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.2019.2910171, IEEE Transactions on Automatic Control

Consensus-based Data-privacy Preserving Data Aggregation

Jianping He[1], Lin Cai[2], Peng Cheng[3], Jianping Pan[2] and Ling Shi[4]

***Abstract*—Privacy-preserving data aggregation in ad hoc networks is a challenging problem, considering the distributed communication and control requirement, dynamic network topology, unreliable communication links, etc. Different from the widely used cryptographic approaches, in this paper, we address this challenging problem by exploiting the distributed consensus technique. We first propose a secure consensus-based data aggregation (SCDA) algorithm that guarantees an accurate sum aggregation while preserving the privacy of sensitive data. Then, we prove that the proposed algorithm converges accurately and is (ϵ, σ)-data-privacy, and the mathematical relationship between ϵ and σ is provided. Extensive simulations have shown that the proposed algorithm has high accuracy and low complexity, and they are robust against network dynamics.**

I. Introduction

Privacy-preserving data aggregation (DA) has attracted great attention with many applications in wireless sensor networks, smart metering systems, cloud computing, etc., [1]–[7]. We consider the applications in distributed networked systems, where data aggregation can be carried out using consensus algorithms [7]. Typical scenarios include the wireless sensor networks where sensors are deployed randomly in an area to monitor the environment, and the sensing data will be aggregated and polled by a remote monitor; or in a smart metering system where the smart meters collect real-time electricity usage and the aggregated usage in an area will be used by the utility company to adjust power supply and enable appropriate demand control. However, these data are often privacy-sensitive [6]. How to ensure accurate data aggregation while preserving privacy is an essential and challenging issue, especially in ad hoc networks.

The ad hoc mode has both pros and cons that should be considered in the design of accurate and privacy-preserving DA. It is well known that in ad hoc networks, centralized algorithm design or optimization solutions are difficult or too costly to implement. Thus, without relying on a centralized controller, an ad hoc network does not suffer from the single-node failure problem and becomes more robust against node failure and link dynamics. On the other hand, without a central trusted authority, it is concerned that some nodes may be compromised or attacked, resulting in the meltdown of the whole network. In addition, dynamic network topology, limited node computing capacity, higher rates of communication errors and losses, and severe delay variations all make privacy-preserving DA more challenging in ad hoc networks. Although privacy-preserving DA has been heavily investigated, existing solutions are typically based on various cryptography techniques, requiring either secure communication channels, pre-established shared keys, a trusted authority, or the combination of them.

Consensus is an important distributed computing method, which has gained much attention in automatic control and signal processing areas [9]–[15], and has been widely used in various networking areas, e.g., time synchronization in sensor networks [16], [17]. Note that an average consensus algorithm can help each node to obtain the average value of all nodes' states in a distributed way, which is a building block of the distributed aggregation algorithm designed in this paper. Recently, Mo and Murray in [20] addressed the privacy-preserving average consensus problem, and they designed a novel Privacy Preservation Average Consensus (PPAC) algorithm to solve the problem. Using PPAC, the privacy-preserving and accurate DA can be achieved in the mean-square sense, while it is more desirable and more challenging to guarantee the privacy and accuracy in a deterministic manner.

To meet the above challenges of DA in ad hoc networks, in this work, we investigate the possibility of not relying on cryptography tools. To enable fully distributed additive data aggregation, we first analyze the conditions on the added noise in the consensus algorithms, which can guarantee that an average consensus can be achieved deterministically. Then, based on the given conditions, we design a secure consensus-based data aggregation (SCDA) algorithm that can achieve (ϵ, σ)-data-privacy and high accuracy in obtaining the sum and the average. Given the accuracy of the aggregation, our solution can be applied to other types of aggregation such as product, variance and other high-order statistics.

The main contributions and approaches of this work are summarized as follows. First, we exploited an average consensus algorithm to solve the privacy-preserving data aggregation (DA) problem in ad hoc networks. We derived a sufficient condition and a necessary condition of the noises added to the consensus process, under which an accurate aggregation is achieved. Based on the sufficient condition, a distributed SCDA algorithm is designed without using any trusted authority, so that the aggregator can obtain the aggregated results from any participating nodes. Second, we proved the convergence of the SCDA. To quantify the degree of the privacy protection, we introduced a novel privacy definition,

1: Dept. of Automation, Shanghai Jiao Tong University, and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai, China jphe@sjtu.edu.cn

2: Dept. of Electrical & Computer Engineering, University of Victoria, BC, Canada cai@ece.uvic.ca; pan@uvic.ca

3: State Key Lab of Industrial Control Technology, Zhejiang University, China pcheng@iipc.zju.edu.cn

4: Dept. of Electric and Computer Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong eesling@ust.hk

This research work is partially sponsored by the National Key R&D Program of China 2017YFE0114600, and Natural Science Foundation of China (NSFC) under grant 61828301, 61761136012, and the Natural Sciences and Engineering Research Council of Canada (NSERC). The work by Ling Shi is supported by a Hong Kong RGC General Research Fund 16208517.

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.2019.2910171, IEEE Transactions on Automatic Control

named (ϵ, σ)-data-privacy, which means that the probability that each node can infer its neighbor nodes' initial states in an ϵ interval is no larger than σ. We also proved that SCDA provides (ϵ, σ)-data-privacy, and the relationship between ϵ and σ has been derived.

The remainder of the paper is organized as follows. System model and problem formulation are presented in Section II. SCDA is proposed and analyzed in Sections III. Simulation evaluation is presented in Section IV, followed by concluding remarks and further research issues in Section V.

II. System Model and Problem Formulation

A. System Model

We consider an ad hoc network where nodes are self-organized into clusters (using an existing clustering algorithm [18]). We focus on a connected cluster with n nodes. The data from the nodes in the cluster are aggregated, while each individual's data should not be revealed to any other node (including the aggregator) or eavesdropper. The aggregator can poll any node in the cluster to acquire the aggregated data.

Two nodes can select each other as neighbors to exchange data with a logical link (a single-hop or multi-hop communication path) between them. Thus, an underlying logical network can be constructed. It should be noted that since a logical link can be a multi-hop communication path, the underlying network may not be equivalent to the physical communication network. The application of logical link is to hide the topology information from privacy attackers, and thus it can enhance privacy protection. For example, even an eavesdropper can eavesdrop all one-hop neighbors' information of node i, it cannot know which part of the information is used in the state update of node i. The underlying network is modeled as an undirected graph, $G = (V, E)$, where V is the set of nodes and E is the set of logical links (edges) between nodes. Let N_i be the neighbor set of node i, where $j \in N_i$ iff $(j, i) \in E$ (neighboring nodes are connected by logical links). Note that the logical links are negotiated in a distributed way, and thus node i knows its neighbor set N_i, but does not know the full topology of the underlying network.

Let $\mathbf{N}^+$ be the set of positive integers. Define the infinite norm as $\parallel x \parallel_\infty = \max\{|x_i|\}$, which is the maximum absolute value of all the elements of vector x. We use $\hat{[\circ]}$ to denote an estimation of $[\circ]$.

B. Problem Formulation

Denote the privacy-sensitive data of each node as $x_i(0)$, which is also called the initial state of node i. In this paper, we consider how to obtain the additive aggregation, i.e., $\sum_{i=1}^n x_i(0)$. The main design objectives are listed below. First, the aggregation should be obtained in a distributed manner, without the knowledge of the whole network topology, i.e., each node in the network (including any attacking node) does not have the full knowledge of the network topology[1]. Second, the computation and communication cost should be minimized. Lastly, each node's initial state should not be known to others (including its neighbors, the aggregator, and eavesdroppers) to preserve privacy, while the aggregation should be accurate.

To achieve the above objectives, we choose to devise the solution based on average consensus which is a well-known distributed algorithm. Given the total number of nodes (n), the sum is easily obtained by multiplying the average by n. [2]

In a nutshell, distributed average consensus computes the average of the initial data by local information exchanges among neighbors (in the underlying network). The state of each node is updated iteratively by taking a weighted sum of its current state and those of its neighbors. If the weights are carefully chosen, the states of all nodes will converge to their average after a number of iterations. To preserve privacy, each state being sent to the neighbors will be added with a noise. Denote by $x_i(k)$ the state of node i at iteration k. The information being sent out at k-th iteration is designed as

$$x_i^+(k) = x_i(k) + \theta_i(k), i \in V, \tag{1}$$

where θ_i is the noise for privacy preservation.

In each iteration, the state is updated as follows.

$$\begin{aligned} x_i(k+1) &= w_{ii}x_i^+(k) + \sum_{j \in N_i} w_{ij}x_j^+(k) \\ &= w_{ii}(x_i(k) + \theta_i(k)) + \sum_{j \in N_i} w_{ij}(x_j(k) + \theta_j(k)) \end{aligned} \tag{2}$$

for $i \in V$, where w_{ij}s are the weights. Here, $\theta_i(k)$ may not be necessary, while it is included to simplify the mathematical expression in both the formulation and proof.

To ensure that average consensus is achieved by the consensus algorithm and that the weights can be obtained in a distributed manner, we use Metropolis weights [8], given by

$$w_{ij} = \begin{cases} (1 + \max\{d_i, d_j\})^{-1}, & j \in N_i, \\ 1 - \sum_{l \in N_i} w_{il}, & i = j, \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

where d_i and d_j are the number of neighbors of node i and j in G, respectively. For a connected graph, a matrix with Metropolis weights is doubly stochastic.

Putting in the matrix form, we have

$$x(k+1) = W(x(k) + \theta(k)), \tag{4}$$

where $x, \theta \in R^n, W \in R^{n \times n}$ satisfying $x = [x_1, x_2, ..., x_n]^T$ and $\theta = [\theta_1, \theta_2, ..., \theta_n]^T$, and W is the matrix with Metropolis weights as its elements. Since W can be obtained in a distributed manner, we give an assumption as follows.

Assumption 2.1: It is assumed that the full knowledge of W_i of node i is unavailable to other nodes in the network, where W_i is the i-th row vector of matrix W.

Define the average state as $\bar{x} = \frac{1}{n}\sum_{i \in V} x_i(0)$. The problem is changed to design the noise process $\theta(k)$, such that

$$\lim_{k \to \infty} x_i(k) = \bar{x}, i \in V. \tag{5}$$

[1]This assumption is not presented in some existing works, e.g., [11], [20], [21], for differential privacy analysis.

[2]Using average consensus, we can obtain the average of $\log x_i$, $(x_i)^k$ (for $k = 2, 3...$) to calculate the product, variance, and other statistics.

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.2019.2910171, IEEE Transactions on Automatic Control

Using the Metropolis weights, W is doubly stochastic and the average consensus can be easily guaranteed when $\theta(k) = 0$ for all k [9], [10]; however, non-zero noise is necessary to preserve privacy. If the aggregation can tolerate some discrepancy, we have more freedom to design the noise process $\theta(k)$. For example, we can choose $\theta(k)$ to be mutually independent with an exponentially decaying co-variance matrix [11]. However, to achieve the exact average consensus, the added $\theta(k)$ has to ensure that the consensus result will not be affected and the privacy can be guaranteed, which implies that $\theta(k)$ must be carefully designed and correlated. In [20] PPAC was designed to guarantee the privacy and the exact average consensus, by adding and subtracting Gaussian and zero-sum noises to the consensus process. It is proved that PPAC has a mean-square convergence rate, i.e., an exact average consensus can be guaranteed by PPAC in the mean-square sense. However, what are the general conditions on the added noise that can guarantee the privacy and the exact average consensus is still an open issue. Different from [20] and [21], the convergence of the average consensus is deterministic, not in the sense of mean square or probability. We will conduct the analysis and design the algorithm to solve this problem.

III. Private and Accurate Data Aggregation

In this section, we first analyze the sufficient conditions and the necessary conditions on the added noise process such that a deterministic average consensus can be achieved. Then, based on the obtained conditions, we propose the SCDA algorithm and analyze its performance in terms of convergence, aggregation accuracy, privacy, and implementation complexity.

A. Algorithm Design

We first present a theorem, which provides a sufficient condition of deterministic average consensus and a theoretical support for our algorithm design.

Theorem 3.1: Considering the linear dynamic system (4), if the added noise vectors are bounded, i.e., $\|\theta(k)\|_\infty \le \alpha\rho^k$ for some $\alpha > 0$ and $\rho \in [0, 1)$, and the sum of all added noises satisfies $\sum_{k=0}^{\infty}\sum_{i=1}^{n}\theta_i(k) = 0$, then (5) holds true. Meanwhile, $\sum_{k=0}^{\infty}\sum_{i=1}^{n}\theta_i(k) = 0$ is a necessary condition.

The proof of Theorem 3.1 is given in Appendix A, where the proof of the convergence can be referred to Theorem 3 of [16]. Based on this theorem, if the noise process $\theta(k)$ satisfies the two conditions that $\|\theta(k)\|_\infty \le \alpha\rho^k$, i.e., exponentially decaying, and $\sum_{k=0}^{\infty}\sum_{i=1}^{n}\theta_i(k) = 0$, i.e., zero-sum, the goals of accurate and fast aggregation can be achieved. The exponentially decaying condition can ensure the convergence of the algorithm. The zero-sum condition ensures that the achieved consensus is an exact average consensus, which guarantees a fully accurate aggregation. Hence, Theorem 3.1 provides general conditions on the added noise which guarantees that an average consensus can be achieved deterministically. From the proof of Theorem 3.1, we have the following corollary.

Corollary 3.2: Consider the linear dynamic system (4). If there are h sub-sequences $\theta(\ell + kh)$ of noise process $\theta(j)$ and each sub-sequence satisfies $\|\theta(\ell+kh)\|_\infty \le \alpha\rho^k$ for some $\alpha > 0$ and $\rho \in [0, 1)$, and the noise process $\theta(\ell)$ satisfies the zero-sum condition, i.e., $\sum_{\ell=0}^{\infty}\sum_{i=1}^{n}\theta_i(\ell) = 0$, then $\lim_{k\to\infty} x_i(k) = \bar{x}$ for $i \in V$, where $\ell = 0, 1, ..., h-1$.

Based on Corollary 3.2, each node can randomly divide the noise adding process into several sub-sequences, such that the correlation between any pair of adjacent added noises is not clear to the other nodes.

Remark 3.3: Considering time-varying networks or directed networks, (4) is changed to $x(k+1) = W(k)(x(k)+\theta(k))$. In this case, an exact average consensus can still be achieved, if the weight matrices $W(k), k = 0, 1, \cdots$, are always doubly stochastic and the added noises satisfy the conditions in Theorem 3.1. Because the row-stochastically can ensure that each input vector (including $x(0)$) will converge to a constant, and the column-stochastically can keep the sum of the input vector unchange with iterations. Then, we can follow the proof of Theorem 3.1 to prove the exact average consensus.

Algorithm 1 : *SCDA Algorithm*

1: Select each element in $\theta_i(0)$ randomly from $[-\frac{\alpha}{2}\rho, \frac{\alpha}{2}\rho]$.
2: Let $x_i^+(0) = x_i(0) + \theta_i(0)$ and transmit $x_i^+(0)$ to its neighbor nodes.
3: Set $\delta_i(0) = \theta_i(0)$.
4: Set $k = 1$.
5: **while** $k <$ Max_Iteration_Number **do**
6: Update $x_i(k)$ with (4) based on $x_i^+(k-1)$ and $x_j^+(k-1)$ received from all neighbor nodes ($\forall j \in N_i$).
7: Select each element of $\delta_i(k)$ randomly or autonomously from $[-\frac{\alpha}{2}\rho^{k+1}, \frac{\alpha}{2}\rho^{k+1}]$, i.e.,

$$|\delta_i(k)| \le \frac{\alpha}{2}\rho^{k+1}, k \ge 1. \tag{6}$$

8: Set $\theta_i(k)$ according to

$$\theta_i(k) = \delta_i(k) - \delta_i(k-1). \tag{7}$$

9: Set $x_i^+(k)$ using (1), and then transmit $x_i^+(k)$ to its neighbor nodes.
10: $k = k + 1$.
11: **end while**

We further design the SCDA algorithm for node i in Algorithm 1. The Max_Iteration_Number in step 5 is given initially. According to our simulation, we can simply let Max_Iteration _Number equal n^2, which is sufficiently large to guarantee an accurate aggregation. We can also let each node terminate the iteration when it finds all its neighbors' states are sufficiently close to its own state, e.g., $|x_i(k) - x_j(k)| \le \varepsilon$ for $\forall j \in N_i$ and a given small ε. SCDA is a fully distributed algorithm. Only the neighbor set N_i is the input of each node i, and after sufficient iterations ($k \ge n^2$), all nodes' updated states could be the output of SCDA. Based on the output, the aggregator can easily achieve the goal of DA. In addition, we can also use the same approach given in [20] to prove that SCDA also converges at least in a mean-square sense.

B. Convergence and Accuracy of SCDA

The following theorem gives the convergence and accuracy of SCDA, and its proof is given in Appendix B.

Theorem 3.4: Using SCDA, we have (5) holds true, i.e., an average is achieved.

For each cluster, every node will achieve an average consensus using the SCDA algorithm, i.e., the aggregator can obtain the average state $\bar{x}$ from any node after the algorithm converges. Then, the sum can be obtained from using $n\bar{x}$, resulting in an accurate sum aggregation.

Remark 3.5: It follows from Theorem 3.4 that for SCDA, there exists $k_0 > 0$ such that $V(x(k)) < \varepsilon$ holds for $\forall k \geq k_0$ and $\varepsilon > 0$, where $V(x(k)) = \max(x(k)) - \min(x(k))$. However, this is not true for PPAC. The reason is that

$$\begin{aligned} V(x(k+1)) &= V(W(x(k)+\theta(k))) \\ &\geq |V(W\theta(k)) - V(Wx(k))|, \end{aligned}$$

where $\Pr\{V(W\theta(k)) \geq M\} > 0$ holds for any $M > 0$ since $f_{\theta_i(k)}(y) > 0$ holds for $\forall \theta_i(k) \in \theta(k)$ and $y \in R$. Thus, one infers that $V(x(k)) < \varepsilon$ cannot be guaranteed by PPAC for any given $k > 0$ with probability 1.

Note that the proof of Theorem 3.4 only used the properties of a doubly stochastic matrix and the results given in Theorem 3.1. SCDA can also be adopted to address the privacy of the asynchronous gossip consensus algorithms which also have the doubly stochastic matrixes in the algorithm dynamic functions, e.g., [13], [14]. However, considering the privacy of more complicated consensus algorithms, e.g., second-order consensus, e.g., [15], it is an open problem.

Remark 3.6: With SCDA, a higher accuracy of DA requires more iterations and an exact DA needs a sufficiently large number of iterations. It should be noticed that the larger communication delays will decelerate the convergence speed of SCDA. Hence, when the delays are not negligible, there is a tradeoff between convergence speed and DA accuracy, and we will discuss how to accelerate the convergence speed of SCDA at the end of this section.

C. Privacy of SCDA

For SCDA, node i only transmits the information sequence $x_i^+(k)$, $k = 0, 1, ...$, to its neighbors. For each message $x_i^+(k)$, there is a noise component $\theta_i(k)$ added to $x_i(k)$. Hence, any neighbor node cannot know the exact value of $x_i(0)$ based on the received information sequence from node i. Meanwhile, note that when $k \geq 1$, $x_i(k)^+$ is an updated state which may be quite different from the initial state $x_i(0)$, since each update is an averaging process among all the information received from its neighbor nodes' states. Define for $\forall j \in N_i$, the information set which is available for node i at iteration k as follows,

$$\begin{aligned} \mathcal{I}_i(k) = &\{x_i(0), x_i^+(0), ..., x_i(k), x_i^+(k); \\ &x_\ell^+(0), ..., x_\ell^+(k), \forall \ell \in N_i\}, \end{aligned}$$

where all the message of node i and the message output of its neighbors are included in $\mathcal{I}_i(k)$, and let $\mathcal{I}_i(\infty) = \lim_{k\to\infty} \mathcal{I}_i(k)$. Suppose that node i cannot listen to all the neighbors' information of node j. This assumption can be guaranteed in the underlying network construction with $N_j \not\subseteq N_i$, and it has been proved to be necessary in [20]. The added noises are assumed to be unknown to each node i, and the initial states of nodes are independent with each other.

Note that if node i does not have any prior information of $x_j(0)$ and no additional information is available for estimation, then it is unlikely to make an accurate estimation on $x_j(0)$ with a high probability. That is, we cannot make an accurate estimation directly if we do not have any information about the initial state of a node. Hence, when node i directly estimates node j's initial state without using any prior or side information, denoted by $\hat{x}_j^0(0)$, it is reasonable to assume

$$\Pr\{\hat{x}_j^0(0) \in [x_j(0) - \epsilon, x_j(0) + \epsilon]\} \ll \sigma, \tag{8}$$

where ϵ and σ are two given small positive constants, and $\sigma = \max_{\nu \in [-\frac{\alpha}{2}\rho, \frac{\alpha}{2}\rho]} \int_{\nu-\epsilon}^{\nu+\epsilon} f_{\theta_j(0)}(y)\mathrm{d}y$. This assumption can be extended to the case when side information may be available. For instance, if it is known that the state $x_j(0)$ is belong to the interval $[-M, M]$ with equal probability, we have

$$\Pr\{\hat{x}_j^0(0) \in [x_j(0) - \epsilon, x_j(0) + \epsilon]\} = \frac{\epsilon}{M}.$$

In this case, (8) still holds if there exists $[v - \epsilon, v + \epsilon]$ such that $f_{\theta_j(0)}(y) \gg \frac{1}{M}$ for $\forall y \in [v - \epsilon, v + \epsilon]$.

Under SCDA, the broadcast information of node j, i.e., $x_j^+(0), x_j^+(1), ..., x_j^+(k) \in \mathcal{I}_i(k)$, is available to node i to infer/estimate the initial value of neighbor node j. Note that the information output, $x_j^+(k)$, equals the weighted sum of the received information in the previous round plus a noise. Based on the information output, node i will take the probability over the space of all noises $\{\theta_j(k)\}_{k=0}^\infty$ (where the space is denoted by Θ) under the condition that $\mathcal{I}_i(\infty)$ is known, to estimate the values of the added noises. Then, using the difference between each information output and the estimated noises, we have $\hat{x}_j(0) = x_j^+(k) - \hat{\theta}_j^k$, where $\hat{\theta}_j^k$ is the estimation of random noise θ_j^k ($\theta_j^k = x_j^+(k) - x_j(0)$). Using this estimation, we have $|\hat{x}_j(0) - x_j(0)| = |\hat{\theta}_i^k - \theta_i^k|$, and

$$\Pr\{|\hat{x}_i(0) - x_i(0)| \leq \epsilon\} = \Pr\left\{|\hat{\theta}_i^k - \theta_i^k| \leq \epsilon\right\}. \tag{9}$$

To evaluate the privacy of SCDA, we give the definition of (ϵ, σ)-data-privacy as follows.

Definition 3.7: A distributed algorithm provides (ϵ, σ)-data-privacy, if, with information set $\mathcal{I}_i(\infty)$, the probability that each node i can successfully estimate its neighbor node j's initial value $x_j(0)$ in a given interval $[x_j(0) - \epsilon, x_j(0) + \epsilon]$ is no larger than σ, i.e.,

$$\sigma = \max_{\hat{\theta}_i^k \in \Theta, k \geq 0} \Pr\{|\hat{\theta}_i^k - \theta_i^k| \leq \epsilon\}. \tag{10}$$

In the above definition, ϵ indicates the estimation accuracy and σ expresses the privacy cost. Given the estimation accuracy ϵ, a smaller value of σ offers a stronger privacy guarantee.

Remark 3.8: For noise-adding privacy preserving solutions, no matter what type of noise distribution is used, there is a chance that an estimated value of the original data is close to the real data, but such a probability cannot be directly measured by differential privacy or the privacy metrics based on mutual information or Fisher information (e.g., given an estimation accuracy, the disclosed probability of initial states cannot be measured by the existing privacy metrics directly). Hence, it motivates us to introduce (ϵ, σ)-data-privacy, which is defined as the probability of ϵ-accurate estimate (the difference of an estimation and the original data is within ϵ) is no larger than σ (the disclosure probability). This definition reveals the relationship between the privacy and the estimation accuracy. Therefore, the propose privacy definition links the disclosure probability and the estimation accuracy directly,

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.2019.2910171, IEEE Transactions on Automatic Control

which is meaningful to quantify the data privacy in the applications of consensus. Indeed, the proposed (ϵ, σ)-data privacy defines a new metric which is critically important for many Internet-of-Things (IoT) applications, and different from the traditional differential privacy extensively studied in the literature, such as [20], [21].

Next, we prove that SCDA provides (ϵ, σ)-data-privacy, and obtain the following theorem with the proof in Appendix C.

Theorem 3.9: SCDA algorithm is (ϵ, σ)-data-private, and the relationship between ϵ and σ satisfies

$$\sigma = \max_{\nu \in [-\frac{\alpha}{2}\rho, \frac{\alpha}{2}\rho]} \int_{\nu-\epsilon}^{\nu+\epsilon} f_{\theta_j(0)}(y) \mathrm{d}y, \tag{11}$$

and $\lim_{\epsilon \to 0} \sigma = 0$, where $f_{\theta_j(0)}(y)$ is the probability density function (PDF) of $\theta_j(0)$.

Remark 3.10: It should be noted that Theorem 3.9 is obtained under the assumption that node i cannot listen to all the neighbors' information of node j. If this assumption is relaxed and node i has the knowledge of N_j, then at any time $k \geq 1$, node i can exactly calculate the value of $\theta_j(k)$ through the following equation,

$$\theta_j(k) = x_j(k) - [w_{jj} x_j(k-1) + \sum_{l \in N_j} w_{jl} x_l^+(k-1)],$$

where all the expressions on the right-hand side are known to node i. Hence, over the time, node i can calculate all of $\theta_j(k), ..., \theta_j(1)$. Then, using the zero-sum property of the noise, node i can calculate $\theta_j(0)$ by $\theta_j(0) = -\sum_{k=1}^{\infty} \theta_j(k)$. Therefore, node i knows the value of $x_j(0)$ through $x_j(0) = x_j^+(0) - \theta_j(0)$, i.e., $x_j(0)$ is released. This result is consistent with Theorem 4 in [20], which proved that the disclosed space of a node with m neighbors is of dimension $m + 1$.

D. Complexity of SCDA

Since each node just calculates a weighted average at each iteration, SCDA has very low computation complexity, in $O(n)$. According to our simulation results, when the underlying network is well connected (e.g., the diameter of the graph is much smaller than n), the consensus can be reached in $O(n)$ iterations. Note that since the number of hops is confined to the diameter of the cluster, we can also let nodes select logical neighbors within a small number of hops (e.g., 1 to 3). Thus, the communication cost is in $O(kn^2)$, where k is the number of iterations which is typically smaller than n for large n. We can further divide the network into more clusters to accelerate the convergence rate, while as a trade-off the aggregator needs to poll more nodes. The latest consensus algorithm proposed in [12] can guarantee that an average consensus is achieved in a few iterations, or nearly linear time. It thus can be applied to guarantee an ultrafast average consensus, which further reduces the communication cost of SCDA.

IV. Performance Evaluation

In this section, simulations are conducted to evaluate the performance of the SCDA.

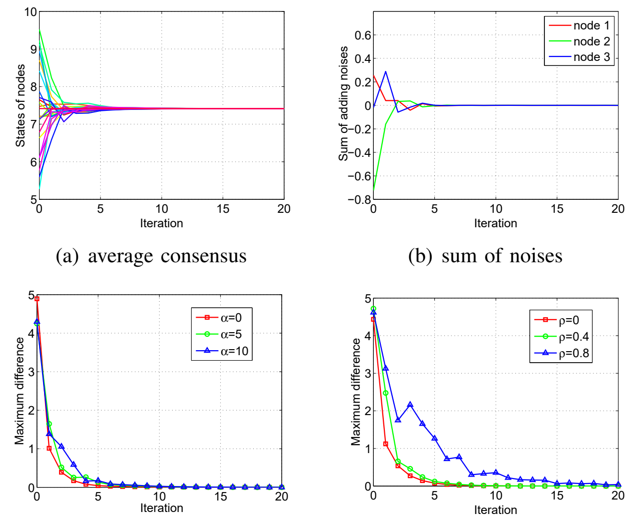
A. Simulation Setup

In the simulation, there are 100 nodes randomly deployed over a $1,000 \times 1,000$ m^2 square area, where the communication range of each node is 300 m. Unless otherwise stated, the whole area is divided into 4 equal-sized sub-areas and the nodes in each sub-area are clustered, i.e., there are 4 clusters.

We set $\alpha = 5$ and $\rho = 0.4$. Define the maximum difference between nodes' states in each cluster by $V(x(k))$. Clearly, a consensus is achieved if $V(x) = 0$.

B. Evaluation of SCDA

Figure 1(a) shows the dynamics of all nodes' states under SCDA. It is observed that the states of all 25 nodes converge exponentially to a constant state, which exactly equals the average of their initial states. This demonstrates that an average consensus can be achieved by SCDA, i.e., the aggregated sum is accurate. Figure 1(b) shows the sum of $\theta(k)$ used by the



(a) average consensus (b) sum of noises

(c) different α (d) different ρ

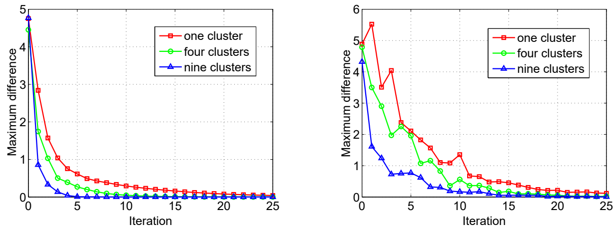
Fig. 1. The performance of SCDA.

nodes, where we randomly select three of them to illustrate. Clearly, the sum of each $\theta_i(k)$ converges to 0, which satisfies the conditions in Theorem 3.1.

Then, we change the values of α and ρ and the convergence of SCDA is shown in Figs. 1(c) and 1(d). When $\alpha = 0$ or $\rho = 0$, it means that the added noise is always 0. It is observed that the convergence rate is slightly affected by adding an exponentially decaying noise as the maximum difference is less than 10^{-4} within 20 iterations. Therefore, SCDA can preserve the privacy with guaranteed accuracy of aggregation.

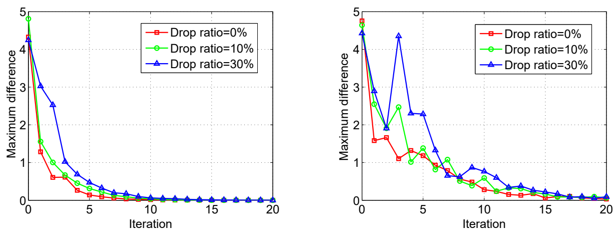
Next we study the convergence of SCDA under different clustering strategies by changing the number of clusters in the network. When there are several clusters in the network, we pick one cluster randomly to illustrate in Fig. 2. From the figure, with more clusters, SCDA has a faster convergence rate as anticipated. Furthermore, note that SCDA can have a fast convergence rate even if the node number of each cluster is large, e.g., with 100 nodes for the one-cluster case in Fig. 2, SCDA can converge to an acceptable accuracy (e.g., 10^{-3}) within 30 iterations.

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.2019.2910171, IEEE Transactions on Automatic Control



(a) $\alpha = 5$ and $\rho = 0.4$ (b) $\alpha = 10$ and $\rho = 0.8$

Fig. 2. The convergence of SCDA with different clusterings.



(a) $\alpha = 5$ and $\rho = 0.4$ (b) $\alpha = 10$ and $\rho = 0.8$

Fig. 3. The robustness of SCDA.

C. Robustness

Note that the communication delay, packet losses and the dynamic change of the topology may occur. Herein, when the delay is larger than a threshold (e.g., the interval between two iterations), the packet will be dropped. At each iteration, if a packet is lost or dropped, it is equivalent to that a logical link is broken at that iteration. In this case, the node just updates its state according to the successfully received neighbor information and adjusts the weights accordingly. In the following simulation, we randomly remove a portion of the logical links at each iteration to investigate the robustness of the proposed solutions.

As shown in Fig. 3, under different drop ratios (the percentage of links being broken in each iteration), SCDA can still converge, although the convergence rate will decrease slightly when the drop ratio becomes larger.

V. Conclusions and Further Discussions

In this paper, we have investigated the privacy-preserving data aggregation problem in ad hoc networks using the average consensus approach. We have proposed the SCDA algorithm to solve the problem. SCDA is simple to implement and can ensure private and accurate aggregation. SCDA does not rely on a centralized controller or a trusted aggregator, and it can be implemented in a distributed manner and robust against the network dynamics. Simulation results have shown that the proposed algorithm has fast convergence rate and high accuracy, and they are robust against network dynamics.

There are still many open issues worth further investigation. In this paper, the underlying network should be a connected, undirected graph. To ensure connectivity, a spanning tree connecting all the nodes in the cluster can be built and the links in the spanning tree should be included in the underlying network. How to deal with permanent node failures needs further investigation. The undirected graph requires bi-directional communications. In case bi-directional logical link cannot be maintained, novel consensus solutions need to be used which are much more complicated. How to apply this work to a more general scenario has attracted attentions, e.g., consider the network with dishonest nodes [19] and the optimal estimation under general noise adding mechanism [22]. Overall, using consensus can be a promising alternative to the heavily investigated privacy-preserving approaches using cryptography techniques in distributed systems. It is also possible to combine these two powerful tools to further enhance privacy and security, which beckons further research.

References

[1] T. Jung, X. Mao, X. Li, S. Tang, W. Gong, and L Zhang, "Privacy-preserving data aggregation without secure channel: Multivariate polynomial evaluation." in *Proc. IEEE Infocom*, 2013.

[2] W. He, X. Liu, H. Nguyen, K. Nahrstedt, and T. Abdelzaher, "PDA: Privacy-preserving data aggregation in wireless sensor networks." in *Proc. IEEE Infocom*, 2007.

[3] S. Ozdemir, and Y. Xiao, "Secure data aggregation in wireless sensor networks: A comprehensive overview." *Computer Networks*, 53(12), 2022-2037, 2009.

[4] J. Shi, Y. Zhang, and Y. Liu, "Prisense: privacy-preserving data aggregation in people-centric urban sensing systems." in *Proc. IEEE Infocom*, 2010.

[5] M. Groat, W. He, and S. Forrest, "KIPDA: k-indistinguishable privacy-preserving data aggregation in wireless sensor networks." in *Proc. IEEE Infocom*, 2011.

[6] Z. Erkin, J. Troncoso-Pastoriza, R. Lagendijk, and F. Perez-Gonzalez, "Privacy-preserving data aggregation in smart metering systems: An overview." *IEEE Signal Proc. Mag.*, 30(2), 75-86, 2013.

[7] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, 95(1): 215–233, 2007.

[8] L. Xiao, S. Boyd and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. IPSN*, 2005.

[9] A. Olshevsky and J. Tsitsiklis, "Convergence speed in distributed consensus and averaging," *SIAM Review*, 53(4): 747–772, 2011.

[10] T. Aysal, M. Coates, and M. Rabbat, "Distributed average consensus with dithered quantization," *IEEE Trans. Signal Proc.*, 56(10): 4905-4918, 2008.

[11] Z. Huang, S. Mitra, and G. Dullerud, "Differentially private iterative synchronous consensus," in *Proc. ACM workshop on Privacy in the Electronic Society*, 2012.

[12] A. Olshevsky, "Average consensus in nearly linear time on fixed graphs and implications for decentralized optimization and multi-agent control," *https://arxiv.org/pdf/1411.4186.pdf*, 2017.

[13] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Information Theory*, 52(6): 2508–2530, 2006.

[14] K. Cai, and H. Ishii, "Quantized consensus and averaging on gossip digraphs," *IEEE Trans. Automatic Control*, 56(9): 2087-2100, 2011.

[15] W. Ren, "Second-order consensus algorithm with extensions to switching topologies and reference models," in *Proc. IEEE ACC*, 2007.

[16] L. Schenato and F. Fiorentin, "Average timesynch: a consensus-based protocol for time synchronization in wireless sensor networks," *Automatica*, 1878–1886, 2011.

[17] J. He, P. Cheng, L. Shi, J. Chen and Y. Sun, "Time synchronization in WSNs: A maximum-value-based consensus approach," *IEEE Trans. Automat. Contr.*, 59(3): 660-675, 2014.

[18] S. Bandyopadhyay, and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks." in *Proc. IEEE Infocom*, 2003.

[19] J. He, L. Cai, P. Cheng, J. Pan and L. Shi, "Distributed privacy-preserving data aggregation against dishonest nodes in network systems," *IEEE Internet of Things Journal*, DOI: 10.1109/JIOT.2018.2834544.

[20] Y. Mo, and R. Murray, "Privacy preserving average consensus," *IEEE Trans Automat Contr.*, 62(2): 753–765, 2017.

[21] E. Nozari, P. Tallapragada, and J. Cortes, "Differentially private average consensus: Obstructions, trade-offs, and optimal algorithm design." *Automatica*, 81: 221-231, 2017.

[22] J. He, L. Cai, and X. Guan, "Preserving Data-Privacy with Added Noises: Optimal Estimation and Privacy Analysis." *IEEE Trans. Information Theory*, 64(8): 5677–5690, 2018.

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.2019.2910171, IEEE Transactions on Automatic Control

Appendix A
The proof of Theorem 3.1

First, we prove that each $x_i(k)$ is bounded by some constant M for $i \in V$. Since W is doubly stochastic, we have $\| W \|_\infty = 1$. Hence,

$$\begin{aligned} &\| x(k+1) \|_\infty = \| W(x(k)+\theta(k)) \|_\infty \\ &\leq \| W \|_\infty \| x(k)+\theta(k) \|_\infty \leq \| x(k) \|_\infty + \| \theta(k) \|_\infty \\ &\leq \| x(0) \|_\infty + \sum_{\ell=0}^{k} \| \theta(\ell) \|_\infty . \end{aligned} \tag{12}$$

Using the condition that $\|\theta(\ell)\|_\infty \leq \alpha\rho^\ell$, it follows

$$\begin{aligned} \| x(k+1) \|_\infty &\leq \| x(0) \|_\infty + \sum_{\ell=0}^{k} \alpha\rho^\ell \\ &\leq \| x(0) \|_\infty + \frac{\alpha}{1-\rho} = M, \end{aligned} \tag{13}$$

which implies that each $x_i(k)$ is bounded by M for all k.

Next, we prove the convergence of (4). The function $V(x(k))$ is nonnegative and has the property that $V(x(k)) = 0$ if and only if all the elements of $x(k)$ have the same values, i.e., $x(k) = C \cdot 1$, where C is a constant and 1 is a vector with all its elements equal to 1.

Note that W^ℓ is still a doubly stochastic matrix for $\ell \in N^+$, and we have $\lim_{\ell\to\infty} W^\ell = \frac{1}{n} 1^T 1$. Since the topology of each cluster is assumed to be connected, we have $W^n > 0$. Then, from Lemma 2 in [16], it follows that, for any vector y,

$$\max\{W^n y\} - \min\{W^n y\} \leq (1-\epsilon)(\max\{y\} - \min\{y\}), \tag{14}$$

where $\epsilon = \max_{j=1}^n \min_{i=1}^n (W^n)_{ij}, \epsilon \in (0,1)$. Hence, we have

$$\begin{aligned} &V(x(k+n)) = \max(x(k+n)) - \min(x(k+n)) \\ &\leq \max(W^n x(k)) - \min(W^n x(k)) \\ &+ \sum_{\ell=0}^{n} [\max(W^\ell \theta(k+n-\ell)) - \min(W^\ell \theta(k+n-\ell))] \\ &\leq (1-\epsilon)V(x(k)) + 2\sum_{\ell=0}^{n} \alpha\rho^{k+n-\ell} \\ &\leq (1-\epsilon)V(x(k)) + 2\alpha \frac{\rho^k(1-\rho^{n+1})}{1-\rho}, \end{aligned} \tag{15}$$

where we used the fact of (14). From (15), one infers that

$$\begin{aligned} &V(x(\ell+hn)) \leq (1-\epsilon)V(x(\ell+(h-1)n)) + \hat{\alpha}(\ell)\rho^{(h-1)n} \\ &\leq (1-\epsilon)^2 V(x(\ell+(h-2)n)) \\ &+ \hat{\alpha}(\ell)[\rho^{(h-1)n} + (1-\epsilon)\rho^{(h-2)n}] \\ &\leq (1-\epsilon)^l V(x(\ell+(h-l)n)) \\ &+ \hat{\alpha}(\ell)[\rho^{(h-1)n} + (1-\epsilon)\rho^{(h-2)n} + ... + (1-\epsilon)^{l-1}\rho^{(h-l)n}] \\ &\leq (1-\epsilon)^h V(x(\ell)) + \hat{\alpha}(\ell) h \max\{\rho^{(h-1)n}, (1-\epsilon)^{(h-1)}\}, \end{aligned}$$

for $\ell = 0,1,...,n-, l = 3,4,...,h$ and $h \in N^+$, where $\hat{\alpha}(\ell) = 2\alpha\rho^\ell \frac{(1-\rho^{n+1})}{1-\rho}$ is a constant. Since $\epsilon \in (0,1)$ and $\rho \in [0,1)$, $\lim_{h\to\infty} V(x(\ell+hn)) = 0$ for $\ell = 0,1,...,n-1$. Clearly, the above equation implies that $\lim_{k\to\infty} V(x(k)) = 0$, i.e.,

$$\lim_{k\to\infty} \max(x(k)) - \min(x(k)) = 0, \tag{16}$$

which means that the differences between elements of $x(k)$ will converge to zero. Then, we will prove that the sum of each column vector of $x(k)$ is a constant, and thus prove that an average consensus can be achieved.

Define $\sum(\circ)$ as the sum of all elements in $(\circ)$. Since W is still a doubly stochastic matrix, we have $\sum(Wx(k)) = \sum(x(k))$. Then, one obtains that

$$\begin{aligned} \sum(x(k)) &= \sum(Wx(k-1) + W\theta(k-1)) \\ &= \sum(x(0) + \sum_{\ell=0}^{k-1} \theta(\ell)). \end{aligned} \tag{17}$$

Taking limiting on both sides of the above equation yields

$$\lim_{k\to\infty} \sum(x(k)) = \sum(x(0) + \lim_{k\to\infty} \sum_{\ell=0}^{k-1} \theta(\ell)) = \sum(x(0)), \tag{18}$$

where we used the condition that $\sum[\sum_{\ell=0}^\infty \theta(\ell)] = 0$. Combining (16) and (18) yields that $\lim_{k\to\infty} x(k) = C \cdot 1 = \bar{x}1$, i.e., an average consensus is achieved.

It notes from (17) that

$$\lim_{k\to\infty} \sum_{i=1}^n x_i(k) = \sum_{i=1}^n x_i(0) + \lim_{k\to\infty} \sum_{i=1}^n \sum_{\ell=0}^{k-1} \theta_i(\ell).$$

If (5) holds, then $\sum_{i=1}^n x_i(\infty) = \sum_{i=1}^n x_i(0)$. It thus follows that the zero-sum condition is the necessary condition to achieve an exact average consensus with (4).

Appendix B
The proof of Theorem 3.4

We just need to prove that the SCDA algorithm can ensure the two conditions in Theorem 3.1.

First, we prove that the first condition, i.e., $\sum_{k=0}^\infty \sum_{i=1}^n \theta_i(k) = 0$, is ensured by SCDA. From step 1 and 4, one infers that $\theta_i(1) + \theta_i(0) = \delta_i(1)$ and $\theta_i(2) + \theta_i(1) + \theta_i(0) = \delta_i(2)$ for any $i \in V$. Then, by mathematical induction, one obtains that $\sum_{k=0}^\infty \theta_i(k) = \lim_{k\to\infty} \delta_i(k)$. From (6), one has that

$$\lim_{k\to\infty} |\delta_i(k)| \leq \lim_{k\to\infty} |\frac{\alpha}{2}\rho^{k+1}| = 0,$$

which implies that $\sum_{k=0}^\infty \theta_i(k) = 0$ for any $i \in V$. Hence, we have $\sum_{k=0}^\infty \sum_{i=1}^n \theta_i(k) = 0$.

Next we prove the added noise, $\theta(k)$, is exponentially decaying, i.e., $\|\theta(k)\|_\infty \leq \alpha\rho^k$. From (7) and (6), one infers

$$\begin{aligned} |\theta_i(k)| &= |\delta_i(k) - \delta_i(k-1)| \leq |\delta_i(k)| + |\delta_i(k-1)| \\ &\leq \frac{\alpha}{2}\rho^{k+1} + \frac{\alpha}{2}\rho^k \leq \alpha\rho^k. \end{aligned} \tag{19}$$

Thus, we have $\|\theta(k)\|_\infty \leq \alpha\rho^k$.

Appendix C
The proof of Theorem 3.9

To prove this theorem, we need to prove that at each iteration k, the probability that each node i can successfully infer that $x_j(0) \in [x_j(0)-\epsilon, x_j(0)+\epsilon]$ is no larger than σ

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.2019.2910171, IEEE Transactions on Automatic Control

using the information set $\mathcal{I}_i(k)$. In the following, we prove this result for each iteration.

At time $k=0$, node i can estimate neighbor j's initial value based on $\mathcal{I}_i(0)$ and use the fact that

$$x_j^+(0) = x_j(0) + \theta_j(0), \tag{20}$$

for estimation. Then, the corresponding estimation is given by

$$x_j^+(0) = \hat{x}_j(0) + \hat{\theta}_j(0). \tag{21}$$

Then, we have

$$\begin{aligned} &\Pr\left\{\hat{x}_j(0) \in [x_j(0)-\epsilon, x_j(0)+\epsilon]\right\} = \Pr\left\{|\hat{\theta}_j(0) - \theta_j(0)| \le \epsilon\right\} \\ &= \Pr\left\{\theta_j(0) \in [\hat{\theta}_j(0)-\epsilon, \hat{\theta}_j(0)+\epsilon]\right\} = \int_{\hat{\theta}_j(0)-\epsilon}^{\hat{\theta}_j(0)+\epsilon} f_{\theta_j(0)}(y)\mathrm{d}y. \end{aligned} \tag{22}$$

Note that $\hat{\theta}_j(0)$ is an estimation and could be any values in $[-\frac{\alpha}{2}\rho, \frac{\alpha}{2}\rho]$. Hence, we have

$$\begin{aligned} &\max_{\hat{x}_j(0)} \Pr\left\{\hat{x}_j(0) \in [x_j(0)-\epsilon, x_j(0)+\epsilon]\right\} \\ &= \max_{\nu \in [-\frac{\alpha}{2}\rho, \frac{\alpha}{2}\rho]} \int_{\nu-\epsilon}^{\nu+\epsilon} f_{\theta_j(0)}(y)\mathrm{d}y \end{aligned} \tag{23}$$

Hence, (ϵ,σ)-data-privacy is ensured at time $k=0$ for SCDA.

At time $k=1$, node i can estimate $x_j(0)$ based on $\mathcal{I}_i(1)$ and use the fact of both (20) and

$$\begin{aligned} \frac{x_j^+(1)}{w_{jj}} &= x_j^+(0) + \frac{1}{w_{jj}}[\sum_{l\in N_j} w_{jl}x_l^+(0) + \theta_j(1)] \\ &= x_j(0) + \theta_j(0) + \frac{1}{w_{jj}}[\sum_{l\in N_j} w_{jl}(x_l(0)+\theta_l(0)) + \theta_j(1)]. \end{aligned} \tag{24}$$

If using (20) only, we also have (23). Then, we consider the estimation with (24). Let $f_{\theta_j'(1)}(z)$ be the PDF of $\theta_j'(1)$, where

$$\begin{aligned} \theta_j'(1) &= \theta_j(0) + \frac{1}{w_{jj}}[\sum_{l\in N_j} w_{jl}x_l^+(0) + \theta_j(1)] \\ &= \theta_j(0) + \frac{1}{w_{jj}}\theta_j(1) + \sum_{l\in N_j}\frac{w_{jl}}{w_{jj}}x_l^+(0) \\ &= \theta_j(0) + \frac{1}{w_{jj}}\theta_j(1) + \theta_j^{''}(1). \end{aligned} \tag{25}$$

Based on (24), one can make estimation, $\frac{x_j^+(1)}{w_{jj}} = \hat{x}_j(0) + \hat{\theta}_j'(1)$. Let $\tilde{\theta}_j(1) = \theta_j(0) + \frac{1}{w_{jj}}\theta_j(1)$. Then, we have

$$\begin{aligned} &\max \Pr\{|\hat{\theta}_j'(1) - \theta_j'(1)| \le \epsilon\} \\ &\le \max \Pr\{|\hat{\theta}_j^{'}(1) - \theta_j^{'}(1)| \le \epsilon | \tilde{\theta}_j(1)\} \\ &= \max \Pr\{|\hat{\theta}_j^{'}(1) - \tilde{\theta}_j(1) - \theta_j^{''}(1)| \le \epsilon\} \\ &\le \max \Pr\{|\hat{\theta}_j^{''}(1) - \theta_j^{''}(1)| \le \epsilon\}, \end{aligned} \tag{26}$$

where $\hat{\theta}_j^{''}(1) = \hat{\theta}_j^{'}(1) - \tilde{\theta}_j(1)$ is viewed as an estimation of $\theta_j^{''}(1)$, and we have used the independence between variables $\theta_j(0) + \frac{1}{w_{jj}}\theta_j(1)$ and $\theta_j^{''}(1)$. Since node i cannot listen to all the neighbors' information of node j, there exists at least one independent variable $x_l^+(0)$ in $\sum_{l\in N_j} w_{jl}x_l^+(0)$ that is unknown to node i (i.e., no information of $x_l^+(0)$ is available to node i) to estimate the value of $\theta_j^{''}(1)$. From (8), we have

$$\Pr\{|\hat{\theta}_j^{''}(1) - \theta_j^{''}(1)| \le \epsilon\} \le \max_{\nu \in [-\frac{\alpha}{2}\rho, \frac{\alpha}{2}\rho]} \int_{\nu-\epsilon}^{\nu+\epsilon} f_{\theta_j(0)}(y)\mathrm{d}y. \tag{27}$$

Combining (26) and (27), we have

$$\begin{aligned} &\Pr\{\hat{x}_j(0) \in [x_j(0)-\epsilon, x_j(0)+\epsilon]\} \\ &\le \max_{\nu \in [-\frac{\alpha}{2}\rho, \frac{\alpha}{2}\rho]} \int_{\nu-\epsilon}^{\nu+\epsilon} f_{\theta_j(0)}(y)\mathrm{d}y. \end{aligned}$$

Then, using (20) and (24) together for estimation, we have

$$\begin{aligned} &\Pr\{\hat{x}_j(0) \in [x_j(0)-\epsilon, x_j(0)+\epsilon]\} \\ &\le \max_{\substack{\nu \in [-\frac{\alpha}{2}\rho, \frac{\alpha}{2}\rho] \\ \mu\in[b_1,B_1]}} \int_{\nu-\epsilon}^{\nu+\epsilon}\int_{\mu-\epsilon}^{\mu+\epsilon} f_{\theta_j(0),\theta_j'(1)}(y,z)\mathrm{d}z\mathrm{d}y \\ &\le \max_{\substack{\nu \in [-\frac{\alpha}{2}\rho, \frac{\alpha}{2}\rho] \\ \mu\in[b_1,B_1]}} \int_{\nu-\epsilon}^{\nu+\epsilon}\int_{\mu-\epsilon}^{\mu+\epsilon} f_{\theta_j'(1)|\theta_j(0)}(z|y) f_{\theta_j(0)}(y)\mathrm{d}z\mathrm{d}y \\ &\le \max_{\nu \in [-\frac{\alpha}{2}\rho, \frac{\alpha}{2}\rho]} \int_{\nu-\epsilon}^{\nu+\epsilon} f_{\theta_j(0)}(y)\mathrm{d}y, \end{aligned} \tag{28}$$

where $[b_1, B_1]$ $(B_1 - b_1 > 0)$ is an interval including all the possible value of $\theta_j'(1)$. The above result means that if we combine the two facts for estimation, it will not enhance the successful estimation probability. Therefore, at time $k=1$, we still have (23) and (ϵ,σ)-data-privacy is still ensured.

At each iteration k, with similar analysis, there are $k+1$ facts (equations) can be used for estimation. Based on the $(k+1)$-th equation, i.e.,

$$\begin{aligned} \frac{x_j^+(k)}{[W^k]_{jj}} &= \frac{1}{[W^k]_{jj}}\left[[W^k]_j x(0) + \sum_{l=0}^{k}[W^{k-l}]_j\theta(l)\right] \\ &= x_j(0) + \tilde{\theta}_j(k) \\ &+ \left[\frac{[W^k]_j}{[W^k]_{jj}}x(0) + \sum_{l=0}^{k}\frac{[W^{k-l}]_j}{[W^k]_{jj}}\theta(l) - x_j(0) - \tilde{\theta}_j(k)\right] \\ &= x_j(0) + \tilde{\theta}_j(k) + \theta_j^{''}(k) = x_j(0) + \theta_j'(k) \end{aligned} \tag{29}$$

where $[W^k]_j$ denotes the j-th row vector of W^k, $[W^k]_j'$ is a vector obtained from setting $[W^k]_{jj} = 0$ for $[W^k]_j$, and $\tilde{\theta}_j(k) = \sum_{l=0}^{k}\frac{[W^{k-l}]_{jj}}{[W^k]_{jj}}\theta_j(l)$. Then, with the similar analysis of (26) and (27), we can obtain the following equation,

$$\begin{aligned} &\Pr\{\hat{x}_j(0) \in [x_j(0)-\epsilon, x_j(0)+\epsilon]\} \\ &\le \Pr\{|\hat{\theta}_j^{''}(k) - \theta_j^{''}(k)| \le \epsilon\} \le \max_{\nu \in [-\frac{\alpha}{2}\rho, \frac{\alpha}{2}\rho]} \int_{\nu-\epsilon}^{\nu+\epsilon} f_{\theta_j(0)}(y)\mathrm{d}y, \end{aligned}$$

where we have used the fact that $\hat{\theta}_j^{''}(k)$ contains the independent variables with no information available to node i. Also, if we combine the equations together, we can prove that the successful estimation probability cannot be increased. That is, (23) holds and (ϵ,σ)-data-privacy is proved at iteration k.

From the above discussion, one concludes that (23) holds and (ϵ,σ)-data-privacy is guaranteed by SCDA. Meanwhile, note that $f_{\theta_j(0)}(\nu)$ is the PDF function of $\theta_j(0)$, it follows that $\lim_{\epsilon\to 0}\sigma = 0$.