Statistical Privacy in Distributed Average Consensus on Bounded Real Inputs

Nirupam Gupta, Jonathan Katz and Nikhil Chopra

***Abstract*—This paper proposes a privacy protocol for distributed average consensus algorithms on bounded real-valued inputs that guarantees statistical privacy of honest agents' inputs against colluding (passive adversarial) agents, if the set of colluding agents is not a vertex cut in the underlying communication network. This implies that privacy of agents' inputs is preserved against t number of arbitrary colluding agents if the connectivity of the communication network is at least $(t+1)$. A similar privacy protocol has been proposed for the case of bounded integral inputs in our previous paper [1]. However, many applications of distributed consensus concerning distributed control or state estimation deal with real-valued inputs. Thus, in this paper we propose an extension of the privacy protocol in [1], for bounded real-valued agents' inputs, where bounds are known apriori to all the agents.**

I. Introduction

Distributed average consensus algorithms (for eg. [2], [3]) can be used in a peer-to-peer network by agents to reach a consensus value, equal to the average of all the agents' inputs. Some of the applications of distributed average consensus include sensor fusion [4], solving economic-dispatch problem in smart grids [5], and peer-to-peer online voting.

Typical distributed average consensus algorithms require the agents to share their inputs (and intermediate states) with their neighbors [2], [3]. This infringes the privacy of agents' inputs, which is undesirable as certain agents in the network may be passive adversarial[1] and non-trustworthy [6], [7], [8], [9], [10], [11].

If the agents' inputs are integers (bounded), privacy in distributed average consensus can be achieved by relying on (information-theoretic) distributed secure multi-party computation protocols [12] or homomorphic encryption-based average consensus [10], [13]. In this paper, we are interested in real-valued inputs with *known* bound, as several applications of distributed average consensus such as distributed Kalman filtering [4], formation control [14] and distributed learning [15]—deal with real-valued agents' inputs.

Several proposals [6], [8], [9] achieve *differential privacy* by having agents obscure their intermediate states (or values) by adding locally generated noise in a particular synchronous distributed average consensus protocol. Adding such local noises induces a loss in accuracy [9], [16] and there is an inherent trade-off between privacy and the achievable accuracy (agents are only able to compute an *approximation* to the exact average value). Schemes in [8], [7] iteratively cancel the noise added over time to preserve the accuracy of the average of all inputs. In the proposed privacy protocol, the random values added by agents to hide their inputs are correlated over space (in context of communication network) than over time, and collectively add up to zero, hence preserving the average value of the inputs. Note that differential privacy guarantees inevitably change if the agents' inputs are bounded by a value *known* to all the agents. In this paper, we are interested in statistical privacy guarantee specifically for the case when inputs have a *known* bound.

Scheme in [17] proposes re-designing of network link weights to limit the *observability* of agents' inputs but every agent's input gets known to its neighbors. The scheme of Gupta et al. [18] assumes a centralized (thus, not distributed), trusted authority that distributes information to all agents each time they wish to run the consensus algorithm.

We note that some of the above solutions [6], [7], [8], [9] require *synchronous* execution of the agents, whereas our privacy protocol is asynchronous (refer Section III). Moreover, this is the first paper, to the best of authors' knowledge, to propose a privacy protocol for distributed average consensus on bounded real-value inputs where bounds are apriori known. It is important to note that prior knowledge of inputs' bounds makes the privacy problem more challenging and renders the existing claims on differential privacy invalid.

A. Summary of Contribution

We develop on our previous works [1], [11] to propose a privacy protocol that guarantees statistical privacy of honest (non-adversarial) agents' inputs against colluding passive adversarial agents in any distributed average consensus over bounded (bounds *known* to all agents) real-valued inputs. In [11] we proposed a general approach for achieving privacy in distributed average consensus protocols for both real-valued and integral inputs. However, the privacy guarantee in [11] is weaker and uses relative entropy (KL-divergence) instead of the more standard statistical distance for privacy analysis. It is to be noted that the privacy approach in [1], [11] for integral inputs is quite similar to the one proposed by Emmanuel et al. [19]. However, [19] only considers a complete network topology which is relaxed in our work. Moreover, we focus on real-valued inputs and thus, the privacy scheme in [19] is not readily applicable. The privacy scheme in [19] has been extended for privacy in distributed optimization by [20] for real-valued agents' costs (equivalent to 'inputs' in our case). However, the privacy analysis in [20] does not provide

This work was partially supported by the National Science Foundation through award ECCS1711554.

Nirupam Gupta (nirupam@umd.edu) and Nikhil Chopra (nchopra@umd.edu) are with the Department of Mechanical Engineering, University of Maryland, College Park, 20742 MD, USA

Jonathan Katz (jkatz@cs.umd.edu) is with the Department of Computer Science, University of Maryland, College Park, 20742 MD, USA

[1] Passive adversarial agents follow the prescribed protocol unlike active adversarial agents, but can use their information to gather information about the inputs of other agents in the network.

any formal quantification on privacy guaranteed, and is not applicable to the case when the inputs are bounded with bounds being known apriori to all the agents.

Our proposed protocol constitutes of two phases:

1) In the first phase, each agent share correlated random values with its neighbors and computes a new, "effective input" based on its original input and the random values.
2) In the second phase, the agents run any (non-private) distributed average consensus protocol (for eg. [2]) to compute the sum of their effective inputs.

By design, the first phase ensures that the average of the agents' effective inputs is equal to the average of their original inputs (under a particular mathematical operator). Therefore, the two-phase approach does not affect the accuracy of the average value of the inputs. Furthermore, the privacy holds in our approach—in a formal statistical sense and under certain conditions, as discussed below—regardless of the average consensus protocol used in the second phase. To prove this we consider the worst-case scenario where all the effective inputs of the honest agents are revealed to the colluding semi-honest parties in the second phase.

The notion of privacy is the same as that used for the case of integral inputs in our earlier work [1], which had been adopted from the literature on secure multi-party computation [21]. Informally, the guarantee is that the entire *view* of the colluding agents throughout the execution of our protocol can be *simulated* by those agents given (1) their original inputs and (2) the average of the original inputs of the honest agents (or, equivalently, the average of the original inputs of all the agents in the network). This holds regardless of the true inputs of the honest agents. As a consequence, this means that the colluding adversarial agents learn nothing about the collective inputs of the honest agents from an execution of the protocol other than the average of the honest agents' inputs, and this holds regardless of any prior knowledge the adversarial agents may have about the inputs of (some of) the honest agents, or the distribution of those inputs. We prove that our protocol satisfies this notion of privacy as long as the set of colluding adversarial agents is not a vertex cut in underlying the communication network.

II. Notation and Preliminaries

We let $\mathbb{R}$ denote the set of non-negative real numbers and $frac(x) \in [0, 1)$ denote the fractional part of $x \in \mathbb{R}$. For any interval $[a, b] \in \mathbb{R}$, $[a, b]^n$ denotes the set of n-dimensional vectors with element taking values in $[a, b]$. We rely on the following basic properties

$$frac(x + y) = frac\left(frac(x) + frac(y)\right)$$
$$frac(-x) = 1 - frac(x),\ x \neq 0$$

If x is an n-dimensional vector, then x_i denotes its ith element and $\sum_i x_i$ simply denotes the sum of all its elements. We use 1_n to denote the n-dimensional vector all of whose elements is 1.

We consider communication networks represented by simple, undirected graphs. That is, the communication links in a network of n agents is modeled via a graph $\mathcal{G} = \{\mathcal{V},\ \mathcal{E}\}$ where the nodes $\mathcal{V} \triangleq \{1, \ldots, n\}$ denote the agents, and there is an edge $\{i, j\} \in \mathcal{E}$ iff there is a direct communication channel between agents i and j. We let N_i denote the set of neighbors of an agent $i \in \mathcal{V}$, i.e., $j \in N_i$ if and only if $\{i,\ j\} \in \mathcal{E}$. (Note that $i \notin N_i$ since $\mathcal{G}$ is a simple graph.)

We say two agents i, j are *connected* if there is a path from i to j; since we consider undirected graphs, this notion is symmetric. We let $p_{i,j}$ denote an arbitrary path between i and j, when one exists. A graph $\mathcal{G}$ is *connected* if every distinct pair of nodes is connected; note that a single-node graph is connected.

Definition 1: (Vertex cut) A set of nodes $\mathcal{V}_{cut} \subset \mathcal{V}$ is a *vertex cut* of a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ if removing the nodes in S (and the edges incident to those nodes) renders the resulting graph unconnected. Then, we say that $\mathcal{V}_{cut}$ *cuts* $\mathcal{V} \setminus \mathcal{V}_{cut}$.

A graph is *k-connected* if the smallest vertex cut of the graph contains k nodes.

Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ be a graph. The *subgraph induced by* $\mathcal{V}' \subset \mathcal{V}$ is the graph $\mathcal{G}' = \{\mathcal{V}', \mathcal{E}'\}$ where $\mathcal{E}' \subset \mathcal{E}$ is the set of edges entirely within $\mathcal{V}'$ (i.e., $\mathcal{E}' = \{\{i, j\} \in \mathcal{E} \mid i, j \in \mathcal{V}'\}$). We say a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ has c *connected components* if its vertex set $\mathcal{V}$ can be partitioned into disjoint sets $\mathcal{V}_1, \ldots, \mathcal{V}_c$ such that (1) $\mathcal{G}$ has no edges between $\mathcal{V}_i$ and $\mathcal{V}_j$ for $i \neq j$ and (2) for all i, the subgraph induced by $\mathcal{V}_i$ is connected. Clearly, if $\mathcal{G}$ is connected then it has one connected component.

For a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, we define its *incidence matrix* $\nabla \in \{-1, 0, 1\}^{|\mathcal{V}| \times |\mathcal{E}|}$ (see [22]) to be the matrix with $|\mathcal{V}|$ rows and $|\mathcal{E}|$ columns in which

$$\nabla_{i,\,e} = \begin{cases} 1 & \text{if } e = \{i,\ j\} \text{ and } i < j \\ -1 & \text{if } e = \{i,\ j\} \text{ and } i > j \\ 0 & \text{otherwise.} \end{cases}$$

Note that $1_n^T \cdot \nabla = 0$. We use $\nabla_{*,e}$ to denote the column of ∇ corresponding to the edge $e \in \mathcal{E}$.

We rely on the following result [22, Theorem 8.3.1]:

Lemma 1: Let $\mathcal{G}$ be an n-node graph with incidence matrix ∇. Then $\text{rank}(\nabla) = n - c$, where c is the number of connected components of $\mathcal{G}$.

A. Problem Formulation

We consider a network of n agents where the communication network between agents is represented by an undirected, simple, connected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$; that is, agents i and j have a direct communication link between them iff $\{i, j\} \in \mathcal{E}$. The communication channel between two nodes/agents is assumed to be both private and authentic; equivalently, in our adversarial model we do not consider an adversary who can eavesdrop on communications between honest agents, or tamper with their communication[2].

Each agent i holds a (private) input s_i. By scaling appropriately[3], we can assume without loss of generality that $s_i \in [0, 1/n)$, where n is the number of agents in the network. We let $s = [s_1, \ldots,\ s_n]^T \in [0, 1/n)^n$. A *distributed average*

[2] Alternately, private and authentic communication can be ensured using standard cryptographic techniques.

[3] Suppose each agent holds a finite real-valued input $x_i \in [0, q)$, $q \in \mathbb{R}^+$, then $s_i = x_i/nq \in [0, 1/n)$.

consensus algorithm is an interactive protocol allowing the agents in the network to each compute the average of the agents' inputs, i.e., after execution of the protocol each agent outputs the value $\bar{s} = \frac{1}{n} \cdot \sum_i s_i$. The value of n is assumed known to all the agents.

We are interested in distributed average consensus algorithms that ensure privacy against an attacker who controls some fraction of the agents in the network. We let $\mathcal{C} \subset \mathcal{V}$ denote the set of passive adversarial, and let $\mathcal{H} = \mathcal{V} \setminus \mathcal{C}$ denote the remaining honest agents. As stated earlier, we assume the adversarial agents are passive and thus run the prescribed protocol. Privacy requires that the entire *view* of the adversarial agents—i.e., the inputs of the adversarial agents as well as their internal states and all the protocol messages they received throughout execution of the protocol—does not leak (significant) information about the original inputs of the honest agents. Note that, by definition, the set of adversarial agents learns $\bar{s}$ (assuming at least one agent is adversarial) from the sum of the inputs of the honest agents can be computed, and so our privacy definition requires that the adversarial agents do not learn anything more than this.

Before giving our formal definition of privacy, we introduce some notation. Let $s_{\mathcal{C}}$ denote a set of inputs held by the agents in $\mathcal{C}$, and $s_{\mathcal{H}}$ a set of inputs held by the agents in $\mathcal{H}$. Fixing some protocol, we let $\mathsf{View}_{\mathcal{C}}(s)$ be a random variable denoting the view of the agents in $\mathcal{C}$ in an execution of the protocol when the agents all begin holding inputs s. Then:

Definition 2: A distributed average consensus protocol is *(perfectly) $\mathcal{C}$-private* if for all $s, s' \in [0, 1/n)^n$ such that $s_{\mathcal{C}} = s'_{\mathcal{C}}$ and $\sum_{i\in\mathcal{H}} s_i = \sum_{i\in\mathcal{H}} s'_i$, the distributions of $\mathsf{View}_{\mathcal{C}}(s)$ and $\mathsf{View}_{\mathcal{C}}(s')$ are identical.

We remark that this definition makes sense even if $|C| = n-1$, though in that case the definition is vacuous since $s_{\mathcal{H}} = \sum_{i\in\mathcal{H}} s_i$ and so revealing the sum of the honest agents' inputs reveals the (single) honest agent's input!

An alternate, perhaps more natural, way to define privacy is to require that for any distribution S (known to the attacker) over the honest agents' inputs, the distribution of the honest agents' inputs conditioned on the attacker's view is identical to the distribution of the honest agents' inputs conditioned on their sum. It is not hard to see that this is equivalent to the above definition.

III. Private Distributed Average Consensus

As described previously, our protocol has a two-phase structure. In the first phase, each agent i computes an "effective input" $\tilde{s}_i$ based on its original input s_i and random values it sends to its neighbors; this is done while ensuring that $frac(\sum_i \tilde{s}_i)$ is equal to $\sum_i s_i$ (see below). In the second phase, the agents use any (correct) distributed average consensus protocol Π to compute $\sum_i \tilde{s}_i$, take its fractional part, and then divide by n. This (as will be shown) gives the correct average $\frac{1}{n} \cdot \sum_i s_i$.

We prove privacy of our algorithm by making a "worst-case" assumption about Π, namely, that it simply reveals all the agents' inputs to all the agents. Such an algorithm is, of course, not at all private; for our purposes, however, this does not violate privacy because Π is run on the agents' *effective* inputs $\{\tilde{s}_i\}$ rather than their true inputs $\{s_i\}$. Therefore, the privacy result holds regardless of the distributed average consensus protocol Π. From now on, then, we let the *view* of the adversarial agents consist of the original inputs of the adversarial agents, their internal states and all the protocol messages they receive throughout execution of the first phase of our protocol, and the vector $\tilde{s} = [\tilde{s}_1, \ldots, \tilde{s}_n]^T$ of all agents' effective inputs at the end of the first phase. Our definition of privacy (cf. Definition 2) remains unchanged.

The first phase of our protocol proceeds as follows:

1) Each agent $i \in \mathcal{V}$ chooses independent, uniform values $r_{ij} \in [0, 1)$ for all $j \in \mathcal{N}_i$, and sends r_{ij} to agent j.
2) Each agent $i \in \mathcal{V}$ computes a mask
$$a_i = frac\left(\sum_{j\in N_i} (r_{ji} - r_{ij})\right), \quad (1)$$
where $a_i \in [0, 1)$.
3) Each agent $i \in \mathcal{V}$ computes effective input
$$\tilde{s}_i = frac(s_i + a_i). \quad (2)$$

Note that
$$frac\left(\sum_i \tilde{s}_i\right) = frac\left(\sum_i s_i + frac\left(\sum_i a_i\right)\right)$$
As $\mathcal{G}$ is undirected, therefore
$$frac\left(\sum_i a_i\right) = frac\left(\sum_i \sum_{j\in N_i} (r_{ji} - r_{ij})\right) = 0$$
Thus, $frac\left(\sum_i \tilde{s}_i\right) = \sum_i s_i$, since $\sum_i s_i < 1$ as $s_i \in [0, 1/n)$, $\forall i$. Hence, correctness of our overall algorithm (i.e., including the second phase) follows.

Note that any two neighboring agents i and j choose values r_{ij} and r_{ji}, respectively, independently. Agents i and j then transmit these values r_{ij} and r_{ji}, respectively to each other in an independent manner as well[4]. Therefore, Step 1 does not require synchronicity between any two agents. Steps 2 and 3 are performed locally, and therefore synchronicity between agents is out of question. Once an agent completes the first-phase, it floods the network with this information regardless of whether any other agent has completed the first-phase or not. As every agent has prior knowledge of the total number of agents, the agents reach an agreement on the completion of the first-phase when $\mathcal{G}$ is connected. *Hence, the first-phase is asynchronous and this implies that the proposed protocol is asynchronous if the distributed average consensus protocol in the second-phase is asynchronous.*

In the second-phase, the agents can use an asynchronous distributed average consensus protocol, such as the randomized gossip algorithm [3], to compute the average value of $\{n\tilde{s}_i\}$, which equal to $\sum_i \tilde{s}_i$.

[4] Agent i transmits r_{ij} regardless of whether it has received r_{ji} or not. Same applies for agent j.

A. Privacy Analysis

We show here that $\mathcal{C}$-privacy holds if $\mathcal{C}$ is not a vertex cut of $\mathcal{G}$ under the assumptions on agents' inputs, network topology and communication links mentioned in Section II-A.

For an edge $e = \{i, j\}$ in the graph with $i < j$, define

$$b_e = frac(r_{ji} - r_{ij}).$$

Let $b = [b_{e_1}, \dots]$ be the collection of such values for all the edges in $\mathcal{G}$. If we let $a = [a_1, \dots, a_n]^T$ denote the masks used by the agents, then we have

$$a = frac(\nabla \cdot b).$$

Since the r_{ij} are uniform and independent in $[0, 1)$, it is easy to see that the values $\{b_e\}_{e \in \mathcal{E}}$ are uniform and independent in $[0, 1)$ as well[5]. Thus, a is uniformly distributed over the vectors in the span of the columns of ∇, which we denote by $L(\nabla)$, with coefficients in $[0, 1)$. The following is easy to prove using the fact that rank$(\nabla) = n-1$ when $\mathcal{G}$ is connected (cf. Lemma 1):

***Lemma** 2:* If $\mathcal{G}$ is connected then a is uniformly distributed over all points in $[0, 1)^n$ subject to the constraint that $frac(\sum_i a_i) = 0$.

(A full proof of Lemma 2 is given in Appendix A.)

Since $\tilde{s}_i = frac(s_i + a_i)$, we have

***Lemma** 3:* If $\mathcal{G}$ is connected, then the effective inputs $\tilde{s}$ are uniformly distributed in $[0, 1)^n$ subject to the constraint that $frac\left(\sum_i \tilde{s}_i\right) = \sum_i s_i$.

The proof of Lemma 3 is given in Appendix B.

The above implies privacy for the case when $\mathcal{C} = \emptyset$, i.e., when there are no adversarial agents. In that case, the view of any agent consists only of the effective inputs $\tilde{s}$, and Lemma 3 shows that the distribution of those values depends only on the sum of the agents' true inputs. Below, we extend this line of argument to the case of nonempty $\mathcal{C}$.

Fix some set $\mathcal{C}$ of passive adversarial agents, and recall that $\mathcal{H} = \mathcal{V} \setminus \mathcal{C}$. Let $\mathcal{E}_\mathcal{C}$ denote the set of edges incident to $\mathcal{C}$, and let $\mathcal{E}_\mathcal{H} = \mathcal{E} \setminus \mathcal{E}_\mathcal{C}$ be the edges incident only to honest agents. Note that now the view of adversarial agents' view contains (information that allows it to compute) $\{b_e\}_{e \in \mathcal{E}_\mathcal{C}}$ in addition to the honest agents' effective inputs $\{\tilde{s}_i\}_{i \in \mathcal{H}}$.

The key observation enabling a proof of privacy is that the values $\{b_e\}_{e \in \mathcal{E}_\mathcal{H}}$ are uniform and independent in $[0, 1)^{|\mathcal{H}|}$ *even conditioned on the values of* $\{b_e\}_{e \in \mathcal{E}_\mathcal{C}}$. Thus, owing to Lemma 2, as long as $\mathcal{C}$ is not a vertex cut of $\mathcal{G}$, an argument as earlier implies that the masks $\{a_i\}_{i \in \mathcal{H}}$ are uniformly distributed in $[0, 1)^{|\mathcal{H}|}$ subject to $frac(\sum_{i \in \mathcal{H}} a_i) = frac(-\sum_{i \in \mathcal{C}} a_i)$ (even conditioned on knowledge of the values $\{b_e\}_{e \in \mathcal{E}_\mathcal{C}}$), and hence the effective inputs $\{\tilde{s}_i\}_{i \in \mathcal{H}}$ are uniformly distributed in $[0, 1)^{|\mathcal{H}|}$ subject to

$$frac\left(\sum_{i \in \mathcal{H}} \tilde{s}_i\right) = frac\left(\sum_{i \in \mathcal{V}} s_i - \sum_{i \in \mathcal{C}} \tilde{s}_i\right)$$

(again, even conditioned on knowledge of the $\{b_e\}_{e \in \mathcal{E}_\mathcal{C}}$). Since the right-hand side of the above equation can be computed from the effective inputs of the adversarial agents, the $\{b_e\}_{e \in \mathcal{E}_\mathcal{C}}$, and the sum of the honest agents' inputs, this implies:

***Theorem** 1:* If $\mathcal{C}$ is not a vertex cut of $\mathcal{G}$, then our proposed distributed average consensus protocol is perfectly $\mathcal{C}$-private.

A formal proof of this theorem is given in Appendix C.

As a corollary, we have

***Corollary** 1:* If $\mathcal{G}$ is $(t+1)$-connected, then for any $\mathcal{C}$ with $|\mathcal{C}| \leq t$ our proposed distributed average consensus protocol is perfectly $\mathcal{C}$-private.

In case the passive adversarial agents do form a vertex cut, in that case the proposed privacy protocol guarantees privacy of each set of honest agents that is not cut by $\mathcal{C}$, in the sense as formally defined[6]. Alternately, for a set of honest agents $\mathcal{H}' \subset \mathcal{H}$ that is not cut by $\mathcal{C}$ the adversarial agents can deduce anything about their collective inputs $\{s_i\}_{i \in \mathcal{H}'}$ other than their sum $\sum_{i \in \mathcal{H}'} s_i$. (refer [1])

IV. Illustration

To demonstrate our proposed distributed average consensus protocol we consider a simple network of 3 agents with $\mathcal{V} = \{1, 2, 3\}$ and $\mathcal{E} = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$, as shown in Fig. 1. Let $s_1 = 0.1$, $s_2 = 0.2$ and $s_3 = 0.15$.

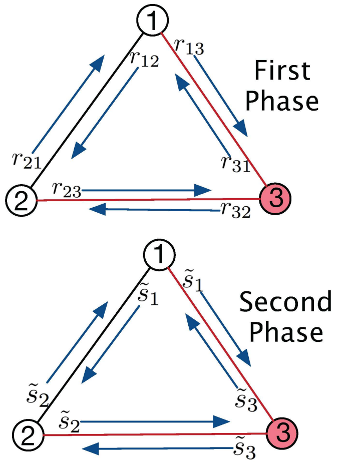

Fig. 1. Arrows (in blue) show the flow of information over an edge.

In first phase, the agents execute the following steps

1) As shown in Fig. 1, all pair of adjacent agents i and j exchange the respective values of r_{ij} and r_{ji} (chosen independently and uniformly in $[0, 1)$) with each other. Consider a particular instance where: $r_{12} = 0.1$, $r_{21} = 0.5$, $r_{23} = 0.7$, $r_{32} = 0.4$, $r_{31} = 0.3$ and $r_{13} = 0.8$.
2) The agents compute their respective masks,

   $$a_1 = frac\left((r_{21} - r_{12}) + (r_{31} - r_{13})\right) = 0.9$$

   Similarly, $a_2 = 0.3$ and $a_3 = 0.8$. (One can verify that $frac(a_1 + a_2 + a_3) = 0$.)
3) The agents compute their respective effective inputs,

   $$\tilde{s}_1 = frac(s_1 + a_1) = frac(0.1 + 0.9) = 0.0$$

[5] If x and y are two independent random variables in $[0, 1)$ with at least one of them being uniformly distributed, then $z = frac(x + y)$ is uniformly distributed in $[0, 1)$.

[6] $\mathcal{C}$ does not cut a set of agents if that set of agents that is connected in the residual graph after removing $\mathcal{C}$ and $\mathcal{E}_\mathcal{C}$.

Similarly, $\tilde{s}_2 = 0.5$ and $\tilde{s}_3 = 0.95$.

After the first phase, each agent uses a (non-private) distributed average consensus protocol Π (an instance shown in Fig. 1) in the second phase to compute $(1/3)\sum_i \tilde{s}_i$ (it can be easily to verified that $frac(\sum_i \tilde{s}_i) = \sum_i s_i = 0.45$).

Let $\mathcal{C} = \{3\}$ and so, $\mathcal{E}_{\mathcal{C}} = \{\{1,\, 3\},\, \{2,\, 3\}\}$. It is easy to see that $\mathcal{C}$ does not cut the graph $\mathcal{G}$ and therefore, for any pair of inputs $s_1 \in [0, 1/3)$ and $s_2 \in [0, 1/3)$ that satisfy $s_1 + s_2 = .3$ the joint distribution of $\tilde{s}_1$ and $\tilde{s}_2$ is uniform over $[0, 1)^2$ such that $frac(\tilde{s}_1 + \tilde{s}_2) = 0.5$ (cf. Lemma 3).

V. Conclusion

In this paper, we propose a general approach (distributed and asynchronous) to ensure privacy of honest agents in any distributed average consensus protocol. The inputs of the agents are assumed to be finite real-values. The proposed approach guarantees (perfect) privacy of honest agents against passive adversarial agents if the set of adversarial agents is not a vertex cut of the underlying communication network. The only information that adversarial agents can get on the inputs of honest agents is their sum (or average).

It is not difficult to see that the privacy protocol proposed in this paper be used for privacy in distributed computation of any function $h : \mathbb{R}^n \to \mathbb{R}$, over agents inputs $\{s_i\}$, of the following form

$$h(s_1, \ldots, s_n) = g\left(\sum_i h_i(s_i)\right).$$

Here, $h_i : \mathbb{R} \to \mathbb{R}, \forall i$ and $g : \mathbb{R} \to \mathbb{R}$. We assume that the functions h_i, $\forall i$ are injective (one-to-one), thus privacy of $h_i(s_i)$ is equivalent to the privacy of s_i. Also, it is reasonable to assume that $h_i(s_i)$ is finite if s_i is finite. For now, let $h_i(s_i) \in [0, 1/n), \forall i$.

Each agent first computes the effective function values $\tilde{h}_i(s_i) = frac(h_i(s_i) + a_i)$ and then uses any (non-private) distributed average consensus on these effective function values to compute $\sum_i \tilde{h}_i(s_i)$. Then, $frac\left(\sum_i \tilde{h}_i(s_i)\right) = \sum_i h_i(s_i)$ as $\sum_i a_i = 0$. Thus, each agent correctly computes the desired function value as

$$g\left(frac\left(\sum_i \tilde{h}_i(s_i)\right)\right) = g\left(\sum_i h_i(s_i)\right) = h(s_1, \ldots, s_n).$$

References

[1] N. Gupta, J. Katz, and N. Chopra, "Information-theoretic privacy in distributed average consensus," *arXiv preprint arXiv:1809.01794*, 2018 (Under review for Automatica).

[2] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.

[3] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE/ACM Transactions on Networking (TON)*, vol. 14, no. SI, pp. 2508–2530, 2006.

[4] R. Olfati-Saber, "Distributed kalman filter with embedded consensus filters," in *44th IEEE Conference on Decision and Control*. IEEE, 2005, pp. 8179–8184.

[5] S. Yang, S. Tan, and J.-X. Xu, "Consensus based approach for economic dispatch problem in a smart grid," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4416–4426, 2013.

[6] Z. Huang, S. Mitra, and G. Dullerud, "Differentially private iterative synchronous consensus," in *Proc. ACM Workshop on Privacy in the Electronic Society*. ACM, 2012, pp. 81–90.

[7] N. E. Manitara and C. N. Hadjicostis, "Privacy-preserving asymptotic average consensus," in *European Control Conference*. IEEE, 2013, pp. 760–765.

[8] Y. Mo and R. M. Murray, "Privacy preserving average consensus," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 753–765, 2017.

[9] 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.

[10] M. Ruan, M. Ahmad, and Y. Wang, "Secure and privacy-preserving average consensus," in *Proceedings of the 2017 Workshop on Cyber-Physical Systems Security and PrivaCy*. ACM, 2017, pp. 123–129.

[11] N. Gupta, J. Katz, and N. Chopra, "Privacy in distributed average consensus," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 9515–9520, 2017.

[12] J. Garay and R. Ostrovsky, "Almost-everywhere secure computation," in *Advances in Cryptology—Eurocrypt 2008*, ser. Lecture Notes in Computer Science. Springer, 2008, pp. 307–323.

[13] R. Lazzeretti, S. Horn, P. Braca, and P. Willett, "Secure multi-party consensus gossip algorithms," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 2014, pp. 7406–7410.

[14] W. Ren and R. W. Beard, *Distributed consensus in multi-vehicle cooperative control*. Springer, 2008.

[15] P. A. Forero, A. Cano, and G. B. Giannakis, "Consensus-based distributed support vector machines," *Journal of Machine Learning Research*, vol. 11, no. 5, pp. 1663–1707, 2010.

[16] P. Braca, R. Lazzeretti, S. Marano, and V. Matta, "Learning with privacy in consensus + obfuscation," *IEEE Signal Processing Letters*, vol. 23, no. 9, pp. 1174–1178, 2016.

[17] S. Pequito, S. Kar, S. Sundaram, and A. P. Aguiar, "Design of communication networks for distributed computation with privacy guarantees," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 1370–1376.

[18] N. Gupta and N. Chopra, "Confidentiality in distributed average information consensus," in *55th IEEE Conf. on Decision and Control*. IEEE, 2016, pp. 6709–6714.

[19] E. A. Abbe, A. E. Khandani, and A. W. Lo, "Privacy-preserving methods for sharing financial risk exposures," *American Economic Review*, vol. 102, no. 3, pp. 65–70, 2012.

[20] S. Gade and N. H. Vaidya, "Private learning on networks," *arXiv preprint arXiv:1612.05236*, 2016.

[21] O. Goldreich, *Foundations of Cryptography: Basic Applications*. Cambridge University Press, 2004, vol. 2.

[22] C. Godsil and G. Royle, *Algebraic Graph Theory*. Springer, 2001.

Appendix

A. Proof of Lemma 2

The proof is obvious for $n = 1$. From now on, $n > 1$ and $\mathcal{G}$ is assumed connected.

Keep in mind that each $\{b_e\}_{e \in \mathcal{E}}$ is *independent and uniformly distributed* in $[0, 1)$. (As for any $e = \{i, j\} \in \mathcal{E}$, the values r_{ij} and r_{ji} are independent and uniform in $[0, 1)$.)

Consider a subset $\mathcal{E}'$ of $\mathcal{E}$ with $n-1$ edges such that $\mathcal{G}' = \{\mathcal{V},\, \mathcal{E}'\}$ is connected (such $\mathcal{E}'$ is guaranteed to exist as $\mathcal{G}$ is connected). Therefore, all the $n-1$ columns of ∇' (incidence matrix of $\mathcal{G}'$) are linearly independent. This implies that all the points in the span (coefficients belonging to $[0, 1)$) of the columns of ∇', given by

$$L(\nabla') = \left(frac(\nabla' \cdot b) \,|\, b \in [0, 1)^{n-1}\right),$$

are equally probable as b is uniformly distributed in $[0, 1)^{n-1}$(note that this claim holds because all the elements of ∇' belong to $\{-1, 0, 1\}$). Alternately,

$$a' = frac(\nabla' \cdot b) = frac\left(\sum_{e \in \mathcal{E}'} \nabla'_{*,e} \cdot b_e\right)$$

is uniformly distributed over $[0,1)^{n-1}$. Furthermore, combining the above with the fact that $1_n^T \cdot \nabla' = 0$ implies that $a' \in L(\nabla') \iff frac(1_n^T a') = 0$ (for all $a' \in [0,1)^n$).

In case $\mathcal{E}' = \mathcal{E}$ (or $\mathcal{E}$ has only $n-1$ edges), the proof concludes here. Otherwise, choose an edge e' from the set of remaining edges $\mathcal{E}' \backslash \mathcal{E}$. Now, $\nabla'_{*,e'}$ can be obtained by linearly combining the columns of ∇' as following

$$\nabla'_{*,e'} = \sum_{e \in \mathcal{E}'} \mu_e \nabla'_{*,e} \tag{3}$$

as $\mathcal{G}'$ is connected[7], where $\mu_e \in \{-1, 0, 1\}$ for all $e \in \mathcal{E}'$.

Define a new set of edges $\mathcal{E}'' = \mathcal{E}' \cup \{e'\}$. From (3), each point a'' of $L(\nabla'')$ (span of the columns of the incidence matrix ∇'' of $\mathcal{G}'' = \{\mathcal{V}, \mathcal{E}''\}$) is given as

$$\begin{aligned} a'' &= frac\left(\sum_{e \in \mathcal{E}'} \nabla'_{*,e} \cdot b_e + \nabla'_{*,e'} b_{e'}\right) \\ &= frac\left(\sum_{e \in \mathcal{E}'} \nabla'_{*,e} \cdot (b_e + \mu_e b_{e'})\right) \end{aligned}$$

As the values $\{b_e\}_{e \in \mathcal{E}'}$ are independent and uniform in $[0,1)$, this implies $\{frac(b_e + \mu_e b_{e'})\}_{e \in \mathcal{E}'}$ is uniformly distributed over all the values in $[0,1)^{n-1}$. Hence, a'' is uniformly distributed over $L(\nabla')$ (and $L(\nabla'')$ is same as $L(\nabla)$).

Same as before, if $\mathcal{E}'' = \mathcal{E}$, the proof concludes. Otherwise, repeat the above procedure by considering another edge from $\mathcal{E} \setminus \mathcal{E}''$, which leads to the same conclusion. This iterative process of including edges stops when all the edges of E have been considered. Ultimately, we reach the conclusion that to express a is uniformly distributed in $L(\nabla')$. (Axiomatically, this also implies that $L(\nabla)$ is same as $L(\nabla')$.)

Hence, a is uniform in $[0,1)^n$ subject to the constraint that $frac(1_n^T a) = 0$ (or $frac(\sum_i a_i) = 0$), as any point $a \in L(\nabla') \iff frac(1_n^T a) = 0$.

B. *Proof of Lemma 3*

Let $\tilde{S}$, S and A represent the random vectors of the agents' effective inputs, true inputs and masks, respectively. $\mathcal{G}$ is assumed connected. For a random variable (or vector) S, $f_S(s)$ denotes its probability density at s.

As $\tilde{s}_i = frac(s_i + a_i)$ and s_i, a_i are independent, we have

$$f_{\tilde{S}}\left(\tilde{s} | S = s\right) = f_A\left(A = frac(\tilde{s} - s)\right)$$

If $frac(\sum_i \tilde{s}_i) = \sum_i s_i$ then $frac(\tilde{s} - s)$ belongs to $L(\nabla)$. Thus, from Lemma 2,

$$f_{\tilde{S}}\left(\tilde{s} | S = s\right) = 1$$

for all the values $\tilde{s}$ in $[0,1)^n$ that satisfy $frac(1_n^T \tilde{s}) = 0$, when $\mathcal{G}$ is connected.

C. *Proof of Theorem 1*

Let $\mathcal{G}_{\mathcal{H}} = \{\mathcal{H}, \mathcal{E}_{\mathcal{H}}\}$ be the graph of honest agents (and edges incident to only honest agents) and $\nabla_{\mathcal{H}}$ be its *incidence matrix*. For a random variable (or vector) S, $f_S(s)$ denotes its probability density at s.

The *view* of adversarial agents in $\mathcal{C}$ consists of honest agents' effective inputs $\tilde{s}_{\mathcal{H}}$ after the first phase of our protocol (considering the "worst-case" scenario where agents' can acquire all the inputs through their internal states in Π) and the values $\{b_e\}_{e \in \mathcal{E}_{\mathcal{C}}}$. Therefore,

$$\mathsf{View}_{\mathcal{C}}(s) = \{\tilde{s}_{\mathcal{H}}, \{b_e\}_{e \in \mathcal{E}_{\mathcal{C}}}\}$$

given the inputs $s \in [0, 1/n)^n$.

Thus, we prove that the joint probability distribution of $\tilde{s}_{\mathcal{H}}$ and $\{b_e\}_{e \in \mathcal{E}_{\mathcal{C}}}$ is the same for any two sets of true inputs s, s', that satisfy $s_{\mathcal{C}} = s'_{\mathcal{C}}$ and $\sum_{i \in \mathcal{V}} s_i = \sum_{i \in \mathcal{V}} s'_i$, when $\mathcal{G}_{\mathcal{H}}$ is connected.

We have,

$$a_i = frac\left(frac\left(\sum_{e \in \mathcal{E}_{\mathcal{H}}} \nabla_{i,e} b_e\right) + frac\left(\sum_{e \in \mathcal{E}_{\mathcal{C}}} \nabla_{i,e} b_e\right)\right)$$

The values $\{frac(\sum_{e \in \mathcal{E}_{\mathcal{H}}} \nabla_{i,e} b_e)\}_{i \in \mathcal{H}}$ lie in the span of $\nabla_{\mathcal{H}}$, denoted by $L(\nabla_{\mathcal{H}})$. Therefore, $\{frac(\sum_{e \in \mathcal{E}_{\mathcal{H}}} \nabla_{i,e} b_e)\}_{i \in \mathcal{H}}$ is uniformly distributed over $[0,1)^{|\mathcal{H}|}$ subject to $frac(\sum_{i \in \mathcal{H}}(\sum_{e \in \mathcal{E}_{\mathcal{H}}} \nabla_{i,e} b_e)) = 0$ when $\mathcal{G}_{\mathcal{H}}$ is connected (cf. Lemma 2).
Thus, it is clear that the masks $\{a_i\}_{i \in \mathcal{H}}$ are uniformly distributed in $[0,1)^{|\mathcal{H}|}$ subject to
$frac(\sum_{i \in \mathcal{H}} a_i) = frac(-\sum_{i \in \mathcal{C}} a_i)$ when $\mathcal{G}_{\mathcal{H}}$ is connected (given the values of $\{b_e\}_{e \in \mathcal{E}_{\mathcal{C}}}$).
Note that random variables $\{b_e\}_{e \in \mathcal{E}_{\mathcal{H}}}$ are uniformly and independently distributed in $[0,1)$, given the values of $\{b_e\}_{e \in \mathcal{E}_{\mathcal{C}}}$, and $a_i = frac(\sum_{e \in \mathcal{E}_{\mathcal{C}}} \nabla_{i,e} b_e)$ for every $i \in \mathcal{C}$. Thus using Lemma 3 above implies, ($\tilde{S}_{\mathcal{H}}$ denotes the random vector of honest agents' effective inputs $\tilde{s}_{\mathcal{H}}$)

$$f_{\tilde{S}_{\mathcal{H}}}\left(\tilde{s}_{\mathcal{H}} | s_{\mathcal{H}}, \{b_e\}_{e \in \mathcal{E}_{\mathcal{C}}}\right) = 1 \tag{4}$$

for all the values $\tilde{s}_{\mathcal{H}}$ in $[0,1)^{|\mathcal{H}|}$ that satisfy

$$frac \sum_{i \in \mathcal{H}} \tilde{s}_i = frac\left(\sum_{i \in \mathcal{V}} s_i - \sum_{i \in \mathcal{C}} \tilde{s}_i\right)$$

when $\mathcal{G}_{\mathcal{H}}$ is connected.

Combining (4) with the fact that the random variables $\{b_e\}_{e \in \mathcal{E}}$ are independent to the true inputs s implies

$$f_{\mathsf{View}_{\mathcal{C}}(s)}\left(\{\tilde{s}_{\mathcal{H}}, \{b_e\}_{e \in \mathcal{E}_{\mathcal{C}}}\}\right) \equiv f_{\mathsf{View}_{\mathcal{C}}(s')}\left(\{\tilde{s}_{\mathcal{H}}, \{b_e\}_{e \in \mathcal{E}_{\mathcal{C}}}\}\right)$$

for all $s, s' \in [0, 1/n)^n$ such that $s_{\mathcal{C}} = s'_{\mathcal{C}}$ and $\sum_{i \in \mathcal{V}} s_i = \sum_{i \in \mathcal{V}} s'_i$ when $\mathcal{G}_{\mathcal{H}}$ is connected.

[7] It follows easily from the fact that there exists a path in $\mathcal{G}'$ between the terminal nodes of the edge e' as $\mathcal{G}'$ is connected.