}_{j \geq 0}$ converges a.s. to a finite random variable Θ , where $\bar{\mathbf{X}}_{\text{avg}}(j) = \frac{1}{N}(\mathbf{1}^T \otimes I_M)\bar{\mathbf{X}}(j)$.

Theorem 1 (a.s. convergence to a finite random vector): Consider the consensus algorithm in (6) with initial state $\bar{\mathbf{X}}(0) \in \mathbb{R}^{NM}$. The weight coefficients satisfy the persistence condition in (9). Assume that the time-varying connectivity graph Laplacian $L(j)$ is independent of communication noise $\varphi_{n,l}(j)$ for $1 \leq n, l \leq N$. If $L(j) = \bar{L} + \tilde{L}(j)$ with mean $\bar{L} = \mathbb{E}[L(j)]$ is such that $\lambda_2(\bar{L}) > 0$, and if $p(l, n) > 0$ for $\{l, n\} \in E(j)$, then there exists an almost sure finite real random vector Θ such that

$$\mathbb{P} \left[\lim_{j \rightarrow \infty} \bar{\mathbf{X}}(j) = \mathbf{1}_N \otimes \Theta \right] = 1.$$

Proof Since the mean connectivity graph \bar{L} is connected with non-zero link probability, for j large enough, each node will receive information from one another and generate its updated local estimate. For a fixed k , let $J_k = \inf\{j | (S_k^j)^c = \emptyset, j \geq 0\}$. Then, $\Gamma(j) = \gamma(j)I_N$ for $j \geq J_k$ and (6) becomes

$$\bar{\mathbf{X}}(j+1) = \bar{\mathbf{X}}(j) - \gamma(j) [L(j) \otimes I_M \bar{\mathbf{X}}(j) + \bar{\Phi}(j)] \quad \text{for } j \geq J_k. \quad (11)$$

Define the average of $\bar{\mathbf{X}}(j)$ as $\bar{\mathbf{X}}_{\text{avg}}(j) = \frac{1}{N}(\mathbf{1}^T \otimes I_M)\bar{\mathbf{X}}(j)$. Multiply both sides of (11) by $\frac{1}{N}(\mathbf{1}^T \otimes I_M)$ and use the

fact that $\mathbf{1}^T L(j) = \mathbf{0}_N$, so that for $(S_k^j)^c = \emptyset$, we have

$$\bar{\mathbf{X}}_{\text{avg}}(j+1) = \bar{\mathbf{X}}_{\text{avg}}(j) - \varepsilon(j) = \bar{\mathbf{X}}_{\text{avg}}(J_k) - \sum_{J_k \leq l \leq j} \varepsilon(l),$$

where $\varepsilon(j) = \frac{\gamma(j)}{N}(\mathbf{1}^T \otimes I_M)\bar{\Phi}(j)$. Assuming that receiver noise is zero-mean and time independent, we obtain

$$\begin{aligned} \mathbb{E}[\|\varepsilon(j)\|^2] &= \frac{\gamma^2(j)}{N^2} \mathbb{E}[\bar{\Phi}(j)^T (\mathbf{1}^T \otimes I_M)^T (\mathbf{1}^T \otimes I_M) \bar{\Phi}(j)] \\ &= \frac{\gamma^2(j)}{N^2} \mathbb{E} \left[\sum_{1 \leq n \leq N} (\phi_n(j))^T \phi_n(j) \right], \end{aligned}$$

where $\phi_n(j) = -\sum_{l \in \Omega_n(j)} \varphi_{n,l}(j)$ denotes the total incoming noise from node $l \in \Omega_n(j)$ to node n and the last step follows from the independence of $\varphi_l(j)$ and $\varphi_n(j)$. By assuming that $\mathbb{E}[\varphi_{l,n}(j)\varphi_{l,n}(j)^T] = \sigma^2 I_M$ for $1 \leq l, n \leq N$, we obtain

$$\mathbb{E}[\|\varepsilon(j)\|^2] \leq \frac{\gamma^2(j)}{N^2} MN(N-1)\sigma^2 = \frac{\gamma^2(j)M(N-1)}{N}\sigma^2.$$

From independence of $\bar{\mathbf{X}}(j)$ and $\bar{\Phi}(j)$ and the independence of noise over time, we then have that

$$\mathbb{E}[\|\bar{\mathbf{X}}_{\text{avg}}(j+1)\|^2] \leq \mathbb{E}[\|\bar{\mathbf{X}}_{\text{avg}}(J_k)\|^2] + \sum_{l \geq J_k} \frac{\gamma^2(l)M(N-1)}{N}\sigma^2 \leq \infty.$$

Denote $\bar{\mathbf{X}}_{\text{avg}}(j) = [\bar{X}_{\text{avg},1}(j) \dots \bar{X}_{\text{avg},M}(j)]^T$. It can be easily seen that

$$\mathbb{E}[(\bar{X}_{\text{avg},m}(j+1))^2] \leq \mathbb{E}[(\bar{X}_{\text{avg},m}(J_k))^2] + \sum_{l \geq J_k} \frac{\gamma^2(l)(N-1)}{N}\sigma^2 \leq \infty.$$

Hence, the sequence $\{\bar{X}_{\text{avg},m}(j)\}$ is an L_2 bounded martingale and thus converges a.s. in L_2 to a finite random scalar θ . Define $\bar{\mathbf{X}}_m(j) = [\mathbf{e}_m^T \bar{\mathbf{x}}_1 \dots \mathbf{e}_m^T \bar{\mathbf{x}}_N]^T$. From the conclusion of Lemma 3, we have that

$\mathbb{P}[\lim_{j \rightarrow \infty} \|\bar{\mathbf{X}}(j) - \mathbf{1}_N \otimes \bar{\mathbf{X}}_{\text{avg}}(j)\| = 0] = 1$, which implies that

$\mathbb{P}[\lim_{j \rightarrow \infty} \|\bar{\mathbf{X}}_m(j) - \bar{X}_{\text{avg},m}(j)\mathbf{1}_N\| = 0] = 1$. Then, we obtain that

$$\mathbb{P}[\lim_{j \rightarrow \infty} \bar{\mathbf{X}}_m(j) = \theta \mathbf{1}_N] = 1 \text{ and the theorem follows.}$$

Theorem 1 shows that the proposed distributed tracking with consensus algorithm will reach consensus almost surely and the consensus estimate $\lim_{j \rightarrow \infty} \bar{\mathbf{x}}(j)$ is a finite random vector Θ . Since the consensus algorithm in (6) falls in the framework of stochastic approximation, we may also analyze the convergence rate of the consensus algorithm based on the ODE method (ordinary difference equation) [37]. The next theorem characterizes an upper bound to the convergence rate of the proposed distributed tracking with consensus algorithm.

Theorem 2 (convergence rate): Consider the consensus algorithm in (6) with initial state $\bar{\mathbf{X}}(0) \in \mathbb{R}^{NM}$. The weight coefficients satisfy the persistence condition in (9) and $\gamma(j) \leq \frac{2}{\lambda_2(\bar{\mathbf{L}}) + \lambda_n(\bar{\mathbf{L}})}$. Assume that the time-varying connectivity graph Laplacian $L(j)$ is independent of communication noise $\varphi_{n,l}(j)$ for $1 \leq n, l \leq N$. For $j \geq J_k$, the effective network graph Laplacian is $L(j) = \bar{\mathbf{L}} + \tilde{\mathbf{L}}(j)$ with mean $\bar{\mathbf{L}} = \mathbb{E}[L(j)]$. If the connectivity graph Laplacian $L(j)$ with mean $\bar{\mathbf{L}} = \mathbb{E}[L(j)]$ is such that $\lambda_2(\bar{\mathbf{L}}) > 0$, and if $p(l, n) > 0$ for $\{l, n\} \in E(j)$, the convergence rate,^e of the proposed consensus algorithm is bounded by $-\lambda_2(\bar{\mathbf{L}}) \left(\frac{1}{J-J_k} \sum_{J_k \leq j \leq J} \gamma(j) \right)$.

Proof For a fixed i , let $J_k = \inf\{j | (S_k^j)^c = \emptyset, j \geq 0\}$. From the asymptotic unbiasedness of Θ , we have $\lim_{j \rightarrow \infty} \mathbb{E}[\bar{\mathbf{X}}(j)] = \mathbf{1}_N \otimes \mathbf{r}$, where $\mathbf{r} = \bar{\mathbf{X}}_{\text{avg}}(J_k)$. For $j \geq J_k$, define $\Xi(j) = I_{NM} - \gamma(j)(\bar{\mathbf{L}} \otimes I_M)$, where $\bar{\mathbf{L}} = \mathbb{E}[L(j)]$. Using the fact that $L(j)$ and $\bar{\mathbf{X}}(j)$ are independent, and $\mathbb{E}[\tilde{\Phi}(j)] = \mathbf{0}_{NM}$, from (6), we have that

$$\mathbb{E}[\bar{\mathbf{X}}(j+1)] = \Xi(j)\mathbb{E}[\bar{\mathbf{X}}(j)] = \prod_{l=J_k}^j \Xi(l)\mathbb{E}[\bar{\mathbf{X}}(J_k)], \quad \forall j \geq J_k. \quad (12)$$

From the persistence condition $\gamma(j) > 0$, $\sum_{j \geq 0} \gamma(j) = \infty$ and $\sum_{j \geq 0} \gamma^2(j) \leq \infty$ [16], it follows that $\gamma(j) \rightarrow 0$. From the mixed-product property of Kronecker product $(A \otimes B)(C \otimes D) = AB \otimes CD$ and $(I_{NM} - \gamma(j)\bar{\mathbf{L}})\mathbf{1}_N = \mathbf{1}_N$ [32], we have

$$\mathbf{1}_N \otimes \mathbf{r} = \Xi(j)(\mathbf{1}_N \otimes \mathbf{r}). \quad (13)$$

From (12) and (13), it can be shown that

$$\begin{aligned} \|\mathbb{E}[\bar{\mathbf{X}}(j)] - \mathbf{1}_N \otimes \mathbf{r}\| &\leq \prod_{J_k \leq l \leq j-1} \bar{\rho}(1 - \gamma(l)\bar{\mathbf{L}}) \|\mathbb{E}[\bar{\mathbf{X}}(J_k)] - \mathbf{1}_N \otimes \mathbf{r}\| \\ &= \prod_{J_k \leq l \leq j-1} (1 - \gamma(l)\lambda_2(\bar{\mathbf{L}})) \|\mathbb{E}[\bar{\mathbf{X}}(J_k)] - \mathbf{1}_N \otimes \mathbf{r}\|, \end{aligned}$$

where last step follows from Lemma 8 of [15] and $\bar{\rho}(\cdot)$ denotes the spectral radius of a matrix. From the assumption on weight coefficient $\gamma(j)$, we have $0 \leq \gamma(l)\lambda_2(\bar{\mathbf{L}}) \leq 1$. Since $1 - \alpha \leq e^{-\alpha}$ for $0 \leq \alpha \leq 1$, we then have that

$$\|\mathbb{E}[\bar{\mathbf{X}}(j)] - \mathbf{1}_N \otimes \mathbf{r}\| \leq \left(e^{-\lambda_2(\bar{\mathbf{L}}) \left(\sum_{J_k \leq l \leq j-1} \gamma(l) \right)} \right) \|\mathbb{E}[\bar{\mathbf{X}}(J_k)] - \mathbf{1}_N \otimes \mathbf{r}\|. \quad (14)$$

Therefore, as $j \rightarrow J$, the convergence rate is bounded by $-\lambda_2(\bar{\mathbf{L}}) \left(\frac{1}{J-J_k} \sum_{J_k \leq l \leq J} \gamma(l) \right)$, which depends on the algebraic connectivity $\lambda_2(\bar{\mathbf{L}})$ and the weights $\gamma(j)$, for $J_k \leq j \leq J$.

Theorem 2 shows that the convergence rate of the proposed algorithm depends on the topology through the algebraic connectivity $\lambda_2(\bar{\mathbf{L}})$ of the effective network graph $\tilde{\mathbf{G}}(j)$ and through weights $\gamma(j)$, for $j \geq J_k$. Since for

$\bar{\mathbf{L}} = \mathbb{E}[L(j)] = \mathbb{E}[L(j)]$ and $L(j) = L(j)$, we have $\bar{\mathbf{L}} = \mathbb{E}[L(j)] = \mathbb{E}[L(j)]$. In (14), $\lambda_2(\bar{\mathbf{L}})$ is the algebraic connectivity of the mean Laplacian corresponding to the time-varying network graphs. For a static network, this reduces to the algebraic connectivity of the static Laplacian \mathbf{L} .

Since the consensus algorithm in (6) is iterative, whose energy consumption is proportional to the time necessary to achieve consensus and inversely proportional to transmit power. From [38,39], for energy-constrained sensor networks, there exists a tradeoff between convergence time that depends on network connectivity and the transmit power of each node necessary to establish the links with the desired reliability. Therefore, we can minimize the energy consumption for consensus process by optimizing transmit power, network topology, and weights $\gamma(j)$.

B Steady-state analysis for noiseless graphs

In this section, we analyze the steady-state performance of the proposed distributed tracking with consensus algorithm. When the filter reaches steady state, the error covariance matrix is time invariant and the corresponding filter gain is constant. Therefore, finding the steady state of the proposed algorithm will help understanding its asymptotic behavior, analyzing error covariance, and filter design. From (8), it can be seen that the propagation of communication noise implies the non-existence of an upper bound to the covariance matrix. Therefore, the covariance matrix in the Kalman filter may not also converge and the filter may not reach steady state. However, time-varying graph assumption does not affect the existence of steady state. Since for $J \rightarrow \infty$, consensus is reached over the network and the outputs of the consensus update $\bar{\mathbf{X}}_n(k, J)$ and $\bar{\mathbf{P}}(k, J)$ depend only on the inputs $\bar{\mathbf{X}}_n(k, 0)$ and $\bar{\mathbf{P}}(k, 0)$ for complete data case with noiseless time-varying graphs (for incomplete data case with noiseless time-varying graphs, this property still holds for some special types of graphs). Hence, the combined system of distributed tracking with consensus can be transformed into a Kalman filter with time-invariant parameters. Therefore, steady state can still be reached [1]. In the following, assuming noiseless time-varying graphs, we start with steady-state analysis for the case with complete data, and then, we extend the results to the case with incomplete data.

1) Complete data with noiseless time-varying graphs

Here, we assume complete data, a scalar target state $x \in \mathbb{R}^1$ (for simplicity) and noiseless time-varying graphs, where the connectivity graph Laplacian $L(i)$ with mean $\bar{\mathbf{L}} = \mathbb{E}[L(j)]$ is such that $\lambda_2(\bar{\mathbf{L}}) > 0$, and $p(l, n) > 0$ for $\{l, n\} \in E(j)$. Note that, since a closed form equation for

$\hat{P}_n(k+1|k)$ cannot be easily obtained when the target state $x \in \mathbb{R}^M$ for $M > 1$, the following derivation would not apply to vector state.

From the result of Theorem 1 for scalar target state, it can be shown that $\lim_{j \rightarrow \infty} \bar{X}(k, j) = \bar{X}_{\text{avg}}(k, 0) \mathbf{1}_N$, where $\bar{X}_{\text{avg}}(k, j) = \frac{1}{N} \mathbf{1}^T \bar{X}(k, j)$. From the definition of $\bar{X}(k, j)$ and $\bar{x}_n(k, 0) = \hat{x}_n(k|k)$, we have for $1 \leq n \leq N$

$$\lim_{j \rightarrow \infty} \bar{x}_n(k, j) = \frac{1}{N} \sum_{n=1}^N \hat{x}_n(k|k). \quad (15)$$

With the assumptions above, the covariance matrix (8) in the $(j+1)$ th consensus iteration after the k th tracking update simplifies to $\bar{P}(k, j+1) = \mathbf{A}(j) \bar{P}(k, j) \mathbf{A}(j)^T$. For complete data case, $L(j) = L(j)$. Since $\mathbf{1}^T L(j) = 0$, from (7), we have $\mathbf{1}^T \mathbf{A}(j) = \mathbf{1}$. Then, we can obtain that

$$\mathbf{1}^T \bar{P}(k, j+1) \mathbf{1} = \mathbf{1}^T \bar{P}(k, j) \mathbf{1}. \quad (16)$$

By applying the result of Theorem 1, we have $\lim_{j \rightarrow \infty} \bar{P}(k, j) = (\bar{X}_{\text{avg}}(k, 0) - x(k))^2 \mathbf{1} \mathbf{1}^T$. Since all the elements in $\lim_{j \rightarrow \infty} \bar{P}(k, j)$ are equal, from (16), it follows that

$$\lim_{j \rightarrow \infty} \bar{P}(k, j) = \frac{\mathbf{1}^T \bar{P}(k, 0) \mathbf{1}}{N^2} \mathbf{1} \mathbf{1}^T = \frac{\sum_{n=1}^N \hat{P}_n(k|k)}{N^2} \mathbf{1} \mathbf{1}^T. \quad (17)$$

Since $\bar{P}_n(k, j)$ is the n th $M \times M$ main diagonal block of $\bar{P}(k, j)$, we have the covariance matrix for node n ($1 \leq n \leq N$) as below:

$$\lim_{j \rightarrow \infty} \bar{P}_n(k, j) = \frac{\sum_{n=1}^N \hat{P}_n(k|k)}{N^2}. \quad (18)$$

From (15) and (18), we have $\bar{x}_n(k, j) = \bar{x}_l(k, j)$ and $\bar{P}_n(k, j) = \bar{P}_l(k, j)$ for $j \rightarrow \infty$ and $1 \leq n, l \leq N$. Then, each node n sets $\bar{x}_n(k|k) = \bar{x}_n(k, j)$ and $\bar{P}_n(k|k) = \bar{P}_n(k, j)$. From (4), we have $\hat{x}_n(k+1|k) = \hat{x}_l(k+1|k)$ and $\hat{P}_n(k+1|k) = \hat{P}_l(k+1|k)$ for $1 \leq n, l \leq N$ and it follows that for $1 \leq n \leq N$

$$\begin{aligned} \hat{x}_n(k+1|k) &= F \hat{x}_n(k|k-1) - F \frac{1}{N} \sum_{q=1}^N [K_q(k)(y_q(k) - H_q \hat{x}_q(k|k-1))], \\ \hat{P}_l(k+1|k) &= Q + \frac{1}{N^2} \sum_{q=1}^N F(I - K_q(k)H_q) \hat{P}_q(k|k-1) F^T. \end{aligned} \quad (19)$$

Let $\hat{x}_n(k+1|k) = \hat{x}(k+1|k)$ and $\hat{P}_n(k+1|k) = \hat{P}(k+1|k)$. Then, the combined system of distributed tracking with consensus can be transformed into a single Kalman filter as follows:

$$\begin{aligned} \hat{x}(k+1|k) &= F \hat{x}(k|k-1) + \frac{F}{N} \sum_{n=1}^N [K_n(k)(y_n(k) - H_n \hat{x}_n(k|k-1))], \\ K_n(k) &= \hat{P}(k|k-1) H_n^T [H_n \hat{P}(k|k-1) H_n^T + R_n]^{-1}, \\ \hat{P}(k+1|k) &= Q + \frac{1}{N^2} \sum_{n=1}^N [F \hat{P}(k|k-1) F^T - F K_n(k) (H_n \hat{P}(k|k-1) H_n^T + R_n) \\ &\quad \times K_n(k)^T F^T]. \end{aligned} \quad (20)$$

Theorem 3 Consider the system dynamics in (1) and (2) and the Kalman filter in (20). Assume that the connectivity graph Laplacian $L(j)$ with mean $\bar{L} = \mathbb{E}[L(j)]$ is such that $\lambda_2(\bar{L}) > 0$, and $p(l, n) > 0$ for $\{l, n\} \in E(j)$. If the pair (F, H_n) is observable for $1 \leq n \leq N$, then the prediction covariance matrix $\hat{P}(k|k-1)$ converges to a constant matrix

$$\lim_{k \rightarrow \infty} \hat{P}(k|k-1) = P,$$

where P is the unique definite solution of the discrete algebraic Riccati equation (DARE)

$$P = Q + \frac{1}{N^2} \sum_{n=1}^N [F P F^T - F P H_n^T (H_n P H_n^T + R_n)^{-1} H_n P F^T]. \quad (21)$$

Proof See Proof of Theorem 4. By setting $m = N$ and $\beta_n(k) = 1$ for $1 \leq n \leq N$, the Kalman filter in (24) can be reduced to the one in (20). Theorem 3 can be considered as a special case of Theorem 4. Thus, it can be proved in a similar manner.

As a consequence of Theorem 3, the local Kalman filter gain converges to

$$\lim_{k \rightarrow \infty} K_n(k) = P H_n^T [H_n P H_n^T + R_n]^{-1}.$$

From (21), it can be seen that $\lim_{N \rightarrow \infty} P = Q$, i.e., as the size of the sensor network N increases, the steady-state covariance P , which in this case is a scalar, will decrease. This implies that if the network size is large enough, asymptotically the tracking is noiseless and follows the target exactly. It is obvious that this result still holds for distributed local Kalman filtering with centralized fusion. However, the distributed tracking with consensus results in the same performance even if the graph is time-varying and it also improves the robustness and scalability due to consensus exchanges. For the assumed scalar case, for example, if $H_n = H$ and $R_n = R$ for $1 \leq n \leq N$, then we have $K = \frac{HP}{H^2 P + R}$ and $P = \frac{-B + \sqrt{B^2 + 4H^2 QR}}{2H^2}$, where $B = \left(1 - \frac{F^2}{N}\right) R - H^2 Q$. This implies that for the same sensing model, each node will have the same Kalman gain K and prediction covariance P in the steady state.

2) Incomplete data with noiseless time-varying graphs

Next, we assume incomplete data, a scalar target state $x \in \mathbb{R}^1$ and noiseless time-varying graphs, where the connectivity graph Laplacian $L(j)$ with mean $\bar{L} = \mathbb{E}[L(j)]$ is such that $\lambda_2(\bar{L}) > 0$, and $p(l, n) > 0$ for $\{l, n\} \in E(j)$. Furthermore, we assume that only m nodes can observe the target and without loss of any generality the index of those m nodes are ordered as $1, 2, \dots, m$, where m is constant and $1 \leq m \leq N$. This implies that the active node set $S_k^0 = \{1, 2, \dots, m\}$, which does not require

further assumptions on the connectivity graph for consensus, since the graph is connected on average and the information can still propagate over the network even if only a fixed number of nodes have observation.

With the assumption of incomplete data and noiseless time-varying graphs, $\mathbf{1}^T \mathbf{L}(j) = 0$ for $J_k \leq j < J$. Then, (17) becomes

$$\lim_{J \rightarrow \infty} \bar{P}(k, J) = \frac{\mathbf{1}^T \bar{P}(k, J_k) \mathbf{1}}{N^2} \mathbf{1} \mathbf{1}^T = \frac{\mathbf{1}^T [\mathbf{A}_0^{J_k-1}] \bar{P}(k, 0) [\mathbf{A}_0^{J_k-1}]^T \mathbf{1}}{N^2} \mathbf{1} \mathbf{1}^T \quad (22)$$

$$= \frac{\sum_{n=1}^m \hat{P}_n(k|k) \beta_n^2(k)}{N^2} \mathbf{1} \mathbf{1}^T,$$

where $[\mathbf{A}_0^{J_k-1}] = \mathbf{A}(J_k - 1) \cdots \mathbf{A}(0)$ and $\beta_n(k) = \sum_{l=1}^N [\mathbf{A}_0^{J_k-1}]_{ln}$ is the n th column sum of $[\mathbf{A}_0^{J_k-1}]$ that depends on time k . The last step of (22) follows from that $\hat{P}_n(k|k) = \varepsilon$ for $m < n \leq N$ and some $\varepsilon > 0$. Then, as in previous subsection, we have for $1 \leq n \leq N$

$$\lim_{J \rightarrow \infty} \bar{P}_n(k, J) = \frac{\sum_{n=1}^m \hat{P}_n(k|k) \beta_n^2(k)}{N^2}, \quad \text{and} \quad (23)$$

$$\lim_{J \rightarrow \infty} \bar{x}_n(k, J) = \frac{1}{N} \sum_{n=1}^m \hat{x}_n(k|k) \beta_n(k).$$

From (23), for $J \rightarrow \infty$, we have $\bar{x}_n(k, J) = \bar{x}_l(k, J)$ and $\bar{P}_n(k, J) = \bar{P}_l(k, J)$ for $1 \leq n, l \leq N$. By setting $\bar{x}_n(k|k) = \bar{x}_n(k, J)$ and $\bar{P}_n(k|k) = \bar{P}_n(k, J)$, from (4), we can obtain recursive update equations for $\hat{P}_n(k+1|k)$ and $\hat{x}_n(k+1|k)$. Furthermore, we also have $\hat{x}_n(k+1|k) = \hat{x}_l(k+1|k)$ and $\hat{P}_n(k+1|k) = \hat{P}_l(k+1|k)$ for $1 \leq n, l \leq N$. Let $\hat{x}_n(k+1|k) = \hat{x}(k+1|k)$ and $\hat{P}_n(k+1|k) = \hat{P}(k+1|k)$. Then, the combined system of distributed tracking with consensus can then be transformed into a single Kalman filter for node n ($1 \leq n \leq m$) as below:

$$\hat{x}(k+1|k) = \frac{F}{N} \sum_{n=1}^m \hat{x}(k|k-1) \beta_n(k) + \frac{F}{N} \sum_{n=1}^m [K_n(k)(y_n(k) - H_n \hat{x}(k|k-1)) \beta_n(k)],$$

$$K_n(k) = \hat{P}(k|k-1) H_n^T [H_n \hat{P}(k|k-1) H_n^T + R_n]^{-1}, \quad (24)$$

$$\hat{P}(k+1|k) = Q + \frac{1}{N^2} \sum_{n=1}^m [F \hat{P}(k|k-1) F^T - F K_n(k) (H_n \hat{P}(k|k-1) H_n^T + R_n) \times K_n(k)^T F^T] \beta_n^2(k),$$

where (20) is a special case of (24) with $m = N$ and $\beta_n(k) = 1$ for $1 \leq n \leq N$.

Theorem 4 Consider the system dynamics in (1) and (2) and the Kalman filter in (24). Assume that m nodes can observe the target and the index of those m nodes are fixed and ordered as $1, \dots, m$. The connectivity graph Laplacian $L(j)$ with mean $\bar{L} = \mathbb{E}[L(j)]$ is such that $\lambda_2(\bar{L}) > 0$, and $p(l, n) > 0$ for $\{l, n\} \in E(j)$. The connectivity graph has switching topologies and is periodic such that $\beta_n(k) = \beta_n$ is time invariant. If the pair (F, H_n)

is observable for $1 \leq n \leq m$, then the prediction covariance matrix $\hat{P}(k|k-1)$ converges to a constant matrix

$$\lim_{k \rightarrow \infty} \hat{P}(k|k-1) = P,$$

where P is the unique definite solution of the discrete algebraic Riccati equation (DARE)

$$P = Q + \frac{1}{N^2} \sum_{n=1}^m [F P F^T - F P H_n^T (H_n P H_n^T + R_n)^{-1} H_n P F^T] \beta_n^2. \quad (25)$$

Proof See Appendix B.

Theorem 4 asserts that if the connectivity graph topology is switching and periodic, the proposed algorithm can still reach steady-state and the steady-state covariance matrix can be obtained by solving (25). The conditions of graph topology assumed in Theorem 4 are strong. However, it may still be applicable in certain situations such as satellite surveillance network in [29], since the existence of a communication link depends on distance between nodes and the trajectories of satellites are pre-determined and periodic, whenever ratios of the orbit periods are rational. As an example, consider the network model in Figure 4. The connectivity graph in Figure 4 is switching and periodic with period equal to 4, and it can be seen that the graph is connected on average. Let $N = 6, m = 4, S_k^0 = \{1, 2, 3, 4\}$ and $\gamma(j) = \frac{1}{j+1}$ for $0 \leq j < J$. After iteration $J_k = 1$, all nodes will have updated local estimates to be shared. In this case, $[\mathbf{A}_0^{J_k-1}]$ becomes

$$[\mathbf{A}_0^{J_k-1}] = \begin{bmatrix} -1 & 1 & 0 & 1 & 0 & 0 \\ 1 & -2 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & -1 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

It can be seen that $[\mathbf{A}_0^{J_k-1}]$ is time-variance of m and the periodic graph topology. Thus, $\beta_n(k) = \beta_n$ is also time invariant and $\frac{1}{N} \sum_{n=1}^m \beta_n = 1$, which follows from the condition for unbiasedness in the consensus estimate $\bar{\mathbf{X}}(k, j)$. As we will see shortly, from the simulation results in Section IV, the filter indeed reaches steady state in this case, and then, the error covariance matrix becomes time invariant and the corresponding filter gain is constant.

4 Numerical examples

In this section, we consider the performance of the proposed distributed tracking with consensus algorithm and compare it with centralized Kalman filter and distributed local Kalman filtering with centralized fusion. The

B

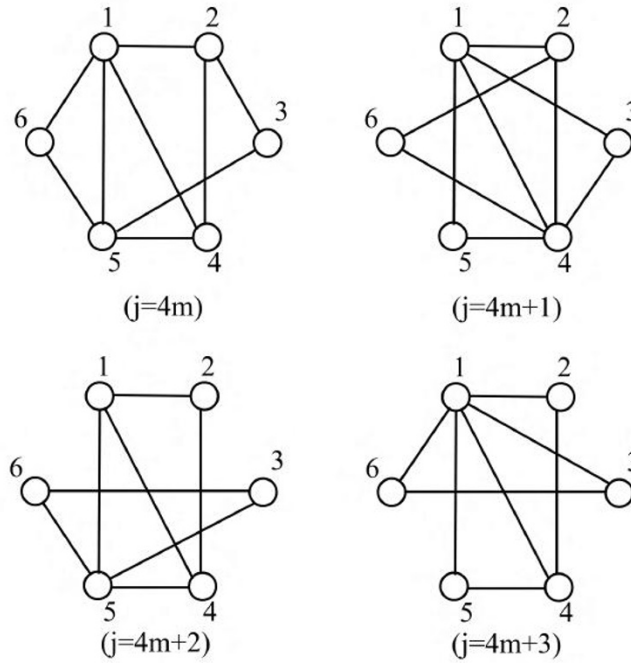


Figure 4 A time-varying graph with switching topologies.

performance of the centralized Kalman filter is well-understood [40] and provides a benchmark performance for distributed local Kalman filtering with centralized fusion. In distributed local Kalman filtering with centralized fusion, all nodes send their filtered estimates to a fusion center. The fusion center then generates a fused estimate $\hat{\mathbf{x}}_{\text{fusion}}(k) = \frac{1}{|S_k^0|} \sum_{n \in S_k^0} \hat{\mathbf{x}}_n(k|k)$.

In the first simulation, we compare the performance of the proposed algorithm with the distributed local Kalman filtering with centralized fusion and the centralized Kalman filter over a random graph with noisy communication links and incomplete data. We consider a random connectivity graph $G(N, p)$ with $N = 20$ and the probability that each link exists $p = 0.5$. The other parameters of the simulation setup are: $F = 1$, $Q = 1$, $\mathbf{x}(0) = 0$, $P_0 = 0$, $R_n = 0.25$, $H_n = 1$, $\sigma_{l,n}^2 = \sigma^2 = 0.1$, $S_k^0 = \{n | 1 \leq n \leq 10, n \in \mathbb{Z}\}$ and $J = 30$.

Figure 5a shows the node estimates of the three algorithms in a time-varying graph with noisy communication links. As we can see, the node estimates of the three algorithms follow the target's trajectory. In Figure 5a, the curve with cross marker denotes the first node's estimate by using distributed tracking with consensus algorithm, the dashed curve denotes the distributed local Kalman filtering with centralized fusion, and the curve with circle marker denotes the centralized Kalman

filter and the solid curve denotes the target's trajectory. Figure 5b compares the resulting mean squared error (MSE) of the three algorithms, where the MSE of the distributed tracking with consensus is defined to be the average MSE over all nodes $\frac{1}{N} \sum_{n=1}^N [(\bar{\mathbf{x}}_n(k, J) - \mathbf{x}(k))^T (\bar{\mathbf{x}}_n(k, J) - \mathbf{x}(k))]$. In Figure 5b, it can be seen that the MSE of the proposed distributed tracking with consensus algorithm is close to that of the distributed local Kalman filtering with centralized fusion. As expected, both of them are higher than the MSE of the centralized Kalman filter, which acts as a benchmark. The results in Figure 5 show that the performance of the proposed distributed tracking with consensus algorithm is close to that of the distributed local Kalman filtering with centralized fusion in a time-varying random graph with noisy communication and incomplete data. Additional communication bandwidth, which depends on graph topology G and number of iterations J , is required for the proposed algorithm due to information exchange among nodes. However, it resolves the bandwidth constraints problem of fusion center for centralized fusion case and has a high level of fault tolerance and reliability. Also, because of its advantages of fully distributed implementation, robustness, and scalability, it may be preferable in practical applications.

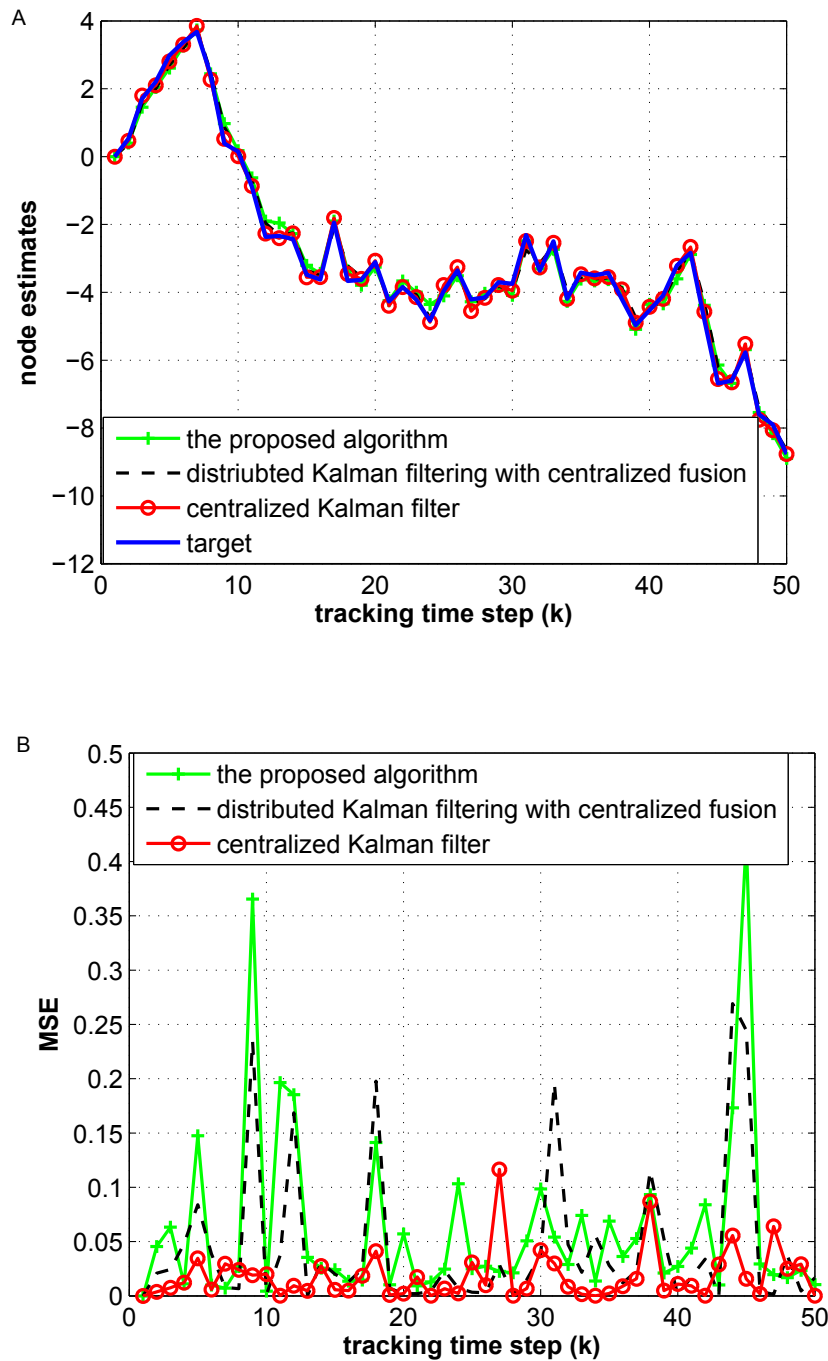


Figure 5 Comparison of the proposed distributed tracking with consensus algorithm with distributed local Kalman filtering with centralized fusion and centralized Kalman filter. (a) Node estimates; (b) mean squared error.

In the second simulation, we consider the two-dimensional tracking problem treated in [22]. The connectivity graph is again assumed to be a random graph $G(N, p)$ with $N = 50$ and the probability that each link exists $p = 0.5$. The probability of each node having an observation at a given time instant is $p_s = 0.9$. The other parameters of the simulation setup are as follows:

$F = I_2 + \varepsilon F_0 + \frac{\varepsilon^2}{2} F_0^2 + \frac{\varepsilon^3}{6} F_0^3$, $F_0 = \begin{bmatrix} 0 & -2 \\ 2 & 0 \end{bmatrix}$, $\varepsilon = 0.015$, $Q = (\varepsilon c_w^2)^2 I_2$, $c_w = 5$, $x(0) = [15, -10]^T$, $H_n = [1, 0]$ for n is odd and $H_n = [0, 1]$ for n is even, $R_n = c_v^2 \sqrt{n}$ for $n = 1, \dots, N$ with $c_v = 30$, $\sigma_{l,n}^2 = \sigma^2 = 1$, $J = 10$. Note that, the target is moving on noisy circular trajectories. The target is not fully observable by an individual node, but is collectively observable by all nodes.

Figure 6a shows the node estimates (trajectory) of the two algorithms over a time-varying graph with incomplete data. In Figure 6a, the curves with markers denote all the node estimates by using distributed tracking with consensus algorithm, while the dashed curve denotes the distributed local Kalman filtering with centralized fusion and the solid curve denotes the target's trajectory. As we can see, both algorithms overcome the impact of partial observations at each node resulting in improved overall observation quality and the node estimates by using distributed tracking with consensus algorithm are noisy due to the communication noise. Note that the estimates are close to the trajectory of the target but with a small gap. That is because the observation noise covariance is rather large at each node. Figure 6b compares the resulting MSE of these algorithms. It can be seen that the mean squared error of the proposed algorithm is slightly higher than that of the distributed Kalman filtering with centralized fusion.

Next, we study the steady-state behavior in the case of time-varying graphs with complete data and noiseless communication. We consider a random connectivity graph $G(N, p)$ with $N = 6$ and the probability that each link exists $p = 0.5$. The other parameters of the simulation setup are as follows: $F = 1, Q = 1, x(0) = 0P_0 = 0.5, R_n = 0.25, \sigma_{l,n}^2 = \sigma^2 = 0, J = 30, H_n = 1$. Figure 7a shows the node consensus estimates $\bar{x}_n(k, J)$ over a random graph with noiseless communication links and complete data. It can be seen that all node estimates $\bar{x}_n(k, J)$ converge to the same value and follow the target state, as asserted by Theorem 1. Figure 7b and 7c shows the node estimates $\bar{x}_n(k, J)$ in the consensus update after the twenty-first tracking update and the variance of all the node estimates, respectively. Here, the variance of all the node estimates is defined as $(k, j) = \mathbb{E}[(\bar{x}_n(k, j) - \mu(k, j))^T(\bar{x}_n(k, j) - \mu(k, j))]$, where $\mu(k, j) = \frac{1}{N} \sum_{n=1}^N \bar{x}_n(k, j)$. From Figure 7b, it can be seen that the node estimates converge to the average which is also confirmed in Figure 7c, where the variance $\text{var}(k, j)$ decreases as consensus iteration number increases and becomes static (around 10^{-17}) after consensus is reached. Figure 8 shows the node estimate variance $\hat{P}_n(k|k-1)$ and Kalman gain $K_n(k)$ of the filter in (20). It can be seen that as the Kalman filter reaches steady state, both the node estimate variance and the Kalman gain converge, as asserted by Theorem 3.

Next, we study the steady-state behavior on a graph with switching topologies and incomplete data and noiseless communication. The assumed parameters in the first simulation setup are as follows:

$F = 1, Q = 1, x(0) = 0, P_0 = 0.5, R_n = 0.25, \sigma_{l,n}^2 = \sigma^2 = 0, N = 6, J = 40, S_k^0 = \{1, 3, 4, 6\}, H_n = 0.5$

for $n = 1, 3$ and $H_n = 1$ for $n = 4, 6$. The connectivity graph Laplacian is

$$L(j) = \begin{cases} L_1 j = 4m \\ L_2 j = 4m + 1 \\ L_3 j = 4m + 2 \\ L_4 j = 4m + 3 \end{cases} \quad \text{for } m = 0, 1, 2, \dots,$$

which is shown in Figure 4. As we can see, the graph is connected on average and $p(l, n) > 0$ for $\{l, n\} \in E(j)$, satisfying the conditions on the connectivity graph Laplacian required in Theorem 1.

Figure 9 shows the prediction covariance matrix $\hat{P}_n(k|k-1)$ and Kalman gain $K_n(k)$ of the filter in (20), respectively. It can be seen that as the Kalman filter reaches the steady state, both the prediction covariance matrix and the Kalman gain converge, as asserted by Theorem 4. Note that the limit of the Kalman gain is different for different nodes in Figure 9 because the observation matrix H_n is different for different nodes.

5 Conclusions

In this paper, we considered the problem of distributed tracking with consensus on a time-varying graph with incomplete data and noisy communication links. We developed a framework consisting of tracking and consensus updates to handle the issues of time-varying network topology and incomplete data. We discussed the conditions for achieving consensus, quantified the convergence rate and analyzed the steady-state performance when applicable. Our simulation results showed that the proposed distributed tracking with consensus algorithm improves the estimation quality at each node and its performance is close to that of the distributed local Kalman filtering with centralized fusion. The proposed algorithm shows advantages of fully distributed implementation, robustness and scalability, which is preferable in practical application.

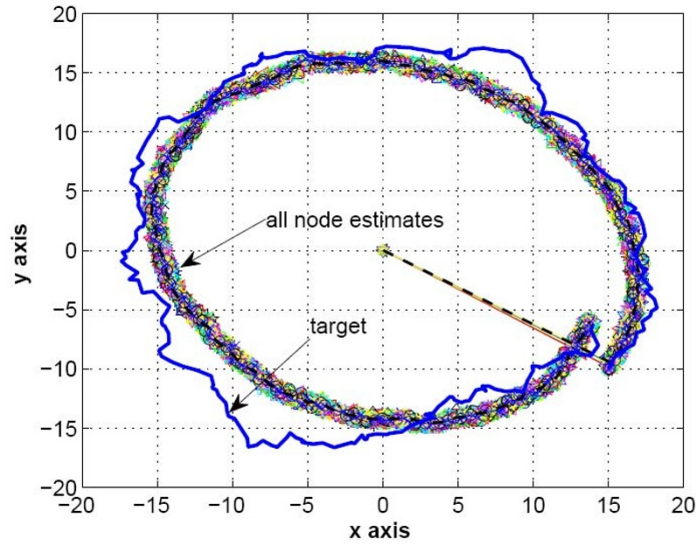
Appendix A

Proof of Lemma 3

Proof Since $\lambda_2(\bar{L}) > 0$ and $p(l, n) > 0$ for $\{l, n\} \in E(j)$, the undirected time-varying connectivity graph $G(j)$ is connected on average with non-zero link probability. For j large enough, each node will receive the information from one another and generate its updated local estimates. For a fixed k , let $J_k = \inf\{j | (S_k^j)^c = \emptyset, j \geq 0\}$. Then, we have the effective network graph is the same as connectivity graph $\tilde{G}(j) = G(j), L(j) = L(j)$ and $\Gamma(j) = \gamma(j) I_N$ for $j \geq J_k$.

Since $\mathbb{P}[\bar{X}(j)|\bar{X}(j-1), \dots, \bar{X}(0)] = \mathbb{P}[\bar{X}(j)|\bar{X}(j-1)]$, the process $\{\bar{X}(j)\}_{j \geq 0}$ is Markov. Define

A



B

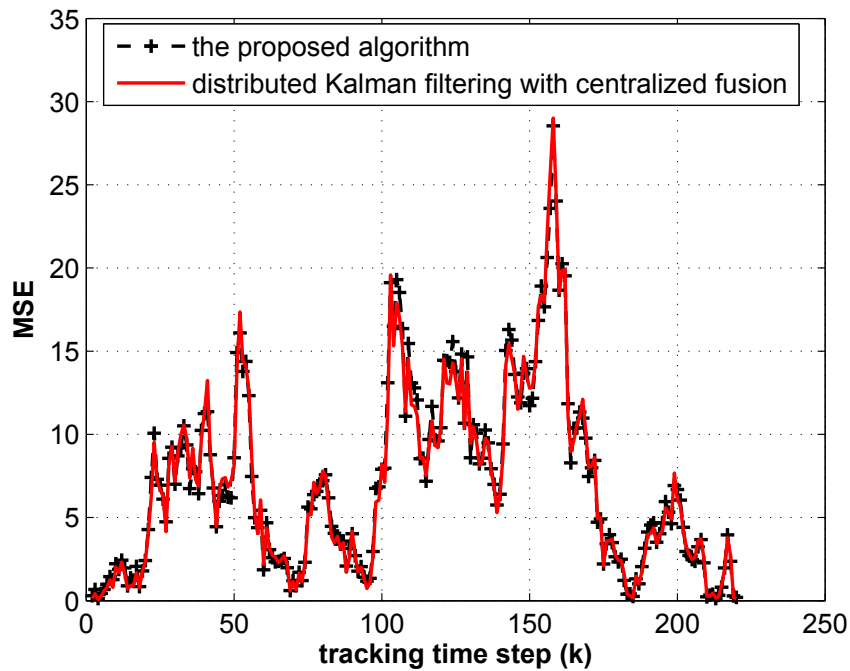


Figure 6 Comparison of the proposed distributed tracking with consensus algorithm and distributed Kalman filter with centralized fusion in a two-dimensional tracking problem. (a) Trajectory; (b) mean squared error.

$V(j, \bar{\mathbf{X}}) = \bar{\mathbf{X}}^T (\bar{\mathbf{L}} \otimes \mathbf{I}_M) \bar{\mathbf{X}}$. Since we assume the graph is undirected and connected on average, $\bar{\mathbf{L}}$ is positive semi-definite. Then, the potential function $V(j, \bar{\mathbf{X}})$ is non-negative. Since $\bar{\mathbf{X}}_{\varepsilon \mathcal{C}}$ is an eigenvector of $\bar{\mathbf{L}} \otimes \mathbf{I}_M$ with zero eigenvalue, $V(j, \bar{\mathbf{X}}) = 0, \bar{\mathbf{X}} \in \mathcal{C}, \lim_{\bar{\mathbf{X}} \rightarrow \mathcal{C}} \sup_{j \geq 0} V(j, \bar{\mathbf{X}}) = 0$.

From Courant-Fisher Theorem [31,41], for $\mathbf{Z} \in \mathbb{R}^{N \times M}$ and $\mathbf{Z} \perp \mathcal{C}$, we have

$$\mathbf{Z}^T (\bar{\mathbf{L}} \otimes \mathbf{I}_M) \mathbf{Z} \geq \lambda_2 (\bar{\mathbf{L}} \otimes \mathbf{I}_M) \mathbf{Z}^T \mathbf{Z}. \quad (26)$$

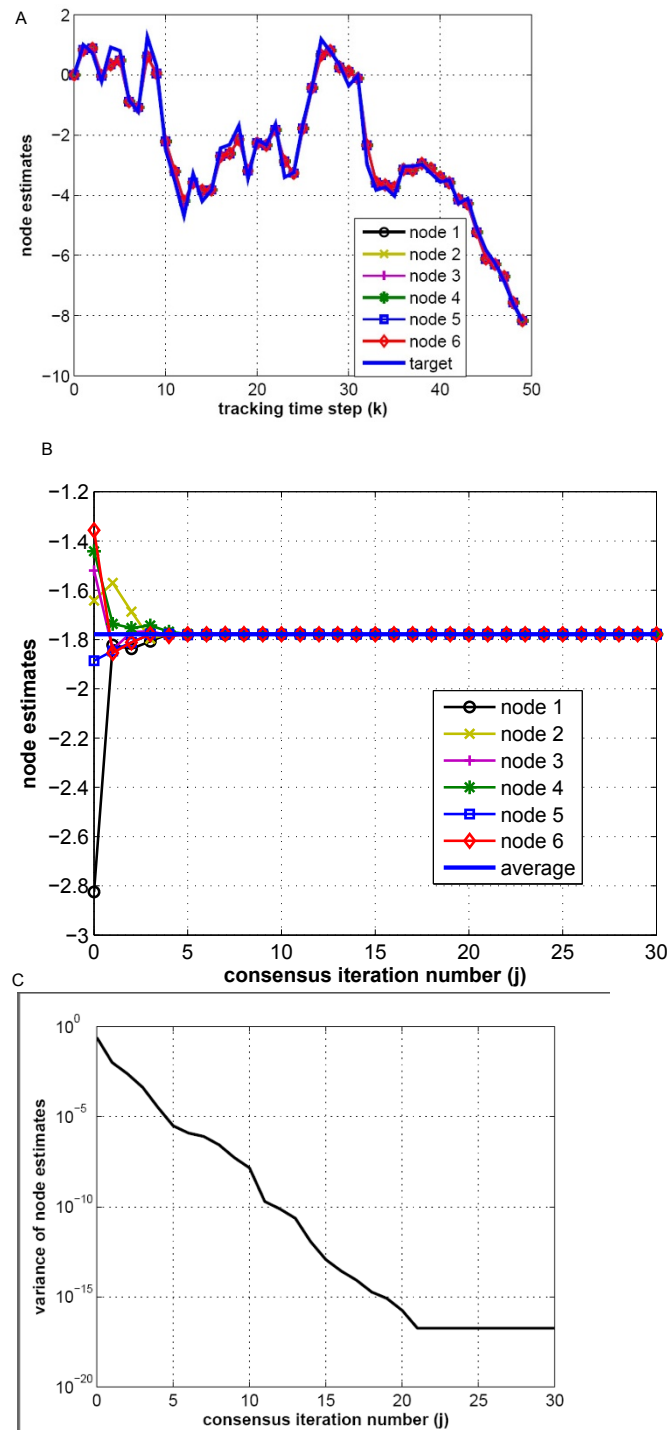


Figure 7 Performance of the distributed tracking with consensus algorithm for complete data and noiseless communication case. (a) Node consensus estimates $\bar{\mathbf{x}}_n(k, j)$ versus tracking time step; (b) node estimates $\bar{\mathbf{x}}_n(k, j)$ versus consensus iteration number; (c) variance of node estimates $\text{var}(k, j)$ versus consensus iteration number.

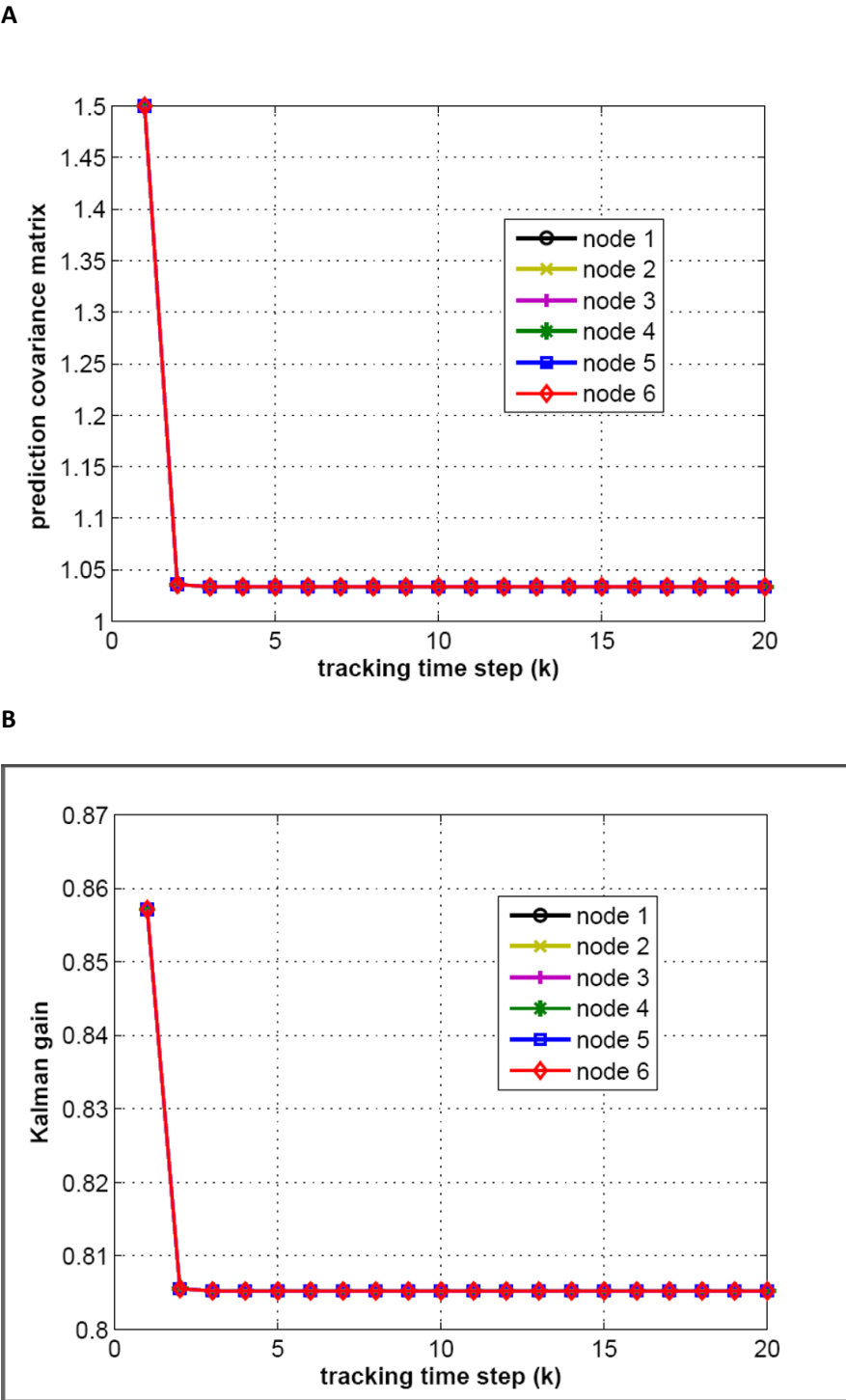


Figure 8 Steady-state performance of the distributed tracking with consensus algorithm for complete data and noiseless communication case. (a) Prediction covariance matrix $\hat{P}_n(k|k-1)$ Kalman gain $K_n(k)$.

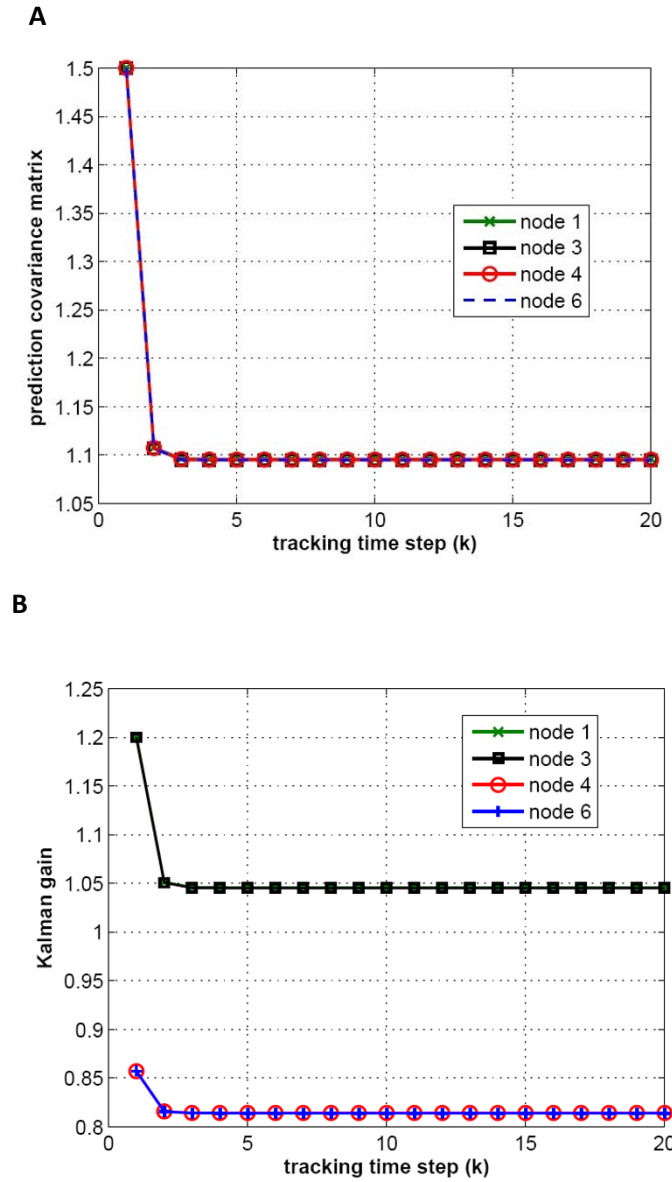


Figure 9 Steady-state performance of the distributed tracking with consensus algorithm for incomplete data and noiseless communication case. (a) Prediction covariance matrix $\hat{P}_n(k|k-1)$; (b) Kalman gain $K_n(k)$.

From Lemma 1 and the complement of the ϵ -neighborhood of a set in (10), we have $\bar{\mathbf{X}} \in V_\epsilon(\mathcal{C}) \Rightarrow \|\bar{\mathbf{X}}_{C^\perp}\| \geq \epsilon$. Then, for $\bar{\mathbf{X}} \in V_\epsilon(\mathcal{C})$, from (26) and the properties of Kronecker product and eigenvalues, we will have

$$\begin{aligned} V(j, \bar{\mathbf{X}}) &= \bar{\mathbf{X}}^T (\bar{\mathbf{L}} \otimes \mathbf{I}_M) \bar{\mathbf{X}} = \bar{\mathbf{X}}_{C^\perp}^T (\bar{\mathbf{L}} \otimes \mathbf{I}_M) \bar{\mathbf{X}}_{C^\perp} + \bar{\mathbf{X}}_C^T (\bar{\mathbf{L}} \otimes \mathbf{I}_M) \bar{\mathbf{X}}_C, \\ &\geq \lambda_2(\bar{\mathbf{L}} \otimes \mathbf{I}_M) \|\bar{\mathbf{X}}_{C^\perp}\|^2 = \lambda_2(\bar{\mathbf{L}}) \|\bar{\mathbf{X}}_{C^\perp}\|^2 \geq \lambda_2(\bar{\mathbf{L}}) \epsilon^2. \end{aligned} \quad (27)$$

Since $\lambda_2(\bar{\mathbf{L}}) > 0$, we get $\inf_{j \geq 0, \bar{\mathbf{X}} \in V_\epsilon(\mathcal{C})} V(j, \bar{\mathbf{X}}) \geq \lambda_2(\bar{\mathbf{L}}) \epsilon^2 > 0$. Consider the generating operator \mathcal{L} and (11). Using the fact that $L(j) = L(j)$ for $j \geq J_k$ we obtain

$$\begin{aligned} \mathcal{L}V(j, \bar{\mathbf{X}}) &= E[\bar{\mathbf{X}}(j+1)^T (\bar{\mathbf{L}} \otimes \mathbf{I}_M) \bar{\mathbf{X}}(j+1) | \bar{\mathbf{X}}(j) = \bar{\mathbf{X}}] - \bar{\mathbf{X}}^T (\bar{\mathbf{L}} \otimes \mathbf{I}_M) \bar{\mathbf{X}}, \\ &= E \left[[\bar{\mathbf{X}} - \gamma(j)(L(j) \otimes \mathbf{I}_M) \bar{\mathbf{X}} - \gamma(j)\Phi(j)]^T (\bar{\mathbf{L}} \otimes \mathbf{I}_M) \right. \\ &\quad \times [\bar{\mathbf{X}} - \gamma(j)(L(j) \otimes \mathbf{I}_M) \bar{\mathbf{X}} - \gamma(j)\Phi(j)] \left. \right] - \bar{\mathbf{X}}^T (\bar{\mathbf{L}} \otimes \mathbf{I}_M) \bar{\mathbf{X}} \quad \text{for } j \geq J_k. \end{aligned}$$

From (6), we have $E[\|\bar{\Phi}(j)\|^2] \leq \eta$. By using the independence of $L(j)$ and $\bar{\Phi}(j)$ with respect to $\bar{X}(j)$ and $\bar{X}^T \bar{L} \bar{X} \leq \lambda_N(\bar{L}) \|\bar{X}_{C^\perp}\|^2$ [32], after some work, we have that

$$\begin{aligned} LV(j, \bar{X}) &= -2\gamma(j)\bar{X}^T(\bar{L} \otimes I_M)^2\bar{X} + \gamma^2(j)\bar{X}^T(\bar{L} \otimes I_M)^3\bar{X} \\ &+ E\left[\gamma^2(j)\left[(\bar{L}(j) \otimes I_M)\bar{X}\right]^T(\bar{L} \otimes I_M)(\bar{L}(j) \otimes I_M)\bar{X}\right] \\ &+ E\left[\gamma^2(j)\bar{\Phi}(j)^T(\bar{L} \otimes I_M)\bar{\Phi}(j)\right], \\ &\leq -2\gamma(j)\bar{X}^T(\bar{L} \otimes I_M)^2\bar{X} + \gamma^2(j)\left[\lambda_N^3(\bar{L})\|\bar{X}_{C^\perp}\|^2 + \lambda_N(\bar{L})E\left[\|(\bar{L}(j) \otimes I_M)\bar{X}\|^2\right]\right. \\ &\quad \left.+ \lambda_N(\bar{L})E\left[\|\bar{\Phi}(j)\|^2\right]\right], \\ &\leq -2\gamma(j)\bar{X}^T(\bar{L} \otimes I_M)^2\bar{X} + \gamma^2(j)\left[\lambda_N^3(\bar{L})\|\bar{X}_{C^\perp}\|^2 + 4N^2\lambda_N(\bar{L})\|\bar{X}_{C^\perp}\|^2 + \lambda_N(\bar{L})\eta\right]. \end{aligned}$$

The last step follows from the fact that all the eigenvalues of $\bar{L}(j)$ are less than $2N$ in absolute value, by the Gershgorin circle theorem. Using the fact that $\bar{X}^T(\bar{L} \otimes I_M)\bar{X} \geq \lambda_2(\bar{L})\|\bar{X}_{C^\perp}\|^2$ from (27), we have

$$\begin{aligned} LV(j, \bar{X}) &\leq -2\gamma(j)\bar{X}^T(\bar{L} \otimes I_M)^2\bar{X} + \gamma^2(j)\left[\lambda_N(\bar{L})\eta + \left(\frac{\lambda_N^3(\bar{L})}{\lambda_2(\bar{L})} + \frac{4N^2\lambda_N(\bar{L})}{\lambda_2(\bar{L})}\right)\right. \\ &\quad \left.\times \bar{X}^T(\bar{L} \otimes I_M)\bar{X}\right], \\ &\leq -2\gamma(j)\varphi(j, \bar{X}) + g(j)[1 + \varphi(j, \bar{X})] \quad \text{for } j \geq j_k, \end{aligned}$$

where $\varphi(j, \bar{X}) = 2\bar{X}^T(\bar{L} \otimes I_M)^2\bar{X}$, $g(j) = \gamma^2(j)\max\left(\lambda_N(\bar{L})\eta, \frac{\lambda_N^3(\bar{L})}{\lambda_2(\bar{L})} + \frac{4N^2\lambda_N(\bar{L})}{\lambda_2(\bar{L})}\right)$.

Then, the theorem follows by using Lemma 2.

Appendix B

Proof of Theorem 4

Proof Step 1: Bound on the error covariance

From (1), it can be easily shown that the controllability matrix has full rank and the system is controllable. Since (F, H_n) is detectable, $\exists K'_n$ such that $(F - K'_n H_n)$ are stable. Consider the suboptimal filter

$$\hat{x}_n(k+1|k) = F\hat{x}_n(k|k-1) + K'_n(y_n(k) - H_n\hat{x}_n(k|k-1)),$$

Since consensus is reached in consensus update part, $\hat{x}_n(k|k-1) = \hat{x}_l(k|k-1) = \hat{x}(k|k-1)$ for $1 \leq n, l \leq N$. Then,

$$\hat{x}(k+1|k) = \frac{1}{N}\left(F\sum_{n=1}^m\beta_n - \sum_{n=1}^m K'_n H_n \beta_n\right)\hat{x}(k|k-1) + \frac{1}{N}\sum_{n=1}^m K'_n y_n(k)\beta_n.$$

It is easily verified that

$$\begin{aligned} \tilde{x}(k+1|k) &= x(k+1) - \hat{x}(k+1|k), \\ &= \left(F - \frac{1}{N}\sum_{n=1}^m K'_n H_n \beta_n\right)\tilde{x}(k|k-1) - \frac{1}{N}\sum_{n=1}^m K'_n v_n(k)\beta_n + w(k), \end{aligned}$$

where the last step follows from the fact that the estimate is unbiased and $\frac{1}{N}\sum_{n=1}^m \beta_n = 1$. Since $(F - K'_n H_n)$ is stable, $\left(F - \frac{1}{N}\sum_{n=1}^m K'_n H_n \beta_n\right)$ is also stable. It follows that the covariance matrix $\Pi(k) = \text{Cov}[\tilde{x}(k|k-1)]$ is bounded, where $\text{Cov}(x)$ denotes the covariance matrix

of x . However, the filter above is suboptimal, so that $P(k|k-1) \leq \Pi(k)$.

Step 2: Monotonicity of the error covariance

Recall that the mapping $f: \hat{P}_n(k|k-1) \rightarrow \hat{P}_n(k+1|k)$ as $\hat{P}_n(k+1|k) = \min_{K_n} g(\hat{P}_n(k|k-1), K_n)$, where

$$g(\hat{P}_n, K_n) = (F - K_n H_n)\hat{P}_n(F - K_n H_n)^T + K_n R_n K_n^T + Q.$$

Thus, if $\hat{P}_n(k|k-1) \geq \hat{P}'_n(k|k-1)$

$$\begin{aligned} \hat{P}_n(k+1|k) &= \min_{K_n} g(\hat{P}_n(k|k-1), K_n) = g(\hat{P}_n(k|k-1), K_n^*) \geq g(\hat{P}'_n(k|k-1), K_n^*) \\ &\geq \min_{K_n} g(\hat{P}'_n(k|k-1), K_n) = \hat{P}'_n(k+1|k). \end{aligned}$$

Therefore, the mapping f from $\hat{P}_n(k|k-1)$ to $\hat{P}_n(k+1|k)$ is monotonic. Because

$\hat{P}(k+1|k) = \frac{1}{N^2} \sum_{n=1}^m \hat{P}_n(k+1|k)\beta_n^2$, the mapping $\hat{f}: \hat{P}(k|k-1) \rightarrow \hat{P}(k+1|k)$ is also monotonic.

Step 3: Use of zero initial covariance

Suppose $\hat{P}(0|-1) = 0$. Then $\hat{P}(1|0) \geq \hat{P}(0|-1) = 0$. But from Step 2 it follows that $\hat{P}(k+1|k) \geq \hat{P}(k|k-1)$, for $k \geq 0$. Since $\{\hat{P}(k|k-1)\}$ is bounded by Step 1, then $\hat{P}(k|k-1) \rightarrow P$ for some $P \geq 0$. Obviously, P must be a stationary point of the covariance update equation, hence solves the DARE.

Step 4: Asymptotic stability of the filter

With \bar{K}_n the stationary gain corresponding to P , the DARE is

$$\begin{aligned} P &= \left(F - \frac{1}{N}\sum_{n=1}^m \bar{K}_n H_n \beta_n\right)P\left(F - \frac{1}{N}\sum_{n=1}^m \bar{K}_n H_n \beta_n\right)^T \\ &\quad + \frac{1}{N^2}\sum_{n=1}^m \bar{K}_n R_n \bar{K}_n^T \beta_n^2 + GG^T, \end{aligned}$$

where $GG^T = Q$. Let v be a left eigenvector of $\left(F - \frac{1}{N}\sum_{n=1}^m \bar{K}_n H_n \beta_n\right)$ with eigenvalue λ . Then

$$(vPv^T) = |\lambda|^2(vPv^T) + \frac{1}{N^2}\sum_{n=1}^m v\bar{K}_n R_n \bar{K}_n^T v^T \beta_n^2 + vGG^T v^T. \quad (28)$$

Since R_n and Q are positive semidefinite, it implies that $|\lambda| \leq 1$. It only remains to show that $|\lambda| = 1$ is impossible. If $|\lambda| = 1$, we have from (28) and the definition of v :

$$v\left(F - \frac{1}{N}\sum_{n=1}^m \bar{K}_n H_n \beta_n\right) = \lambda v, \quad v\bar{K}_n = 0, \quad \text{and} \quad vG = 0,$$

which gives that $v[\lambda I - F, G] = 0$. This contradicts the assumption that (F, G) is stabilizable.

Step 5: Nonzero initial covariances

Suppose we use the stationary suboptimal filter $K'_n \equiv \bar{K}_n$ to obtain the estimate $\hat{x}(k|k-1)$. We show that its error covariance converges to P .

Defining $\tilde{x}(k|k-1) \triangleq x(k) - \hat{x}(k|k-1)$, we obtain

$$\tilde{x}(k|k-1) = \left(F - \frac{1}{N} \sum_{n=1}^m \bar{K}_n H_n \beta_n \right) \tilde{x}(k|k-1) - \frac{1}{N} \sum_{n=1}^m \bar{K}_n v_n(k) \beta_n + w(k).$$

Since $\left(F - \frac{1}{N} \sum_{n=1}^m \bar{K}_n H_n \beta_n \right)$ is stable with eigenvalue $|\lambda| < 1$, it follows from above results on stationary behavior that $\Pi(k) \equiv \text{Cov}[\tilde{x}(k|k-1)] \rightarrow \tilde{P} \geq 0$, where \tilde{P} is the unique non-negative solution of the Lyapunov equation:

$$\tilde{P} = \left(F - \frac{1}{N} \sum_{n=1}^m \bar{K}_n H_n \beta_n \right) \tilde{P} \left(F - \frac{1}{N} \sum_{n=1}^m \bar{K}_n H_n \beta_n \right)^T + \frac{1}{N^2} \sum_{n=1}^m \bar{K}_n R_n \bar{K}_n^T \beta_n^2 + Q.$$

However, substituting \bar{K}_n this is just the DARE which is satisfied by P , hence $\tilde{P} = P$. Now, $\hat{x}(k|k-1)$ is suboptimal so that $P(k|k-1) \leq \Pi(k) \rightarrow \tilde{P}$. On the other hand, by monotonicity of mapping $\hat{f}: \hat{P}(k|k-1) \rightarrow \hat{P}(k+1|k)$, it follows that $P(k|k-1) \geq P^0(k|k-1) \rightarrow P$, where $P^0(k|k-1)$ is the covariance for $P(0|k-1) = 0$. Hence, $P(k|k-1) \rightarrow P$.

Endnotes

^aThe assumption here is that at the beginning of the consensus update process, the filtered estimates at different nodes are statistically uncorrelated. ^bNote that, for $n \in (S_k^j)^c$ and $\sum_{l=1}^N \mathbf{A}_{n,l}(j) = 0$, node n does not receive information from any node that has local tracking estimate. Then, $\bar{\mathbf{x}}_n(k, j+1) = \bar{\mathbf{x}}_n(k, j)$. ^cNote that, similar results on the unbiasedness of consensus estimate was obtained in [29]. ^dFor practical consideration, due to energy constraints of sensor networks, the time period J for consensus process is not too long such that the nodes can still efficiently obtain new information from the source [38]. Simulation results in Section IV show how the algorithm performs in this case. ^eNote that the convergence rate calculated here is for the period of $J_k \leq j \leq J$, where $J \gg 1$ is the number of consensus iterations. From persistence condition (9), $\lim_{j \rightarrow \infty} \gamma(j) = 0$. Then $\gamma(j)$ is very close to zero and the convergence speed can be assumed negligible for $j \geq J$ and J large enough [26,42].

Acknowledgements

This research was supported in part by the Space Vehicles Directorate of the Air Force Research Laboratory (AFRL) and in part by the National Science foundation (NSF) under the grant CCF-0830545.

Author details

¹Communications and Information Sciences Lab (CISL), Department of Electrical and Computer Engineering, University of New Mexico, Albuquerque, NM, USA ²Air Force Research Laboratory (AFRL), Space Vehicles Directorate, Kirtland AFB, NM 87117, USA

Competing interests

The authors declare that they have no competing interests.

Received: 4 January 2011 Accepted: 23 November 2011

Published: 23 November 2011

References

1. BDO Anderson, JB Moore, *Optimal Filtering* (Prentice-Hall, Englewood Cliffs, 1979)
2. DB Reid, An algorithm for tracking multiple targets. *IEEE Trans Autom Control* **24**(6), 843–854 (1979). doi:10.1109/TAC.1979.1102177
3. Y Bar-Shalom, TE Fortmann, *Tracking and Data Association* (Academic Press, Boston, 1988)
4. Y Bar-Shalom, XR Li, *Multitarget-Multisensor Tracking: Principles and Techniques* (YBS Publishing, Storrs, 1995)
5. M Fogel, N Burkhart, H Ren, J Schiff, M Meng, K Goldberg, Automated tracking of pallets in warehouses: Beacon layout and asymmetric ultrasound observation models, in *Third IEEE Conference on Automation Science and Engineering (CASE)*, Scottsdale, AZ, (September, 2007)
6. J Wilson, V Bhargava, A Redfern, P Wright, A wireless sensor network and incident command system for urban firefighting, in *Proceedings of the 4th Annual International Conference on Mobile and Ubiquitous Systems: Networking & Services (MobiQuitous)*, Philadelphia, PA, (2007)
7. A Jadbabaie, J Lin, AS Morse, Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans Autom Control* **48**(6), 988–1001 (2003). doi:10.1109/TAC.2003.812781
8. W Ren, RW Beard, E Atkins, A survey of consensus problems in multi-agent coordination, in *Proceedings of American Control Conference*, Portland, OR, 1859–1864 (June, 2005)
9. R Olfati-Saber, RM Murray, Consensus problems in networks of agents with switching topology and time-delays. *IEEE Trans Autom Control* **49**(9), 1520–1533 (2004). doi:10.1109/TAC.2004.834113
10. DB Kingston, RW Beard, Discrete-time average-consensus under switching network topologies, in *Proceedings of American Control Conference*, Minneapolis, MN, 3551–3556 (June 2006)
11. L Xiao, S Boyd, Fast linear iterations for distributed averaging. *Syst Control Lett.* **53**, 65–78 (2004). doi:10.1016/j.sysconle.2004.02.022
12. Y Hatano, M Mesbahi, Agreement over random networks. *IEEE Trans Autom Control* **50**(11), 1867–1872 (2005)
13. A Kashyap, T Basar, R Srikant, Quantized consensus. *Automatica* **43**(7), 1192–1203 (2007). doi:10.1016/j.automatica.2007.01.002
14. F Fagnani, S Zampieri, Average consensus with packet drop communication. *SIAM J Control Optim.* **48**(1), 102–133 (2009). doi:10.1137/060676866
15. S Kar, JMF Moura, Sensor networks with random links: topology design for distributed consensus. *IEEE Trans Signal Process.* **56**(7), 3315–3326 (2008)
16. S Kar, JMF Moura, Distributed consensus algorithms in sensor networks: link failures and channel noise. *IEEE Trans Signal Process.* **57**(1), 355–369 (2009)
17. M Huang, JH Manton, Coordination and consensus of networked agents with noisy measurement: stochastic algorithms and asymptotic behavior. *SIAM J Control Optim.* **48**(1), 134–161 (2009). doi:10.1137/06067359X
18. T Li, Asymptotically unbiased average consensus under measurement noises and fixed topologies, in *Proceedings of the 17th IFAC World Congress*, Seoul, Korea, 2867–2873 (July, 2008)
19. T Li, JF Zhang, Mean square average consensus under measurement noises and fixed topologies: necessary and sufficient conditions. *Automatica* **45**(8), 1929–1936 (2009)
20. T Li, JF Zhang, Consensus conditions of multi-agent systems with time-varying topologies and stochastic communication noises. *IEEE Trans Autom Control* **55**(9), 2043–2057 (2010)
21. R Olfati-Saber, Distributed Kalman filter with embedded consensus filters, in *Proceedings of the 44th IEEE Conference on Decision and Control*, Seville, Spain, 8179–8184 (December, 2005)

22. R Olfati-Saber, Distributed Kalman filtering for sensor networks, in *Proceedings of the 46th IEEE Conference on Decision and Control*, New Orleans, LA, 5492–5498 (December, 2007)
23. P Altkisson, A Rantzer, Distributed Kalman filtering using weighted averaging, in *Proceedings of the 17th Symposium on Mathematical Theory of Networks and Systems*, Kyoto, Japan, (July 2006)
24. U Khan, JMF Moura, Distributing the Kalman filter for large-scale systems. *IEEE Trans Signal Process.* **56**(10), 4919–4935 (2008)
25. Y Hong, J Hu, L Gao, Tracking control for multi-agent consensus with an active leader and variable topology. *Automatica* **42**(7), 1177–1182 (2006). doi:10.1016/j.automatica.2006.02.013
26. C Mosquera, R Lopez-Valcarce, SK Jayaweera, Stepsize sequence design for distributed average consensus. *IEEE Signal Process Lett.* **17**(2), 169–172 (2010)
27. SK Jayaweera, Y Ruan, RS Erwin, Distributed Tracking with Consensus on Noisy Time-varying Graphs with Incomplete Data, in *10th International Conference on Signal Processing (ICSP 10)*, Beijing, China, (October, 2010)
28. Y Ruan, SK Jayaweera, C Mosquera, Performance Analysis of Distributed Tracking with Consensus on Noisy Time-varying Graphs, in *5th Advanced Satellite Multimedia System Conference and the 11th Signal Processing for Space Communications Workshop*, Sardinia, Italy, (September, 2010)
29. SK Jayaweera, *Distributed space-object tracking and scheduling with a satellite-assisted collaborative space surveillance network (SSN)*. Final project report: AFRL Summer Faculty Fellowship Program, (July, 2009)
30. Y Cao, W Ren, Y Li, Distributed discrete-time coordinated tracking with a time-varying reference state and limited communication. *Automatica* **45**(5), 1299–1305 (2009)
31. FRK Chung, *Spectral Graph Theory* (American Mathematical Society, Providence, 1997)
32. AJ Laub, *Matrix Analysis for Scientists and Engineers* (Society for Industrial and Applied Mathematics, SIAM, 2004)
33. M Tinati, T Rezaei, Multi-target tracking in wireless sensor networks using distributed joint probabilistic data association and average consensus filter, in *Proceedings of the 2009 International Conference on Advanced Computer Control*, Singapore, (January, 2009)
34. O Boric-Lubecke, J Lin, B Park, C Li, W Massagram, V Lubecke, A Host-Madsen, Battlefield triage life signs detection techniques, in *Proceedings of the SPIE Defense and Security Symposium*, vol. 6947. Orlando, FL, 69470J.1–69470J.10 (April, 2008)
35. C Soto, B Song, A Chowdhury, Distributed multi-target tracking in a self-configuring camera network, in *Proceedings of IEEE Computer Vision and Pattern Recognition*, Miami, FL, 1486–1493 (June, 2009)
36. M Nevel'son, R Has'minskii, *Stochastic Approximation and Recursive Estimation* (American Mathematical Society, Providence, 1973)
37. N Ghasemi, S Dey, J Baras, *Stochastic Average Consensus Filter for Distributed HMM Filtering: Almost Sure convergence* (Institute for Systems Research Technical Reports, May 2010)
38. S Barbarossa, G Scutari, A Swami, Achieving Consensus in Self-Organizing Wireless Sensor Networks: the Impact of Network Topology on Energy Consumption, in *Proceedings of IEEE ICASSP*, Honolulu, HI, USA, (April, 2007)
39. S Sardellitti, S Barbarossa, A Swami, Average consensus with minimal energy consumption: optimal topology and power consumption, in *18th European Signal Processing Conference*, Aalborg, Denmark, (August, 2010)
40. BSY Rao, H Durrant-Whyte, JA Sheen, A fully decentralized multi-sensor system for tracking and surveillance. *Int J Robot Res.* **12**(1), 20–44 (1993). doi:10.1177/027836499301200102
41. B Mohar, Y Alavi, G Chartrand, OR Oellermann, AJ Schwenk, The Laplacian spectrum of graphs. *Graph Theory Comb Appl.* **2**, 871–898 (1991)
42. H Kushner, G Yin, *Stochastic Approximation and Recursive Algorithms and Applications*, 2nd edn. (Springer, New York, 2003)

doi:10.1186/1687-6180-2011-110

Cite this article as: Ruan et al.: Distributed tracking with consensus on noisy time-varying graphs with incomplete data. *EURASIP Journal on Advances in Signal Processing* 2011 **2011**:110.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com