Differentially Private Consensus With an Event-Triggered Mechanism

Lan Gao, Shaojiang Deng, and Wei Ren, *Fellow, IEEE*

***Abstract*—This paper studies the differentially private consensus problem of multiagent networks by employing a distributed event-triggered mechanism such that not only agents can protect the privacy of their initial states from information disclosure, but the execution efficiency of the whole network can be improved. First, we propose a distributed event-triggered mechanism for a differentially private consensus algorithm such that frequent real-time communication and controller updates can be avoided. Second, we propose a distributed event-triggering condition that only depends on local information and local parameters, which can effectively avoid global information collection. Third, the convergence analysis of the mean-square average consensus is given to explain the sufficiency of the proposed event-triggered mechanism and event-triggering condition. Furthermore, we establish the statistic properties of the convergent accuracy that the expectation of the convergence point converges to the average value of all agents' initial states exactly and the disturbance variance is bounded with an explicit expression. In addition, we further give the differential privacy analysis that each agent can flexibly select its own privacy level to prevent information disclosure. Finally, simulation results are given to illustrate the effectiveness of the proposed mechanism and the correctness of the theoretical results.**

***Index Terms*—Differentially private consensus, distributed event-triggered mechanism, multiagent network.**

I. Introduction

The CONSENSUS and cooperation problems of multiagent networks have received increasing attention in recent years from various fields including multirobot coordination [1]–[3]; distributed filtering and estimation [4], [5]; sensor fusion [6], [7]; feature-based map merging [8], [9]; and distributed tracking [10]–[12]. Generally, a consensus algorithm requires agents to share their individual states with their neighbors and, in some cases, even their local inputs [13]. This might be very dangerous for privacy disclosure because if some malicious adversaries are able to listen to the exchanged messages, then they could infer local inputs, individual states, sensitive responses, and even the final agreement value of the network. In light of these scenarios, the requirement of the privacy preservation poses a new challenge in the consensus study of multiagent networks.

From the viewpoint of privacy, the participating agents may not want to disclose their initial or current state values while communicating with each other to reach an agreement. For example, a group of agents might want to rendezvous at a certain location while keeping their initial locations secret to others due to some particular reasons. In another practical scenario, a group of individuals might want to vote for a common decision on some subject while they do not want to reveal their exact personal opinions in the meantime [14].

In the context of privacy preservation, the notion of the differential privacy first introduced in [15] has gained significant popularity due to its rigorous formulation and proven security properties, including the resilience to postprocessing and side information, and the independence from the models of adversaries [16]. Based on the work under continual observation [17], the notion of the differential privacy is introduced into the average consensus study of multiagent networks [18]. Then, relevant problems including filtering estimation [19] and distributed optimization [20], [21] have been further studied in recent years. Specifically, an interesting result that the exact average consensus cannot be accomplished under the differential privacy mechanism is established in [22], where a more accurate convergence point in expectation and a distributed privacy level for each agent are given. Then, the convex optimization problem of the differential privacy is further discussed in [23] and [24].

Regarding the execution efficiency of the network evolution, the aforementioned differential privacy consensus algorithms have a common deficiency: real-time communication and controller updates. In other words, each agent has to collect its neighbors' information and actuating its controller updates every time instant, which may be infeasible for agents only equipped with small and capability-limited embedded microprocessors [25]. Therefore, how to design and develop a proper control strategy to avoid real-time communication becomes a new challenge for researchers.

The advent of the event-triggered mechanism offers a new viewpoint on how information should be collected and transmitted [26]. Under an event-triggered mechanism, an agent transmits its local state to its neighbors only when it is nec-

Manuscript received July 15, 2017; revised November 16, 2017; accepted January 4, 2018. Date of publication January 23, 2018; date of current version March 14, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61672119; in part by the Graduate Scientific Research and Innovation Foundation of Chongqing, China, under Grant CYB16045; in part by the National Science Foundation under Grant CMMI-1537729; and in part by the 111 Project (B17048). Recommended by Associate Editor L. Shi. *(Corresponding author: Wei Ren.)*

L. Gao and S. Deng are with the College of Computer Science, Chongqing University, Chongqing 400044, China (e-mail: langao@ucr.edu; sj_deng@cqu.edu.cn).

W. Ren is with the Department of Electrical and Computer Engineering, University of California, Riverside, CA 92521 USA (e-mail: ren@ece.ucr.edu).

Digital Object Identifier 10.1109/TCNS.2018.2795703

essary, that is, only when a measurement of the local agent's state error reaches a specified threshold [27]. A novel event-triggering condition based on the norms of the state and the state error is presented in [28], where the measurement received at the controller is held constant until a new measurement arrives. When this happens, the error is set to zero and starts increasing until it triggers a new measurement update. Obviously, the real-time communication and controller updates are avoided, and then the execution efficiency will be largely improved [26]. The event-triggered control strategy was first implemented into the study of the average consensus of multiagent networks in [29] and [30]. From then on, a growing large number of research results on discrete-time systems [31]; leader–follower consensus [32], [33]; trajectory tracking [34], [35]; quantized sampled-data communication [36], [37]; observer-based feedback [38], [39]; and second-order dynamics [40], [41] have been published in recent years.

In this paper, we focus on improving the execution efficiency of differentially private consensus by utilizing the event-triggered mechanism in order to effectively avoid frequent real-time communication and controller updates. Note that the interplay of these two ideas is not trivial. As mentioned previously, all of the existing works on the differential privacy consensus assume that the communication and controller updates are real time and continuous. Also, the existing works on the event-triggered consensus only focus on the common consensus problem without accounting for the individual privacy preservation. To achieve the differential privacy consensus under an event-triggered mechanism, the existing works cannot be applied directly and there are several new significant challenges that need to be overcome. The biggest challenge is how to design a communication algorithm based on an event-triggered strategy such that the convergent accuracy and the differential privacy can be well preserved. Furthermore, some statistic characteristics should be employed to describe the stochastic convergence process due to the existence of Laplacian random noise. In addition, the measurement error and the event-triggering condition need to be redesigned under the consideration of the Laplacian random noise and the preselected privacy level. Finally, to avoid global information collection, a distributed event-triggering condition that only depends on local information and local parameters should be designed.

The main contribution of this paper is to design a novel communication algorithm that successfully combines the benefits of the differential privacy consensus and the event-triggered mechanism such that not only can the individual privacy be well preserved but the execution efficiency of the whole network can be largely improved. More specifically, we first propose a distributed event-triggered algorithm for the differential privacy consensus such that information collection and exchange depend on nonperiodic sporadic sampling instead of real-time sampling. Second, we propose a distributed event-triggering condition that only depends on local information and local parameters, which can effectively avoid global information collection. Third, the convergence analysis of the mean-square average consensus is given to explain the sufficiency of the proposed algorithm and event-triggering condition. Furthermore, since the transmitted messages are corrupted with Laplacian random noise to achieve the differential privacy, we establish the statistic properties of the convergent accuracy that the expectation of the convergence point converges to the average value of all agents' initial states exactly, and the disturbance variance is bounded with an explicit expression. In addition, we further discuss the differential privacy level from the view of privacy preservation that each agent can flexibly select its own privacy level to prevent information disclosure.

The remainder of this paper is organized as follows: Section II declares some preliminary knowledge and background about the graph theory, probability theory, and differential privacy consensus; Section III provides the detailed event-triggered algorithm for the differential privacy consensus; Section IV gives the main results including the mean-square consensus analysis, the accuracy on the convergence point, and the differential privacy analysis with the preselected privacy level; some numerical simulations are given in Section V to illustrate the main results; and, finally, this paper concludes in Section VI.

II. Preliminaries and Background

A. Notations

The following standard notations are used throughout this paper. The set of all natural numbers, positive integers, real numbers, and non-negative real numbers are, respectively, denoted by $\mathbb{N}$, $\mathbb{N}^+$, $\mathbb{R}$, and $\mathbb{R}_{\geq 0}$. The absolute value of the real number x is denoted by $|x|$. Let $\mathbf{1}_N$ and $\mathbf{0}_N$ be, respectively, a 1 vector and a 0 vector containing N elements, and I_N be an N-dimension unity matrix. The transposes of a vector v and a matrix M are denoted by v^T and M^T, respectively. The average of any given vector x is denoted by $\mathrm{Ave}(x)$. The probability density function, probability, expectation, and variance of a random variable X are denoted by $f(X)$, $\mathbb{P}\{X\}$, $\mathbb{E}[X]$, and $\mathbb{V}[X]$, respectively.

B. Algebraic Graph Theory

Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$ be an undirected graph with N nodes, in which $\mathcal{V} = \{1, 2, ..., N\}$ is the node set, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set, and $W = (w_{ij}) \in \mathbb{R}^{N\times N}$ is the weighted adjacency matrix of $\mathcal{G}$. An edge $e_{ji} = (v_j, v_i)$ represents that node j can reach node i or node i can directly receive information from node j. Here, W is a symmetric matrix, that is, the communication channels between network nodes are two way. If $e_{ij} \in \mathcal{E}$, that is, there is a communication channel between node i and node j, then they are called neighbors of each other and accordingly $w_{ij} = w_{ji} > 0$; otherwise, $w_{ij} = w_{ji} = 0$. The neighbor set of node i is denoted by $\mathcal{N}_i$. Let $N_i = |\mathcal{N}_i|$ denote the number of neighbors of node i. The Laplacian matrix $L = (l_{ij}) \in \mathbb{R}^{N\times N}$ associated with the adjacency matrix W is defined by $l_{ij} = -w_{ij}, i \neq j$ and $l_{ii} = \sum_{j=1,j\neq i}^{N} w_{ij}$.

C. Probability Theory

The following lemmas about the probability theory will be used in our analysis.

Lemma 1: [42] For a random variable X obeying Laplace distribution, that is, $X \sim \mathrm{Laplace}(\mu, b)$, then the Laplace probability density function is given by

$$\mathcal{L}(x) = \frac{1}{2b}\exp\left(-\frac{|x-\mu|}{b}\right)$$

where μ is the mean and b is the scale parameter. Then, we have the expectation $\mathbb{E}[X] = \mu$ and the variance $\mathbb{V}[X] = 2b^2$.

Lemma 2: [43] Considering a random variable X with finite expected value μ and finite nonzero variance σ^2, then for any scalar $k > 0$, the following Chebyshev inequality holds:

$$\mathbb{P}\{|X-\mu| \geq k\sigma\} \leq \frac{1}{k^2}.$$

D. Differential Privacy Consensus

The privacy-preserving consensus means to develop a control algorithm to protect the agents' states from disclosure while the agents can communicate with their neighbors and update their states to reach an agreement [14]. We consider adversaries inside or outside the network that do not interfere with the algorithm execution but seek to steal information about the input values, individual states, or the agreement trajectory of the network. Regarding this condition, the notion of the differential privacy is employed to meet the privacy requirement. The following differentially private consensus algorithm in discrete time is proposed in [18] and [22]:

$$\theta_i(t+1) = \theta_i(t) + hu_i(t) + s_i\eta_i(t) \tag{1}$$

where $\theta_i(t) \in \mathbb{R}$ is the internal state of agent i, $\eta_i(t) \in \mathbb{R}$ is a random noise generated by agent i at time t from a Laplace distribution, $h > 0$ is the step size, and $s_i > 0$ is the noise parameter for agent i. The controller $u_i(t)$ is defined as

$$u_i(t) = \sum_{j\in\mathcal{N}_i} w_{ij}(x_j(t) - x_i(t)) \tag{2}$$

where the transmitted message $x_i(t)$ is defined as

$$x_i(t) = \theta_i(t) + \eta_i(t),\ i = 1,\ldots,N. \tag{3}$$

Remark 1: The privacy concern of agents can be local (e.g., some or all of the agents do not want to reveal their local inputs to the outside world) or global (e.g., all agents do not want to reveal their agreement value to agents outside the network). The existence of noise parameter s_i provides a chance for each agent to choose its own privacy level without affecting other agents [22].

III. Problem Statement and Algorithm Design

A. Algorithm Design With an Event-Triggered Mechanism

Though the algorithm (1)–(3) well prevents the information disclosure due to the addition of random noise at each execution procedure. There exists a nonnegligible fact that each agent has to collect all its neighbors' states at every time instant, which means that real-time information communication has to be remained [25]. In other words, the agents must cope with heavy computation and communication load, which is not available for agents only equipped with restricted microprocessors and energy batteries.

The advent of the event-triggered mechanism offers a new point of view on how information could be sampled and transmitted. To introduce the event-triggered strategy, we first assume that the sequence of event times for each agent i is $0 = t_0^i, t_1^i, t_2^i, \ldots$, and the agent broadcasts its state only at its own event times. Then, the real-time measurements from neighbors are not available for each agent i. Thus, we redesign the controller (2) and the transmitted message (3) by utilizing the last measurements received from each neighbor $j \in \mathcal{N}_i$ as follows:

$$u_i(t) = \sum_{j\in\mathcal{N}_i} w_{ij}(x_j(t_{k_j}^j) - x_i(t_{k_i}^i)),\ t \in [t_{k_i}^i, t_{k_i+1}^i) \tag{4}$$

where

$$x_i(t_{k_i}^i) = \theta_i(t_{k_i}^i) + s_i\eta_i(t_{k_i}^i). \tag{5}$$

Note that $x_i(t_{k_i}^i)$ and $x_j(t_{k_j}^j)$ represent the transmitted messages of agent i and its neighbors at their last event times, respectively. Also $\theta_i(t_{k_i}^i)$ and $\eta_i(t_{k_i}^i)$ represent the internal state and the additional random noise of agent i at its event time, respectively.

Remark 2: Each agent i executes triggering only at its individual event time $t_{k_i}^i$, and then, generates and transmits the message $x_i(t_{k_i}^i)$ to its neighbors. Meanwhile, agent i updates its controller by utilizing its own and its neighbors' transmitted messages only when an event is triggered at agent i or its neighbors. That is, in time interval $[t_{k_i}^i, t_{k_i+1}^i)$, the controller of each agent i will remain unchanged as a constant until its next triggering time instant $t_{k_i+1}^i$ comes or an event is triggered at its neighbors. Note that the transmitted message in (5) has an additional noise parameter s_i compared with (3). The reasons are twofold: 1) this is a key trick for the design of measurement error in the next section such that the measurement error can be automatically reset to zero when an event is triggered; and 2) the algorithms (1), (4), and (5) have better universality compared with the algorithms (1)–(3). Specifically, when all s_i equal to zero, that is, when all agents no longer need privacy preservation, then this differentially private average algorithm will degenerate to the common average consensus algorithm with an event-triggered mechanism [29], [30].

Regarding the differential privacy consensus with an event-triggered mechanism, we here provide a formal and detailed algorithm to describe the communication process. Assume that each agent i has a memory that can store its own instant internal state $\theta_i(t)$, transmitted message $x_i(t_{k_i}^i)$, and its neighbors' transmitted messages $x_j(t_{k_j}^j), j \in \mathcal{N}_i$. Furthermore, the initial internal states of all agents are given by $\theta(0) = (\theta_1(0), \ldots, \theta_N(0))^T$, and all the initial event time t_0^i are initialized to 0. At the start time, all agents initialize their memory and Laplacian noise, and broadcast their own $x_i(0)$ to their neighbors. Then, each agent implements the following algorithm at each time instant.

Remark 3: The event-triggered mechanism plays a key role in reducing communication frequency and controller updates. As we can see, only when an event is triggered at agent i, (5) is calculated and broadcast to the neighbors of agent i. Meanwhile,

Algorithm 1: The Description of Communication Algorithm.

1: Agent i updates its own internal state according to algorithm $\theta_i(t+1) = \theta_i(t) + hu_i(t) + s_i\eta_i(t)$
2: Generates the Laplacian noise $\eta_i(t+1)$
3: Judges whether an event occurs or not
4: **if** an event is triggered **then**
5: updates the latest event time, i.e., $t^i_{k_i} = t+1$
6: generates the latest transmitted message $x_i(t^i_{k_i}) = \theta_i(t^i_{k_i}) + s_i\eta_i(t^i_{k_i})$
7: updates $x_i(t^i_{k_i})$ stored in the local memory of agent i with the transmitted message
8: updates the controller $u_i(t+1) = \sum_{j=1}^N w_{ij}(x_j(t^j_{k_j}) - x_i(t^i_{k_i}))$
9: broadcasts $x_i(t^i_{k_i})$ to the neighbors of agent i
10: **else**
11: keeps local memory and controller constant
12: keeps silent
13: **end if**
14: Agent i detects whether messages from its neighbors are received or not
15: **if** a message from neighbor j is received **then**
16: updates $x_j(t^j_{k_j})$ stored in the local memory of agent i with the transmitted message
17: updates the controller $u_i(t+1) = \sum_{j=1}^N w_{ij} (x_j(t^j_{k_j}) - x_i(t^i_{k_i}))$
18: **else**
19: keeps local memory and controller constant
20: **end if**

once the event is triggered at agent i or agent i receives the transmitted messages from its neighbors, the agent first updates the copies of its own or its neighbors' latest transmitted messages stored in the local memory, and then, updates its controller (4) utilizing the updated local copies of these messages. Therefore, real-time communication and calculation are avoided compared with the common differential privacy consensus algorithm [18], [22]. In this event-triggered mechanism, the key is to judge when an event should be triggered. We hence need to design an effective event-triggering condition, which will be introduced in the next subsection.

B. Design of Event-Triggering Condition

To introduce the event-triggering condition, we first need to define a variable named measurement error as follows, where $t \in [t^i_{k_i}, t^i_{k_i+1})$:

$$\begin{aligned} e_i(t) &= x_i(t^i_{k_i}) - x_i(t) \\ &= \theta_i(t^i_{k_i}) + s_i\eta_i(t^i_{k_i}) - \theta_i(t) - s_i\eta_i(t). \end{aligned} \tag{6}$$

Note that $x_i(t) = \theta_i(t) + s_i\eta_i(t)$ can be called as a pretransmission message. That is, if the current time t is not an event time, the state $x_i(t)$ will not be transmitted even though it has been calculated. Actually, (6) roughly describes the degree that the current pretransmission message deviates from the transmitted message at its last event time. Once the measurement error reaches a threshold prescribed in advance, the event is triggered and the measurement error is reset to zero automatically since $e_i(t) = x_i(t^i_{k_i}) - x_i(t^i_{k_i}) = 0$ at this event time.

Let $A = (a_{ij})$ with $a_{ij} = hw_{ij} \geq 0$, for $i \neq j$, and $a_{ii} = 1 - \sum_{j=1, j\neq i}^N a_{ij}$. Note that here we assume $a_{ii} > 0$ (by selecting proper h or w_{ij}). Hence, A is stochastic since A satisfies $A\mathbf{1} = \mathbf{1}$. Substituting (4)–(6) into (1), the algorithm (1) can be rewritten as

$$\begin{aligned} \theta_i(t+1) &= \theta_i(t) + h\sum_{j\in\mathcal{N}_i} w_{ij}(x_j(t^j_{k_j}) - x_i(t^i_{k_i})) + s_i\eta_i(t) \\ &= \theta_i(t) + \sum_{j\in\mathcal{N}_i} a_{ij}(\theta_j(t^j_{k_j}) - \theta_i(t^i_{k_i})) \\ &\quad + \sum_{j\in\mathcal{N}_i} a_{ij}(s_j\eta_j(t^j_{k_j}) - s_i\eta_i(t^i_{k_i})) + s_i\eta_i(t) \\ &= \theta_i(t) + \sum_{j=1}^N a_{ij}\theta_j(t^j_{k_j}) - \theta_i(t^i_{k_i}) \\ &\quad + \sum_{j=1}^N a_{ij}s_j\eta_j(t^j_{k_j}) - s_i\eta_i(t^i_{k_i}) + s_i\eta_i(t) \\ &= -e_i(t) + \sum_{j=1}^N a_{ij}\theta_j(t^j_{k_j}) + \sum_{j=1}^N a_{ij}s_j\eta_j(t^j_{k_j}). \end{aligned} \tag{7}$$

Because each agent can only obtain its neighbors' transmitted messages, then the event should be calculated only depending on local information available to each agent. We propose the following event-triggering condition to determine the next event time:

$$t^i_{k_i+1} = \inf\left\{t \in \mathbb{N}, t > t^i_{k_i} \,|\, f\left(e_i(t), x(t)\right) \geq 0\right\} \tag{8}$$

where

$$f\left(e_i(t), x(t)\right) = e_i^2(t) - \frac{a_{ii}^2}{16}\sum_{j\in\mathcal{N}_i} a_{ij}\left(x_j(t^j_{k_j}) - x_i(t^i_{k_i})\right)^2. \tag{9}$$

Remark 4: Note that (9) only depends on local information and local parameters. In this design, not only the required state information is local but the key parameters are also local, which implies that the proposed algorithm (1), (4), (5) under event-triggering condition (8) can be implemented successfully in real communication environment where global information is not available.

IV. Main Results

A. Mean-Square Consensus Analysis

Due to the existence of the Laplacian random noise, the system (1) becomes a stochastic system instead of a deterministic one. In this section, we mainly focus on the mean-square consensus analysis of the algorithm (1), (4), (5) under the event-triggering condition (8).

Definition 1: [44] For any given initial state $x(0)$, a stochastic system is said to asymptotically achieve the mean-square consensus if there is a random variable x^* such that

$$\lim_{t\to\infty} \mathbb{E}[x_i(t) - x^*]^2 = 0, i = 1, 2, \ldots, N.$$

Theorem 1: Consider the multiagent network (1) with the control input (4) under the event-triggering condition (8). Assume that the communication graph is undirected and connected, and the transmitted message (5) is corrupted by the Laplacian noise $\eta_i(t) \sim \text{Lap}(b_i(t))$ with $b_i(t) = c_i q_i^t, c_i > 0, s_i, q_i \in (0, 1)$. Then, for any agent $i \in \{1, 2, \ldots, N\}$

$$\lim_{t\to\infty} \mathbb{E}[V(t)] = 0 \quad \forall\, \theta_i(0) \in \mathbb{R}$$

where $V(t) = \sum_{i=1}^N (\theta_i(t) - \frac{1}{N}\sum_{j=1}^N \theta_j(t))^2$ is the energy function of the consensus error.

Proof: See Appendix. ■

Remark 5: This paper focuses on the fully distributed design in both the differentially private consensus algorithm and the event-triggering conditions. Note that extending the fully distributed design to the case of directed graphs is very challenging. First, this is partly due to the fact that the symmetric structure of undirected graphs plays an important role in the theoretical analysis. Second, the symmetric structure of undirected graphs plays a key role in the derivation of fully distributed event-triggering conditions. In fact, the asymmetric structure of directed graphs might lead to a poor result that the key parameters of the event-triggering conditions are dependent on the algebraic connectivity of the Laplacian matrix L, which means that the fully distributed design (not only the consensus algorithm itself but the event-triggering conditions) in this paper will be broken since the Laplacian matrix L depends on the global information of the whole network.

Remark 6: Regarding Zeno behaviors (occurring infinite times in a finite-time interval), it is worth emphasizing that these behaviors only might happen in a hybrid system rather than in a discrete-time system. Note that our main algorithm is modeled as a discrete-time iteration with a constant sampling interval, which means that the smallest event time interval in this paper is actually one step size. That is, the event will be triggered at each sampling time instant in the worst-case scenario, and even so, the number of event-triggering times is still finite in a finite-time interval. Thus, the Zeno behaviors will not happen, that is, the triggering events will not blow up in this paper.

B. Accuracy Analysis

In general, the common average consensus algorithm [13] can converge to the average of the initial states deterministically. However, the algorithm $(1), (4), (5)$ cannot surely reach the exact initial average due to the intrinsic property of the differential privacy mechanism [22]. Therefore, we further establish the statistic properties of the agreement value corresponding to the algorithm $(1), (4), (5)$ under the event-triggered mechanism in this section.

Definition 2: [18] For any given initial state $x(0), p \in (0, 1), r \in \mathbb{R}_{\geq 0}$, a stochastic system is said to achieve (p, r)-accuracy if the agreement trajectory converges to a random variable x^* with a bounded dispersion r with probability at least $1 - p$.

Corollary 2: The proposed differential consensus system achieves

$$\left(p, \frac{1}{N}\sqrt{\frac{2}{p}\sum\nolimits_{i=1}^{N} \frac{s_i^2 c_i^2}{1 - q_i^2}}\right)$$

accuracy and the convergence point θ_∞ is an unbiased estimate of their initial state average $\text{Ave}(\theta(0))$.

Proof: Equations (1) and (4)–(6) can be written in a compact matrix form as

$$\theta(t+1) = (I - hL)\theta(t) + (I - hL)S\eta(t) - hLe(t) \quad (10)$$

where $\theta(t) = (\theta_1(t), \ldots, \theta_N(t))^T$, $\eta(t) = (\eta_1(t), \ldots, \eta_N(t))^T$, $S = \text{diag}(s_1, \ldots, s_N)$, and $e(t) = (e_1(t), \ldots, e_N(t))^T$.

Let $J_N = \left(\frac{1}{N}\right) \mathbf{1}\mathbf{1}^T \in \mathbb{R}^{N\times N}$. Note that $J_N L = \mathbf{0}$. Multiplying J_N by the both sides of (10), we have $J_N \theta(t+1) = J_N(\theta(t) + S\eta(t))$, which is equivalent to

$$\begin{aligned}\frac{1}{N}\sum_{i=1}^{N} \theta_i(t+1) &= \frac{1}{N}\sum_{i=1}^{N} \theta_i(t) + \frac{1}{N}\sum_{i=1}^{N} s_i\eta_i(t)\\ &= \frac{1}{N}\sum_{i=1}^{N} \theta_i(0) + \frac{1}{N}\sum_{k=0}^{t}\sum_{i=1}^{N} s_i\eta_i(k)\end{aligned}$$

i.e.,

$$\text{Ave}(\theta(t)) = \text{Ave}(\theta(0)) + \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{t-1} s_i\eta_i(k). \quad (11)$$

Then, we get the convergence point $\theta_\infty = \lim_{t\to\infty} \text{Ave}(\theta(t))$.

Since the noise $\eta_i(t) \sim \text{Lap}(b_i(t))$ with $b_i(t) = c_i q_i^t, q_i \in (0, 1)$, according to Lemma 1, we have $\mathbb{E}[\eta_i(t)] = 0, \mathbb{V}[\eta_i(t)] = \mathbb{E}[\eta_i^2(t)] = 2c_i^2 q_i^{2t}$. Thus, we can further obtain

$$\begin{aligned}\mathbb{E}[\theta_\infty] &= \mathbb{E}[\text{Ave}(\theta(0))]\\ \mathbb{V}[\theta_\infty] &= \frac{2}{N^2}\sum_{i=1}^{N} \frac{s_i^2 c_i^2}{1 - q_i^2}. \quad (12)\end{aligned}$$

By Lemma 2, we have

$$\mathbb{P}\{|\theta_\infty - \text{Ave}(\theta(0))| < r\} \geq 1 - \frac{\mathbb{V}[\theta_\infty]}{r^2}.$$

Choosing $r = \frac{1}{N}\sqrt{\frac{2}{p}\sum_{i=1}^{N} \frac{s_i^2 c_i^2}{1-q_i^2}}$, we thus have $\mathbb{P}\{|\theta_\infty - \text{Ave}(\theta(0))| < r\} \geq 1 - p$. The proof is completed. ■

Remark 7: Actually, Definition 2 and Corollary 2 show that the agreement value θ_∞ in this paper is not the exact initial average. Instead, it is just a random variable falling into the nearby range of the initial average with a bounded deviation even if the expectation of θ_∞ equals to the initial average.

C. Differential Privacy Analysis

In this section, we will give some definitions and analysis on the notion of the differential privacy introduced in [15],

[18], and [22]. For a multiagent network, the agents asymptotically converge to an agreement value via exchanging messages with each other. During this information exchange, the transmitted messages constitute the observable parts for adversaries. By contrast, the internal states and the additional random noise are not available for adversaries. For simplicity, we denote all possible observation sequence and noise sequence as $x = \{x(0), x(1), \ldots\}$ and $\eta = \{\eta(0), \eta(1), \ldots\}$, where $x(t) = (x_1(t), \ldots, x_N(t))^T$ and $\eta(t)$ is defined after (10). Given an initial state $\theta(0)$ defined after (10), x is uniquely determined by the noise sequence η due to the algorithm (5). Thus, we denote $X_{\theta(0)}(\eta) = \{x(0), x(1), \ldots\}$ as the corresponding observation sequence. Considering the internal state sequence and noise sequence, we denote $E_{\theta(0)}(\eta) = \{(\theta(0), \eta(0)), (\theta(1), \eta(1)), \ldots\}$ as the corresponding execution sequence. Then, for another initial state $\theta'(0)$ with the noise sequence η', the corresponding observation sequence and execution sequence can be denoted as $X_{\theta'(0)}(\eta')$ and $E_{\theta'(0)}(\eta')$, respectively.

Definition 3: [15] For any given $\delta \in \mathbb{R}_{\geq 0}$, the vectors x, x' are called δ-adjacent if there exists one $k \in \{1, 2, \ldots, N\}$ such that

$$|x_i - x_i'| \leq \begin{cases} \delta, & i = k \\ 0, & i \neq k \end{cases}$$

for $i \in \{1, 2, \ldots, N\}$.

Definition 4: [18], [22] For any given pair of δ-adjacent initial states $\theta(0), \theta'(0)$, a stochastic system is said to preserve the ϵ-differential privacy if for any sets of observation sequence and noise sequence $\mathcal{O}, \Omega \subset (\mathbb{R}^N)^{\mathbb{N}}$

$$\mathbb{P}\{X_{\theta(0)}(\eta) \in \mathcal{O} | \eta \in \Omega\} \leq e^{\epsilon\delta} \mathbb{P}\{X_{\theta'(0)}(\eta') \in \mathcal{O} | \eta' \in \Omega\}.$$

Remark 8: Intuitively speaking, the differential privacy mechanism ensures that the presence or absence of any individual agent has no significant effect on the output of the execution algorithm (1)–(3). That is, any pair of initial states $\theta(0)$ and $\theta'(0)$ only one component being different will lead to the same observation sequence in large probability. Consequently, the adversaries who steal information from the observation sequence cannot infer and threaten the privacy of initial states of the individual participants.

Corollary 3: The proposed algorithm (1), (4), (5) preserves the ϵ_i-differential privacy for agent $i \in \{1, \ldots, N\}$ with

$$\epsilon_i = \frac{q_i}{c_i(q_i + s_i - 1)}.$$

From the view of the whole network, the privacy level is $\epsilon = \max_i(\epsilon_i)$.

Proof: Considering any pair of δ-adjacent initial states $\theta(0), \theta'(0)$, we assume $\theta_k(0) = \theta_k'(0) + \delta$ for some agent $k \in \{1, \ldots, N\}$ and $\theta_i(0) = \theta_i'(0)$ for all $i \neq k$. Then, we define a bijection between $E_{\theta(0)}(\eta)$ and $E_{\theta'(0)}(\eta')$, where

$$\eta_i'(t) = \begin{cases} \eta_i(t) + \frac{\delta}{s_i}(1 - s_i)^t, & i = k \\ \eta_i(t), & i \neq k. \end{cases}$$

Letting

$$X_{\theta(0)}(\eta) = \{x(0), \ldots, x(T)\} = \{\rho_0, \ldots, \rho_T\}$$
$$X_{\theta'(0)}(\eta') = \{x'(0), \ldots, x'(T)\} = \{\rho_0', \ldots, \rho_T'\}$$

we can easily get $X_{\theta(0)}(\eta) = X_{\theta'(0)}(\eta')$ according to the mathematical induction under the aforementioned bijection.

Note that the equality $X_{\theta(0)}(\eta) = X_{\theta'(0)}(\eta')$ implies that the two observation sequences corresponding to the δ-adjacent initial states are indistinguishable for malicious adversaries. That is, the adversaries cannot infer any agent's initial state by observing and analyzing the difference between the outputs corresponding to the two δ-adjacent initial states.

Next, we first calculate the joint probability density function as follows:

$$f(X_{\theta(0)}(\eta) \in \mathcal{O}) = \prod_{t=0}^{T} f(\rho_t | \rho_0, \ldots, \rho_{t-1})$$

$$f(X_{\theta'(0)}(\eta') \in \mathcal{O}) = \prod_{t=0}^{T} f(\rho_t' | \rho_0', \ldots, \rho_{t-1}').$$

For any given initial state $\theta(0)$, the transmitted message $x(t)$ is uniquely determined by $\eta(t)$ and the random noise $\eta_i(t)$ belongs to the i.i.d Laplacian distribution $\eta_i(t) \sim \text{Lap}(b_i(t))$. Thus, we have the joint Laplace probability density function

$$f(X_{\theta(0)}(\eta) \in \mathcal{O}) = \prod_{t=0}^{T}\prod_{i=1}^{N} \mathcal{L}(\eta_i(t))$$

$$f(X_{\theta'(0)}(\eta') \in \mathcal{O}) = \prod_{t=0}^{T}\prod_{i=1}^{N} \mathcal{L}(\eta_i'(t)). \tag{13}$$

Therefore, we further have as $T \to \infty$

$$\begin{aligned} \frac{f(X_{\theta(0)}(\eta) \in \mathcal{O})}{f(X_{\theta'(0)}(\eta') \in \mathcal{O})} &= \frac{\prod_{t=0}^{T}\prod_{i=1}^{N} \mathcal{L}(\eta_i(t))}{\prod_{t=0}^{T}\prod_{i=1}^{N} \mathcal{L}(\eta_i'(t))} \\ &= \frac{\prod_{t=0}^{T} \mathcal{L}(\eta_k(t))}{\prod_{t=0}^{T} \mathcal{L}(\eta_k'(t))} = \prod_{t=0}^{T} e^{\frac{|\eta_k'(t)| - |\eta_k(t)|}{c_k q_k^t}} \\ &\leq \prod_{t=0}^{T} e^{\frac{|\eta_k'(t) - \eta_k(t)|}{c_k q_k^t}} = \prod_{t=0}^{T} e^{\frac{\delta}{c_k}\left(\frac{1-s_k}{q_k}\right)^t} \\ &= e^{\epsilon_k \delta}. \end{aligned} \tag{14}$$

Then, integrating the both sides, we obtain the probability

$$\mathbb{P}\{X_{\theta(0)}(\eta) \in \mathcal{O}\} \leq e^{\epsilon_k \delta} \mathbb{P}\{X_{\theta'(0)}(\eta') \in \mathcal{O}\} \tag{15}$$

where $\epsilon_k = \frac{q_k}{c_k(q_k + s_k - 1)}$, $q_k \in (1 - s_k, 1)$, which implies that the privacy level of agent k is ϵ_k. Note that agent k can be any agent in the network, which implies that each agent i can select its own privacy level ϵ_i. The proof is completed. ■

Remark 9: Note that the practical significance of the assumption $\theta_k(0) = \theta_k'(0) + \delta$ in the proof of Corollary 3 means that the maximum difference between two initial state sets $\theta(0)$ and $\theta'(0)$ is δ in the worst-case scenario, where the other $N - 1$ agents collude with each other to attack any agent k. Corollary 3 implies that the proposed consensus algorithm can

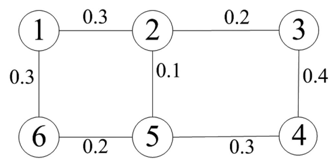


Fig. 1. Weighted interaction network with six agents.

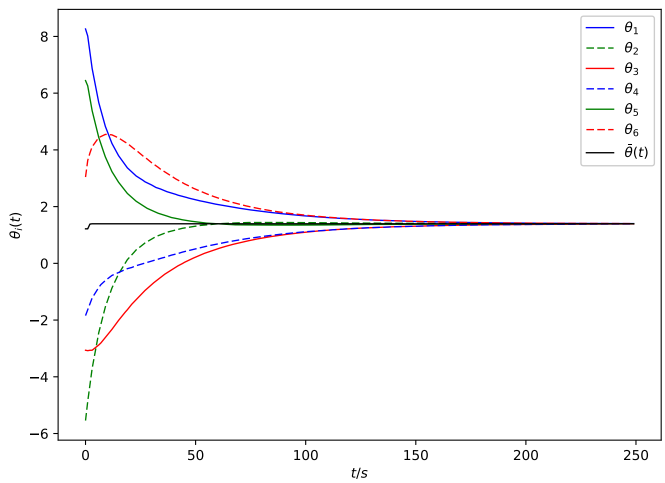


Fig. 2. Internal state evolution of agent i, $i = 1, 2, \ldots, 6$.

ensure that the δ-adjacent initial state sets $\theta(0)$ and $\theta'(0)$ can produce a pair of similar distributions whose statistical difference is dependent on the prescribed ϵ_i for any given δ.

Remark 10: According to the maximum information principle in the security literature, we assume that the adversaries can listen to all possible transmitted message sequences, that is, the agents evolve and transmit their messages at every time instant. The aforementioned results declare that the differential privacy properties can be well preserved even if the assumption is in an extreme condition. Therefore, the adversaries who can only listen to a part or a subset of all possible transmitted message sequence $X_{\theta(0)}(\eta) = \{x(0), x(1), \ldots\}$ under the event-triggered mechanism cannot infer any agent's initial state. In other words, the proposed event-triggered strategy has no impact on the privacy preservation while the execution efficiency of the whole network is largely improved.

V. Simulations

In this section, we provide some simulations to illustrate the proposed event-triggered mechanism. Considering the communication network with graph $\mathcal{G}$ given in Fig. 1, and the initial internal state $\theta(0) = (8.2632, -5.5434, -3.0639, -1.8427, 6.4439, 3.0425)$, which is randomly generated in the interval $[-10, 10]$. As for the parameter values, we set $c_i = 0.2, q_i = 0.1,$ and $s_i = 0.99$.

Fig. 2 shows the internal state evolutions of all agents. It is worth emphasizing that the simulation here does not achieve a common average consensus in which the real-time average

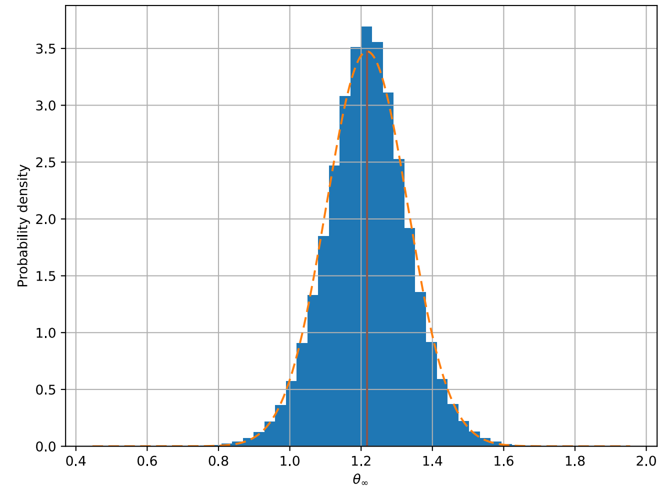


Fig. 3. Histogram of agreement values.

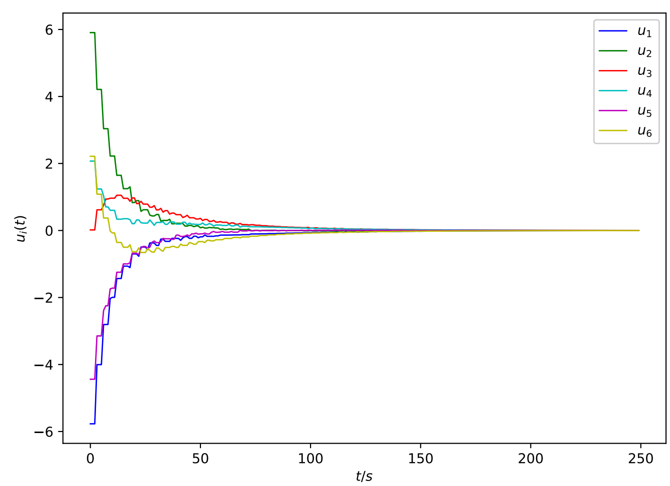


Fig. 4. Controller evolution of agent i, $i = 1, 2, \ldots, 6$.

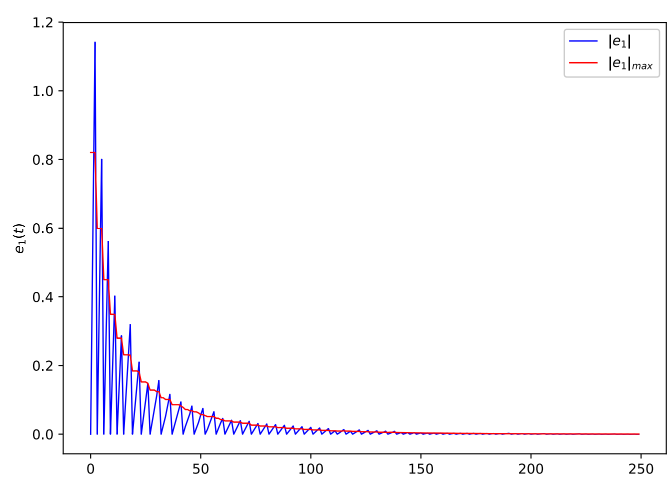


Fig. 5. Evolution of the measurement error and threshold.

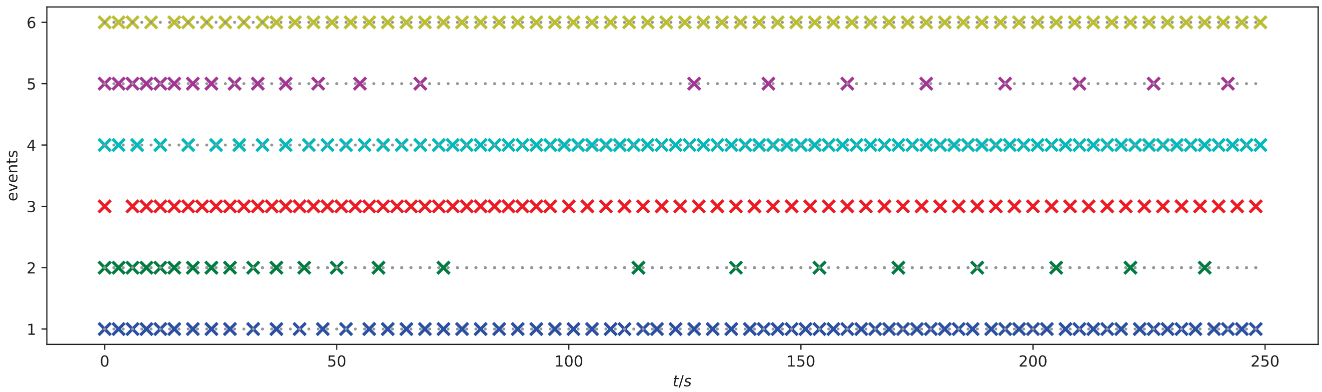


Fig. 6. Event triggering times of agent i, $i = 1, 2, \ldots, 6$.

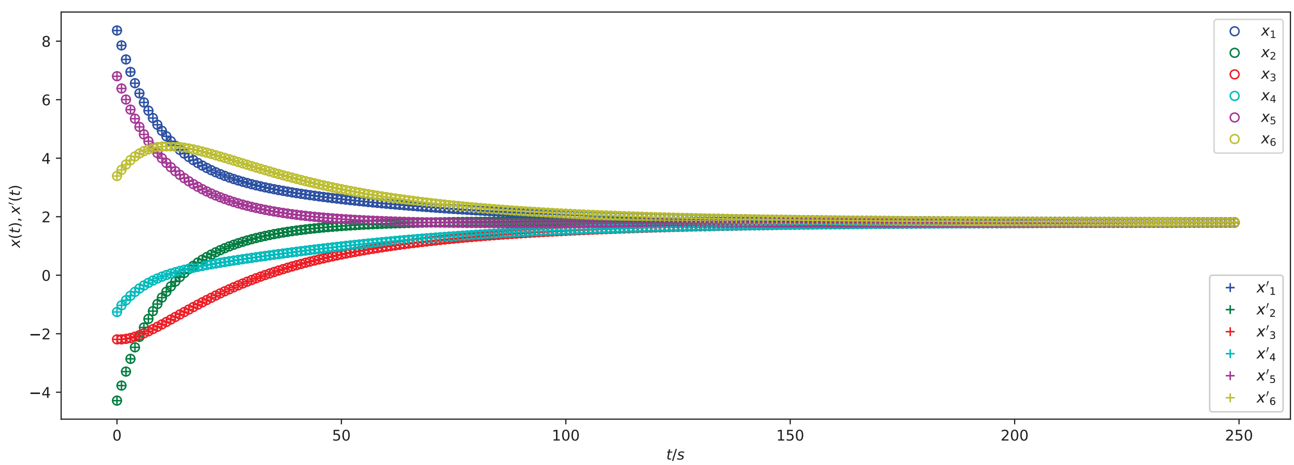


Fig. 7. Observation sequences $x(t), x'(t)$ of δ-adjacent initial states $\theta(0)$ and $\theta'(0)$.

state of all agents remains constant and this constant value equals to the initial average state. In contrast, the real-time average state in this paper is time varying and the final agreement value does not always equal to the initial average state. Specifically, the real-time average state $\bar{\theta}(t)$ denoted by a black line in Fig. 2 is obviously disturbed at initial time, and then, tends to be stable after some iterations. The statistical distribution of agreement values with 10^6 runs is shown in Fig. 3, which shows that the agreement values mainly fall into the range $[0.8, 1.6]$ and the histogram appears to be a bell-shaped curve with a mean exactly at the true initial average indicated by a brown line. Note that the reason that the real-time average state $\bar{\theta}(t)$ is first disturbed, and then, stable is due to the addition of the Laplace noise with a 0-mean and a diminishing scale function $b_i(t) = c_i q_i^t, q_i \in (0, 1)$ (0-mean ensures that all the additional noise will offset each other in distribution and $b_i(t)$ ensures that the additional noise is diminishing as time goes on).

Fig. 4 shows the controller state evolutions of all agents, from which we can see that the control signals are piecewise constants. The evolution of the measurement error of agent 1 is shown in Fig. 5, in which $|e_1(t)|$ is the measurement error of agent 1 and $|e_{1,\max}(t)| = \frac{a_{ii}}{4}\sqrt{\sum_{j\in\mathcal{N}_i} a_{ij}(x_j(t^j_{k_j}) - x_i(t^i_{k_i}))^2}$

TABLE I
STATISTICS OF EVENT TIMES

Agent No.	Event Times	Total Times	Rate
1	72	250	28.8%
2	23	250	9.2%
3	69	250	27.6%
4	76	250	30.4%
5	23	250	9.2%
6	64	250	25.6%

is the specified maximum threshold. In Fig. 6, the events of each agent are marked in time interval $[0, 250]$, from which we can see that the sampling is sporadic rather than consecutive. Regarding of the execution efficiency, we further count the event times or the communication frequency for each agent as described in Table I. The average communication frequency is 53.5 compared with the total time 250, which implies the average communication rate between agents is just 21.4%.

To illustrate the differential privacy properties, we assume that the agents communicate with each other at every time instant. Also the δ-adjacent initial states $\theta(0)$ and $\theta'(0)$ are same except agent 1's state. For example, let $\theta(0) = (8.2632,$

$-5.5434, -3.0639, -1.8427, 6.4439, 3.0425)$ versus $\theta'(0) = (0, -5.5434, -3.0639, -1.8427, 6.4439, 3.0425)$. Then, we get $\delta = 8.2632$. Fig. 7 visually shows the observation sequences $x(t)$ and $x'(t)$ corresponding to the δ-adjacent initial states $\theta(0)$ and $\theta'(0)$, respectively. We can obviously see that these two observation sequences are exactly fitted, which implies that they are indistinguishable for malicious adversaries.

VI. Conclusion

In this paper, we studied the privacy preserving problem of initial states for multiagent networks by employing the differential privacy mechanism and the distributed event-triggering strategy. We first developed a differentially private consensus algorithm combined with an event-triggering strategy against frequent real-time communication and controller updates. Furthermore, we proposed a distributed event-triggering condition that only depends on local state information and local parameters. In addition, the convergence analysis of the mean-square average consensus was given to explain the sufficiency of the proposed algorithm and event condition. Finally, we also established the statistic properties of the convergent accuracy and differential privacy. In the future, we intend to further investigate issues such as the effect of the communication topology on the convergence rate, switching topologies, and time delays in differentially private consensus with an event-triggered mechanism.

Appendix

A. Proof of Theorem 1

Proof: Defining the consensus error

$$\delta_i(t) = \theta_i(t) - \frac{1}{N}\sum_{j=1}^{N}\theta_j(t)$$

then combining with (7), we have

$$\delta_i(t+1) = T_1 - \frac{1}{N}\sum_{j=1}^{N}\theta_j(t) - \frac{1}{N}\sum_{j=1}^{N}s_j\eta_j(t)$$

where

$$T_1 = -e_i(t) + \sum_{j=1}^{N}a_{ij}\theta_j(t^j_{k_j}) + \sum_{j=1}^{N}a_{ij}s_j\eta_j(t^j_{k_j}).$$

Consider the Lyapunov functional candidate $V(t) = \sum_{i=1}^{N}\delta_i^2(t)$. Note that for any $i, j \in \{1, 2, \ldots, N\}$

1) $\theta_i(t)$ and $\eta_j(t)$ are independent of each other;
2) for $i \neq j$, $\eta_i(t)$ and $\eta_j(t)$ are independent of each other;
3) the Laplacian noise mean $\mathbb{E}[\eta_i(t)] = 0$.

Thus, we have

$$\mathbb{E}[\Delta V] = \mathbb{E}\left[V(t+1) - V(t)\right] = \mathbb{E}[\Delta V_1 + \Delta V_2 + \Delta V_3] \quad (16)$$

where

$$\Delta V_1 = \sum_{i=1}^{N}\left(T_1^2 - \theta_i^2(t)\right)$$

$$\Delta V_2 = \sum_{i=1}^{N}\left(\frac{1}{N^2}\sum_{j=1}^{N}s_j^2\eta_j^2(t) - \frac{2}{N}s_i^2\eta_i^2(t)\right)$$

$$\Delta V_3 = \frac{2}{N}\sum_{i=1}^{N}\left(e_i(t) + \theta_i(t) - \sum_{j=1}^{N}a_{ij}\theta_j(t^j_{k_j})\right)\sum_{j=1}^{N}\theta_j(t).$$

Now we simplify the term $\mathbb{E}[\Delta V_1]$ as

$$\mathbb{E}[\Delta V_1] = \mathbb{E}\left[\sum_{i=1}^{N}\left(T_2 + \left(\sum_{j=1}^{N}a_{ij}s_j\eta_j(t^j_{k_j})\right)^2 - 2e_i(t)\sum_{j=1}^{N}a_{ij}s_j\eta_j(t^j_{k_j})\right)\right] \quad (17)$$

where

$$T_2 = e_i^2(t) - 2\sum_{j=1}^{N}a_{ij}e_i(t)\theta_j(t^j_{k_j}) + \left(\sum_{j=1}^{N}a_{ij}\theta_j(t^j_{k_j})\right)^2 - \theta_i^2(t).$$

Since

$$\left(\sum_{j=1}^{N}a_{ij}\theta_j(t^j_{k_j})\right)^2 = \sum_{j=1}^{N}a_{ij}^2\theta_j^2(t^j_{k_j}) + 2\sum_{j=1}^{N-1}\sum_{l>j}a_{ij}a_{il}\theta_j(t^j_{k_j})\theta_l(t^l_{k_l}) \quad (18)$$

$$\sum_{j=1}^{N}a_{ij}e_i(t)\theta_j(t^j_{k_j}) = \sum_{j=1,j\neq i}^{N}a_{ij}e_i(t)\theta_j(t^j_{k_j}) + \left(1 - \sum_{j=1,j\neq i}^{N}a_{ij}\right)e_i(t)\theta_i(t^i_{k_i})$$
$$= \sum_{j=1,j\neq i}^{N}a_{ij}e_i(t)\left(\theta_j(t^j_{k_j}) - \theta_i(t^i_{k_i})\right) + e_i(t)\theta_i(t^i_{k_i}) \quad (19)$$

we have

$$T_2 = T_3 + T_4 + T_5 + T_6$$

where

$$T_3 = \sum_{j=1}^{N}a_{ij}^2\theta_j^2(t^j_{k_j}) + \sum_{j=1}^{N-1}\sum_{l>j}a_{ij}a_{il}\left(\theta_j^2(t^j_{k_j}) + \theta_l^2(t^l_{k_l})\right)$$

$$T_4 = \sum_{j=1}^{N-1}\sum_{l>j}a_{ij}a_{il}\left(-\theta_j^2(t^j_{k_j}) - \theta_l^2(t^l_{k_l}) + 2\theta_j(t^j_{k_j})\theta_l(t^l_{k_l})\right)$$

$$T_5 = -2 \sum_{j=1, j\neq i}^{N} a_{ij} e_i(t) \left(\theta_j(t^j_{k_j}) - \theta_i(t^i_{k_i})\right)$$

$$T_6 = e_i^2(t) - \theta_i^2(t) - 2e_i(t)\theta_i(t^i_{k_i}).$$

Since

$$T_3 = \sum_{j=1}^{N} a_{ij}^2 \theta_j^2(t^j_{k_j}) + \sum_{j=1}^{N} \sum_{l=1, l\neq j} a_{ij} a_{il} \theta_j^2(t^j_{k_j})$$

$$= \sum_{j=1}^{N} a_{ij} \theta_j^2(t^j_{k_j}) \tag{20}$$

$$T_4 = -\sum_{j=1}^{N-1} \sum_{l>j} a_{ij} a_{il} \left(\theta_j(t^j_{k_j}) - \theta_l(t^l_{k_l})\right)^2$$

$$= -\sum_{j=1, j\neq i}^{N-1} \sum_{l>j, l\neq i} a_{ij} a_{il} \left(\theta_j(t^j_{k_j}) - \theta_l(t^l_{k_l})\right)^2$$

$$- \sum_{j=1, j\neq i}^{N} a_{ij} a_{ii} \left(\theta_j(t^j_{k_j}) - \theta_i(t^i_{k_i})\right)^2$$

$$\leq - \sum_{j=1, j\neq i}^{N} a_{ij} a_{ii} \left(\theta_j(t^j_{k_j}) - \theta_i(t^i_{k_i})\right)^2$$

$$T_5 \leq \sum_{j=1, j\neq i}^{N} a_{ij} \left(\frac{1}{\alpha_i} e_i^2(t) + \alpha_i \left(\theta_j(t^j_{k_j}) - \theta_i(t^i_{k_i})\right)^2\right) \tag{21}$$

$$T_6 = e_i^2(t) - \left(\theta_i(t^i_{k_i}) + s_i\eta_i(t^i_{k_i}) - e_i(t) - s_i\eta_i(t)\right)^2$$

$$- 2e_i(t)\theta_i(t^i_{k_i})$$

$$= -\theta_i^2(t^i_{k_i}) - s_i^2\eta_i^2(t^i_{k_i}) - 2\theta_i(t^i_{k_i}) s_i\eta_i(t^i_{k_i})$$

$$+ 2s_i\eta_i(t^i_{k_i}) e_i(t) - s_i^2\eta_i^2(t) + 2\theta_i(t^i_{k_i}) s_i\eta_i(t)$$

$$+ 2s_i\eta_i(t^i_{k_i}) s_i\eta_i(t) - 2e_i(t) s_i\eta_i(t)$$

we have

$$\sum_{i=1}^{N} T_2 \leq \sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} a_{ij} \left(\frac{1}{\alpha_i} e_i^2(t) \right.$$

$$\left. + (\alpha_i - a_{ii}) \left(\theta_j(t^j_{k_j}) - \theta_i(t^i_{k_i})\right)^2\right)$$

$$+ \sum_{i=1}^{N} \left(-s_i^2\eta_i^2(t^i_{k_i}) - 2\theta_i(t^i_{k_i}) s_i\eta_i(t^i_{k_i})\right.$$

$$+ 2s_i\eta_i(t^i_{k_i}) e_i(t) - s_i^2\eta_i^2(t) + 2\theta_i(t^i_{k_i}) s_i\eta_i(t)$$

$$\left. + 2s_i\eta_i(t^i_{k_i}) s_i\eta_i(t) - 2e_i(t) s_i\eta_i(t)\right). \tag{22}$$

Using similar simplification as in (18), (20) and (19), (21), we have

$$\left(\sum_{j=1}^{N} a_{ij} s_j \eta_j(t^j_{k_j})\right)^2$$

$$\leq \sum_{j=1}^{N} a_{ij} s_j^2 \eta_j^2(t^j_{k_j}) - \sum_{j=1, j\neq i}^{N} a_{ij} a_{ii} \left(s_j\eta_j(t^j_{k_j}) - s_i\eta_i(t^i_{k_i})\right)^2 \tag{23}$$

$$- 2e_i(t) \sum_{j=1}^{N} a_{ij} s_j \eta_j(t^j_{k_j})$$

$$\leq \sum_{j=1, j\neq i}^{N} a_{ij} \left(\frac{1}{\alpha_i} e_i^2(t) + \alpha_i \left(s_j\eta_j(t^j_{k_j}) - s_i\eta_i(t^i_{k_i})\right)^2\right)$$

$$- 2e_i(t) s_i \eta_i(t^i_{k_i}). \tag{24}$$

Substituting (22)–(24) into (17), we have

$$\mathbb{E}[\Delta V_1] = \sum_{i=1}^{N} \left(T_1^2 - \theta_i^2(t)\right)$$

$$\leq \mathbb{E}\left[\sum_{i=1}^{N} \left\{\frac{2}{\alpha_i} e_i^2(t) - \sum_{j=1, j\neq i}^{N} a_{ij}(a_{ii} - \alpha_i)\left[\left(\theta_j(t^j_{k_j}) - \theta_i(t^i_{k_i})\right)^2\right.\right.\right.$$

$$\left.\left.\left. + \left(s_j\eta_j(t^j_{k_j}) - s_i\eta_i(t^i_{k_i})\right)^2\right] + s_i^2\eta_i^2(t)\right\}\right]. \tag{25}$$

Since the noise $\eta_i(t) \sim \mathrm{Lap}(b_i(t))$ with $b_i(t) = c_i q_i^t, q_i \in (0,1)$, according to Lemma 1, we have $\mathbb{E}[\eta_i^2(t)] = \mathbb{V}[\eta_i(t)] = 2c_i^2 q_i^{2t}$, and thus

$$\mathbb{E}[\Delta V_2] = \sum_{i=1}^{N} \left(\frac{2}{N^2} \sum_{j=1}^{N} s_j^2 c_j^2 q_j^{2t} - \frac{4}{N} s_i^2 c_i^2 q_i^{2t}\right)$$

$$= -\frac{2}{N} \sum_{i=1}^{N} s_i^2 c_i^2 q_i^{2t}$$

$$\leq 0 \tag{26}$$

$$\mathbb{E}[\Delta V_3]$$

$$= \frac{2}{N} \mathbb{E}\left[\sum_{i=1}^{N} \left(\theta_i(t^i_{k_i}) + s_i\eta_i(t^i_{k_i}) - \theta_i(t) - s_i\eta_i(t)\right.\right.$$

$$\left.\left. + \theta_i(t) - \sum_{j=1}^{N} a_{ij}\theta_j(t^j_{k_j})\right) \sum_{j=1}^{N} \theta_j(t)\right]$$

$$= \frac{2}{N} \mathbb{E}\left[\left(\sum_{i=1}^{N} \theta_i(t^i_{k_i}) - \sum_{j=1}^{N} \theta_j(t^j_{k_j})\right) \sum_{j=1}^{N} \theta_j(t)\right]$$

$$= 0. \tag{27}$$

Combining (16) with (25)–(27), we have

$$\begin{aligned}\mathbb{E}[\Delta V] &= \mathbb{E}[\Delta V_1 + \Delta V_2 + \Delta V_3] \\ &\le \mathbb{E}\left[\sum_{i=1}^{N}\left\{\frac{2}{\alpha_i}e_i^2(t) - \sum_{j\in\mathcal{N}_i}^{N} a_{ij}(a_{ii}-\alpha_i)\left[\left(\theta_j(t_{k_j}^j) - \theta_i(t_{k_i}^i)\right)^2 \right.\right.\right. \\ &\qquad \left.\left.\left. + \left(s_j\eta_j(t_{k_j}^j) - s_i\eta_i(t_{k_i}^i)\right)^2\right] + s_i^2\eta_i^2(t)\right\}\right] \\ &\le \mathbb{E}\left[\sum_{i=1}^{N} g(e_i(t), x(t_k)) + \sum_{i=1}^{N} s_i^2\eta_i^2(t)\right] \end{aligned} \tag{28}$$

where

$$\begin{aligned} & g(e_i(t), x(t_k)) \\ &= \frac{2}{\alpha_i}e_i^2(t) - \frac{1}{2}\sum_{j\in\mathcal{N}_i} a_{ij}(a_{ii}-\alpha_i)\left(x_j(t_{k_j}^j) - x_i(t_{k_i}^i)\right)^2. \end{aligned}$$

Then, enumerating and calculating the sums of both sides in (28), we can further have as $t \to \infty$

$$\begin{aligned} & \mathbb{E}[V(t)] - \mathbb{E}[V(0)] \\ & \le \mathbb{E}\left[\sum_{i=1}^{N}\sum_{k=0}^{t-1} g(e_i(k), x(t_k))\right] + 2\sum_{i=1}^{N}\frac{c_i^2 s_i^2}{1-q_i^2}. \end{aligned} \tag{29}$$

Note that the event-triggering condition (8) is implemented, we have

$$e_i^2(t) \le \frac{1}{4}\sum_{j\in\mathcal{N}_i} a_{ij}\alpha_i(a_{ii}-\alpha_i)\left(x_j(t_{k_j}^j) - x_i(t_{k_i}^i)\right)^2$$

where $\alpha_i = a_{ii}/2$. Thus, we further have $g(e_i(t), x(t_k)) \le 0$.

Since $\mathbb{E}[g(e_i(t), x(t_k))] \le 0, \mathbb{E}[V(t)] \ge 0$, from (29), it can be seen that the terms $\mathbb{E}[V(t)]$ and $\mathbb{E}[g(e_i(t), x(t_k))]$ are both bounded. Therefore, there must exist a positive real number $0 < c < 1$ before achieving consensus such that

$$\mathbb{E}[V(t+1)] \le \mathbb{E}\left[(1-c)V(t) + \sum_{i=1}^{N} s_i^2\eta_i^2(t)\right]. \tag{30}$$

As $t \to \infty$, the contribution of the first term in (30) converges to 0, and the second term in (30) given by $\sum_{i=1}^{N}\mathbb{E}[s_i^2\eta_i^2(t)] = 2\sum_{i=1}^{N} c_i^2 s_i^2 q_i^{2t}$ also converges to 0. Thus, for any $i \in 1, 2, \ldots, N$, we have $\lim \mathbb{E}[V(t)] = 0, t \to \infty$. The proof is completed. ■

References

[1] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1465–1476, Sep. 2004.

[2] W. Ren and E. M. Atkins, "Distributed multi-vehicle coordinated control via local information exchange," *Int. J. Robust Nonlinear Control*, vol. 17, no. 1011, pp. 1002–1033, 2007.

[3] W. Ren, "On consensus algorithms for double-integrator dynamics," *IEEE Trans. Autom. Control*, vol. 53, no. 6, pp. 1503–1509, Jul. 2008.

[4] R. Olfati-Saber, "Distributed Kalman filter with embedded consensus filters," in *Proc. 44th IEEE Conf. Decision Control/Eur. Control Conf.*, 2005, pp. 8179–8184.

[5] W. Ren, R. W. Beard, and D. B. Kingston, "Multi-agent Kalman consensus with relative uncertainty," in *Proc. Amer. Control Conf.*, 2005, pp. 1865–1870.

[6] L. Xiao, B. Stephen, and L. Sanjay, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. 4th Int. Symp. Inf. Process. Sensor Netw.*, 2005, pp. 63–70.

[7] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proc. 44th IEEE Conf. Decis. Control/Eur. Control Conf.*, 2005, pp. 6698–6703.

[8] R. Aragues, J. Cortes, and C. Sagues, "Distributed consensus on robot networks for dynamically merging feature-based maps," *IEEE Trans. Robot.*, vol. 28, no. 4, pp. 840–854, Aug. 2012.

[9] R. Aragues, J. Cortes, and C. Sagues, "Distributed consensus algorithms for merging feature-based maps with limited communication," *Robot. Auton. Syst.*, vol. 59, no. 3, pp. 163–180, 2011.

[10] J. Hu and G. Feng, "Distributed tracking control of leader–follower multi-agent systems under noisy measurement," *Automatica*, vol. 46, no. 8, pp. 1382–1387, 2010.

[11] Z. Li, X. Liu, W. Ren, and L. Xie, "Distributed tracking control for linear multiagent systems with a leader of bounded unknown input," *IEEE Trans. Autom. Control*, vol. 58, no. 2, pp. 518–523, Feb. 2013.

[12] Y. Wen, Z. Wang, Z. Zuo, C. Yang, and H. Shi, "Nodes selection strategy in cooperative tracking problem," *Automatica*, vol. 74, pp. 118–125, 2016.

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

[14] Y. Mo, and R. M. Murray, "Privacy preserving average consensus," *IEEE Trans. Autom. Control*, vol. 62, no. 2, pp. 753–765, Feb. 2017.

[15] C. Dwork, "Differential privacy: A survey of results," in *Proc. Int. Conf. Theory Appl. Models Comput.*, 2008, pp. 1–19.

[16] C. Dwork and L. Jing, "Differential privacy and robust statistics," in *Proc. 41st Annu. ACM Symp. Theory Comput.*, 2009, pp. 371–380.

[17] C. Dwork, M. Naor, T Pitassi, and G. Rothblum, "Differential privacy under continual observation," in *Proc. 42nd ACM Symp. Theory Comput.*, 2010, pp. 725–724.

[18] Z. Huang, S. Mitra, and G. Dullerud, "Differentially private iterative synchronous consensus," in *Proc. ACM Workshop Privacy Electron. Soc.*, 2012, pp. 81–90.

[19] J. L. Ny and G. J. Pappas, "Differentially private filtering," *IEEE Trans. Autom. Control*, vol. 59, no. 2, pp. 341–354, Feb. 2014.

[20] Z. Huang, Y. Wang, S. Mitra, and G. Dullerud, "On the cost of differential privacy in distributed control systems," in *Proc. 3rd Int. Conf. High Confidence Netw. Syst.*, 2014, pp. 105–114.

[21] Z. Huang, S. Mitra, and N. Vaidya, "Differentially private distributed optimization," *Proc. Int. Conf. Distrib. Comput. Netw.*, 2015, Art. no. 4.

[22] E. Nozari, P. Tallapragada, and J. Cortes, "Differentially private average consensus with optimal noise selection," *IFAC-PapersOnLine* vol. 48, no. 22, pp. 203–208, 2015.

[23] E. Nozari, P. Tallapragada, and J. Cortes, "Differentially private distributed convex optimization via functional perturbation," *IEEE Trans. Control Netw. Syst.*, doi: 10.1109/TCNS.2016.2614100

[24] S. Han, T. Ufuk, and J. George, "Differentially private distributed constrained optimization," *IEEE Trans. Autom. Control*, vol. 62, no. 1, pp. 50–64, Jan. 2017.

[25] K. J. Astrom and B. M. Bernhardsson, "Comparison of Riemann and Lebesgue sampling for first order stochastic systems," in *Proc. IEEE 41st Conf. Decision Control*, 2002, pp. 2011–2016.

[26] A. Albert, "Comparison of event-triggered and time-triggered concepts with regard to distributed control systems," *Embedded World*, Nurnberg, pp. 235–252, Feb. 17–19, 2004.

[27] X. Wang and M. D. Lemmon, "Event-triggered broadcasting across distributed networked control systems," in *Proc. Amer. Control Conf.*, 2008, pp. 3139–3144.

[28] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Trans. Autom. Control*, vol. 52, no. 9, pp. 1680–1685, Sep. 2007.

[29] D. V. Dimarogonas and K. H. Johansson, "Event-triggered control for multi-agent systems," in *Proc. 28th China Decision Control*, 2009, pp. 7131–7136.

[30] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson, "Distributed event-triggered control for multi-agent systems," *IEEE Trans. Autom. Control*, vol. 57, no. 5, pp. 1291–1297, May 2012.

[31] E. Alina, D. V. Dimarogonas, and K. J. Kyriakopoulos, "Event-triggered control for discrete-time systems," *Amer. Control Conf.*, 2010, pp. 4719–4724.

[32] L. Gao, X. Liao, and H. Li, "Pinning controllability analysis of complex networks with a distributed event-triggered mechanism," *IEEE Trans. Circuits Syst. II, Expr. Briefs*, vol. 61, no. 7, pp. 541–545, Jul. 2014.

[33] H. Li, X. Liao, T. Huang, and W. Zhu, "Event-triggering sampling based leader-following consensus in second-order multi-agent systems," *IEEE Trans. Autom. Control*, vol. 60, no. 7, pp. 1998–2003, Jul. 2015.

[34] P. Tallapragada and N. Chopra, "On event triggered tracking for nonlinear systems," *IEEE Trans. Autom. Control*, vol. 58, no. 9, pp. 2343–2348, Sep. 2013.

[35] R. Postoyan, M. C. Bragagnolo, E. Galbrun, J. Daafouz, J. Nesic, and E. B. Castelan, "Event-triggered tracking control of unicycle mobile robots," *Automatica*, vol. 52, pp. 302–308, 2015.

[36] L. Gao, X. Liao, H. Li, and G. Chen, "Event-triggered control for multi-agent network with limited digital communication," *Nonlinear Dyn.*, vol. 82, no. 4, pp. 1659–1669, 2015.

[37] H. Li, G. Chen, T. Huang, and Z. Dong, "Event-triggered distributed average consensus over directed digital networks with limited communication bandwidth," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 3098–3110, Dec. 2016.

[38] J. Zhang and G. Feng, "Event-driven observer-based output feedback control for linear systems," *Automatica*, vol. 50, no. 7, pp. 1852–1859, 2014.

[39] H. Zhang, G. Feng, H. Yan, and Q. Chen, "Observer-based output feedback event-triggered control for consensus of multi-agent systems," *IEEE Trans. Ind. Electron.*, vol. 61, no. 9, pp. 4885–4894, Sep. 2014.

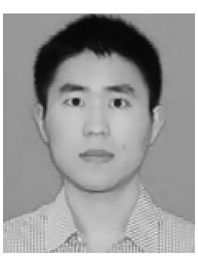
[40] D. Xie, D. Yuan, J. Lu, and Y. Zhang, "Consensus control of second-order leader–follower multi-agent systems with event-triggered strategy," *Trans. Inst. Meas. Control*, vol. 35, no. 4, pp. 426–436, 2013.

[41] N. Mu, X. Liao, and T. Huang, "Leader-following consensus in second-order multiagent systems via event-triggered control with nonperiodic sampled data," *IEEE Trans. Circuits Syst. II, Expr. Briefs*, vol. 62, no. 10, pp. 1007–1011, Oct. 2015.

[42] A. Papoulis and S. U. Pillai, *Probability, Random Variables, and Stochastic Processes.* New York, NY, USA: McGraw-Hill, 2002.

[43] R. Durrett, *Probability: Theory and Examples*. Cambridge, U.K.: Cambridge Univ. Press, 2010.

[44] M. Huang and J. H. Manton, "Coordination and consensus of networked agents with noisy measurements: Stochastic algorithms and asymptotic behavior," *SIAM J. Control Optim.*, vol. 48, no. 1, pp. 134–161, 2009.



Lan Gao received the B.S. degree in information and computing science from the Department of Information and Computing Science, Shengli College, China University of Petroleum, Dongying, China, in 2012. He is currently working toward the Ph.D. degree in computer science and technology with the College of Computer Science and Technology, Chongqing University, Chongqing, China.

His current research interests include multiagent networks, differential privacy, and distributed optimization.

Shaojiang Deng received the B.S. degree in computer science and technology from the College of Computer Science and Technology, Chongqing Jianzhu University, Chongqing, China, in 1993, and the Ph.D. degree in computer science and technology from the College of Computer Science and Technology, Chongqing University, Chongqing, in 2005.

He is currently a Professor with the College of Computer Science and Technology, Chongqing University. His research interests include neural networks, wireless-sensor networks, and wireless body-area networks.

Wei Ren (F'16) received the Ph.D. degree in electrical engineering from Brigham Young University, Provo, UT, USA, in 2004.

He is currently a Professor with the Department of Electrical and Computer Engineering, University of California, Riverside (UC Riverside), Riverside, CA, USA. Prior to joining the UC Riverside in 2011, he was a faculty member with Utah State University and a Postdoctoral Fellow with the University of Maryland, College Park, MD, USA. His research interests include distributed control of multiagent systems and autonomous control of unmanned vehicles. He is an author of two books *Distributed Coordination of Multi-agent Networks* (Springer-Verlag, 2011) and *Distributed Consensus in Multi-vehicle Cooperative Control* (Springer-Verlag, 2008).

Dr. Ren was the recipient of the Antonio Ruberti Young Researcher Prize in 2017 and the National Science Foundation CAREER Award in 2008. He is currently an Associate Editor for the *Automatica* and *Systems and Control Letters*.