This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2018.2867341, IEEE Transactions on Automatic Control

Sensing-Based Distributed State Estimation for Cooperative Multi-Agent Systems

Cheolhyeon Kwon, *Member, IEEE,* and Inseok Hwang, *Member, IEEE*

***Abstract*—Distributed estimation has proven to be suitable for many Multi-Agent System (MAS) applications, yet it relies heavily on information exchange via a costly and vulnerable communication network. This paper proposes a sensing-based distributed estimation algorithm which enables a local monitoring agent to expand its estimation capabilities beyond its sensing range without needing communication overhead. The key to expanding the limited sensing range is to incorporate the MAS's cooperative control protocol, allowing the monitoring agent to infer the state of out-of-range agents from the behavior of in-range agents that may interact with them. Then, the state estimation for out-of-range agents is performed through a Bayesian approach which considers the correlation of state estimates between in-range and out-of-range agents. This approach of only taking sensor measurements of local monitoring agents without inter-agent communications can successfully compensate for the existing communication-based distributed estimation methods. The performance of the proposed sensing-based distributed estimation algorithm is theoretically verified and demonstrated with numerical simulations of a multi-vehicle formation flight example.**

I. Introduction

FOLLOWING recent advances in networked communication and control technologies, Multi-Agent System (MAS) models have been employed in numerous network applications to accomplish more elaborate tasks at a lower cost than a single complex system [1]–[3]. A MAS is comprised of multiple intelligent agents interacting within various environments where the individual agents are capable of operating independently, but collaborate with each other in a distributed manner [4], [5]. However, the lack of a central supervisor and the limitations in inter-agent network capabilities make it challenging for the distributed schemes to coordinate MASs and achieve their cooperative goals [6], [7].

Most of the MAS research that addresses this challenge can be categorized into control oriented and estimation oriented works. On the control side, different distributed control techniques for the MAS have been developed such as semi-decentralized consensus algorithms using partial information [8] and local control strategies with a limited sensor view [9]. In addition, theoretical frameworks for flocking and vehicle formation control in the presence of limited communication have been developed in [10] and [11], respectively. On the other hand, various distributed estimation techniques have been developed with different applications, including but not limited to wireless sensor networks [12], electric power grids [13], mobile robots [14], etc. The Bayesian approach is suitable for the most distributed estimation problems that appropriately integrate the 'prior' knowledge (e.g., agent dynamics, network topology, initial conditions) and the 'posterior' information (e.g., sensor measurements, transmitted data over communication network), whereas the non-Bayesian approaches are relevant especially when the system dynamics is not available [15], [16]. In particular, the distributed Kalman filtering structure has proven to be effective in implementing different distributed estimation algorithms, e.g., the Kalman consensus filtering [17], [18], the diffusion strategy for distributed Kalman filtering [19], [20], complex filter dynamics modeling [21], [22], as well as the combination of two or more of those algorithms [23], [24].

The main idea of the distributed estimation is to exchange information between agents, enabling individual agents to have better awareness of the MAS states. Therefore, the existing techniques essentially build upon the inter-agent communication network through which large data can be transmitted and used for each agent's estimation process. A large data transmission through the communication network, however, is nontrivial in the MAS as it is susceptible to packet loss [25], [26] and delay [27], [28] as well as malicious cyber attack injections [29]–[31], and could consume considerably more energy than processing data locally [32]. To reduce the amount and frequency of communication, the distributed consensus estimation with the limited communication rate is discussed [33], and a gossip-based approach is proposed where local information exchange occurs randomly [34], [35]. Further, event-triggered consensus approaches can significantly alleviate the usage of both communication and computation resources [36], [37]. Nevertheless, none of them has attempted to avoid the use of a communication network which could be inherently vulnerable and costly.

This paper seeks a distributed estimation method that does not count on the communication between agents. Specifically, we are interested in a sensing-based estimation approach where a local agent only takes measurements of its neighbors using on-board sensors, thereby eliminating the data communication along with its associated issues (e.g., packet drops, communication delays, cyber attacks on communication). Despite the advantage of no communication, individual agents have less information available to them when they rely solely on their sensor measurements instead of communication data. This comes with the difficulty in expanding the estimation scope beyond the limited sensing range, while making its performance comparable with existing communication-based

Manuscript submitted October 00, 2017.

The authors are with the School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN, 47907 USA (email: kwonc@purdue.edu, ihwang@purdue.edu)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2018.2867341, IEEE Transactions on Automatic Control

methods. It is mathematically challenging to track a distant target agent that is outside of the monitoring agent's limited sensing range, which would have been directly available if communication is allowed. Such limited information could degrade the MAS coordination and even threaten the entire MAS's safety in the case of faulty agent behavior [38], [39]. Under this concern, we argue that the cooperative control information of the MAS can be exploited to compensate for the sensing-based estimation framework. To get a better insight, let us consider the distant agents which may not be within the sensing range of the local monitoring agent. While the distant agents in general are not directly observable from the monitoring agent, they will interact with their neighbors according to the embedded cooperative control protocol. Then, information about an unobservable agent can be indirectly obtained from observed agents whose behaviors are affected by interacting with the unobservable agent. Based on this idea, we develop a sensing-based distributed estimation algorithm that makes use of the control protocol information as the 'prior' knowledge instead of the communication data for the 'posterior' information within the Bayesian estimation framework. To the best of the our knowledge, this is the first attempt to develop a distributed estimation algorithm without relying on communication in a multi-agent network environment, a kind of paradigm changing innovation over the existing communication-based distributed estimation techniques, and thus granting advantages for preventing the various issues inherent in data communications.

The MAS considered in this paper is modeled as a group of stochastic linear systems in which individual agents have time invariant linear dynamics with disturbances and noisy sensor measurements. Due to these uncertainties, state estimates of neither in-range agents nor out-of-range agents are deterministic. Thus, we present them as probability distributions. To track a target agent beyond a local monitoring agent's sensing range, our estimation algorithm recursively predicts and updates the probability distributions using: (i) the observed behavior of in-range agents; and (ii) their correlations with out-of-range agents based on the agent's dynamics, sensing-based MAS network, and cooperative control protocol information. We would like to note that the similar concept of sensing-based distributed estimation has been discussed in the authors' preliminary work [40]. However, substantially new mathematical derivations are carried out in this paper to account for how the exact inter-agent interaction information can be used for the estimation, whereas the previous paper has only taken care of the aggregated impact of the interactions that results in the approximated estimation. Further, the complete theoretical verification is presented for the stability of the proposed distributed estimation algorithm and extensive numerical simulations are performed to validate the algorithm performance. The augmented sensing-based estimates, attained locally by each monitoring agent, could be particularly useful when the communication is costly or vulnerable, and can successfully complement the existing communication-based distributed estimation methods.

The rest of this paper is organized as follows: In Section II, we present a description of the MAS, including the dynamics, control protocol, and network topology of inter-agent sensing. Section III provides a detailed derivation of the sensing-based distributed state estimation algorithm subdivided into the prediction and update steps. The stability of the proposed algorithm is analytically proved in Section IV, followed by the numerical demonstration with an example of the cooperative MAS, a multi-UAS formation flight, in Section V. Conclusions and future work are given in Section VI.

II. Cooperative Multi-Agent System Model

The MAS considered in this paper is modeled as a group of stochastic linear time-invariant systems, in which individual agents are interacting in accordance with cooperative control protocols. The homogeneous dynamics of each agent is given by:

$$x_i(k+1) = Ax_i(k) + Bu_i(k) + w_i(k), \quad \forall i \in \{1, 2, \cdots, N\} \tag{1}$$

where $x_i \in \mathbb{R}^n$ and $u_i \in \mathbb{R}^p$ are the state and control input of the i-th agent, respectively; $w_i \in \mathbb{R}^n$ is the disturbance assumed to be zero mean white Gaussian noise with a positive definite covariance matrix $Q_i \succ \mathbf{0}$; A and B are the homogeneous system matrices of appropriate dimensions for each agent; and $k \in \mathbb{N} = \{0, 1, 2, \cdots\}$ denotes discrete-time index. The state of an agent may represent continuous quantities regarding the MAS dynamics such as position, attitude, temperature, voltage, etc. It is assumed that the matrix pair (A, B) satisfies the controllability condition.

Given the knowledge of the agents' dynamics, multi-agent cooperation is achieved using information exchanges among neighboring agents, which can be enabled by communication link and/or taking measurements from on-board sensors. The inter-agent network topology can be described by a graph $G = (V, E)$, which consists of a set of vertices $V = \{1, 2, \cdots, N\}$ indexed by each agent and a set of edges $E \subset V \times V$. This research considers the sensing-based network in that the j-th agent neighbors the i-th agent, $(j, i) \in E$, indicating it stays in the sensing range and can be observed by agent i. Note that the graph G for a sensing-based network is a class of directed graphs due to the different sensing capability of each agent, i.e., $(j, i) \in E$ does not guarantee that agent j is able to observe agent i. Furthermore, E can change over time in a dynamic network with a varying topology. However, in this research, we consider the MAS with the fixed network topology and the varying topology case will be included in our future research.

The control input for each agent depends on its neighboring agents' configuration and thus has the following structure:

$$u_i(k) = f_i\left(x_i(k), \left\{x_j(k) \middle| (i, j) \in \Omega_i\right\}\right) \tag{2}$$

where the function $f_i : \mathbb{R}^n \times \mathbb{R}^{n|\Omega_i|} \to \mathbb{R}^p$ prescribes the cooperative control protocol for agent i; Ω_i is the set of agent i's neighbors (including itself) under the given G for sensing-based network; and $|\Omega_i|$ denotes its cardinality. Thus, Ω_i denotes the set of the observable agents from the perspective of agent i, directly affecting agent i's behavior along with agent i itself. To account for cooperative coordination between agents,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2018.2867341, IEEE Transactions on Automatic Control

we consider a linear state feedback control protocol f_i defined by:

$$f_i\left(x_i, \{x_j|(i,j) \in \Omega_i\}\right) := Kx_i + b_i + K \sum_{j\in\Omega_i} (x_i - x_j) \quad (3)$$

where $K \in \mathbb{R}^{p\times n}$ is a control gain matrix for the interactions with the neighboring agents and $b_i \in \mathbb{R}^p$ is a reference control input for agent i. Under this control protocol, each interaction among the neighboring agents collectively exhibits the cooperation goal of the MAS. Eq. (3) can represent most cooperative control strategies from the general consensus algorithms [3] to specific applications including but not limited to rendezvous [41], flocking behavior [10], formation control [11], [42], etc.

In reality, the state feedback control (3) cannot be directly applicable to the MAS with noisy observation which prevents the agent from accessing the exact state information. Therefore, while retaining the structure of (3), we replace the true states with the corresponding sensor measurements. The measurement set for the i-th agent is denoted by $\{z_{ij}(k)|j \in V\}$ and is given by:

$$z_{ij}(k) = c_{ij}\left(x_j(k) + v_{ij}(k)\right), \quad \forall j \in V \quad (4)$$

where subscript "ij" of z indicates "the state of agent j measured by agent i". $v_{ij} \in \mathbb{R}^n$ are the sensor noises in agent i assumed to be independent and identically distributed (i.i.d.) Gaussian random variables with zero mean and covariance $R_i \succ \mathbf{0}$, and c_{ij} is a Boolean variable that indicates the availability of the sensor observation from agent i to agent j such that:

$$c_{ij} := \begin{cases} 1 & \text{if } \ j \in \Omega_i \\ 0 & \text{otherwise} \end{cases}$$

Using this definition, it is clear that $\Omega_i \equiv \{j|c_{ij} = 1\}$ and $(j,i) \in E \Longleftrightarrow c_{ij} = 1$. Applying (4) to (3), the control input for agent i is given by:

$$\begin{aligned} u_i(k) &= Kz_{ii}(k) + b_i(k) + K\sum_{j\in\Omega_i} z_{ii}(k) - K\sum_{j\in\Omega_i} z_{ij}(k) \\ &= K_{\Omega_i} z_{ii}(k) + b_i(k) - K\sum_{j\in\Omega_i} z_{ij}(k) \end{aligned} \quad (5)$$

where $K_{\Omega_i} := K + \sum_{j\in\Omega_i} K_{ij} = (1 + |\Omega_i|)\,K$ aggregates all the feedback gain matrices associated with the interactions on agent i, including itself. Plugging (5) into (1), the individual agent state evolution is governed by the following dynamics:

$$x_i(k+1) = Ax_i(k) + B\left(\mathcal{K}_i Z_i(k) + b_i(k)\right) + w_i(k), \qquad i \in V \quad (6)$$

where the block gain matrices $\mathcal{K}_i$, $i \in V$ are:

$$\begin{aligned} \mathcal{K}_1 &:= \begin{bmatrix} K_{\Omega_1} & -K & \cdots & -K \end{bmatrix} \\ \mathcal{K}_2 &:= \begin{bmatrix} -K & K_{\Omega_2} & \cdots & -K \end{bmatrix} \\ &\vdots \\ \mathcal{K}_N &:= \begin{bmatrix} -K & -K & \cdots & K_{\Omega_N} \end{bmatrix} \end{aligned}$$

and $Z_i(k) := \begin{bmatrix} z_{i1}^{\mathrm{T}}(k) & z_{i2}^{\mathrm{T}}(k) & \cdots & z_{iN}^{\mathrm{T}}(k) \end{bmatrix}^{\mathrm{T}}$ arrays all the measurements of agent i in a single vector.

In order for all the agents to monitor their cooperative interactions, we consider a unified multi-agent network where each agent can be treated as a "monitoring agent" to track and check other agents as well as a "target agent" to be observed and interact with.

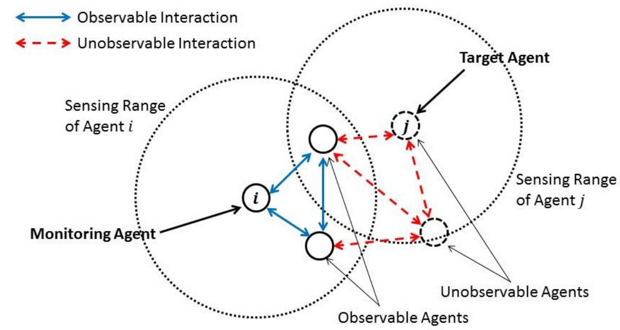


Fig. 1. Availability of Observation from a Local Monitoring Agent with a Limited Sensing Range

However, as shown in Figure 1, the limited sensing range of the on-board sensors usually provides the monitoring agents with only partial awareness about their neighbors. Therefore, the main challenge of the sensing-based MAS is to track the distant agents whose behavior is not directly observable and further affected by other unobserved agents. Even if the target is inside the local monitor's sensing range, its control actions could still be affected by the out-of-range agents, making it difficult to track. This is mathematically formulated by the following problem.

Problem 1. *Distributed Sensing-based State Estimation:* For a given measurement set $\{Z_i(0), Z_i(1), \cdots, Z_i(k)\}$ of a local monitoring agent i, observing its neighboring agents Ω_i up to time step k, we estimate the state $x_j(k)$ of target agent $j \in V$, for both $j \in \Omega_i$ and $j \notin \Omega_i$ cases.

III. Algorithm Development: Sensing-Based Distributed Estimation

This section presents a distributed estimation algorithm for each monitoring agent, which can keep track of distant agents beyond its sensing range. To compensate for "lacking sensor measurements" for the out-of-range agents, our estimator model is supplemented with the following information.

Assumption 1. The estimator embedded in each agent is assumed to have full information of the MAS dynamics and network topology, i.e., A, B, and G, and the cooperative control protocols K and b_i, $\forall i \in V$.

Assumption 1 indicates the cooperative MAS in which all the agents belong to a collaborative team and share important information such as the agent dynamics, network topology, and control protocols. This information is the prior knowledge for the state estimation, and thus can be implemented once at the initialization of the proposed estimation algorithm while not necessarily being updated thereafter. Based on this, we consider the set of agent states that are unobservable from the

i-th agent, denoted by $\{x_j(k) | j \notin \Omega_i\}$, and represent them as probability distributions. Their state estimate is initialized by the given probability measure $\mathcal{P}_j : \mathfrak{B}(\mathbb{R}^n) \to [0, 1]$ on $(\mathbb{R}^n, \mathfrak{B}(\mathbb{R}^n))$ where $\mathfrak{B}(\bullet)$ represents a *Borel measure*. In this paper, we further assume that the probability measure $\mathcal{P}_j$ is Gaussian, or equivalently, the following.

Assumption 2. Each unobservable state vector $x_j,\ j \notin \Omega_i$ is assumed to be a multivariate Gaussian distribution with mean $\bar{x}_j$ and positive definite covariance Σ_j. This is denoted as $x_j \sim \mathcal{N}_n(\bar{x}_j, \Sigma_j)$ where the subscript n represents the dimension of random vector.

This assumption is justified by the Central Limit Theorem which states that many probability distributions in practical systems can be approximated by Gaussian distributions [43]. Then, from the perspective of a monitoring agent i, the initial probability density function (pdf) of each agent's state $x_j(0), \forall j \in V$ is described by:

$$p[x_j(0)|Z_i(0)] = \begin{cases} \mathcal{N}_n(x_j(0); \bar{x}_j^0, \Sigma_j^0) & \text{if} \quad j \notin \Omega_i \\ \mathcal{N}_n(x_j(0); z_{ij}(0), R_i) & \text{if} \quad j \in \Omega_i \end{cases} \tag{7}$$

where $p[\bullet|\bullet]$ denotes a conditional pdf, and $\bar{x}_j^0$ and Σ_j^0 are the given initial mean and covariance, respectively. Note that there is no need for an initial set-up when the target agent j is inside the sensing range of monitoring agent i since it is provided by sensor measurements $z_{ij}(0)$.

Let $Z_i^k := \{Z_i(0), Z_i(1), \cdots, Z_i(k)\}$ denote the set of measurements collected by agent i up to time step k. Using a *Bayesian approach*, the distributed estimation problem can be posed as follows. Suppose, at time k, we have the state (conditional) pdfs $p[x_j(k)|Z_i^k], \forall j \in V$. We need to compute the pdfs $p[x_j(k+1)|Z_i^{k+1}], \forall j \in V$ at time $k+1$ using the new measurement vector $Z_i(k+1)$. It is worth noting that agent j is not necessarily to be the neighbor of monitoring agent i. By Bayes' Theorem we have:

$$\begin{aligned} p[x_j(k+1)|Z_i^{k+1}] = &\frac{p[Z_i(k+1)|x_j(k+1), Z_i^k]}{p[Z_i(k+1)|Z_i^k]} \\ &\times p[x_j(k+1)|Z_i^k] \end{aligned} \tag{8}$$

In particular, Eq. (8) has a similar structure to the Kalman filter for the case of Gaussian processes. Therefore, we develop a set of N Kalman filter-like linear estimators to estimate the states of individual agents, including the monitoring agent i itself.

At time step $k+1$, a local monitoring agent i acquires a new measurement $Z_i(k+1)$ with the posterior state pdfs of the previous time step k:

$$p[x_j(k)|Z_i^k] = \mathcal{N}_n(x_j(k); {}^i\hat{x}_j(k), {}^i\Sigma_j(k)), \quad \forall j \in V \tag{9}$$

where ${}^i\hat{x}_j(k)$ and ${}^i\Sigma_j(k)$ are the posterior mean and posterior covariance of the state of agent j, estimated by agent i at time k, defined as:

$$\begin{aligned} {}^i\hat{x}_j(k) &:= \mathbb{E}[x_j(k)|Z_i^k] \\ {}^i\Sigma_j(k) &:= \mathbb{E}\left[(x_j(k) - {}^i\hat{x}_j(k))(x_j(k) - {}^i\hat{x}_j(k))^{\mathrm{T}}\right] \end{aligned}$$

where $\mathbb{E}[\bullet]$ (or $\mathbb{E}[\bullet|\bullet]$) denotes the (conditional) expectation. Due to the dynamic interactions between individual agents over time, the state estimates of individual agents are correlated with each other, given as the following cross-covariance:

$${}^i\Sigma_{jl}(k) := \mathbb{E}\left[(x_j(k) - {}^i\hat{x}_j(k))(x_l(k) - {}^i\hat{x}_l(k))^{\mathrm{T}}\right], \quad j \neq l$$

Note that superscript "i" of the above mean, covariance, and cross-covariance indicates that "the state estimate information is computed in a local monitoring agent i, using the sensor measurement set Z_i^k only". Based on this information, each monitoring agent individually processes the pdfs through the following *prediction* and *update* steps.

A. Prediction Step

We first compute the predicted state pdfs $p[x_j(k+1)|Z_i^k], \forall j \in V$, i.e., the last term of the right hand side of (8). From (6), this can be represented by:

$$\begin{aligned} p[x_j(k+1)|Z_i^k] = p\big[Ax_j(k) + &B\left(\mathcal{K}_j Z_j(k) + b_j(k)\right) \\ &+ w_j(k)\big|Z_i^k\big] \end{aligned} \tag{10}$$

Since the pdf $p[x_j(k)|Z_i^k]$ is computed in the previous time step and the disturbance $w_j(k)$ is independent of Z_i^k, the only unknown probability term in the right-hand side of (10) is $p[Z_j(k)|Z_i^k]$, i.e., the pdf of the measurement set collected at agent j from the perspective of the monitoring agent i. In order to compute this, let us consider the pdfs of individual measurement segments in $Z_j(k)$, i.e., $p[z_{jl}(k)|Z_i^k]$, $l \in V$. Applying the posterior estimate information (9) into (4), we have the following Gaussian pdf of $z_{jl}(k)$ for each l, conditioned on Z_i^k:

$$p[z_{jl}(k)|Z_i^k] = \mathcal{N}_n\left(z_{jl}(k); c_{jl}{}^i\hat{x}_l(k), c_{jl}\left({}^i\Sigma_l(k) + R_j\right)\right) \tag{11}$$

And the correlations between $z_{jl}(k)$, $l \in V$ are determined by the corresponding cross-covariances between the estimates of agent j and agent l, i.e., ${}^i\Sigma_{jl}(k)$, $l \in V$, along with their availability of the observation from agent j, i.e., c_{jl}. Combining (11) for all $l \in V$ together, the conditional pdf of $Z_j(k)$ is Gaussian and can be represented by:

$$p[Z_j(k)|Z_i^k] = \mathcal{N}_{nN}\left(Z_j(k); {}^i\hat{Z}_j(k), {}^i\Sigma_j^Z(k)\right) \tag{12}$$

where the mean and covariance are respectively:

$$\begin{aligned} {}^i\hat{Z}_j(k) :=& \mathbb{E}[Z_j(k)|Z_i^k] \\ =& \begin{bmatrix} c_{j1}{}^i\hat{x}_1^{\mathrm{T}}(k) & c_{j2}{}^i\hat{x}_2^{\mathrm{T}}(k) & \cdots & c_{jN}{}^i\hat{x}_N^{\mathrm{T}}(k) \end{bmatrix}^{\mathrm{T}} \end{aligned} \tag{13}$$

$$\begin{aligned} {}^i\Sigma_j^Z(k) :=\ & \mathbb{E}\left[\left(Z_j(k) - {}^i\hat{Z}_j(k)\right)\left(Z_j(k) - {}^i\hat{Z}_j(k)\right)^{\mathrm{T}}\right] \\ =& \begin{bmatrix} c_{j1}\left({}^i\Sigma_1(k) + R_j\right) & c_{j1}c_{j2}{}^i\Sigma_{12}(k) & \cdots & c_{j1}c_{jN}{}^i\Sigma_{1N}(k) \\ c_{j2}c_{j1}{}^i\Sigma_{21}(k) & c_{j2}\left({}^i\Sigma_2(k) + R_j\right) & \cdots & c_{j2}c_{jN}{}^i\Sigma_{2N}(k) \\ \vdots & \vdots & \ddots & \vdots \\ c_{jN}c_{j1}{}^i\Sigma_{N1}(k) & c_{jN}c_{j2}{}^i\Sigma_{N2}(k) & \cdots & c_{jN}\left({}^i\Sigma_N(k) + R_j\right) \end{bmatrix} \end{aligned} \tag{14}$$

Applying (12) to (10) yields the Gaussian pdf $p[x_j(k+1)|Z_i^k]$ whose mean is the predicted state estimate ${}^i\hat{x}_j^-(k+1)$ given by:

$$\begin{aligned}{}^i\hat{x}_j^-(k+1) &:= \mathbb{E}[x_j(k+1)|Z_i^k] \\ &= A^i\hat{x}_j(k) + B\left(\mathcal{K}_j{}^i\hat{Z}_j(k) + b_j(k)\right)\end{aligned} \tag{15}$$

and the predicted error covariance ${}^i\Sigma_j^-(k+1)$ is:

$$\begin{aligned}{}^i\Sigma_j^-(k+1) &:= \mathbb{E}\big[(x_j(k+1) - {}^i\hat{x}_j^-(k+1)) \\ &\qquad (x_j(k+1) - {}^i\hat{x}_j^-(k+1))^{\mathrm{T}}\big] \\ &= A^i\Sigma_j(k)A^{\mathrm{T}} + B\mathcal{K}_j{}^i\Sigma_j^Z(k)\mathcal{K}_j^{\mathrm{T}}B^{\mathrm{T}} \\ &\quad + A^iP_j(k)\mathcal{K}_j^{\mathrm{T}}B^{\mathrm{T}} + B\mathcal{K}_j{}^iP_j^{\mathrm{T}}(k)A^{\mathrm{T}} + Q_j\end{aligned} \tag{16}$$

where the block matrix $P_j \in \mathbb{R}^{n\times nN}$ is defined by:

$${}^iP_j(k) := \begin{bmatrix} c_{j1}{}^i\Sigma_{j1}(k) & c_{j2}{}^i\Sigma_{j2}(k) & \cdots & c_{jN}{}^i\Sigma_{jN}(k)\end{bmatrix} \tag{17}$$

Clearly, the both mean and covariance terms are strongly dependent on the network topology of the MAS for which an appropriate design of G will be discussed in Section IV.

B. Update Step

We now update the predicted state pdfs $p[x_j(k+1)|Z_i^k], \forall j \in V$ to the posterior pdfs $p[x_j(k+1)|Z_i^{k+1}]$ using $Z_i(k+1)$. Recalling (8), the pdf $p[Z_i(k+1)|Z_i^k]$ cannot be computed in the same way as the formal Kalman filter does. This is mainly because the measurement Z_i may not explicitly include the state information of the target agent j when $j \notin \Omega_i$. Further, Z_i contains two types of information: *state information* of the agents within the sensing range, and *binary information* which indicates whether the agent is within the sensing range or not. For example, $z_{ij} = \mathbf{0}$ indicates in most cases that the agent j is out of sensing range, not that the actual state of agent j is being around $\mathbf{0}$. To avoid the disorientation this binary information may provide, we introduce an operator $\Gamma_i : \mathbb{R}^{nN} \to \mathbb{R}^{n|\Omega_i|}$ that only incorporates the measurement of the state for those within the sensing range:

$$\Gamma_i(Z_i) := \begin{bmatrix} z_{im_1}^{\mathrm{T}} & z_{im_2}^{\mathrm{T}} & \cdots & z_{im_{|\Omega_i|}}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}} \tag{18}$$

where $\{m_1, m_2, \cdots m_{|\Omega_i|}\} = \Omega_i$.

Then, we alternatively consider the pdf $p\left[\Gamma_i\left(Z_i(k+1)\right)|Z_i^k\right]$ instead of $p[Z_i(k+1)|Z_i^k]$ that can be represented by the following Gaussian distribution:

$$\begin{aligned}&p\left[\Gamma_i\left(Z_i(k+1)\right)\big|Z_i^k\right] = \\ &\mathcal{N}_{n|\Omega_i|}\left(\Gamma_i\left(Z_i(k+1)\right); \Gamma_i\left({}^i\hat{Z}_i(k+1)\right), {}^i\Sigma_i^{\Gamma}(k+1)\right)\end{aligned} \tag{19}$$

whose mean is given by:

$$\begin{aligned}\Gamma_i\left({}^i\hat{Z}_i(k+1)\right) &:= \mathbb{E}\left[\Gamma_i\left(Z_i(k+1)\right)\big|Z_i^k\right] \\ &= \begin{bmatrix} {}^i\hat{x}_{m_1}^-(k+1) \\ {}^i\hat{x}_{m_2}^-(k+1) \\ \vdots \\ {}^i\hat{x}_{m_{|\Omega_i|}}^-(k+1)\end{bmatrix} \\ &= \begin{bmatrix} A^i\hat{x}_{m_1}(k) + B\left(\mathcal{K}_{m_1}{}^i\hat{Z}_{m_1}(k) + b_{m_1}(k)\right) \\ A^i\hat{x}_{m_2}(k) + B\left(\mathcal{K}_{m_2}{}^i\hat{Z}_{m_2}(k) + b_{m_2}(k)\right) \\ \vdots \\ A^i\hat{x}_{m_{|\Omega_i|}}(k) + B\left(\mathcal{K}_{m_{|\Omega_i|}}{}^i\hat{Z}_{m_{|\Omega_i|}}(k) + b_{m_{|\Omega_i|}}(k)\right)\end{bmatrix}\end{aligned} \tag{20}$$

and the corresponding covariance is:

$$\begin{aligned}{}^i\Sigma_i^{\Gamma}(k+1) :=& \mathbb{E}\left[\left(\Gamma_i\left(Z_i(k+1)\right) - \Gamma_i\left({}^i\hat{Z}_i(k+1)\right)\right)\right. \\ &\left.\times\left(\Gamma_i\left(Z_i(k+1)\right) - \Gamma_i\left({}^i\hat{Z}_i(k+1)\right)\right)^{\mathrm{T}}\right] \\ =& \begin{bmatrix} {}^i\Sigma_{m_1}^-(k+1) + R_i & {}^i\Sigma_{m_1m_2}^-(k+1) & \cdots & {}^i\Sigma_{m_1m_{|\Omega_i|}}^-(k+1) \\ {}^i\Sigma_{m_2m_1}^-(k+1) & {}^i\Sigma_{m_2}^-(k+1) + R_i & \cdots & {}^i\Sigma_{m_2m_{|\Omega_i|}}^-(k+1) \\ \vdots & \vdots & \ddots & \vdots \\ {}^i\Sigma_{m_{|\Omega_i|}m_1}^-(k+1) & {}^i\Sigma_{m_{|\Omega_i|}m_2}^-(k+1) & \cdots & {}^i\Sigma_{m_{|\Omega_i|}}^-(k+1) + R_i\end{bmatrix}\end{aligned} \tag{21}$$

where the off-diagonal terms ${}^i\Sigma_{jl}^-(k+1)$, $j \neq l$ are given by:

$$\begin{aligned}{}^i\Sigma_{jl}^-(k+1) &:= \mathbb{E}\left[(x_j(k+1) - {}^i\hat{x}_j^-(k+1))\right. \\ &\qquad \left.(x_l(k+1) - {}^i\hat{x}_l^-(k+1))^{\mathrm{T}}\right] \\ &= A^i\Sigma_{jl}(k)A^{\mathrm{T}} + B\mathcal{K}_j{}^i\Sigma_{jl}^Z(k)\mathcal{K}_l^{\mathrm{T}}B^{\mathrm{T}} \\ &\quad + A^iP_{jl}(k)\mathcal{K}_l^{\mathrm{T}}B^{\mathrm{T}} + B\mathcal{K}_jP_{lj}^{\mathrm{T}}(k)A^{\mathrm{T}}\end{aligned} \tag{22}$$

with

$${}^iP_{jl}(k) := \begin{bmatrix} c_{l1}{}^i\Sigma_{j1}(k) & c_{l2}{}^i\Sigma_{j2}(k) & \cdots & c_{lN}{}^i\Sigma_{jN}(k)\end{bmatrix} \tag{23}$$

$$\begin{aligned}{}^i\Sigma_{jl}^Z(k) :=&\ \mathbb{E}\left[\left(Z_j(k) - {}^i\hat{Z}_j(k)\right)\left(Z_l(k) - {}^i\hat{Z}_l(k)\right)^{\mathrm{T}}\right] \\ =& \begin{bmatrix} c_{j1}c_{l1}{}^i\Sigma_1(k) & c_{j1}c_{l2}{}^i\Sigma_{12}(k) & \cdots & c_{j1}c_{lN}{}^i\Sigma_{1N}(k) \\ c_{j2}c_{l1}{}^i\Sigma_{21}(k) & c_{j2}c_{l2}{}^i\Sigma_2(k) & \cdots & c_{j2}c_{lN}{}^i\Sigma_{2N}(k) \\ \vdots & \vdots & \ddots & \vdots \\ c_{jN}c_{l1}{}^i\Sigma_{N1}(k) & c_{jN}c_{l2}{}^i\Sigma_{N2}(k) & \cdots & c_{jN}c_{lN}{}^i\Sigma_N(k)\end{bmatrix}\end{aligned} \tag{24}$$

Using (18), the posterior mean of $x_j(k+1)$, i.e., ${}^i\hat{x}_j(k+1) := \mathbb{E}[{}^i\hat{x}_j(k+1)|Z_i^{k+1}]$, $\forall j \in V$ can be updated as:

$$\begin{aligned}{}^i\hat{x}_j(k+1) = {}^i\hat{x}_j^-(k+1) + L_{ij}(k+1)\Big(&\Gamma_i\left(Z_i(k+1)\right) \\ &-\Gamma_i\left({}^i\hat{Z}_i(k+1)\right)\Big)\end{aligned} \tag{25}$$

where $L_{ij}(k+1) \in \mathbb{R}^{n\times n|\Omega_i(k+1)|}$ is the estimator gain for the state estimation of agent j computed in the monitoring

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2018.2867341, IEEE Transactions on Automatic Control

agent i. In the context of optimal estimators such as Kalman filter, L_{ij} is designed to minimize the mean-square error of the state estimate, i.e., $\mathbb{E}\left[\left\|x_j(k+1) - {}^i\hat{x}_j(k+1)\right\|^2\right]$. This is equivalent to minimizing the trace of the posterior covariance matrix ${}^i\Sigma_j(k+1)$, i.e., tr $\left[{}^i\Sigma_j(k+1)\right]$. By the definition of ${}^i\Sigma_j$, we have:

$$\begin{aligned}{}^i\Sigma_j(k+1) =&\mathbb{E}\Bigg[\Big(x_j(k+1) - {}^i\hat{x}_j^-(k+1) - L_{ij}(k+1)\\ &\times\left(\Gamma_i\left(Z_i(k+1)\right) - \Gamma_i\left({}^i\hat{Z}_i(k+1)\right)\right)\Big)\\ &\times\Big(x_j(k+1) - {}^i\hat{x}_j^-(k+1) - L_{ij}(k+1)\\ &\times\left(\Gamma_i\left(Z_i(k+1)\right) - \Gamma_i\left({}^i\hat{Z}_i(k+1)\right)\right)\Big)^{\mathrm{T}}\Bigg]\end{aligned} \tag{26}$$

Substituting the covariances from (16) and (21) into (26) yields:

$$\begin{aligned}{}^i\Sigma_j(k+1) =& {}^i\Sigma_j^-(k+1) - \mathbb{E}\Bigg[\left(x_j(k+1) - {}^i\hat{x}_j^-(k+1)\right)\\ &\times\left(\Gamma_i\left(Z_i(k+1)\right) - \Gamma_i\left({}^i\hat{Z}_i(k+1)\right)\right)^{\mathrm{T}}\Bigg]\\ &\times L_{ij}^{\mathrm{T}}(k+1) - L_{ij}(k+1)\mathbb{E}\Bigg[\Big(\Gamma_i\left(Z_i(k+1)\right)\\ &-\Gamma_i\left({}^i\hat{Z}_i(k+1)\right)\Big)\left(x_j(k+1) - {}^i\hat{x}_j^-(k+1)\right)^{\mathrm{T}}\Bigg]\\ &+ L_{ij}(k+1){}^i\Sigma_i^{\Gamma}(k+1)L_{ij}^{\mathrm{T}}(k+1)\end{aligned} \tag{27}$$

The rest of (27) are the cross-covariances between the predicted estimates of x_j and $\Gamma_i(Z_i)$. Since the sensor noises v_{ij} in $\Gamma_i(Z_i)$ are uncorrelated with x_j, it can be shown that:

$$\begin{aligned}\mathbb{E}&\left[(x_j(k+1) - {}^i\hat{x}_j^-(k+1))(z_{im}(k+1) - {}^i\hat{z}_{im}(k+1))^{\mathrm{T}}\right]\\ &\equiv {}^i\Sigma_{jm}^-(k+1), \qquad \forall m\in\Omega_i\end{aligned}$$

where ${}^i\hat{z}_{ij}(k+1) := \mathbb{E}[z_{ij}(k+1)|Z_i^k]$, $j\in V$. Then, Eq. (27) can be rewritten as:

$$\begin{aligned}{}^i\Sigma_j(k+1) =&{}^i\Sigma_j^-(k+1) - S_{ij}(k+1)L_{ij}^{\mathrm{T}}(k+1)\\ &- L_{ij}(k+1)S_{ij}^{\mathrm{T}}(k+1)\\ &+ L_{ij}(k+1){}^i\Sigma_i^{\Gamma}(k+1)L_{ij}^{\mathrm{T}}(k+1)\end{aligned} \tag{28}$$

where the block matrix $S_{ij}\in\mathbb{R}^{n\times n|\Omega_i|}$ is defined by:

$$\begin{aligned}S_{ij}(k+1) := \big[{}^i\Sigma_{jm_1}^-(k+1) \quad {}^i\Sigma_{jm_2}^-(k+1) \quad \cdots\\ \cdots \quad {}^i\Sigma_{jm_{\Omega_i}}^-(k+1)\big]\end{aligned} \tag{29}$$

Practically, S_{ij} implicates the correlations between the target agent j and the observed agents $\{m_1, m_2, \cdots m_{|\Omega_i|}\} = \Omega_i$. Similar to (28), the other cross-covariances ${}^i\Sigma_{jl}(k+1)$, $j\neq l$ are given by:

$$\begin{aligned}{}^i\Sigma_{jl}(k+1) =&{}^i\Sigma_{jl}^-(k+1) - S_{ij}(k+1)L_{il}^{\mathrm{T}}(k+1)\\ &- L_{ij}(k+1)S_{il}^{\mathrm{T}}(k+1)\\ &+ L_{ij}(k+1){}^i\Sigma_i^{\Gamma}(k+1)L_{il}^{\mathrm{T}}(k+1)\end{aligned} \tag{30}$$

By taking the derivative of the trace of ${}^i\Sigma_j(k+1)$ in (28) with respect to $L_{ij}(k+1)$ and setting it equal to zero, the *optimal* observer gain can be computed as:

$$L_{ij}(k+1) = S_{ij}(k+1)\left({}^i\Sigma_i^{\Gamma}(k+1)\right)^{-1} \tag{31}$$

This gain is used to update the state estimate of agent j by the monitoring agent i with respect to the observed measurement $Z_i(k+1)$ at time step $k+1$.

The overall recursive structure of the proposed sensing-based distributed state estimation algorithm is summarized in Algorithm 1.

Algorithm 1: Sensing-Based Distributed State Estimation for a Cooperative MAS with Fixed Network Topology

---

Initialization: given the information of the MAS dynamics A, B, G, K, and b_i, $\forall i\in V$,

- Set ${}^i\hat{x}_j(0) = \bar{x}_j^0$, ${}^i\Sigma_j(0) = \Sigma_j^0$ for the agents $j\notin\Omega_i$
- Set ${}^i\hat{x}_j(0) = z_{ij}(0)$, ${}^i\Sigma_j(0) = R_i$ for the agents $j\in\Omega_i$
- $p[x_j(0)|Z_i(0)]$, $\forall j\in V$ from (7)

for $k=0$ to the termination time T

a) Prediction step
  for $j=1$ to the total agent number N
  1) Construct ${}^i\hat{Z}_j(k)$ and ${}^i\Sigma_j^Z(k)$ using (13) and (14) respectively
  2) Compute ${}^i\hat{x}_j^-(k+1)$ using (15)
  3) Compute ${}^i\Sigma_j^-(k+1)$ using (16) and (17)
  Output
  $$p[x_j(k+1)|Z_i^k] = \mathcal{N}_n(x_j; {}^i\hat{x}_j^-(k+1), {}^i\Sigma_j^-(k+1))$$
  end for
b) Update step
  for $j=1$ to the total agent number N
  4) Compute ${}^i\Sigma_{jl}^-(k+1)$, $\forall l\in V$ using (22), (23), (24)
  5) Construct $\Gamma_i\left({}^i\hat{Z}_i(k+1)\right)$ and ${}^i\Sigma_i^{\Gamma}(k+1)$ using (20) and (21) respectively
  6) Solve for $L_{ij}(k+1)$ from (29) and (31)
  7) Compute ${}^i\hat{x}_j(k+1)$ using (25)
  8) Compute ${}^i\Sigma_j(k+1)$ using (28)
  9) Compute ${}^i\Sigma_{jl}(k+1)$, $\forall l\in V$ using (30)
  Output
  $$p[x_j(k+1)|Z_i^{k+1}] = \mathcal{N}_n(x_j; {}^i\hat{x}_j(k+1), {}^i\Sigma_j(k+1))$$
  end for
c) Store ${}^i\hat{x}_j(k+1)$, ${}^i\Sigma_j(k+1)$, and ${}^i\Sigma_{jl}(k+1)$, $\forall j,l\in V$ for the next time step iteration

end for

IV. Stability Analysis

In this section, we present a formal stability analysis of the developed sensing-based distributed state estimation algorithm. To guarantee the stability of the estimation algorithm for each monitoring agent, the condition for the MAS network

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2018.2867341, IEEE Transactions on Automatic Control

topology is first addressed. The idea is mainly based on the discrete-time version of the Lyapunov Theorem where the globally asymptotic stability is guaranteed by the existence of a radially unbounded, positive definite (possibly time-varying) Lyapunov function $\mathcal{V}$ which is strictly decreasing over time.

For the stochastic dynamical system such as the MAS considered in this paper, the estimation stability can be interpreted in terms of the supermartingales of Lyapunov functions [44], and correspondingly by the following statement.

Proposition 1. For the state estimation of the stochastic dynamical system, the estimation error e is globally asymptotically stable in the sense of Lyapunov if there exist a Lyapunov function $\mathcal{V}$ that satisfies the following conditions:

$$\begin{cases} \mathcal{V}(e(k),k) = 0, & e(k) = \mathbf{0} \\ \mathcal{V}(e(k),k) > 0, & e(k) \neq \mathbf{0} \\ \mathcal{V}(e(k),k) \to \infty, & e(k) \to \infty \end{cases}, \quad \forall k \tag{32}$$

$$\Delta\mathcal{V}(k+1,k) < 0, \quad \forall k \tag{33}$$

where $\Delta\mathcal{V}(k+1,k) := \mathcal{V}(\mathbb{E}\left[e(k+1)|e(k)\right], k+1) - \mathcal{V}(e(k),k)$.

This is well supported by intuition since a supermartingale tends to decrease in time and is in analogy with Lyapunov functions that are decreasing. Therefore, the focus of this section is to find a suitable candidate Lyapunov function of the state estimation error satisfying the above conditions.

From the structure of the proposed estimation algorithm, the state estimation error has the prediction and update phases, respectively, defined by:

$$\begin{cases} {}^ie_j^-(k) := x_j(k) - {}^i\hat{x}_j^-(k) \\ {}^ie_j(k) := x_j(k) - {}^i\hat{x}_j(k) \end{cases}, \quad \forall j \in V$$

Further, by aggregating the state estimation errors of the individual agents, we can define the following:

$$\begin{aligned} {}^ie^-(k) &:= \begin{bmatrix} {}^ie_1^{-\mathrm{T}}(k) & {}^ie_2^{-\mathrm{T}}(k) & \cdots & {}^ie_N^{-\mathrm{T}}(k) \end{bmatrix}^{\mathrm{T}} \\ {}^ie(k) &:= \begin{bmatrix} {}^ie_1^{\mathrm{T}}(k) & {}^ie_2^{\mathrm{T}}(k) & \cdots & {}^ie_N^{\mathrm{T}}(k) \end{bmatrix}^{\mathrm{T}} \end{aligned}$$

Regarding the prediction state estimation error, subtracting (15) from (1) gives the prediction error dynamics of the state estimate of agent j computed in the monitoring agent i:

$$\begin{aligned} {}^ie_j^-(k+1) =& A{}^ie_j(k) + BK({}^ie_j(k) + v_{jj}(k)) + w_j(k) \\ &+ BK\sum_{l\in\Omega_j}({}^ie_j(k) + v_{jj}(k) - {}^ie_l(k) - v_{jl}(k)) \end{aligned} \tag{34}$$

and its mean dynamics is given by:

$$\begin{aligned} \mathbb{E}[{}^ie_j^-(k+1)] = A\mathbb{E}[{}^ie_j(k)] &+ (1+|\Omega_j|)BK\mathbb{E}[{}^ie_j(k)] \\ &- BK\sum_{l\in\Omega_j}\mathbb{E}[{}^ie_l(k)] \end{aligned} \tag{35}$$

Concatenating (35) for all agents $j \in V$ together, the conditional expectation of the total prediction error can be represented as:

$$\begin{aligned} \mathbb{E}[{}^ie^-(k+1)|e(k)] =& (I_N\otimes A){}^ie(k) + ((I_N+\mathcal{D})\otimes BK) \\ &\times {}^ie(k) - (\mathcal{C}\otimes BK){}^ie(k) \\ =& ((I_N\otimes A) + ((I_N+\mathcal{L})\otimes BK)){}^ie(k) \end{aligned} \tag{36}$$

where $\otimes$ denotes the Kronecker product that assembles the entire MAS dynamics; I_N denotes the N dimensional identity matrix; $\mathcal{D} := \mathrm{diag}(d_1, d_2, \cdots, d_N)$ is the degree matrix of G with elements $d_i = |\Omega_i|$ and zero off-diagonal elements; $\mathcal{C}$ is the adjacency matrix of G defined as:

$$\mathcal{C} := \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1N} \\ c_{21} & c_{22} & \cdots & c_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ c_{N1} & c_{N2} & \cdots & c_{NN} \end{bmatrix}$$

; and $\mathcal{L} \in \mathbb{R}^{N\times N}$ is the graph Laplacian of G defined as:

$$\mathcal{L} := \mathcal{D} - \mathcal{C}$$

Similarly, subtracting (25) from (1) yields the mean dynamics of the update error for the state estimation of agent j by the monitoring agent i:

$$\mathbb{E}[{}^ie_j(k)] = \mathbb{E}[{}^ie_j^-(k)] - L_{ij}(k) \begin{bmatrix} \mathbb{E}[{}^ie_{m_1}^-(k)] \\ \mathbb{E}[{}^ie_{m_2}^-(k)] \\ \vdots \\ \mathbb{E}[{}^ie_{m_{|\Omega_i|}}^-(k)] \end{bmatrix} \tag{37}$$

Here, L_{ij}, designed in (31), consists of S_{ij} from (29) and ${}^i\Sigma_i^\Gamma$ from (21), which can be respectively expressed by:

$$S_{ij}(k) = \begin{bmatrix} {}^i\Sigma_{j1}^-(k) & {}^i\Sigma_{j2}^-(k) & \cdots & {}^i\Sigma_{jN}^-(k) \end{bmatrix} \left(\mathcal{C}_i^{\mathrm{T}} \otimes I_n\right) \tag{38}$$

$${}^i\Sigma_i^\Gamma(k) = (\mathcal{C}_i\otimes I_n)\left({}^i\Sigma^-(k) + I_N\otimes R_i\right)\left(\mathcal{C}_i^{\mathrm{T}}\otimes I_n\right) \tag{39}$$

where ${}^i\Sigma^-(k) \in \mathbb{R}^{nN\times nN}$ represents the total MAS prediction error covariance from the monitoring agent i:

$${}^i\Sigma^-(k) := \begin{bmatrix} {}^i\Sigma_1^-(k) & {}^i\Sigma_{12}^-(k) & \cdots & {}^i\Sigma_{1N}^-(k) \\ {}^i\Sigma_{21}^-(k) & {}^i\Sigma_2^-(k) & \cdots & {}^i\Sigma_{2N}^-(k) \\ \vdots & \vdots & \ddots & \vdots \\ {}^i\Sigma_{N1}^-(k) & {}^i\Sigma_{N2}^-(k) & \cdots & {}^i\Sigma_N^-(k) \end{bmatrix}$$

and $\mathcal{C}_i \in \mathbb{R}^{|\Omega_i|\times N}$ is the graph observation matrix of agent i, designed by:

$$\mathcal{C}_i := \begin{bmatrix} h_1 & h_2 & \cdots & h_{|\Omega_i|} \end{bmatrix}^{\mathrm{T}}$$

where $h_q \in \mathbb{R}^N$, $q = \{1, 2, \cdots, |\Omega_i|\}$ are the non-zero column vectors of the matrix $\mathrm{diag}(c_{i1}, c_{i2}, \cdots, c_{iN})$. Integrating (37) altogether, the conditional expectation of the entire update error is given by:

$$\begin{aligned} \mathbb{E}[{}^ie(k)|{}^ie^-(k)] = \Big(&(I_N\otimes I_n) - {}^i\Sigma^-(k)\left(\mathcal{C}_i^{\mathrm{T}}\otimes I_n\right) \\ &\times \left((\mathcal{C}_i\otimes I_n)\left({}^i\Sigma^-(k) + I_N\otimes R_i\right)\left(\mathcal{C}_i^{\mathrm{T}}\otimes I_n\right)\right)^{-1} \\ &\times (\mathcal{C}_i\otimes I_n)\Big){}^ie^-(k) \end{aligned} \tag{40}$$

Combining (36) and (40), the complete evolution of the estimation error over the one prediction-update cycle can be written in the following compact form:

$$\mathbb{E}[{}^ie(k+1)|{}^ie(k)] = \mathcal{F}_i(k+1)\mathcal{A}{}^ie(k) \tag{41}$$

where

$$\mathcal{A} := (I_N \otimes A) + ((I_N + \mathcal{L}) \otimes BK)$$

$$\begin{aligned}\mathcal{F}_i(k) := \Big(& (I_N \otimes I_n) - {}^i\Sigma^-(k)\left(\mathcal{C}_i^{\mathrm{T}} \otimes I_n\right) \\ & \times \left((\mathcal{C}_i \otimes I_n)\left({}^i\Sigma^-(k) + I_N \otimes R_i\right)\left(\mathcal{C}_i^{\mathrm{T}} \otimes I_n\right)\right)^{-1} \\ & \times (\mathcal{C}_i \otimes I_n)\Big)\end{aligned}$$

Without loss of generality, $\mathcal{A}$ is designed to be non-singular for the closed-loop dynamical systems.

Before verifying the stability of (41), we consider the following lemma.

Lemma 1. Define ${}^i\Sigma \in \mathbb{R}^{nN \times nN}$ to be the total MAS estimation error covariance of the proposed algorithm, as computed by a local monitoring agent i:

$${}^i\Sigma(k) := \begin{bmatrix} {}^i\Sigma_1(k) & {}^i\Sigma_{12}(k) & \cdots & {}^i\Sigma_{1N}(k) \\ {}^i\Sigma_{21}(k) & {}^i\Sigma_2(k) & \cdots & {}^i\Sigma_{2N}(k) \\ \vdots & \vdots & \ddots & \vdots \\ {}^i\Sigma_{N1}(k) & {}^i\Sigma_{N2}(k) & \cdots & {}^i\Sigma_N(k) \end{bmatrix}$$

Then, ${}^i\Sigma(k)$ is positive definite and bounded for all $k > N$ if the following system is observable:

$$\begin{aligned} x(k+1) &= \mathcal{L}x(k) \\ y(k) &= \mathcal{C}_i x(k) \end{aligned} \tag{42}$$

Proof. First, the positive definiteness of ${}^i\Sigma(k)$ can be shown by induction. Suppose ${}^i\Sigma(k-1) \succ \mathbf{0}$. From (16) and (22), the total prediction error covariance at time step k is given by:

$${}^i\Sigma^-(k) = \mathcal{A}{}^i\Sigma(k-1)\mathcal{A}^{\mathrm{T}} + \mathcal{Q} + \mathcal{R} \tag{43}$$

where $\mathcal{Q}$ and $\mathcal{R}$ respectively represent the total disturbance and process noise covariances defined as:

$$\mathcal{Q} := \begin{bmatrix} Q_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & Q_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & Q_N \end{bmatrix}, \ \mathcal{R} := \begin{bmatrix} \mathcal{R}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathcal{R}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathcal{R}_N \end{bmatrix}$$

where $\mathcal{R}_i := \left(|\Omega_i|^2 + |\Omega_i| - 1\right) BKR_iK^{\mathrm{T}}B^{\mathrm{T}}, \forall i \in V$. Since $\mathcal{Q} \succ \mathbf{0}$ and $\mathcal{R} \succeq \mathbf{0}$, ${}^i\Sigma^-(k) \succ \mathbf{0}$ hold. Then, from (28) and (30), we have:

$$\begin{aligned}{}^i\Sigma(k) = {}^i\Sigma^-(k) - & {}^i\Sigma^-(k)\left(\mathcal{C}_i^{\mathrm{T}} \otimes I_n\right) \\ & \times \left((\mathcal{C}_i \otimes I_n)\left({}^i\Sigma^-(k) + I_N \otimes R_i\right)\left(\mathcal{C}_i^{\mathrm{T}} \otimes I_n\right)\right)^{-1} \\ & \times (\mathcal{C}_i \otimes I_n){}^i\Sigma^-(k)\end{aligned} \tag{44}$$

Using the matrix inversion lemma [45], Eq. (44) can be rewritten as:

$$\begin{aligned}{}^i\Sigma(k) = \Big(& \left({}^i\Sigma^-(k)\right)^{-1} + \left(\mathcal{C}_i^{\mathrm{T}} \otimes I_n\right)\Big((\mathcal{C}_i \otimes I_n) \\ & \times (I_N \otimes R_i)\left(\mathcal{C}_i^{\mathrm{T}} \otimes I_n\right)\Big)^{-1}(\mathcal{C}_i \otimes I_n)\Big)^{-1}\end{aligned} \tag{45}$$

Here, $(\mathcal{C}_i \otimes I_n)(I_N \otimes R_i)\left(\mathcal{C}_i^{\mathrm{T}} \otimes I_n\right) \succ \mathbf{0}$ and ${}^i\Sigma^-(k) \succ \mathbf{0}$ in (45) guarantee the positive definiteness of ${}^i\Sigma(k)$. Therefore, given ${}^i\Sigma(0) \succ \mathbf{0}$, one can assure ${}^i\Sigma(k) \succ \mathbf{0}$, $\forall k$.

Second, the boundedness of ${}^i\Sigma(k)$ can be addressed by the optimal design of the observer gain, resulting in the minimum error variance. That is, any sub-optimal estimator is not of minimum variance which cannot be smaller than ${}^i\Sigma(k)$. To this end, we consider the weighted observabilty grammian matrix ${}^i\mathcal{O} \in \mathbb{R}^{nN \times nN}$ of the matrix pair $((\mathcal{C}_i \otimes I_n), \mathcal{A})$, defined by :

$$\begin{aligned}{}^i\mathcal{O}(k) := \sum_{t=k-N}^{k} & \left(\mathcal{A}^{t-k+N}\right)^{\mathrm{T}}\left(\mathcal{C}_i^{\mathrm{T}}\mathcal{C}_i \otimes I_n\right)\left(I_N \otimes R_i^{-1}\right) \\ & \times \left(\mathcal{C}_i^{\mathrm{T}}\mathcal{C}_i \otimes I_n\right)\mathcal{A}^{t-k+N}\end{aligned} \tag{46}$$

where the weight $(\mathcal{C}_i \otimes I_n)\left(I_N \otimes R_i^{-1}\right)\left(\mathcal{C}_i^{\mathrm{T}} \otimes I_n\right) \succ \mathbf{0}$. Here, ${}^i\mathcal{O}(k)$ has full rank nN, i.e., invertible, if and only if the pair $((\mathcal{C}_i \otimes I_n), \mathcal{A})$ is (uniformly) observable. Considering the structure of $\mathcal{A}$, this is equivalent to the observability of $(\mathcal{C}_i, \alpha I_N + \beta(I_N + \mathcal{L}))$ with arbitrary real numbers α and β. Since $(\alpha + \beta) I_N$ makes no difference to the observability, ${}^i\mathcal{O}(k)$ is of full rank for all $k > N$ if and only if the system (42) is observable. Further, we define:

$${}^i\tilde{\mathcal{O}}(k) := \left(\mathcal{A}^{-N}\right)^{\mathrm{T}}{}^i\mathcal{O}(k)\left(\mathcal{A}^{-N}\right) \tag{47}$$

This matrix is closely related to the observability grammian as we can equivalently assure the full rank of ${}^i\tilde{\mathcal{O}}(k)$ with the same observability condition of ${}^i\mathcal{O}(k)$. Using the inverse of ${}^i\tilde{\mathcal{O}}(k)$, the following non-Bayesian estimator can be designed to estimate the MAS states from a local monitoring agent i:

$$\begin{aligned}{}^i\tilde{x}(k) = \left({}^i\tilde{\mathcal{O}}(k)\right)^{-1}\sum_{t=k-N}^{k} & \left(\mathcal{A}^{t-k}\right)^{\mathrm{T}}\left(\mathcal{C}_i^{\mathrm{T}}\mathcal{C}_i \otimes I_n\right) \\ & \times \left(I_N \otimes R_i^{-1}\right) Z_i(t)\end{aligned} \tag{48}$$

where ${}^i\tilde{x} := \begin{bmatrix} {}^i\tilde{x}_1^{\mathrm{T}} & {}^i\tilde{x}_2^{\mathrm{T}} & \cdots & {}^i\tilde{x}_N^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ is the state estimate of the entire MAS using the estimator in (48). Let ${}^i\tilde{\Sigma}$ be the error covariance corresponding to this estimator. Through the evaluations of (46) and (48) along with the true state evolution

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2018.2867341, IEEE Transactions on Automatic Control

(6), ${}^{i}\tilde{\Sigma}$ is reduced to:

$$
\begin{aligned}
{}^{i}\tilde{\Sigma}(k) = &\left({}^{i}\tilde{\mathcal{O}}(k)\right)^{-1} + \left({}^{i}\tilde{\mathcal{O}}(k)\right)^{-1}\left(\sum_{t=k-N}^{k-1}\left(\sum_{s=k-N}^{t}\left(\mathcal{A}^{s-k}\right)^{\mathrm{T}}\right.\right.\\
&\times\left.\left(\mathcal{C}_i^{\mathrm{T}}\mathcal{C}_i\otimes I_n\right)\left(I_N\otimes R_i^{-1}\right)\left(\mathcal{C}_i^{\mathrm{T}}\mathcal{C}_i\otimes I_n\right)\mathcal{A}^{s-k}\right)\\
&\times\mathcal{A}^{k-j-1}\left(\mathcal{Q}+\mathcal{R}\right)\left(\mathcal{A}^{k-j-1}\right)^{\mathrm{T}}\left(\sum_{s=k-N}^{t}\left(\mathcal{A}^{s-k}\right)^{\mathrm{T}}\right.\\
&\times\left.\left.\left(\mathcal{C}_i^{\mathrm{T}}\mathcal{C}_i\otimes I_n\right)\left(I_N\otimes R_i^{-1}\right)\left(\mathcal{C}_i^{\mathrm{T}}\mathcal{C}_i\otimes I_n\right)\mathcal{A}^{s-k}\right)\right)\\
&\times\left({}^{i}\tilde{\mathcal{O}}(k)\right)^{-1}
\end{aligned}
\tag{49}
$$

Clearly, the individual terms in (49) are all finite and their sum yields the bounded covariance ${}^{i}\tilde{\Sigma}(k)$, $\forall k > N$. Then, due to the sub-optimality of (48), ${}^{i}\Sigma(k) \preceq {}^{i}\tilde{\Sigma}(k)$, $\forall k$. Therefore, there exists an upper bound of ${}^{i}\Sigma$ due to the observability condition of the system (42). Similar discussions on the estimation boundedness proof can be found in [46], [47] ■

Now we are ready to show the stability for the total MAS estimation error of our estimation algorithm.

Theorem 1. Given the multi-agent dynamics (1), the control protocol (3), and the sensing-based network topology G along with the location of the local monitoring agent i in the MAS network, the proposed sensing-based distributed state estimation algorithm is globally asymptotically stable in the sense of Lyapunov if the system (42) is observable.

Proof. Let us consider the Lyapunov function $\mathcal{V}:\mathbb{R}^{nN}\times\mathbb{N}\rightarrow\mathbb{R}$ of the estimation error defined as follows:

$$
\begin{aligned}
\mathcal{V}\left({}^{i}e(k),k\right) &:= {}^{i}e^{\mathrm{T}}(k)\left({}^{i}\Sigma(k)\right)^{-1}{}^{i}e(k)\\
\Delta\mathcal{V}\left(k+1,k\right) &:= \mathcal{V}\left(\mathbb{E}[{}^{i}e(k+1)|{}^{i}e(k)],k+1\right)\\
&\qquad -\mathcal{V}\left({}^{i}e(k),k\right)
\end{aligned}
\tag{50}
$$

From Lemma 1, ${}^{i}\Sigma(k)$ is positive definite and bounded, and this is equivalent to $\left({}^{i}\Sigma(k)\right)^{-1}\succ\mathbf{0}$, $\forall k$. Therefore, $\mathcal{V}$ is a quadratic function and is satisfying the positive definiteness condition (32). On the other hand, applying (41) to $\Delta\mathcal{V}$ gives:

$$
\begin{aligned}
\Delta\mathcal{V}\left(k+1,k\right) &= \mathbb{E}[{}^{i}e(k+1)|{}^{i}e(k)]^{\mathrm{T}}\left({}^{i}\Sigma(k+1)\right)^{-1}\\
&\quad\times\mathbb{E}[{}^{i}e(k+1)|{}^{i}e(k)] - {}^{i}e^{\mathrm{T}}(k)\left({}^{i}\Sigma(k)\right)^{-1}{}^{i}e(k)\\
&= {}^{i}e^{\mathrm{T}}(k)\left(\mathcal{A}^{\mathrm{T}}\mathcal{F}_i^{\mathrm{T}}(k+1)\left({}^{i}\Sigma(k+1)\right)^{-1}\right.\\
&\quad\left.\times\mathcal{F}_i(k+1)\mathcal{A}-\left({}^{i}\Sigma(k)\right)^{-1}\right){}^{i}e(k)
\end{aligned}
\tag{51}
$$

From (44) and the definition of $\mathcal{F}_i$, ${}^{i}\Sigma(k+1)$ can be obtained by:

$$
{}^{i}\Sigma(k+1) = \mathcal{F}_i(k+1){}^{i}\Sigma^{-}(k+1) \tag{52}
$$

And using (45), multiplying ${}^{i}\Sigma(k+1)$ to the left and the right sides of $\left({}^{i}\Sigma(k+1)\right)^{-1}$ yields:

$$
\begin{aligned}
{}^{i}\Sigma(k+1) &= {}^{i}\Sigma(k+1)\left({}^{i}\Sigma^{-}(k+1)\right)^{-1}{}^{i}\Sigma(k+1)\\
&+{}^{i}\Sigma(k+1)\left(\mathcal{C}_i^{\mathrm{T}}\otimes I_n\right)\left(\left(\mathcal{C}_i\otimes I_n\right)\left(I_N\otimes R_i\right)\left(\mathcal{C}_i^{\mathrm{T}}\otimes I_n\right)\right)^{-1}\\
&\times\left(\mathcal{C}_i\otimes I_n\right){}^{i}\Sigma(k+1)
\end{aligned}
\tag{53}
$$

Plugging (52) to (53), we have:

$$
\begin{aligned}
{}^{i}\Sigma(k+1) &= \mathcal{F}_i(k+1){}^{i}\Sigma^{-}(k+1)\mathcal{F}_i^{\mathrm{T}}(k+1)+\mathcal{F}_i(k+1)\\
&\times{}^{i}\Sigma^{-}(k+1)\left(\mathcal{C}_i^{\mathrm{T}}\otimes I_n\right)\left(\left(\mathcal{C}_i\otimes I_n\right)\left(I_N\otimes R_i\right)\right.\\
&\times\left.\left(\mathcal{C}_i^{\mathrm{T}}\otimes I_n\right)\right)^{-1}\left(\mathcal{C}_i\otimes I_n\right){}^{i}\Sigma^{-}(k+1)\mathcal{F}_i^{\mathrm{T}}(k+1)
\end{aligned}
\tag{54}
$$

which can be compactly written as:

$$
{}^{i}\Sigma(k+1) = \mathcal{F}_i(k+1)\left({}^{i}\Sigma^{-}(k+1)+\mathcal{W}(k+1)\right)\mathcal{F}_i^{\mathrm{T}}(k+1) \tag{55}
$$

where

$$
\begin{aligned}
\mathcal{W}(k+1) &:= {}^{i}\Sigma^{-}(k+1)\left(\mathcal{C}_i^{\mathrm{T}}\otimes I_n\right)\left(\left(\mathcal{C}_i\otimes I_n\right)\left(I_N\otimes R_i\right)\right.\\
&\times\left.\left(\mathcal{C}_i^{\mathrm{T}}\otimes I_n\right)\right)^{-1}\left(\mathcal{C}_i\otimes I_n\right){}^{i}\Sigma^{-}(k+1)
\end{aligned}
$$

It is obvious $\mathcal{W}(k+1)\succeq\mathbf{0}$, $\forall k$ from the above definition. Recalling ${}^{i}\Sigma^{-}(k+1)$ presented in (43), $\left({}^{i}\Sigma(k+1)\right)^{-1}$ can be rewritten by:

$$
\begin{aligned}
\left({}^{i}\Sigma(k+1)\right)^{-1} &= \left(\mathcal{F}_i^{\mathrm{T}}(k+1)\right)^{-1}\left(\mathcal{A}{}^{i}\Sigma(k)\mathcal{A}^{\mathrm{T}}+\mathcal{Q}+\mathcal{R}\right.\\
&\left.+\mathcal{W}(k+1)\right)^{-1}\left(\mathcal{F}_i(k+1)\right)^{-1}
\end{aligned}
\tag{56}
$$

and applying this to (51) results in:

$$
\begin{aligned}
\Delta\mathcal{V}\left(k+1,k\right) &= -{}^{i}e^{\mathrm{T}}(k)\left(\left({}^{i}\Sigma(k)\right)^{-1}-\mathcal{A}^{\mathrm{T}}\left(\mathcal{A}{}^{i}\Sigma(k)\mathcal{A}^{\mathrm{T}}\right.\right.\\
&\left.\left.+\mathcal{Q}+\mathcal{R}+\mathcal{W}(k+1)\right)^{-1}\mathcal{A}\right){}^{i}e(k)\\
&= -{}^{i}e^{\mathrm{T}}(k){}^{i}\mathcal{U}(k+1){}^{i}e(k)
\end{aligned}
\tag{57}
$$

where

$$
\begin{aligned}
{}^{i}\mathcal{U}(k+1) &:= \left({}^{i}\Sigma(k)\right)^{-1}-\mathcal{A}^{\mathrm{T}}\left(\mathcal{A}{}^{i}\Sigma(k)\mathcal{A}^{\mathrm{T}}+\mathcal{Q}+\mathcal{R}\right.\\
&\left.+\mathcal{W}(k+1)\right)^{-1}\mathcal{A}
\end{aligned}
$$

What follows is to check the condition (33) in (57). Multiplying ${}^{i}\Sigma(k)$ to the left and the right sides of ${}^{i}\mathcal{U}(k+1)$ gives:

$$
\begin{aligned}
{}^{i}\Sigma(k){}^{i}\mathcal{U}(k+1){}^{i}\Sigma(k) &= {}^{i}\Sigma(k)-{}^{i}\Sigma(k)\mathcal{A}^{\mathrm{T}}\left(\mathcal{A}{}^{i}\Sigma(k)\mathcal{A}^{\mathrm{T}}\right.\\
&\left.+\mathcal{Q}+\mathcal{R}+\mathcal{W}(k+1)\right)^{-1}\mathcal{A}{}^{i}\Sigma(k)
\end{aligned}
\tag{58}
$$

and using the matrix inversion lemma, we get:

$$
\begin{aligned}
{}^{i}\Sigma(k){}^{i}\mathcal{U}(k+1){}^{i}\Sigma(k) &= \left(\left({}^{i}\Sigma(k)\right)^{-1}+\mathcal{A}^{\mathrm{T}}\left(\mathcal{Q}+\mathcal{R}\right.\right.\\
&\left.\left.+\mathcal{W}(k+1)\right)^{-1}\mathcal{A}\right)^{-1}
\end{aligned}
\tag{59}
$$

Lastly, multiplying $\left({}^{i}\Sigma(k)\right)^{-1}$ back to the both sides of (59)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2018.2867341, IEEE Transactions on Automatic Control

returns:

$$
{}^{i}\mathcal{U}(k+1) = \left({}^{i}\Sigma(k)\right)^{-1}\left(\left({}^{i}\Sigma(k)\right)^{-1} + \mathcal{A}^{\mathrm{T}}\left(\mathcal{Q}+\mathcal{R}\right.\right.
$$
$$
\left.\left. +\mathcal{W}(k+1)\right)^{-1}\mathcal{A}\right)^{-1}\left({}^{i}\Sigma(k)\right)^{-1} \tag{60}
$$

Since $\left({}^{i}\Sigma(k)\right)^{-1} \succ \mathbf{0}$ and $\mathcal{Q}+\mathcal{R}+\mathcal{W}(k+1) \succ \mathbf{0}$ in (60), one can verify ${}^{i}\mathcal{U}(k+1) \succ \mathbf{0}$, $\forall k$, which guarantees $\Delta\mathcal{V} < 0$ all the time. This completes the proof of the estimation stability by the existence of $\mathcal{V}$ satisfying (32) and (33). ■

V. Application to Multi-Vehicle Formation Flight

In this section, the effectiveness of the proposed distributed estimation algorithm is demonstrated with an illustrative MAS example, a multi-vehicle formation flight scenario.

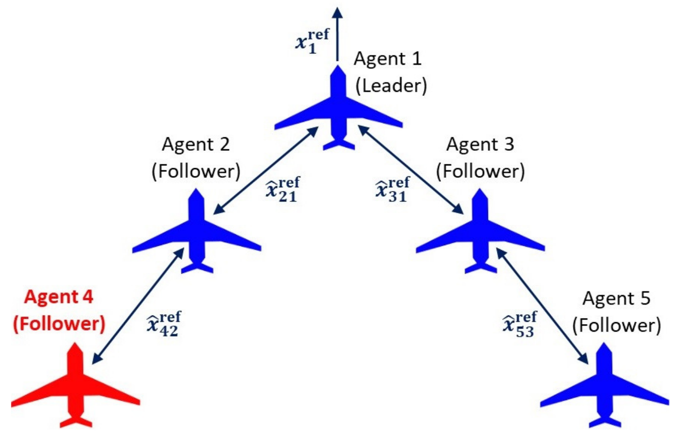


Fig. 2. Multi-Vehicle Formation Flight Configuration

A. MAS Model

For the simulation, we focus on the position of each vehicle in two dimensions, denoted as X and Y measured in meters. Each vehicle state is represented by the vector $x = \left[X\ \dot{X}\ Y\ \dot{Y}\right]^{\mathrm{T}}$ and is governed by the dynamics in (1), where

$$
A = \begin{bmatrix} 1 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{bmatrix}, B = \begin{bmatrix} \frac{T_s^2}{2} & 0 \\ T_s & 0 \\ 0 & \frac{T_s^2}{2} \\ 0 & T_s \end{bmatrix}
$$

where T_s denotes the sampling time, set to 1 second. In the following simulations, "time step k" for integer k refers to the time kT_s, which for $T_s = 1$ second is simply k seconds. All the disturbances and sensor noises for each vehicle are assumed to be i.i.d. zero mean Gaussian distributions with identity covariance matrices, i.e., $Q_i = I_4$ and $R_i = I_4$ respectively [48]. Given this linear Gaussian process, it is mathematically straightforward that any linear state estimator, including the proposed one, retains the Gaussian distribution of the state estimate, satisfying Assumption 2.

The formation flight mission considered in this paper is composed of five vehicle agents as shown in Figure 2, where agent 1 is the leader and agents 2-5 are the followers. The leader has its own reference command to guide the flight path and the followers' control protocol is to maintain a rigid V-formation while following the leader. This cooperation objective can be achieved by close interactions between the neighboring vehicles, including both leader-follower and follower-follower.

Accordingly, the cooperative control protocols are formulated as follows:

$$
\begin{cases} u_1(k) = K\left(x_1(k) - x_1^{\mathrm{ref}}(k)\right) \\ \qquad\qquad + K\displaystyle\sum_{j\in\Omega_1}\left(x_1(k) - x_j(k) - x_{1j}^{\mathrm{ref}}\right) \\ u_i(k) = K\displaystyle\sum_{j\in\Omega_i}\left(x_i(k) - x_j(k) - x_{ij}^{\mathrm{ref}}\right), \quad \forall i \in \{2,\cdots,5\} \end{cases}
$$

where x_1^{ref} is the desired reference state trajectory applied to agent 1 (the leader); and x_{ij}^{ref} $\forall i \in V = \{1,\cdots,5\}$ is the desired relative position and velocity of agent i with respect to the neighboring agent j. Additionally, the feedback control gain is designed by [11] such that it stabilizes the overall formation dynamics:

$$
K = \begin{bmatrix} -0.2263 & -0.4712 & 0 & 0 \\ 0 & 0 & -0.2263 & -0.4712 \end{bmatrix}
$$

Note that the information regarding the vehicle dynamics, formation network, cooperative control protocols and reference control input can be initially uploaded to individual vehicles without further patching via communication in the middle of flight, as indicated by Assumption 1

Under some mission scenarios where the communication is costly or vulnerable (for example, the multi-vehicle surveillance over an adversarial area that prohibits the communication between vehicles due to the possibility of eavesdropping [31]), we alternatively consider on-board vehicle sensors whose sensor measurements are used for achieving the cooperative formation controls between each agent. Then, the above true state-based feedback control is transformed into the following sensor observation-based control protocol control:

$$
\begin{cases} u_1(k) = \left(1+|\Omega_1|\right)Kz_{11}(k) + b_1(k) - K\displaystyle\sum_{j\in\Omega_1} z_{1j}(k) \\ u_i(k) = |\Omega_i|Kz_{ii}(k) + b_i(k) - K\displaystyle\sum_{j\in\Omega_i} z_{ij}(k) \end{cases}
$$

where $b_1(k) = -Kx_1^{\mathrm{ref}}(k) - K\sum_{j\in\Omega_1} x_{1j}^{\mathrm{ref}}$ and $b_i(k) = -K\sum_{j\in\Omega_i} x_{ij}^{\mathrm{ref}}$, $\forall i \in \{2,\cdots,5\}$. Clearly, the resulting control protocols are well-matched to the structure of (5). The on-board sensors for individual agents, however, have limited sensing ranges as shown in Figure 3 where the circles represent the sensing ranges and the arrows indicate the available observations. For example, agent 4, to be considered as a monitoring agent in the following simulations, is only able to observe the state of agent 2 and there is no direct observation to the out-of-range agents (agents 1, 3, and 5) making them difficult to estimate.

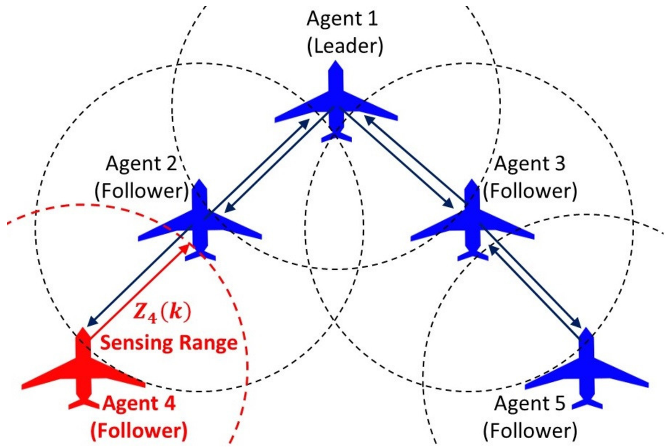


Fig. 3. On-board Sensor Observations for Multi-Vehicle Formation Flight

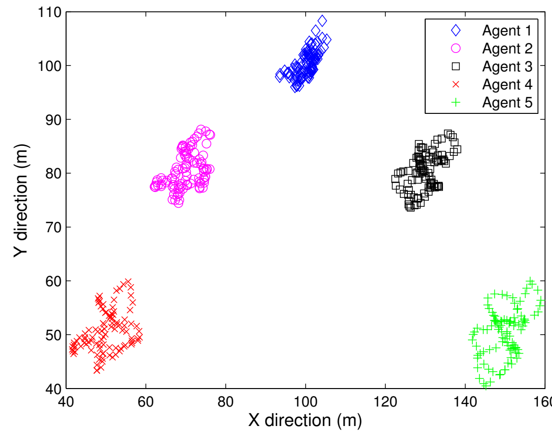


Fig. 4. Traces of Relative Vehicle Agent Positions along the Reference Trajectory Coordinate

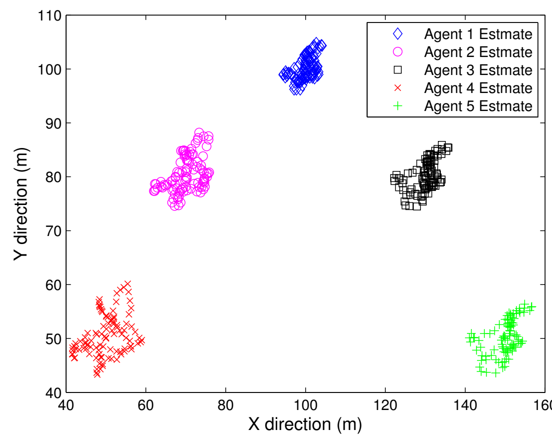


Fig. 5. Proposed Sensing-based Distributed Estimates from the Observation of Agent 4

B. Estimation Stability Test via Theorem 1

Intuitively, the more observations a local monitor can acquire, the better performance we can expect for our algorithm toward the entire MAS state estimation. Theorem 1 at the minimum allows for checking the estimation stability for the given formation flight network topology in advance of the actual simulation. Specifically, the above vehicle sensing-based network along with the formation control protocols form the following Laplacian matrix:

$$\mathcal{L} = \begin{bmatrix} 3 & -1 & -1 & 0 & 0 \\ -1 & 2 & 0 & -1 & 0 \\ -1 & 0 & 2 & 0 & -1 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix}$$

and the graph observation matrix for agent 4 is designed as:

$$\mathcal{C}_4 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

For the matrix pair $(\mathcal{C}_4, \mathcal{L})$, the constructed observability matrix is of full rank, implying the corresponding system (42) is fully observable. Consequently, it is guaranteed by Theorem 1 that the proposed algorithm stably performs the distributed state estimation for this formation flight example.

C. Simulation Results

The state of each vehicle agent in the simulations is randomly initialized using the given Gaussian distributions. The cooperative control protocol, subject to the stochastic dynamics, then regulates the agents' states according to the specified formation shape between the agents while the leader agent follows the desired reference trajectory. Figure 4 represents the traces of relative vehicle positions along the reference trajectory over 100 seconds simulation time.

From the perspective of agent 4, the proposed sensing-based estimation algorithm tracks the position of each vehicle agent, both inside the sensing range (i.e., agents 2 and 4) and outside the sensing range (i.e., agents 1, 3, and 5) as shown in Figure 5.

Without direct access, our distributed estimation algorithm can indirectly estimate the state of the out-of-range agents based on the sensing network topology and cooperative control protocol information. Figure 6 presents the root mean square (RMS) estimation errors of the individual agent estimates from agent 4's observation. The errors are averaged from Monte Carlo simulations with 100 runs of 100 seconds each. Throughout the simulation, it is demonstrated that the proposed sensing-based distributed state estimation algorithm successfully performs the state estimation for each agent even beyond the sensing range which leads to the convergence (or, stochastic boundedness) of estimation error.

We note that the estimation errors of individual agents are dependent on their correlations with the observed agents. In particular, the correlation between the local measurements and the target agent state is taken into account in the update step of our estimation algorithm. This correlation strongly relies on the dynamical interactions between the target agent and the observed agents. In this simulation example, the closer interaction of agent 1 (target) with agent 2 (observed) results in higher correlation with local measurements of (monitoring)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2018.2867341, IEEE Transactions on Automatic Control

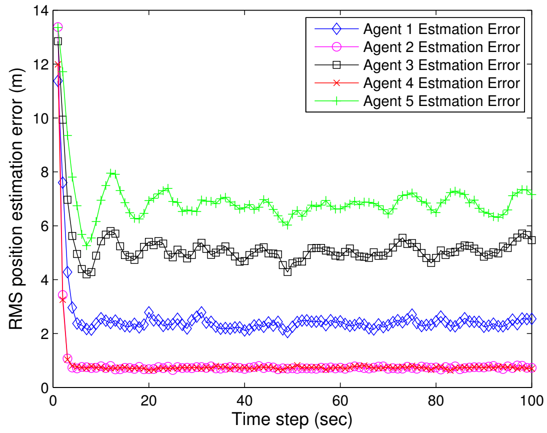


Fig. 6. Root Mean Square (RMS) Estimation Errors for Each Agent Estimate from the Perspective of Agent 4 (Monte Carlo Simulations with 100 Runs)

agent 4, generating smaller estimation errors than those that have less interactions with agent 2, i.e., agents 3 and 5. Figure 7 presents the estimation error ellipses for the state estimates of individual agents 1∼5 from the perspective of agent 4, providing more insight on the statistical confidence of each estimation. Here, the individual error ellipse corresponds to the confidence level of 95% that the actual estimation errors are within the ellipse region. Clearly, the farther the target gets from the monitoring agent, the bigger the estimation error ellipse (agent 4 $<$ agent 2 $<$ agent 1 $<$ agent 3 $<$ agent 5), as we would expect. Further theoretical assessment on the measurements' correlation in terms of the agents' interaction is interesting work and will be included in our future research.

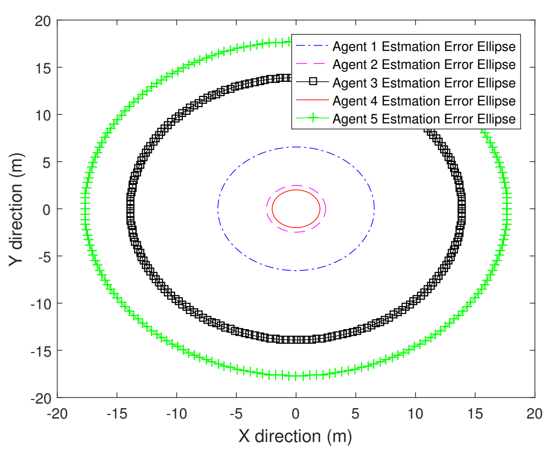


Fig. 7. Position Estimation Error Ellipse with the Confidence of 95%

To further demonstrate the performance of the proposed algorithm, we have performed additional simulations under the following different disturbance and sensor noise scenarios whose results are shown in Figures 8 and 9, respectively:

- Case 1: Sensor noise covariance for agent 5 (R_5) is bigger than all the other agents' disturbance and noise covariances
- Case 2: Disturbance covariance for agent 2 (Q_2) is bigger than all the other agents' disturbance and noise covariances

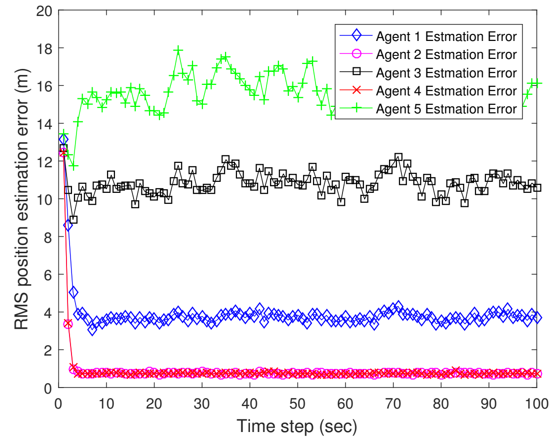


Fig. 8. RMS Estimation Errors from the Perspective of Agent 4 when $R_5 = 100I_4$ while Others are I_4 (Monte Carlo Simulations with 100 Runs)

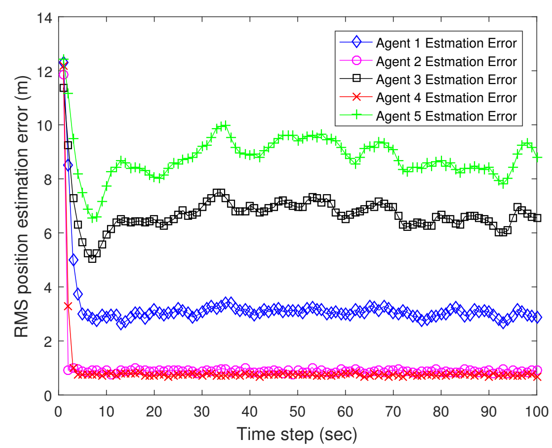


Fig. 9. RMS Estimation Errors from the Perspective of Agent 4 when $Q_2 = 100I_4$ while Others are I_4 (Monte Carlo Simulations with 100 Runs)

In Case 1, it is observed that the position estimation error for agent 5 is increased approximately 2 times (∼8m → ∼16m) as a consequence of the 100 times bigger noise covariance of sensor measurement for agent 5 ($R_5 = 100I_4$). Furthermore, increasing the sensor noise of agent 5 influences the estimation accuracy for the other agents, especially degrading the estimation performance for agent 3 neighboring agent 5. This is mainly attributed to the inter-agent interactions that are based on noisy sensor observations, providing inaccurate information to the proposed estimation algorithm. In Case 2, the estimation error for agent 2 is not significantly increased under the 100 times bigger disturbance ($Q_2 = 100I_4$) since agent 2 is within the sensing-range of agent 4 and thus directly observable. But the estimation accuracy for the other out-range agents degrades due to the disturbed inter-agent interactions, similar to Case 1.

TABLE I
PERFORMANCE COMPARISON BETWEEN 'GEOMETRY-BASED' LOCAL KALMAN FILTERING, PROPOSED 'SENSING-BASED' DISTRIBUTED ESTIMATION, AND 'COMMUNICATION-BASED' KALMAN CONSENSUS FILTERING WITHIN FULLY CONNECTED NETWORK

Distributed Estimation Algorithms / Performance Measures	Geometry-based [49]	Sensing-based	Communication-based [17]
Average RMS Estimation Error for Each Local Monitoring Agent	6.1449 m	3.3394 m	0.9557 m
Total Sensing Loads over Network (# of Sensor Measurements)	13 per time step (Local Observations)	13 per time step (Local Observations)	5 per time step (Agent Itself Only)
Total Communication Loads over Network (# of Data Packet Transmissions)	0 per time step (No Communication)	0 per time step (No Communication)	40 per time step (Fully Connected)

For a comparative analysis, we carry out two other estimation methods performing the same simulation scenario. The first method is essentially equivalent to a local Kalman filtering, i.e., each agent only processes its own sensor observation without communicating with other agents [49]. In the simulation, we supplement the local Kalman filter with the geometry information of the flight formation, known to each agent, which is called the 'geometry-based' approach. By this way, each vehicle can infer the positions of out-of-sensing range vehicles by reconstructing the formation geometry based on the local Kalman filtering estimate of its own position. Similar to the proposed sensing-based distributed estimation, the geometry-based method does not rely on communication. However, using the geometry information alone cannot not rigorously account for the dynamical interactions between the agents, while our proposed estimation algorithm uses 'cooperative control protocol' information to account for the dynamical interactions between the agents. This critical difference significantly improves the estimation performance of our proposed algorithm over the geometry-based algorithm as shown in the simulation results. Figure 10 compares the average RMS estimation errors of all the agents from a local monitoring agent (agent 4 in this case) using different methods, where the geometry-based approach is shown to cause larger errors than our sensing-based distributed state estimation algorithm.

Next, the 'communication-based' method we consider is indeed the Kalman consensus filtering, which is one of the most common distributed Kalman filtering techniques [18]. In the simulation, the standard Kalman consensus filter proposed in [17] is implemented to each vehicle within a fully connected communication network such that each agent locally estimates its own state and error covariance, and transmits them to other agents through the communication link. Therefore, the state estimate from any agent in the MAS can be directly accessed, which leads to small estimation errors as shown in Figure 10.

Table I presents the performance comparison between the proposed sensing-based, the geometry-based, and the conventional communication-based estimation algorithms applied to the example simulation scenario of multi-vehicle formation flight. The performance measures are the communication load as well as the estimation error, both of which are desired to be reduced in practice. At the cost of some estimation error, our sensing-based distributed estimation algorithm provides the

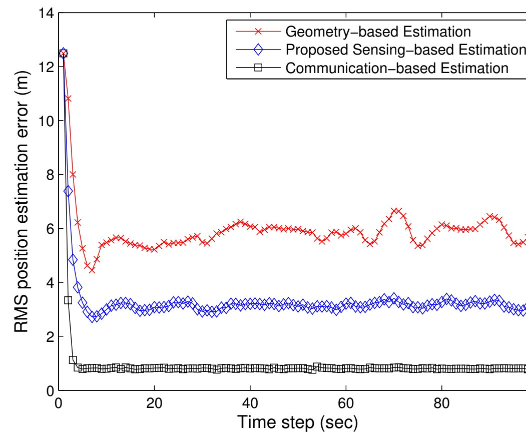


Fig. 10. Average RMS Estimation Error Comparison between: i) Geometry-based; ii) Proposed Sensing-based; and iii) Communication-based Distributed State Estimation (Monte Carlo Simulations with 100 Runs)

clear advantage of reducing energy consumption in data transmission and reduces the potential risks of data compromise, taking care of the vulnerability of the existing communication-based methods.

VI. CONCLUSION

This paper has considered distributed estimation for cooperative Multi-Agent Systems (MASs), which has been mostly studied using communication-based information exchange between individual agents. In seeking to address the cost and vulnerability issues inherent in the communication network, a sensing-based estimation approach was considered where each agent processes only local observations using on-board sensors. Specifically, we have proposed a sensing-based distributed state estimation algorithm to address the limited sensing range of a local monitoring agent and extend its estimation scope beyond the sensing range without needing communication. Leveraging the cooperative control protocol information over the sensor observations, the proposed algorithm recursively predicts and updates the out-of-range agent state estimates by examining their possible interactions with the observed behavior of the in-range agents. Further theoretical analysis has derived the conditions for the sensing network topology of the MAS to assure the stability of the proposed algorithm, and numerical simulation has demonstrated the performance of the algorithm in comparison to other methods. Our approach has been shown to provide the advantage of augmenting distributed estimation capability by explicitly considering the dynamical interactions between agents instead of through a communication network. The augmented sensing-based estimation could be used as either redundant back-up of communication-based estimation methods or a stand-alone distributed state estimation algorithm in communication-denied environments, leading to more reliable awareness of the MAS. Future work includes: i) scalability analysis for large-scale MAS estimation and further approximation for computationally more efficient implementation; ii) sensing-based estimation subject to time-varying network topology that likely happens in mobile MAS applications; iii) the development of a hybrid scheme where the existing communication-based and the proposed sensing-based approaches are intelligently fused in a way that maximizes the estimation performance while minimizing the communication overhead; and iv) integration of the proposed estimation algorithm into the cooperative control protocol to facilitate the MAS coordination.

REFERENCES

[1] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 427–438, Feb. 2013.

[2] R. Murray, "Recent research in cooperative control of multivehicle systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, no. 5, pp. 571–583, Sep. 2007.

[3] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[4] S. Knorn, Z. Chen, and R. Middleton, "Overview: Collective control of multiagent systems," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 4, pp. 334–347, Dec. 2016.

[5] B. Sinopoli, C. Sharp, L. Schenato, S. Schaffert, and S. Sastry, "Distributed control applications within sensor networks," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1235–1246, Aug. 2003.

[6] C. Chong and S. Kumar, "Sensor networks: evolution, opportunities, and challenges," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, Aug. 2003.

[7] J. Baillieul and P. Antsaklis, "Control and communication challenges in networked real-time systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 9–28, Jan. 2007.

[8] E. Semsar-Kazerooni and K. Khorasani, "Optimal consensus algorithms for cooperative team of agents subject to partial information," *Automatica*, vol. 44, no. 11, pp. 2766–2777, Nov. 2008.

[9] Z. Lin, M. Broucke, and B. Francis, "Local control strategies for groups of mobile autonomous agents," *IEEE Transactions on Automatic Control*, vol. 49, no. 4, pp. 622–629, Apr. 2004.

[10] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, Mar. 2006.

[11] J. Fax and R. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, Sep. 2004.

[12] L. Doherty, K. Pister, and L. Ghaoui, "Convex position estimation in wireless sensor networks," in *Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*. IEEE, Apr. 2001, pp. 1655–1663.

[13] Y. Huang, S. Werner, J. Huang, N. Kashyap, and V. Gupta, "State estimation in electric power grid," *IEEE Signal Processing Magazine*, pp. 33–43, Sep. 2012.

[14] M. Daigle, X. Koutsoukos, and G. Biswas, "Distributed diagnosis in formations of mobile robots," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 353–369, Apr. 2007.

[15] G. Mateos and G. B. Giannakis, "Distributed recursive least-squares: Stability and performance analysis," *IEEE Transactions on Signal Processing*, vol. 60, no. 7, pp. 3740–3754, 2012.

[16] A. Nedić, A. Olshevsky, and C. A. Uribe, "Fast convergence rates for distributed non-bayesian learning," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 5538–5553, 2017.

[17] R. Olfati-Saber, "Kalman-consensus filter: optimality, stability, and performance," in *48th IEEE Conference on Decision and Control*. IEEE, Dec. 2009, pp. 7036–7042.

[18] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, "Distributed kalman filtering based on consensus strategies," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 622–633, May 2008.

[19] F. Cattivelli and A. Sayed, "Diffusion strategies for distributed kalman filtering and smoothing," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2069–2084, Sep. 2010.

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

[21] G. Battistelli, L. Chisci, G. Mugnai, A. Farina, and A. Graziano, "Consensus-based linear and nonlinear filtering," *IEEE Transactions on Automatic Control*, vol. 60, no. 5, pp. 1410–1415, 2015.

[22] S. Wang, W. Ren, and J. Chen, "Fully distributed dynamic state estimation with uncertain process models," *IEEE Transactions on Control of Network Systems*, 2017.

[23] X. Yang, W.-A. Zhang, M. Z. Chen, and L. Yu, "Hybrid sequential fusion estimation for asynchronous sensor network-based target tracking," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 2, pp. 669–676, 2017.

[24] R. Deshmukh, O. Thapliyal, C. Kwon, and I. Hwang, "Distributed state estimation for a stochastic linear hybrid system over a sensor network," *IET Control Theory & Applications*, 2018.

[25] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poola, and S. Sastry, "Foundations of control and estimation over lossy networks," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 163–187, Jan. 2007.

[26] V. Gupta, V. Hassibi, and R. Murray, "On sensor fusion in the presence of packet-dropping communication channels," in *44th IEEE Conference on Decision and Control*. IEEE, Dec. 2005, pp. 3547–3552.

[27] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.

[28] Y. Zhang, F. Li, and Y. Chen, "Leader-following-based distributed kalman filtering in sensor networks with communication delay," *Journal of the Franklin Institute*, vol. 354, no. 16, pp. 7504–7520, 2017.

[29] S. Fang, Y. Liu, and P. Ning, "Wireless communications under broadband reactive jamming attacks," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 3, pp. 394–408, May 2016.

[30] S. Weerakkody, X. Liu, S. Son, and B. Sinopoli, "A graph-theoretic characterization of perfect attackability for secure design of distributed control systems," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 1, pp. 60–70, Mar. 2017.

[31] R. Schaefer, H. Boche, and H. Poor, "Secure communication under channel uncertainty and adversarial attacks," *Proceedings of the IEEE*, vol. 103, no. 10, pp. 1796–1813, Oct. 2015.

[32] V. Shnayder, M. Hempstead, B. Chen, G. Werner-Allen, and M. Welsh, "Simulating the power consumption of large-scale sensor network applications," in *2nd International Conference on Embedded Networked Sensor Systems*. ACM, Nov. 2004, pp. 188–200.

[33] T. Li, M. Fu, L. Xie, and J. Zhang, "Distributed consensus with limited communication data rate," *IEEE Transactions on Automatic Control*, vol. 56, no. 2, pp. 279–292, Feb. 2011.

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

[35] S. Kar and J. Moura, "Gossip and distributed kalman filtering: weak consensus under weak detectability," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1766–1784, Apr. 2011.

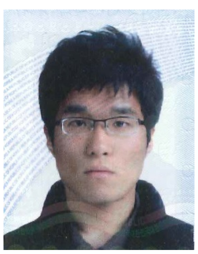
[36] Z.-G. Wu, Y. Xu, Y.-J. Pan, H. Su, and Y. Tang, "Event-triggered control for consensus problem in multi-agent systems with quantized relative state measurements and external disturbance," *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2018.

[37] L. Ding, Q.-L. Han, X. Ge, and X.-M. Zhang, "An overview of recent advances in event-triggered consensus of multiagent systems," *IEEE transactions on cybernetics*, vol. 48, no. 4, pp. 1110–1123, 2018.

[38] M. Guo, D. Dimarogonas, and K. Johansson, "Distributed real-time fault detection and isolation for cooperative multi-agent systems," in *American Control Conference*. IEEE, Jun. 2012, pp. 5270–5275.

[39] J. Chen, S. Kher, and A. Somani, "Distributed fault detection of wireless sensor networks," in *Proceedings of the 2006 workshop on Dependability issues in wireless ad hoc networks and sensor networks*. ACM, Sep. 2006, pp. 65–72.

[40] C. Kwon, D. Kun, and I. Hwang, "Augmented sensing-based state estimation for cooperative multi-agent systems," in *American Control Conference (ACC), 2015*. IEEE, 2015, pp. 3792–3797.
[41] J. Lin, A. Morse, and B. Anderson, "The multi-agent rendezvous problem," in *42th IEEE Conference on Decision and Control*. IEEE, Dec. 2003, pp. 1508–1513.
[42] W. Ren, "Consensus strategies for cooperative control of vehicle formations," *IET Control Theory and Applications*, vol. 1, no. 2, pp. 505–512, Mar. 2007.
[43] R. V. Hogg, J. W. McKean, and A. T. Craig, *Introduction to Mathematical Statistics*, 6th ed. Pearson Education, Inc., 2005.
[44] R. Bucy, "Stability and positive supermartingales," *Journal of Differential Equations*, vol. 1, no. 2, pp. 151–155, Apr. 1965.
[45] W. W. Hager, "Updating the inverse of a matrix," *SIAM Review*, vol. 31, no. 2, pp. 221–239, Jun. 1989.
[46] J. Deyst and C. Price, "Conditions for asymptotic stability of the discrete minimum-variance linear estimator," *IEEE Transactions on Automatic Control*, vol. 31, no. 6, pp. 702–705, Dec. 1968.
[47] J. Deyst, "Correction to "conditions for asymptotic stability of the discrete minimum-variance linear estimator"," *IEEE Transactions on Automatic Control*, vol. 18, no. 5, pp. 562–563, Dec. 1973.
[48] C. Kwon, W. Liu, and I. Hwang, "Analysis and design of stealthy cyber attacks on unmanned aerial systems," *AIAA Journal of Aerospace Information Systems*, vol. 11, no. 8, pp. 525–539, Aug. 2014.
[49] R. Olfati-Saber, "Distributed kalman filtering for sensor networks," in *Decision and Control, 2007 46th IEEE Conference on*. IEEE, 2007, pp. 5492–5498.



Cheolhyeon Kwon (S'12−M'17) received the B.S. degree in Mechanical and Aerospace engineering from Seoul National University, Seoul, Korea, in 2010, and the M.S. and Ph.D. degrees in the School of Aeronautics and Astronautics from Purdue University, West Lafayette, IN, in 2013 and 2017, respectively. He is currently a visiting assistant professor in the School of Aeronautics and Astronautics at Purdue University, West Lafayette, IN. His research interests include control and estimation for dynamical cyber-physical systems (CPS), along with networked autonomous vehicles, air traffic control systems, sensors and communication networks. Specifically, his recent work aims at the cyber-secure and high assurance CPS design, inspired by control and estimation theory perspective, with applications to aerospace systems such as unmanned aircraft systems (UAS).

Inseok Hwang received the Ph.D. degree in Aeronautics and Astronautics from Stanford University, and is currently a professor in the School of Aeronautics and Astronautics at Purdue University. His research interests lie in modeling, estimation, and control of cyber-physical systems (CPS) and their applications to safety critical systems such as aircraft/spacecraft/unmanned aerial systems (UAS), air traffic management, and multi-agent systems. For his research, he leads the Flight Dynamics and Control/Hybrid Systems Laboratory at Purdue University. He received the NSF CAREER award in 2008, was selected as one of the nations brightest young engineers by the National Academy of Engineering (NAE) in 2008, and received the AIAA Special Service Citation in 2010. He is an Associate Fellow of AIAA, and a member of the IEEE Control Systems Society and the IEEE Aerospace and Electronic Systems Society. He is currently an associate editor of the IEEE Transactions on Aerospace and Electronic Systems and the Asian Journal of Control.