Privacy-Preserving Push-Sum Average Consensus via State Decomposition

Xiaomeng Chen ⊙, Lingying Huang ⊙, Kemi Ding ⊙, Subhrakanti Dey ⊙, and Ling Shi ⊙, Fellow, IEEE

Abstract—Average consensus is extensively used in distributed networks for computation and control, where all the agents constantly communicate with each other and update their states in order to reach an agreement. Under a general average consensus algorithm, information exchanged through wireless or wired communication networks could lead to the disclosure of sensitive and private information. In this article, we propose a privacy-preserving push-sum approach for directed networks that can protect the privacy of all agents while achieving average consensus simultaneously. Each node decomposes its initial state arbitrarily into two substates, and their average equals to the initial state, guaranteeing that the agent’s state will converge to the accurate average consensus. Only one substate is exchanged by the node with its neighbors over time, and the other one is reserved. That is to say, only the exchanged substate would be visible to an adversary, preventing the initial state information from leakage. Different from the existing state-decomposition approach, which only applies to undirected graphs, our proposed approach is applicable to strongly connected digraphs. In addition, in direct contrast to offset-adding-based privacy-preserving push-sum algorithm, which is vulnerable to an external eavesdropper, our proposed approach can ensure privacy against both an honest-but-curious node and an external eavesdropper. A numerical simulation is provided to illustrate the effectiveness of the proposed approach.

Index Terms—Average consensus, multiagent systems, privacy preserving, state decomposition.

I. INTRODUCTION

With increasing applications in smart grids, smart buildings, and intelligent transportation systems, etc, the cooperative distributed algorithm has been a heated research topic during the last decade. When all the components of a network reach a common agreement, we say that the distributed system reaches a consensus. One of the most commonly adopted consensus algorithm is the average consensus algorithm, where each agent aims to reach the average of their initial values. The convergence of average consensus is first proved by DeGroot [1] and further studied by other researchers (see, e.g., [2] and [3]). Typical applications of average consensus include distributed sensor fusion [4], load balancing in parallel computing [5], and coordinated control [6].

Conventional average consensus algorithm requiring each node to exchange their state information with the neighboring nodes in order to reach the average consensus is not desirable if the participating nodes have sensitive and private information. In addition, by hacking into communication links, an external eavesdropper has access to state information, which is exchanged through wireless or wired communication networks. As the number of privacy leakage events is increasing, there is an urgent need to preserve privacy of each agent in distributed systems.

Several approaches have been proposed in recent years to protect privacy. The main idea of most existing privacy-preserving approaches is to mask signals by adding noises. Nozari et al. [7] proposed a differentially private consensus algorithm by adding some uncorrelated noises. However, it cannot converge to the exact average due to the tradeoff between the privacy and accuracy. To improve this tradeoff, Mo and Murray [8] devised a new mechanism where exchanged information is masked by correlated noises, and the convergence to the correct average is also guaranteed. Another strand of research emerged recently is observability-based privacy-preserving approaches. Alaeddini et al. [9] guaranteed the privacy protection by minimizing the information about a certain node from the observability perspective.

None of the aforementioned approaches, however, is suitable for directed graphs with weak topological restrictions. To preserve privacy of nodes interacting on an unbalanced graph, Charalambous et al. [10] proposed an offset-adding privacy-preserving approach, and Gao et al. [11] protected privacy by adding randomness on edge weights, both of which are only effective against honest-but-curious nodes. To improve resilience to external eavesdroppers, Hadjicostis and Garcia [12] employed homomorphic encryption to maintain privacy, relying on a trusted node. As a consequence, this approach requires a large amount of computation and communication, which may be inapplicable for systems with limited resources. Aiming at protecting privacy against both honest-but-curious nodes and eavesdroppers, Wang [13] proposed a privacy-preserving mechanism in which the state of a node is decomposed into two substates. However, it is inapplicable to directed graphs and only considers the eavesdropper that is unaware of the entire information about the network. To summarize, the main challenges arising in the design and analysis of privacy-preserving average consensus algorithm are as follows.

1) Communication over directed graphs: In practice, the information flows among sensors may not be bidirectional due to the different communication ranges, e.g., the coordinated vehicle control problem [14] and the economic dispatch problem [15]. Owing to the unbalanced interactions over agents, the approaches in [7], [8], and [13] fail to achieve average consensus in such scenarios.

Authorized licensed use limited to: University of California, San Francisco. Downloaded on July 28, 2023 at 20:45:49 UTC from IEEE Xplore. Restrictions apply.
2) **Definition of privacy preserving:** The privacy notion of differential privacy is not suitable for the privacy-preserving mechanism that is not based on noise injection. Therefore, a new privacy notion is needed to measure the privacy degree.

3) **Existence of eavesdroppers:** Different from honest-but-curious nodes, an external eavesdropper is a stronger adversary that has access to more information. It is difficult to protect privacy against an external eavesdropper with low computation load.

In this article, based on the concepts introduced in [13], we propose a state-decomposition-based privacy-preserving algorithm for directed networks that can maintain the privacy of all agents while guaranteeing the accuracy of the average consensus at the same time. In addition, we specify the estimation strategy of the eavesdropper that knows more information than [13] and provide analytical results on its estimation performance. The main contributions of this article are summarized as follows.

1) We propose a novel privacy-preserving push-sum algorithm for strongly connected digraphs, which addresses explicitly the constraints imposed by the topology of the communication network (Algorithm 3). Furthermore, using coefficients of ergodicity [16], we prove the convergence of our proposed approach to the exact average of the initial value (Theorem 1).

2) Different from the privacy notion in [10] and [12], which only considers the exact initial value, we define the privacy preservation against honest-but-curious nodes, where the privacy of each node is preserved if honest-but-curious nodes have infinite uncertainty (see Definition 4 for more details) on the initial value based on the accessible information. Moreover, we prove that our proposed approach can preserve privacy of each node against honest-but-curious nodes for certain topological conditions (Theorems 2 and 3).

3) We analyze the privacy-preserving performance of the proposed algorithm in the presence of an eavesdropper (see Definition 2) and prove that the estimation error of the eavesdropper cannot be bounded in probability (Theorem 4). In contrast, the estimation error converges to zero under the existing privacy-preserving approaches in [10].

**Notations:** In this article, \( \mathbb{N} \) and \( \mathbb{Z}_+ \) represent the sets whose components are natural numbers and positive integers. \( N'(\mu, \sigma^2) \) denotes the Gaussian distribution with mean \( \mu \) and covariance \( \sigma^2 \). \( U(a, b) \) denotes the uniform distribution over the interval \( (a, b) \). For an arbitrary vector \( x \), we denote its \( i \)th element by \( x_i \). For an arbitrary matrix \( M \), we denote its element in the \( i \)th row and \( j \)th column by \( M_{ij} \). \( |x| \) denotes the greatest integer less than or equal to \( x \), and \( \lfloor x \rfloor \) denotes the smallest integer larger than or equal to \( x \). Finally, “w.p. \( p \)” stands for “with probability \( p \)” in this article.

## II. PRELIMINARIES

### A. Network Model

We consider a directed graph (digraph) \( G = (\mathcal{V}, \mathcal{E}) \) with \( N \) nodes, where \( \mathcal{V} = \{1, 2, \ldots, N\} \) denotes the node set and \( \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V} \) denotes the edge set, respectively. A communication link from node \( i \) to node \( j \) is denoted by \( (j, i) \in \mathcal{E} \), indicating that node \( i \) can send messages to node \( j \). The self-loop is not included in the directed graph \( G \).

The nodes who can directly send messages to node \( i \) are represented as in-neighbors of node \( i \), and the set of these nodes is denoted as \( N_{i}^{\text{in}} = \{j \in \mathcal{V} \mid (i, j) \in \mathcal{E}\} \). Similarly, the nodes who can directly receive messages from node \( i \) are represented as out-neighbors of node \( i \), and the set of these nodes is denoted as \( N_{i}^{\text{out}} = \{j \in \mathcal{V} \mid (j, i) \in \mathcal{E}\} \).

### Algorithm 1: General Push-Sum Algorithm.

**Step 1.** Node \( i \in \mathcal{V} \) initializes \( x_{i,1}(0) = x_i(0) \) and \( x_{i,2}(0) = 1 \). The coupling weight between node \( j \) and node \( i \) is denoted as \( p_{ji} \) and the self-weight of node \( i \) is denoted as \( p_{ii} \).

**Step 2.** At iteration \( k \):

1) Node \( i \) randomly chooses a set of weights, \( \{p_{ji}(k) \sim U(0, 1) \mid j \in N_{i}^{\text{out}} \cup \{i\}\} \), and then normalizes them such that \( \sum_{j=1}^{N} p_{ji}(k) = 1 \). Also, \( p_{ji}(k) \) is set to be 0 if \( (j, i) \notin N_{i}^{\text{out}} \).

2) Node \( i \) computes \( p_{ji}(k)x_{i,1}(k) \) and \( p_{ji}(k)x_{i,2}(k) \), and sends them to its out-neighbors \( j \in N_{i}^{\text{out}} \).

3) After receiving the information from its in-neighbors \( j \in N_{i}^{\text{in}} \), node \( i \) updates \( x_{i,j} \) as follows:

\[
x_{i,j}(k+1) = \sum_{j \in N_{i}^{\text{in}}(i)} p_{ij}(k)x_{j,j}(k), \quad l = 1, 2.
\]

4) Node \( i \) computes the estimated average

\[
\hat{x}_{i}^{\text{ave}}(k+1) = \frac{x_{i,1}(k+1)}{x_{i,2}(k+1)}.
\]

In this article, we consider that the initial value of each node is not equal to the average of them\(^1\), but each node runs the proposed privacy-preserving push-sum algorithm (Algorithm 3) to infer their average.

**Assumption I:** The digraph \( G \) is assumed to be strongly connected with \( N \) nodes, where \( N > 2 \). In other words, there exists at least one directed path from any node \( i \) to any node \( j \) in the digraph with \( i \neq j \).

### B. General Push-Sum Algorithm

The push-sum algorithm, introduced originally in [17], aims to achieve average consensus for each node communicating on a directed graph with relatively weak topological restrictions. Consider a network of \( N \) nodes, where each node has a private initial state, termed as \( x_i(0) \) for node \( i \in \mathcal{V} \). Without loss of generality, we assume that the initial state is a scalar. In the push-sum algorithm, each node generates two values, \( x_{i,1}(k) \) and \( x_{i,2}(k) \), both of which are updated in the same way. The algorithm is described as follows.

It is known that the general push-sum algorithm can reach the exact average if the network is strongly connected and the matrix \( \mathbf{P}(k) \) is column stochastic, where \( \mathbf{P}(k) = [p_{ij}(k)] \) with \( p_{ij}(k) \) defined in [18], Algorithm 1.

### C. Privacy Leakage

In this article, we consider two types of adversaries, which are defined as follows.

**Definition I:** An honest-but-curious adversary is a node that follows the system’s protocol and attempts to infer the private information of other nodes under the knowledge of its received data. In addition, it can

\(^1\)The probability of the event that the initial value of a node equals to the average value of all nodes is zero if each initial value is a continuous random variable.
Algorithm 2: Eavesdropper Estimating Algorithm.

Step 1. The external eavesdropper initializes \( s_1(0) = x_{i,l}^0(0) \) and \( s_2(0) = x_{i,l}^1(0) \).

Step 2. At iteration \( k \):

1) The eavesdropper computes \( p_{ij}(k) = 1 - \sum_{j \neq i} p_{ji}(k) \), and updates \( s_1(k) \) and \( s_2(k) \) as follows:

\[
s_1(k+1) = s_1(k) + x_{ij}^0(k + 1) - \sum_{j \in N^i \cup \{i\}} p_{ij}(k)x_{ij}^0(k),
\]

where \( l = 1, 2 \).

2) The initial value of node \( i \) is estimated by

\[
\hat{x}_i^0(k) = \frac{s_1(k)}{s_2(k)}.
\]

Remark 1: Algorithm 2 constructs an effective observer for an eavesdropper to estimate the initial value of node \( i \), which mimics the system under the general push-sum Algorithm 1. By adopting Algorithm 2, the eavesdropper can successfully estimate the initial value of each node which runs the general push-sum algorithm and existing privacy-preserving approaches in [10] as \( \lim_{k \to \infty} x_i^0(k) = x_i(0) \), where the theoretical analysis is easily to be obtained and the simulation results are shown in Section IV. For different average consensus algorithms, the estimation algorithms of the eavesdroppers would be different accordingly. To the best of our knowledge, Algorithm 2 in this article is the first proposed algorithm that is effective for directed graphs.

Algorithm 3: Privacy-Preserving Push-Sum Algorithm.

Initialization:

1) Node \( i \in V \) randomly generates an initial substate value \( x_i^0(0) \) from \( U((-M,M)) \), where \( M > 0 \) is a pre-defined value, and initializes \( x_i^0(0) = 2x_i(0) - x_{i,l}^0(0) \), \( x_{i,l}^0(0) = 0 \) and \( x_{i,l}^1(0) = 2 \), where \( x_i(0) \) denotes the private initial state of node \( i \).

Weight generation:

1) For \( k = 0 \), node \( i \in V \) randomly chooses a set of weights, \( \{p_{ij}(0) \mid j \in N^i \cup \{i\}\} \) from \( N(0,M) \), and then normalizes them such that \( \sum_{j = 1}^{N} p_{ij}(0) + \alpha_i(0) = 1 \). Further, \( p_{ji}(k) \) is set to be 0 if \( j \notin N^i \).

2) For \( k \geq 1 \), node \( i \) first generates \( \alpha_i(k) \) from \( U(0,1) \), then it chooses a set of weights, \( \{p_{ij}(k) \sim U(0, 1) \mid j \in N^i \cup \{i\}\} \), and normalizes them such that \( \sum_{j = 1}^{N} p_{ij}(k) = 1 - \alpha_i(k) \). Further, \( p_{ji}(k) \) is set to be 0 if \( j \notin N^i \).

State update: For all \( k \geq 0 \)

1) Node \( i \in V \) computes \( p_{ij}(k)x_{ij}^0(k) \) and \( p_{ij}(k)x_{ij}^1(k) \), and sends them to its out-neighbors \( j \in N^i \).

2) After receiving the information from its in-neighbors \( j \in N^i \), node \( i \) updates its two substates \( x_{ij}^0(k + 1) \) and \( x_{ij}^1(k + 1) \) as follows:

\[
\begin{align*}
x_{ij}^0(k + 1) &= \sum_{j \in N^i \cup \{i\}} p_{ij}(k)x_{ij}^0(k) + x_{ji}^0(k), \\
x_{ij}^1(k + 1) &= \alpha_i(k)x_{ij}^1(k),
\end{align*}
\]

with \( i \in V, l = 1, 2 \).

3) Node \( i \) computes the estimated average

\[
\hat{x}_{i,l}(k + 1) = \frac{x_{i,l}^0(k + 1)}{x_{i,l}^1(k + 1)}. \]

B. Convergence Analysis

In this section, we prove convergence of Algorithm 3.

Under Algorithm 3, as shown in Fig. 1, we can view the substate \( x_{ij}^0(k), l \in \{1, 2\} \) as the state of a virtual node \( \hat{v}_l \), which can only communicate with the actual node \( i \). In other words, all substates compose a digraph \( G' \) with 2N nodes, and the sum of the initial values of all nodes in \( G' \) is twice of that in the original graph \( G \).

Using matrix–vector notation, the iteration rule in (1) can be rewritten as follows:

\[
\begin{align*}
x_1(k + 1) &= \tilde{P}(k)x_1(k) \\
x_2(k + 1) &= \tilde{P}(k)x_2(k)
\end{align*}
\]
where
\[ x_1(k) = [x_{1,1}^0(k), ..., x_{N,1}^0(k)], \]
\[ x_2(k) = [x_{1,2}^0(k), ..., x_{N,2}^0(k)], \]
\[ \hat{P}(k) = \begin{bmatrix} P(k) & I_{N \times N} \end{bmatrix} \]
with \( P(k) = [p_{1,1}(k)] \) and \( \Lambda(k) = \text{diag}(\alpha_1(k), ..., \alpha_N(k)) \) [see definitions of \( p_{1,1}(k) \) and \( \alpha_i(k) \) in Algorithm 3]. Then, we can obtain
\[
\begin{align*}
&x_1(k+1) = \hat{P}(k) \cdots \hat{P}(1) x_1(1) \\
&x_2(k+1) = \hat{P}(k) \cdots \hat{P}(1) x_2(1).
\end{align*}
\] (3)

Let \( T_k \) denote the product
\[ T_k = \hat{P}(k)\hat{P}(k-1) \cdots \hat{P}(1). \] (4)

We can easily check that matrix \( T_k \) is column stochastic since it is the product of column-stochastic matrices. Next, we use the coefficient of ergodicity [16] to establish the convergence of Algorithm 3.

**Definition 3:** For a column-stochastic matrix \( P_i \), the coefficient of ergodicity \( \delta(P_i) \) is defined as follows:
\[
\delta(P_i) = \max_{j} \max_{i_1, i_2} |P_i(j, i_1) - P_i(j, i_2)|.
\]

From Definition 3, it can be seen that \( \delta(P_i) \) characterizes how different two columns of \( P_i \) are; in particular, \( \delta(P_i) = 0 \) if and only if the columns of \( P_i \) are identical.

**Lemma 1:** Suppose Assumption 1 holds. The coefficient of ergodicity \( \delta(T_k) \) converges almost surely to zero.

**Proof:** Define
\[
W_i(k) = \prod_{t=1}^{Nt} \hat{P}(k) \quad \forall t \in \mathbb{Z}_+.
\]
where \( N \) is the number of actual nodes in the network. Note that there are paths from each virtual node \( v^i \) to its corresponding actual node \( i \) and the actual nodes in \( \mathcal{V} \) are strongly connected. Hence, the digraph corresponding to the matrix \( \hat{P}(k) \) (diagonal excluded) has paths from all nodes to the actual nodes for any \( k \in \mathbb{Z}_+ \). Furthermore, since \( \hat{P}(k) \) has positive diagonal entries at the location of the actual nodes for any \( k \in \mathbb{Z}_+ \), any node \( i \) or \( i^0 \) has at least one path of length \( N \) to any actual node in \( \mathcal{V} \). Thus, there is at least one row with all entries strictly positive in \( W_i \), \( \forall t \in \mathbb{Z}_+ \). Let \( \alpha_1 > 0 \) be the minimum value of the entry in such a row. Thus, we have \( \alpha_1 > e^N \) for all \( t \in \mathbb{Z}_+ \), where \( \epsilon = \min_{i,j \in \mathcal{V}, p_{ij}(k) > 0} |P_{ij}(k)| \).

Next, following the standard results on coefficients of ergodicity (see, e.g., [16] and [20]), we can obtain that if \( y^t = x^t W_i \), then
\[
\max_{i} y_i - \min_{i} y_i \leq (1 - \alpha_t)(\max_{i} x_i - \min_{i} x_i).
\]

Meanwhile, for a generic column stochastic matrix \( P_c \), we have that if \( y^t = x^t P_c \), then
\[
\max_{i} y_i - \min_{i} y_i \leq (\max_{i} x_i - \min_{i} x_i).
\]

The forward product \( T_k = \hat{P}(k) \hat{P}(k-1) \cdots \hat{P}(1) \) can be assembled in blocks of length \( N \) i.e.,
\[
T_k = \prod_{t=k+1}^{k+N} \hat{P}(t) \prod_{t=1}^{[k/N]} W_t
\]
where \( \prod_{t=k+1}^{k+N} \hat{P}(t) = 1 \).

Iterating, we have that if \( y^t = x^t T_k \), then
\[
\max_{i} y_i - \min_{i} y_i \leq \prod_{t=1}^{[k/N]} (1 - \alpha_t)(\max_{i} x_i - \min_{i} x_i) \]
\[
\leq (1 - e^N)^{[k/N]}(\max_{i} x_i - \min_{i} x_i).
\]

By varying \( x \) among the vectors of the canonical basis, we have
\[
\max_{i} T_k|_{i} - \min_{i} T_k|_{i} \leq (1 - e^N)^{[k/N]} \quad \forall i, j \in \{1, 2, ..., N\},
\]
i.e., \( \delta(T_k) \leq (1 - e^N)^{[k/N]} \). As \( k \to \infty \), \( (1 - e^N)^{[k/N]} \to 0 \) happens with probability 1, and then, we have \( \delta(T_k) \to 0 \) which finishes the proof.

The following theorem shows that the privacy-preserving push-sum algorithm converges to the exact average.

**Theorem 1:** For a digraph satisfying Assumption 1, under Algorithm 3, the estimated average \( \bar{x}_{\text{est}}(k+1) \) will converge to the average of all initial values \( \sum_{i=1}^{N} x_i(0) / N \) with probability one. Also, \( x_{c,1}^0(k+1) / x_{i,2}^0(k+1) \) will converge to the average with probability one.

**Proof:** Since \( \hat{P}(0) \) is column stochastic, we have
\[
1^\top x_1(1) = 1^\top \hat{P}(0) x_1(0) = 1^\top x_1(0) = 2 \sum_{i=1}^{N} x_i(0)
\]
\[
1^\top x_2(1) = 1^\top \hat{P}(0) x_2(0) = 1^\top x_2(0) = 2 N.
\]

Lemma 1 implies that \( \delta(T_k) \to 0 \), i.e., \( T_k \), tends to have identical columns with probability one. Thus, we have \( \hat{P}(\infty) \cdots \hat{P}(1) = v^1 \), where \( v = [v_i] \) is a stochastic vector. Therefore, the estimated average can be obtained as follows:
\[
\bar{x}_{\text{est}}(t) = \frac{x_{c,1}^0(t)}{x_{i,2}^0(t)} = \frac{\hat{P}(\infty) \cdots \hat{P}(1) x_1(1)}{\hat{P}(\infty) \cdots \hat{P}(1) x_2(1)}
\]
\[
= \frac{v^1 \cdot x_1(1)}{v^1 \cdot x_2(1)} = \frac{v_1 \cdot x_1(1)}{v_1 \cdot x_2(1)}
\]
\[
= \frac{2 \sum_{i=1}^{N} x_i(0)}{N} = \frac{\sum_{i=1}^{N} x_i(0)}{N} \quad \text{w.p.1.}
\] (6)

Furthermore,
\[
\frac{x_{1,1}^0(\infty)}{x_{1,2}^0(\infty)} = \frac{\hat{P}(\infty) \cdots \hat{P}(1) x_1(1)}{\hat{P}(\infty) \cdots \hat{P}(1) x_2(1)}
\]
\[
= \frac{v^1 \cdot x_1(1)}{v^1 \cdot x_2(1)} = \frac{v_1 \cdot x_1(1)}{v_1 \cdot x_2(1)}
\]
\[
= \frac{2 \sum_{i=1}^{N} x_i(0)}{N} = \frac{\sum_{i=1}^{N} x_i(0)}{N} \quad \text{w.p.1.}
\] (7)

**Remark 2:** It is worth noting that we assume that the system \( \mathcal{G} \) with \( N \) nodes is expanded to the system \( \mathcal{G}' \) with \( 2N \) nodes for the convenience of convergence analysis. In fact, the number of system nodes does not change. Thus, there is no need to build an additional communication structure under the proposed algorithm, which is desirable if the communication resource is limited.

**C. Privacy-Preserving Performance Analysis Against Honest-but-Curious Nodes**

In this section, we prove that Algorithm 3 protects privacy against honest-but-curious nodes.

According to Definition 1, we consider a set of honest-but-curious nodes \( \mathcal{L} \) aiming to infer the initial value of node \( i \in \mathcal{L} \) based on
the information accessible to it, where $L = \mathcal{V} \setminus \mathcal{A}$ denotes the set of legitimate nodes. Under Algorithm 3, the information set accessible to the set of honest-but-curious nodes $\mathcal{A}$ at time $k$ can be defined as

$$\mathcal{I}_A(k) = \{I_n(k) \mid a \in \mathcal{A}\}$$  \(\text{(8)}\)

where

$$I_n(k) \triangleq \{x_{n,i}^{a,k}(j), x_{n,j}^{a,k}(j), p_{jn}(k), p_{jn}(k) x_{n,j}^{a,k}(j) \mid p \in N_{n,i}^a, j \in \mathcal{V}, l = 1, 2\}.$$ 

Given time instant $\kappa \in \mathbb{N}$, the honest-but-curious nodes $\mathcal{A}$ obtain such a set of information sequence $\mathcal{I}_A(0 : \kappa) = \cup_{i \in \kappa} \mathcal{I}_A(k_i)$ for any feasible set $\mathcal{I}_A(0 : \kappa)$, the term $\Delta(\mathcal{I}_A(0 : \kappa), i)$ denotes the set of all initial values $x_i(0)$ at node $i$ that there exists a set of $x_{n,i}^{a,k}(0)$, $x_{n,j}^{a,k}(0)$, $n \in L$, and a sequence of $p_{jn}(k)$, $j \in \mathcal{V}$, $k = 0, 1, \ldots, \kappa$ such that the sequence of $x_{n,i}^{a,k}(k)$, $x_{n,j}^{a,k}(k)$, $p_{jn}(k)$, $p_{jn}(k) x_{n,j}^{a,k}(k)$, $a \in \mathcal{A}, p \in N_{n,i}^a, j \in \mathcal{V}, l = 1, 2$ generated by Algorithm 3 is equal to that in the adversary information set $\mathcal{I}_A(0 : \kappa)$.

The set $\Delta(\mathcal{I}_A(0 : \kappa), i)$ includes all possible initial values of node $i$ that can generate $\mathcal{I}_A(0 : \kappa)$ in (8). The diameter of $\Delta(\mathcal{I}_A(0 : \kappa), i)$ is defined as follows:

$$\text{Diam}(\mathcal{I}_A(0 : \kappa)) = \sup_{x_i(0), x_i(0) \in \Delta(\mathcal{I}_A(0 : \kappa), i)} |x_i(0) - x_i(0)|.$$ 

**Definition 4:** The privacy of node $i \in \mathcal{L}$ is preserved against a set of honest-but-curious nodes $\mathcal{A}$ if, for any $\kappa \in \mathbb{N}$, $\text{Diam}(\mathcal{I}_A(0 : \kappa)) = \infty$ for any feasible $\mathcal{I}_A(0 : \kappa)$.

**Definition 4** shares a similar idea to the uncertainty-based privacy notion in [21], which is inspired from the notion of $l$-diversity [22]. In $l$-diversity, the diversity of the discrete-valued sensitive data is measured by the number of different valuations for the data, and a larger diversity leads to a larger uncertainty on the sensitive data. In our problem, we view the continuous-valued $x_i(0)$ as the sensitive data, whose diversity is measured by the diameter of the set $\Delta(\mathcal{I}_A(0 : \kappa), i)$. A larger diameter represents a larger diversity/uncertainty.

**Remark 3:** The abovementioned uncertainty-based notion is advantageous to other privacy notions to a certain degree since it does not require any specific statistical model for the system inputs and outputs. In practical scenarios, the inputs of the system and the communication weights may not follow any specific probabilistic distribution. The probability-based privacy notion might, thus, become restrictive or even unrealistic. Furthermore, the proposed privacy notion is closely related to nonstrong observability [23] in control theory, where a dynamic system is said to be not strongly observable if at least one entry of the initial state is unobservable, i.e., cannot be uniquely determined. Note that in our problem setting, we say that the privacy is preserved if the diameter of set $\Delta(\mathcal{I}_A(0 : \kappa), i)$ is infinite for any feasible $\mathcal{I}_A(0 : \kappa)$ for any $\kappa \in \mathbb{N}$, achieving the largest possible diversity. That is to say, the adversary is not able to find a unique value or even a meaningful range of $x_i(0)$, which is similar to an unobservability-based privacy notion and is more stringent than the privacy definition in [8] and [11], which adopt the noise-adding approach to preserve privacy, since the estimated value could reside in a range around the true initial value.

Without loss of generality, we consider the scenario on how to protect the initial value of node $i$.

**Theorem 2:** For a digraph satisfying Assumption 1, the privacy of node $i$ can be preserved against a set of honest-but-curious nodes $\mathcal{A}$ under Algorithm 3 if $N_{n,i}^a \cup N_{n,i}^m \nsubseteq \mathcal{A}$.

**Proof:** Since $N_{n,i}^a \cup N_{n,i}^m \nsubseteq \mathcal{A}$, there exists at least one node $m$ that belongs to $N_{n,i}^a \cup N_{n,i}^m$ but not $\mathcal{A}$. Fix any $\kappa \in \mathbb{N}$ and any feasible information set $\mathcal{I}_A(0 : \kappa)$. We denote $\{x_{n,1}(0)^{a,k}, x_{n,1}(0)^{a,k}, p_{jn}(k) \mid n \in L, j \in \mathcal{V}, k = 0, 1, \ldots, \kappa\}$ as an arbitrary set of initial substrate values and weights that satisfies $\mathcal{I}_A(0 : \kappa)$. Hence, we have where $x_i(0)^{a,k} = (x_{n,1}(0)^{a,k} + x_{n,1}(0)^{a,k})/2$. We then denote $x_i(0)^{a,k}$ as $x_i(0)^{a,k} = x_i(0)^{a,k} + e$, where $e$ is an arbitrary real number.

Next we show that there exists a set of values $\{x_{n,1}^a(0), x_{n,1}(0)^{a,k}, p_{jn}(k), n \in L, j \in \mathcal{V}, k = 0, 1, \ldots, \kappa\}$ that makes $x_i(0)^{a,k} \in \Delta(\mathcal{I}_A(0 : \kappa))$. (Note that the valuation of the initial value of node $i$ should still guarantee the convergence to the original average after altering $x_i(0)^{a,k}$ to $x_i(0)^{a,k} - e$, i.e., the sum of all nodes’ initial values does not change.) The initial substrate values $x_{n,1}^a(0), x_{n,1}^a(0)$ denoted as follows, which satisfy $x_{n,1}^a(0) + x_{n,1}^a(0) = 2x_{n,1}^a(0)$ \(\forall n \in L\).

$$\begin{align*}
x_{n,1}^a(0) &= x_{i,1}^a(0), x_{i,1}^a(0) = x_{i,1}^a(0) + 2e \quad (9)
\end{align*}$$

Then we derive the division into two situations: $m \in N_{n,i}^a$ and $m \notin N_{n,i}^a$. 

**Situation I:** Consider $m \in N_{n,i}^a$, then the information set sequence accessible to set $A$ evaluates to $\mathcal{I}_A(0 : \kappa)$ under the initial substrate values in (9) and the following weights:

$$\begin{align*}
\alpha_n(k)^{a,k} &= \alpha_n(k)^{a,k} \quad \forall n \in L, k = 0, \ldots, \kappa
p_{mn}(k)^{a,k} &= (p_{mn}(k)^{a,k}x_{n,1}^a(0) + 2e)/x_{n,1}^a(0) \quad (10)
\end{align*}$$

**Situation II:** Consider $m \notin N_{n,i}^a$, then the information set sequence accessible to set $A$ evaluates to $\mathcal{I}_A(0 : \kappa)$ under the initial substrate values in (9) and the following weights:

$$\begin{align*}
\alpha_n(k)^{a,k} &= \alpha_n(k)^{a,k} \quad \forall n \in L, k = 0, \ldots, \kappa
p_{mn}(k)^{a,k} &= (p_{mn}(k)^{a,k}x_{n,1}(0)^{a,k} - 2e)/x_{n,1}(0)^{a,k} \quad (11)
\end{align*}$$

**Summarizing Situations I and II,** we have that $x_i(0)^{a,k} = x_i(0)^{a,k} + e \in \Delta(\mathcal{I}_A(0 : \kappa))$, then

$$\text{Diam}(\Delta(\mathcal{I}_A(0 : \kappa)) \geq \sup_{e \in \mathbb{C}} |x_i(0)^{a,k} - (x_i(0)^{a,k} + e)| = \sup_{e \in \mathbb{C}} |e| = \infty.$$ 

The abovementioned analysis holds for any $\kappa \in \mathbb{N}$ and any feasible $\mathcal{I}_A(0 : \kappa)$. Therefore, by **Definition 4**, the privacy of node $i$ preserved against a set of honest-but-curious nodes $\mathcal{A}$ if node $i$ has at least one legitimate neighbor node $m \notin \mathcal{A}$. 

**Remark 4:** The values of $x_{1,2}(0)$ and $x_{2,1}(0)$ are known to set $A$ since each node $q$ initializes $x_{q,2}(0) = 0$ and $x_{2,1}(0) = 2 \forall q \in \mathcal{V}$. Nevertheless, the knowledge of $x_{1,2}(0)$ and $x_{2,1}(0)$ does not help the adversary to infer $x_i(0)$ since the transmitted data $p_{ij}(k)x_{n,1}(0) \in \mathcal{N}_{n,i}$ is equal to zero under any $p_{ij}(k)$.
Note that depending on the value of \( c \), the weights \( p_{mn}(0)^r, p_{mj}(0)^r, p_{mj}(0)^r \) in (10) and (11) could be outside the range (0, 1). To ensure \( (x_i(0)^r + e) \in \Delta(Z_0(0 : k)) \) for arbitrary real number \( c \), the weights at \( k \) should be unrestricted, which is consistent with the weight selection in Step 1.2 of Algorithm 3.

**Theorem 3:** For a digraph satisfying Assumption 1, the initial value of any node \( i \) can be uniquely inferred in an asymptotic sense by a set of honest-but-curious nodes \( A \) if \( N^{\text{out}} \cup N^{\text{in}}_{i} \subseteq A \).

Proof: Since \( N^{\text{out}} \cup N^{\text{in}}_{i} \subseteq A \), i.e., all the neighbors of node \( i \) belong to set \( A \), we have
\[
z_i(k + 1) - z_i(k) = \sum_{m \in N^{\text{in}}_{i}} p_{mi}(k)x_{m,i}^{\alpha}(k) - \sum_{m \in N^{\text{out}}_{i}} p_{im}(k)x_{i,m}^{\alpha}(k)
\]
where \( z_i(k) = x_{i,i}^{\alpha}(k) + x_{i,l}^\beta(k), l \in [1, 2] \).

With the information accessible to set \( A \) and \( z_2(0) = 2 \), \( z_2(k) \) can be easily computed as follows:
\[
z_2(k) = z_2(0) + \sum_{l=0}^{k-1} \left( \sum_{m \in N^{\text{in}}_{i}} p_{mi}(k)x_{m,i}^{\alpha}(k) - \sum_{m \in N^{\text{out}}_{i}} p_{im}(k)x_{i,m}^{\alpha}(k) \right).
\]
As \( k \) goes to infinity, average consensus will be achieved asymptotically, i.e.,
\[
\lim_{k \to \infty} z_2(k) = \lim_{k \to \infty} z_2(0) \quad \text{and} \quad \lim_{k \to \infty} z_{1}(k) = \lim_{k \to \infty} \frac{z_2(k)x_{i,i}^{\alpha}(k)}{x_{i,i}^{\alpha}(k)}.
\]

Then, \( z_1(0) \) can be obtained by
\[
z_1(0) = \lim_{k \to \infty} z_1(k) = \frac{1}{\sum_{l=0}^{k-1} \left( \sum_{m \in N^{\text{in}}_{i}} p_{mi}(k)x_{m,i}^{\alpha}(k) - \sum_{m \in N^{\text{out}}_{i}} p_{im}(k)x_{i,m}^{\alpha}(k) \right)}.
\]

Therefore, \( x_i(0) \) is uniquely estimated by set \( A \) in an asymptotic sense if \( N^{\text{out}} \cup N^{\text{in}}_{i} \subseteq A \).

**Remark 5:** Theorems 2 and 3 imply that the single neighbor configuration should be avoided in order to preserve privacy, which is also indicated in other privacy-preserving approaches, such as [11], [17], and [19].

**D. Privacy-Preserving Performance Analysis Against External Eavesdroppers**

In this section, we show that an eavesdropper defined in Definition 2 fails to estimate the initial value of each node when the system runs Algorithm 3.

**Definition 5:** The privacy of node \( i \) is preserved against an eavesdropper defined in Definition 2 if for any \( c > 0 \), there always exists a constant \( 0 < \delta < 1 \) and a natural number \( k_1 \), such that for all \( k > k_1 \),
\[
\Pr(|\hat{x}_i^p(k) - x_i(0)| > c) > \delta.
\]

Intuitively, this means that there is no guarantee the estimation error will be bounded by any finite \( c \), i.e., the estimated value cannot converge to any neighborhood of the actual value with probability 1.

**Theorem 4:** For a digraph satisfying Assumption 1, the privacy of node \( i \) can be preserved by Algorithm 3 against an external eavesdropper defined in Definition 2.

Proof: The main idea of this proof is to show that for any \( c > 0 \), there always exists a constant \( 0 < \delta < 1 \) and a natural number \( k_1 \), such that for all \( k > k_1 \),
\[
\Pr(|\hat{x}_i^p(k) - x_i(0)| > c) > \delta.
\]
Without the knowledge of existing substates, an external eavesdropper would treat the exchange information \( p_{ij}(k)x_{i,j}^\alpha(k) \) as \( p_{ij}(k)x_{i,j}^\alpha(k) \). We then employ Algorithm 2 to estimate the initial value of node \( i \). The resulting estimation error is denoted as \( |\hat{x}_i^p(k) - x_i(0)| = |(a_\beta - \gamma_k)/2 - \beta_k| \), where \( \beta_k = \alpha_k(k - 1)x_{i,i}^{\alpha}(k - 1), \gamma_k = \hat{\alpha}_x(k - 1)x_{i,i}^{\alpha}(k - 1) - \alpha_k(k - 1)x_{i,i}^{\alpha}(k - 1), \hat{d} = \sum_{i=1}^{N} x_i(0)/N, \) and \( a = x_i(0) - \hat{d} \). Without loss of generality, we assume \( a > 0 \); otherwise, one can replace \( a \) and \( \gamma_k \) by \( -a \) and \( -\gamma_k \), respectively, and follow the same analysis hereinafter.

1) From the convergence analysis in (6), we have
\[
\lim_{k \to \infty} \alpha_k(k - 1)(x_{i,i}^{\alpha}(k - 1) - \hat{x}_i^p(k - 1)) = 0
\]
i.e., \( \lim_{k \to \infty} \gamma_k = 0 \). Hence, there exists a natural number \( k_2 \) such that for all \( k > k_2, |\gamma_k| < a \). Then, for any \( k > k_2 \) and \( \beta_k \in (2 - \delta_1, 2 - \delta_2) \), where \( 0 < \delta_2 < \delta_1 < \frac{a}{c - a} \), we have
\[
|\hat{x}_i^p(k) - x_i(0)| > |(a_\beta - a)/(2 - \beta_k)|.
\]
Therefore, for \( k > k_2 \)
\[
\Pr(|\hat{x}_i^p(k) - x_i(0)| > c, \beta_k \in (2 - \delta_1, 2 - \delta_2)) > \Pr(|(a_\beta - a)/(2 - \beta_k)| > c, \beta_k \in (2 - \delta_1, 2 - \delta_2)).
\]

2) If \( \beta_k \in (2 - \delta_1, 2 - \delta_2), \) we can obtain \( |(a_\beta - a)/(2 - \beta_k)| > c \), i.e.,
\[
\Pr(|(a_\beta - a)/(2 - \beta_k)| > c, \beta_k \in (2 - \delta_1, 2 - \delta_2)) = 1.
\]

3) Since \( p_{ij}(k)\forall j \in N^{\text{out}}_{i} \cup \{i\} \) are random variables from the same distribution over the range (0,1) for any \( k \geq 1 \), one has \( p_{ij}(k) \in (1/2^{1/2}, 1) \) with a nonzero probability, where \( k_3 = [\log_{1/2^{1/2}}(0.25k^N)/(N^{1/2})] \) and \( \zeta_1 = 1.4^{1/2} - 2^{1/2} + (1 - e^N)k_1 < 1 \).

Denote \( \min_{i \in V} \Pr(p_{ij}(k) \in (1/2^{1/2}, 1)\forall j \in N^{\text{in}}_{i} \cup \{i\}, k \geq 1) = p_1 \). We attempt to prove
\[
\Pr(x_{i,i}^{\alpha}(k - 1) > 2 - \delta_2) \geq p_1^{k_1N} \forall k > k_3N + 1.
\]
From (3), we have \( x_{i,i}^{\alpha}(k - 1) = 2\hat{P}(k - 2 : 1)\) \( [\begin{array}{c} 1 \\ 0 \end{array}]^T \)
\[
\text{implying that} \ x_{i,i}^{\alpha}(k - 1) = 2\sum_{j=1}^{N} [\hat{P}(k - 2 : 1)]_{ij}, \text{ where} \ 
\hat{P}(k - 2 : 1) = \hat{P}(k - 2) \hat{P}(k - 3) \ldots \hat{P}(2) \hat{P}(1) \text{ with} \ 
\hat{P}(k : k) = \hat{P}(k).
\] Thus, in order to prove inequality (13), it suffices to prove
\[
\Pr\left(\min_{j} \left\{ \hat{P}(k - 2 : 1)_{ij} \right\} > 1 - 0.5\delta_2 \right) \geq p_1^{k_1N} \forall k > k_3N + 1.
\]
Since the digraph \( G \) is strongly connected, there must exist a path from node \( l \) to node \( i \). Denote the path as \( v_1 \rightarrow v_2 \rightarrow \cdots \rightarrow v_r \), where \( v_1 = l, v_r = i, \) and \( r \leq N \). Then, we have
\[
\left\{ \hat{P}(k - 2 : k - k_3N - 1) \right\}_{ij} \geq \zeta_2
\]
where $\zeta_2 = p_{41}(k-2)p_{41}(k-3)\ldots p_{41}(k-k_3N + r - 2)\cdot p_{v_{y_{v-1}}} (k-k_3N + r - 3)\ldots p_{v_{y_{v2}}}(k-k_3N)p_{v_{y_{v1}}}(k-k_3N - 1)$.

From inequality (5), we can obtain $\min_j [\hat{P}(k-2 : k-k_3N - 1)]_{ij} \geq \zeta_2 (1 - e^{-N})^{k_3}$.

If $p_{41}(k-2), \ldots, p_{41}(k-k_3N + r - 2), p_{v_{y_{v-1}}} (k-k_3N + r - 3), \ldots, p_{v_{y_{v2}}}(k-k_3N - 1)$ are all in the range $(\zeta_1^{k_3N}, 1)$, we have

$$\zeta_2 > \frac{1 - 0.5\delta_2}{N - 1} + (1 - e^{-N})^{k_3}$$

i.e., $\min_j [\hat{P}(k-2 : k-k_3N - 1)]_{ij} > \frac{1 - 0.5\delta_2}{N}$.

Hence, for all $k > k_3N + 1$

$$\Pr \left( \min_j [\hat{P}(k-2 : k-k_3N - 1)]_{ij} > \frac{1 - 0.5\delta_2}{N} \right) \geq p_1^{k_3N}.$$ 

Moreover, since $\hat{P}(k-k_3N - 2 : 1)$ is column stochastic and $\hat{P}(k-2 : 1) = \hat{P}(k-2 : k-k_3N - 1)\hat{P}(k-k_3N - 2 : 1)$, we can conclude that if $\min_j [\hat{P}(k-2 : k-k_3N - 1)]_{ij} > \frac{1 - 0.5\delta_2}{N}$, one has $\min_j [\hat{P}(k-2 : 1)]_{ij} > \frac{1 - 0.5\delta_2}{N}$. Therefore,

$$\Pr \left( \min_j [\hat{P}(k-1 : 1)]_{ij} > \frac{1 - 0.5\delta_2}{N} \right) \geq p_1^{k_3N},$$

which implies that inequality (13) holds.

4) Since for any $k \geq 1$, $\alpha_i (k-1) \sim U(0, 1)$ and $x_i^{n_2}(k) < 2N$, we have $\forall k > k_3N + 1$,

$$\Pr \left( \alpha_i (k-1) \in \left( \frac{2 - \delta_1}{x_i^{n_2}(k-1)}, \frac{2 - \delta_2}{x_i^{n_2}(k-1)} \right) \right) = \frac{\delta_1 - \delta_2}{2 N}.$$ 

Hence,

$$\Pr \left( \beta_k \in (2 - \delta_1, 2 - \delta_2) \right) > \delta \quad \forall k > k_3N + 1,$$

where $\delta = \frac{(\delta_1 - \delta_2)^{k_3N}}{2N}$.

Therefore, for all $k > k_i = \max\{k_2, k_3N + 1\}$

$$\Pr \left( |\hat{x}_i^n(k) - x_i(0)| > c \right) \geq \Pr \left( |\hat{x}_i^n(k) - x_i(0)| > c, \beta_k \in (2 - \delta_1, 2 - \delta_2) \right) > \Pr \left( |(a\beta_k - a)/(2 - \beta_k)| > c, \beta_k \in (2 - \delta_1, 2 - \delta_2) \right) = \Pr \left( |(a\beta_k - a)/(2 + \beta_k)| > c, \beta_k \in (2 - \delta_1, 2 - \delta_2) \right) > \Pr \left( \beta_k \in (2 - \delta_1, 2 - \delta_2) \right) = \Pr \left( \beta_k \in (2 - \delta_1, 2 - \delta_2) \right) > \delta,$$

i.e., the privacy of node $i$ is preserved against an eavesdropper which runs Algorithm 2.

**IV. SIMULATIONS**

In this section, we illustrate the effectiveness of our proposed algorithm and show its advantages over existing privacy-preserving approaches on directed graphs. We first verify the convergence performance of our proposed algorithm communicating on a strongly connected digraph with $N = 5$ nodes, shown in Fig. 2. The initial values for all nodes are chosen from $U(0, 50)$, $c$ is set to be 500 and $M$ is set to be 100. The evolution of the network and the convergence of $\hat{x}_i^n$ are shown in Fig. 3. It is shown that the convergence is achieved, which is consistent with the theoretical results. Moreover, Fig. 4 shows that the estimation error cannot be always bounded by $c$, i.e., the privacy of node 5 is preserved against an external eavesdropper.

Unlike the privacy-preserving approach in [10] and [11], where the convergence only happens after $k = L + 1$ ($L$ is a randomly chosen integer), our proposed approach starts converging to the average at the iteration $k = 1$. Fig. 5 shows the evolution of mean square error (mse) under different privacy-preserving approaches, where $L$ is set to be 10. According to Fig. 5, our proposed approach converges faster than others and the mse is much smaller before $k \leq L$.

Next, we investigate the privacy-preserving performance of our proposed approach. Without loss of generality, we assume that an eavesdropper is interested in the initial value of node 5 and applies Algorithm 2 to estimate it.

Fig. 6 shows that the evolution of estimated average and the eavesdropper’s estimated value under the privacy-preserving approach in [10]. It can be seen that as the network converges and the estimated
Evolution of mse under different privacy-preserving approaches.

average value reaches the true average value, the eavesdropper’s estimated value also converges to the initial value of node 5. Therefore, the error between $\tilde{x}_5(0)$ and $x_5(0)$ is bounded, i.e., the privacy of node 5 cannot be preserved against an eavesdropper in [10].

It can be seen in Table I that, besides homomorphic encryption [12], our proposed algorithm can be effective in a number of situations. Furthermore, our proposed algorithm is built on the simple multiplications and addition steps (can be computed in $O(1)$ time). By comparison, the implementation of [12] requires extra cryptosystems to perform complicated modular exponentiation steps (see [24] for more details about its computation and time complexity). Hence, in contrast to privacy-preserving approach in [12], our proposed algorithm covers a wider range of applications.

V. CONCLUSION

In this article, we proposed a privacy-preserving push-sum algorithm based on state decomposition for systems interacting on directed graphs. While protecting privacy from honest-but-curious nodes, our approach can guarantee the convergence to exact average. Moreover, in contrast to the offset-adding approach, our proposed approach can prevent an external eavesdropper from estimating the initial value of each node. Furthermore, our proposed algorithm has lower computation and communication complexity, which can be easily implemented in practice. Future work includes analyzing the convergence rate of the privacy-preserving algorithm and studying other types of consensus problems.

REFERENCES

[1] M. H. DeGroot, “Reaching a consensus,” J. Amer. Stat. Assoc., vol. 69, no. 345, pp. 118–121, 1974.
[2] S. Chatterjee and E. Seneta, “Towards consensus: Some convergence theorems on repeated averaging,” J. Appl. Probability, vol. 14, no. 1, pp. 89–97, 1977.
[3] J. Tsitsiklis, D. Bertsekas, and M. Athans, “Distributed asynchronous deterministic and stochastic gradient optimization algorithms,” IEEE Trans. Autom. Control, vol. 31, no. 9, pp. 803–812, Sep. 1986.
[4] L. Xiao, S. Boyd, and S. Lall, “A scheme for robust distributed sensor fusion based on average consensus,” in Proc. Int. Symp. Inf. Process. Sensor Netw., 2005, pp. 63–70.
[5] J. E. Boillat, “Load balancing and poisson equation in a graph,” Concurrency: Pract. Experience, vol. 2, no. 4, pp. 289–313, 1990.
[6] W. Ren and R. W. Beard, “Consensus seeking in multiagent systems under dynamically changing interaction topologies,” IEEE Trans. Autom. Control, vol. 50, no. 5, pp. 655–661, May 2005.
[7] E. Nozari, P. Tallapragada, and J. Cortés, “Differentially private average consensus: Obstructions, trade-offs, and optimal algorithm design,” Automatica, vol. 81, pp. 221–231, 2017.
[8] Y. Mo and R. M. Murray, “Privacy preserving average consensus,” IEEE Trans. Autom. Control, vol. 62, no. 2, pp. 753–765, Feb. 2017.
[9] A. Alaeddini, K. Morgansen, and M. Mesbahi, “Adaptive communication networks with privacy guarantees,” in Proc. Amer. Control Conf., 2017, pp. 4460–4465.
[10] T. Charalambous, N. E. Manitara, and C. N. Hadjicostis, “Privacy-preserving average consensus over digraphs in the presence of time delays,” in Proc. 57th Ann. Allerton Conf. Commun., Control, Comput., 2019, pp. 238–245.
[11] H. Gao, C. Zhang, M. Ahmad, and Y. Wang, “Privacy-preserving average consensus on directed graphs using push-sum,” in Proc. IEEE Conf. Commun. Netw. Secur., 2018, pp. 1–9.
[12] C. N. Hadjicostis and A. D. Dominguez-Garcia, “Privacy-preserving distributed averaging via homomorphically encrypted ratio consensus,” IEEE Trans. Autom. Control, vol. 65, no. 9, pp. 3887–3894, Sep. 2020.
[13] Y. Wang, “Privacy-preserving average consensus via state decomposition,” IEEE Trans. Autom. Control, vol. 64, no. 11, pp. 4711–4716, Nov. 2019.
[14] R. Ghachbelou, A. Pascoal, C. Silvestre, and I. Kaminer, “Coordinated path following control of multiple wheeled robots with directed communication links,” in Proc. 44th IEEE Conf. Decis. Control, 2005, pp. 7084–7089.
[15] S. Yang, S. Tan, and J.-X. Xu, “Consensus based approach for economic dispatch problem in a smart grid,” IEEE Trans. Power Syst., vol. 28, no. 4, pp. 4416–4426, Nov. 2013.
[16] E. Seneta, Non-Negative Matrices and Markov Chains. Berlin, Germany: Springer, 2006.
[17] D. Kempe, A. Dobra, and J. Gehrke, “Gossip-based computation of aggregate information,” in Proc. 44th Ann. IEEE Symp. Found. Comput. Sci., 2003, pp. 482–491.
[18] P. Rezaeinia, B. Gharesifard, T. Linder, and B. Touri, “Push-sum on random graphs: Almost sure convergence and convergence rate,” IEEE Trans. Autom. Control, vol. 65, no. 3, pp. 1295–1302, Mar. 2020.
[19] M. Ruan, H. Gao, and Y. Wang, “Secure and privacy-preserving consensus,” IEEE Trans. Autom. Control, vol. 64, no. 10, pp. 4035–4049, Oct. 2019.
[20] J. Hajnal and M. Bartlett, “Weak ergodicity in non-homogeneous Markov chains,” in Mathematical Proceedings of the Cambridge Philosophical Society, Cambridge, U.K.: Cambridge Univ. Press, 1958, pp. 233–246.
[21] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, “k-L-diversity: Privacy beyond $k$-anonymity,” ACM Trans. Knowl. Discov. Data, vol. 1, no. 1, pp. 1–52, 2007.
[22] M. L. J. Hautus, “Strong detectability and observers,” Linear Algebra Appl., vol. 50, pp. 353–368, 1983.
[23] O. Goldreich, “Foundations of cryptography: Volume 1,” in Basic Tools. Cambridge, U.K.: Cambridge Univ. Press, 2007.