# Privacy in Distributed Average Consensus [⋆]

**Nirupam Gupta [*] Jonathan Katz [**] Nikhil Chopra [*]**

[*] *Department of Mechanical Engineering, University of Maryland, College Park, 20742 MD, USA (e-mails: nirupam@umd.edu, nchopra@umd.edu).*
[**] *Department of Computer Science, University of Maryland, College Park, 20742 MD, USA (e-mail: jkatz@cs.umd.edu)*

**Abstract:** Distributed average consensus refers to computing average of inputs held by multiple agents communicating with each other over peer-to-peer network. Cooperation amongst agents is imperative for any distributed average consensus protocol as each agent has to share its input with other agents, which are usually the adjacent(neighboring) agents. That being said, privacy issues could discourage some agents from participating in such protocols. This paper proposes a novel distributed privacy mechanism that preserves privacy of the collection of honest agents' inputs as long as the colluding semi-honest agents do not form a vertex cut. The proposed privacy mechanism does not alter the average of agents' inputs, hence it does not provide privacy against what is already lost by knowing the average of the inputs. It poses minimal additional computation and communication costs, requires no alteration of the distributed consensus protocol and promises a highly scalable practical solution for privacy in distributed average consensus. The privacy achieved is quantified using Kullback-Leibler divergence (KL-divergence) and limitations are discussed analytically for two cases; case i) inputs are continuous random variables, and case ii) inputs are discrete random variables.

*Keywords:* Privacy, distributed mechanism, distributed average consensus, semi-honest agents.

## 1. INTRODUCTION

Distributed average consensus is an important protocol for decision making in a peer-to-peer network architecture. However, any distributed average consensus protocol requires agents in the network to share their input values with other agents in order for each agent to compute the average of the agents' inputs. Sharing of actual input values infringes the confidentiality of the agents and this has motivated researchers to develop privacy mechanisms that prevent an agent's input from getting revealed to any other agent in the network or an eavesdropper recording the exchanges between the agents during the consensus protocol. Some of the average information consensus applications with critical confidentiality issues are: online voting though social networks (peer-to-peer communication), total energy-consumption in a power-grid, medical surveys, tracking online product sell, multi-agent rendezvous systems and online census.

This problem has been addressed by Pequito et al. (2014) using a control theoretic perspective, in which the authors re-design the communication link weights (graph-Laplacian based protocol) of the network to minimize the dimension of the observable subspace, but an agent is still able to procure the inputs of its neighboring agents. Manitara and Hadjicostis (2013) preserve privacy of an agent's input against some participating agents by adding noise to the outgoing agent's messages for particular number of iteration in a graph-Laplacian based distributed average consensus protocol. In comparison to these works, our proposed distributed privacy mechanism can preserve privacy of agents' inputs against an eavesdropper listening to the messages shared between agents during the course of any distributed average consensus protocol. Further, privacy of honest agents' inputs can also be preserved against colluding semi-honest agents [1] if the set of semi-honest agents does not form a vertex cut.

In recent years, researchers have also explored $\epsilon-$ differentially private (DP) mechanisms for distributed average consensus like Huang et al. (2012); Mo and Murray (2014); Nozari et al. (2015). Mo and Murray (2014) propose an $(\epsilon, \delta)-$ DP mechanism [2] . Differential privacy protects privacy of data against the risk of concomitant information leakage due to the knowledge of the output of any query made on the data. In this specific case of distributed average consensus, the data is distributed amongst agents (their inputs) and so, every agent makes queries to their neighboring agents about their inputs and states (values generated during an iterative average consensus protocol) and then every agent distributively obtains the average of agents' inputs. In this paper, we address the issue of additional privacy loss of agents' inputs only due to the sharing of agents' states and inputs during the distributed average consensus protocol and not due to the public

[⋆] This work was supported by the Naval Air Warfare Center Aircraft Division - Pax River, MD under contract N00421132M022.

[1] Semi-honest agents are passive adversarial agents that strictly follow the prescribed protocol, but also try to determine other agents secret information or inputs.

[2] $\epsilon-$ differential privacy is stronger than $(\epsilon, \delta)-$ differential privacy because if a randomized mechanism is $\epsilon-$ DP implies it is $(\epsilon, \delta)-$ DP for all non-negative $\delta$.

knowledge of the average of agents' inputs. The reason being that we do not intend to perturb the average inputs that is imperative for providing $\epsilon$-differential privacy for the inputs Dwork et al. (2014).

From Proposition 6.1 of Nozari et al. (2015), for $\epsilon-$ DP distributed consensus algorithm like the ones discussed in Huang et al. (2012); Nozari et al. (2015) (state values corrupted with Laplacian noise), the variance of the converged agents' states around the true average is inversely proportional to $\epsilon^2$. This means, higher the $\epsilon$ ($\Rightarrow$ choppier Laplacian distribution around $0$) better is the consensus accuracy, but as mentioned in Section 3.3 of Dwork et al. (2014), smaller the $\epsilon$ ($\Rightarrow$ flatter Laplacian distribution around $0$) better is the privacy. Hence, the tradeoff. Let $x$ and $\eta(0)$ be the input value and added initial noise of a cooperating agent, respectively. If $x$ could take only quantized real values with step value $\Delta$, then smaller variance of $\eta(0)$ about zero would mean higher $Pr(\, x-\Delta < x(0)+\eta(0) < x+\Delta\,)$ and better chances of guessing $x(0)$, given $x(0)+\eta(0)$. This implies, the tradeoff between privacy and accuracy worsens when the agents' inputs are quantized values (e.g. integers). Hence, if we are not worried about the loss of privacy of inputs due to the public knowledge of their average, then a differentially private mechanism is not an ideal solution. Having said that, one can always use any DP distributed average consensus algorithm on top of the proposed privacy mechanism without affecting the differential privacy results.

The existing secure multi-party computation (MPC) protocols in Ben-Or et al. (1988); Goldreich et al. (1987); Chaum et al. (1988), are capable of computing any general function $f$ over multiple agents' inputs $x_i, i = 1, \ldots, n$ (index $i$ represents an agent) securely against $t < n/2$ semi-honest agents. However, for all the MPC protocols the communication cost for every agent increases as $n$ increases as these protocols require every pair of $n$ agents to communicate with each other (refer to Section 3.2 in Lindell and Pinkas (2009)). This requirement limits the scalability of MPC protocols and makes it very difficult to implement them for distributed computation in large scale peer-to-peer networks. Although our proposed privacy mechanism is only applicable to computing average of the agents' inputs, but unlike MPC protocols we do not require all the agents to communicate with each other and the communication cost for any single agent is just linearly dependent on the number of its neighboring agents. Thus, for certain network topologies (like simple cycle graphs) the communication cost of implementing the proposed privacy mechanism practically remains constant for an agent even when the total number of participating agents in the network is increasing.

*Paper Contribution:* We consider two cases; case i) the agents' inputs are real and case ii) the agents' inputs are bounded integers with known bounds. In both cases, inputs of any two agents are assumed independent from each other and their prior probability distribution is assumed public. Inspired from the MPC protocol by Ben-Or et al. (1988), we propose a novel distributed privacy mechanism for both the cases to preserve privacy (as defined in Section 4) of a collection of honest agents' inputs as long as the semi-honest agents do not form a vertex cut. We first present the analysis of privacy against an eavesdropper that is listening to messages being exchanged *during* any average consensus protocol, and then utilize this analysis to analyze privacy against $t$ colluding semi-honest agents. The proposed distributed privacy mechanism does not alter the average of the inputs in either case (refer to Claim 1 and 2) and communication cost imposed on each agent depends linearly on its number of neighbors. However, unlike differential privacy the privacy provided by the proposed mechanism is conditioned by the graph topology as subsequently discussed in detail in Section 4.

Throughout this paper, the following **assumptions** hold:

**A1**: Every agent in the network is cooperative and follows the protocols honestly without injecting false information.

**A2**: Every agent has knowledge of the privacy mechanism.

**A3**: The network topology is undirected [3] (defined in Subsection 2.2) at the time when privacy mechanism is executed and is known to all the agents.

**A4**: The communication channel is *secure* between any two agents. [4]

## 2. NOTATIONS AND REQUISITES

$\mathbb{N}$, $\mathbb{R}$, $\mathbb{R}^n$, $\mathbb{R}^{n\times m}$, $\mathbb{S}^n$ and $\mathbb{S}^n_+$ represent the set of natural numbers, real scalars, real $n \in \mathbb{N}$-dimensional vectors, $n$ by $m$ real-valued matrices, $n$-dimensional symmetric matrices and $n$-dimensional symmetric positive semi-definite matrices, respectively. $0_n$ and $1_n$ represent $n$-dimensional vectors with all the elements equal to 0 and 1, respectively. For matrix $M$, $\mathscr{N}(M)$ and $\mathscr{R}(M)$ represent its null-space and range, respectively. $(\cdot)^T$, $\det(\cdot)$, $\det^*(\cdot)$ and $(\cdot)^\dagger$ represents transpose, determinant, pseudo-determinant and generalized inverse, respectively [5]. $M^{1/2}$ represents square-root of a square matrix $M$. For $x \in \mathbb{R}^n$, $\|x\|$ represents its 2-norm and $Diag(x) \in \mathbb{S}^n$ represents a diagonal matrix with $i$-th diagonal element being the $i$-th element of the vector $x$.

For any $x \in \{0, \ldots, q-1\}^n$, $x$ mod $p$ represents the modulo operation on each element of $x$ by $p$. For any finite set $S$, $|S|$ denotes its cardinality.

### *2.1 Probability & Information Theory*

For a **continuous random variable or vector** $X$, the probability density is represented as $p_X(x)$. supp$(X)$ represents support of $X$. $P_{X,Y}(x, y)$ and $p_X(x|Y = y)$ represent the joint probability density $X$ and $Y$ and conditional probability density for $X$ given $Y = y$, respectively. If $X \sim \mathcal{N}(\mu, \sigma^2)$ (Gaussian distribution), then

$$p_X(x) = (1/\sqrt{2\pi\sigma^2})e^{\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)}$$

$E(X)$ and $\Sigma_X = E(XX^T)$ represents the mean and covariance matrix of a random vector $X$. $Pr(X \in S|Y = y)$ represents the conditional probability of random vector

[3] Network topology need not be static and can change over time. Average consensus can be reached even if graph is not connected at each point in time

[4] In other words, if the communication channels between any two agents is *secure*, then it cannnot be read or tempered by any other agent (refer to Ben-Or et al. (1988)).

[5] For a singular $M \in \mathbb{S}^n$, pseudo-determinant $\det^*(M) = \lim_{\alpha\to 0} \frac{\det(M+\alpha I)}{\alpha^{n-rank(M)}}$ and $MM^\dagger M = M$.

$X$ taking value in set $S$ for a given $Y = y$. The KL-divergence between two probability distributions with respective probability densities $p_X$ and $p_Y$ is given as

$$0 \leq D_{KL}(p_X || p_Y) = \int_{x \in \mathbb{R}^n} p_X(x) \log \frac{p_X(x)}{p_Y(x)} dx$$

For a **discrete random variable or vector** $X$ taking values in finite set $\{0, \dots, p-1\}$, $p \in \mathbb{N}$, the probability mass is represented as $P_X(x)$. Other notations are similar to that of continuous random variable or vector and are self explanatory. If $X \sim \mathcal{U}\{0, \dots, p-1\}$, then $P_X(x) = 1/p$, $\forall x \in \{0, \dots, p-1\}$. The KL-divergence between two probability distributions with respective probability masses $P_X$ and $P_Y$ is given as

$$0 \leq D_{KL}(P_X || P_Y) = \sum_{x \in \{0, \dots, p-1\}^n} P_X(x) \log \frac{P_X(x)}{P_Y(x)}$$

### *2.2 Graph Theory*

Consider a simple undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $n$ nodes. $\mathcal{V} \triangleq \{1, \dots, n\}$ is the set of all nodes in the graph. Let $A = [e_{ij}]$ be the adjacency matrix of the graph with $e_{ij} = e_{ji} \in \{0, 1\}$ (also referred to as edge weights), where $(i, j) \in \mathcal{V} \times \mathcal{V}$. $\mathcal{E}$ is the set of unordered pairs,

$$\mathcal{E} = \{\{i, j\} \in \mathcal{V} \times \mathcal{V} | e_{ij} = e_{ji} = 1\}.$$

$N_i$ represents the set of neighbors of a node $i \in \mathcal{V}$. A node $j \in N_i$ if and only if $\{i, j\} \in \mathcal{E}$ and $j \neq i$.

*Definition 1.* (Connected graph) Undirected graph $\mathcal{G}$ is connected if there exists a path between every pair of nodes.

The graph-Laplacian $L$ is defined as $L = D - A$, where $D = Diag([d_1, \dots, d_n]^T)$ is the out-degree matrix, with out-degree of each node $d_i = \sum_{j \in \mathcal{V}} e_{ij}$. Graph-Laplacian $L \in \mathbb{S}^n_+$ for an undirected graph $\mathcal{G}$ with $rank(L) \leq n-1$ and $1^1_n L = 0$. Being symmetric, $L$ can be written as (spectral theorem)

$$L = U \, Diag\left([\mu_1, \dots, \mu_n]^T\right) U^T$$

where $\mu_1 \geq \dots \geq \mu_n = 0$ are eigenvalues of $L$, and $U = (v_1, \dots, v_n)$ is the orthonormal matrix with $v_i$ being the unit eigenvector for the corresponding eigenvalue $\mu_i$, $i \in \{1, \dots, n\}$. $\nabla \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{E}|}$ represents the oriented incident matrix of any $\mathcal{G}$ with

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

where, $e$ represents an element in $\mathcal{E}$. Let $\nabla_{\{i,j\}}$ denote the column of $\nabla$ associated with edge $\{i, j\} \in \mathcal{E}$. Clearly, $1_n^T \nabla = 0$. rank$(\nabla) = n - c$, where $c$ is the number of connected components of graph $\mathcal{G}$. The graph-Laplacian $L = \nabla \nabla^T$. If $\mathcal{G}$ is connected, then $\mathcal{N}(L) = span\{1_n\}$.

**Note:** Throughout this paper, the network topology at the time of execution of privacy mechanism will be represented by a simple undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $0-1$ edge weights and $c < n$ connected components. $\mathcal{V} = \{1, \dots, n\}$ denotes set of agents and $\mathcal{E}$ denotes set of links between agents.

## 3. PRIVACY MECHANISM

In this section we propose the privacy mechanism. First, in Subsection 3.1 we present it for case i) when the agents' inputs are real valued, and then in Subsection 3.2 for case ii) when the agents' inputs are bounded integers.

### *3.1 Real Valued Inputs*

For the case when $x_i \in \mathbb{R}$, $\forall i \in \mathcal{V}$, the privacy mechanism consists of the following 4 steps (in order of execution).

(1) Each agent $i$ chooses a secret random value of random variable $r_{ij} \sim \mathcal{N}(0, \sigma^2)$ independently[6] for every $j \in N_i$.
(2) Every pair of neighboring agents $i$ and $j$ share the value of $r_{ij}$ and $r_{ji}$, respectively with each other securely, this is protected from any eavesdropper.
(3) Each agent $i$ computes its respective *mask*
$$a_i = \sum_{j \in N_i} (r_{ji} - r_{ij}) \tag{1}$$
(4) Each agent $i$ generates the *masked input* $\tilde{x}_i = x_i + a_i$.

Once these steps have been executed, each agent uses its *masked input* rather than the actual input to compute the global average of the agents' inputs using any distributed average consensus protocol.

*Claim 1.* As graph $\mathcal{G}$ is undirected, thus $\sum_{i \in \mathcal{V}} a_i = 0$. This implies, $\sum_{i \in \mathcal{V}} \tilde{x}_i = \sum_{i \in \mathcal{V}} x_i$.

### *3.2 Integer Valued Inputs*

Let $x_i \in \{0, \dots, q-1\}$, $\forall i \in \mathcal{V}$ and $p \in \mathbb{N}$, such that $n(q-1) \leq p-1$. The privacy mechanism is similar to the one outlined in Subsection 3.1, but with minor modifications.

(1) Each agent $i$ chooses a secret random value of $r_{ij} \sim \mathcal{U}\{0, \dots, p-1\}$ independently for every $j \in N_i$.
(2) Every pair of neighboring agents $i$ and $j$ exchange $r_{ij}$ and $r_{ji}$, respectively with each other securely.
(3) Each agent $i$ computes its respective *mask*
$$a_i = \left( \sum_{j \in N_i} (r_{ji} - r_{ij}) \right) \bmod p \tag{2}$$
(4) Each agent $i$ generates the *masked input* $\tilde{x}_i = (x_i + a_i) \bmod p$.

*Claim 2.* As graph $\mathcal{G}$ is undirected,

$\left(\sum_{i \in \mathcal{V}} a_i\right) \bmod p = 0$. This implies,

$$\left( \sum_{i \in \mathcal{V}} \tilde{x}_i \right) \bmod p = \left\{ \sum_{i \in \mathcal{V}} x_i + \sum_{i \in \mathcal{V}} a_i \right\} \bmod p = \sum_{i \in \mathcal{V}} x_i$$

This is true because we assume $x_i \leq q-1$,which guarantees that $\sum_{i \in \mathcal{V}} x_i \leq p-1$.

*Remark 1.* Each agent $i$ uses its *masked input* $\tilde{x}_i$ rather than the actual input $x_i$ to compute the average using any distributed average consensus protocol. Once, every agent has the value of the average of all *masked inputs* $\sum_{i \in \mathcal{V}} \tilde{x}_i / n$, it can locally compute $\sum_{i \in \mathcal{V}} x_i / n$ (follows from Claim 2).

[6] Mean need not be zero, subsequent results hold even if $r_{ij} \sim \mathcal{N}(\nu, \sigma^2)$, $\nu > 0$.

**Henceforth**,

(1) ‘input vector’ will always refer to the $n$-dimensional vector of agents’ inputs $[x_1, \dots, x_n]^T$.
(2) agents follow the proposed privacy mechanism in Section 3 for both cases.

## 4. PRIVACY ANALYSIS (AGAINST EAVESDROPPER)

As mentioned in Section 1, in this paper we are only concerned with privacy loss of the agents’ inputs given the *masked inputs*, and not what is already incurred from the knowledge of their global average. In this section we quantify the privacy of agents’ inputs for both the cases against an eavesdropper listening to the message exchanges between agents during the consensus protocol.

### *4.1 Real Valued Inputs*

In this subsection we deal with case (i), where the inputs can take values in $\mathbb{R}$. The input vector can be modeled as continuous random vector $X = [X_1, \dots, X_n]^T$ consisting of $n$ independent random variables $X_i$, $i = 1, \dots, n$ with density $p_X(x)$ such that $\text{supp}(X) = \mathbb{R}^n$.

*Remark 2.*

$$Pr(X \in H_z | Z = z) = 1 \tag{3}$$

where, $H_z = (x \in \mathbb{R}^n | 1_n^T x = z)$. This concomitant information leakage is unavoidable as the proposed privacy mechanism does not alter the output of any distributed average consensus protocol (refer to the definition of privacy in Lindell and Pinkas (2009)).

The *masked input* vector can also be modeled as a continuous random vector $\tilde{X} = X + A$, where $A$ is a continuous random vector with each element $A_i$ being the continuous random variable associated with *mask* $a_i$ of agent $i \in \mathcal{V}$. $A$ can be written as the following linear transformation (refer to Step 3 in Subsection 3.1).

$$A = \nabla B \tag{4}$$

where $B = [B_{\{i,j\}}]$ is $|\mathcal{E}|$-dimensional continuous random vector with

$$B_{\{i,j\}} = R_{ji} - R_{ij},\ \{i, j\} \in \mathcal{E} \,\&\, i < j$$

$R_{ij}$ and $R_{ji}$ are identical and independent random variables associated with the undirected edge $\{i, j\}$ and have probability distribution $\mathcal{N}(0, \sigma^2)$ (refer to Step 1 in Subsection 3.1). Thus, $\Sigma_B = E(BB^T) = 2\sigma^2 Diag(1_{|\mathcal{E}|})$ and this makes $A$ a multivariate normally distributed random vector with

$$E(A) = 0,\ \Sigma_A = 2\sigma^2 L \tag{5}$$

From (4), it is clear that $A \in \mathscr{R}(\nabla) = \mathscr{R}(L)$ [7]. Thus from (5), random vector $A$ can assume take values in $\mathscr{R}(L) = \mathscr{R}(L^{1/2})$, with the following probability density [8].

$$P_A(a) = \frac{1}{\sqrt{\det^*(2\pi\Sigma_A)}} \exp\left(-\frac{1}{2} a^T \Sigma_A^\dagger a\right) \tag{6}$$

The generalized inverse $\Sigma_A^\dagger = \frac{1}{2\sigma^2} L^\dagger$, where $L^\dagger$ is given as (refer to Gutman and Xiao (2004); Ben-Israel and Greville (2003)),

$$L^\dagger = U\, Diag\left([1/\mu_1, \dots, 1/\mu_{n-c}, 0_c^T]^T\right) U^T \tag{7}$$

The value of the pseudo-determinant is

$\det^*(2\pi\Sigma_A) = (4\pi\sigma^2)^{n-c} \prod_{k=1}^{n-c} \mu_k$.

*Definition 2.* ($\delta$-*companions*) Two input vectors $x^1 \in \mathbb{R}^n$ and $x^2 \in \mathbb{R}^n$ are $\delta$-*companions* ($\delta > 0$) if $\|x^1 - x^2\| \le \delta$ and $p_{\tilde{X}}(\tilde{x}|X = x^1) > 0 \Leftrightarrow p_{\tilde{X}}(\tilde{x}|X = x^2) > 0$.

As $X$ and $A$ are independent, we get

$$p_{\tilde{X}}(\tilde{x}|X = x) = p_A(\tilde{x} - x) \tag{8}$$

Thus, if $x^1$ and $x^2$ are $\delta$-*companions*, then $1_n^T x^1 = 1_n^T x^2$.

*Claim 3.* For a given $\tilde{x}$, any two input vectors in $C_{\tilde{x}} = (\tilde{x} - a \in \mathbb{R}^n | a \in \mathscr{R}(L))$ separated by Euclidean distance $\le \delta$ are $\delta$-*companions.*

*Definition 3.* ($(\delta, \epsilon)$-*distinguishable* [9]) Two $\delta$-*companion* input vectors $x^1$ and $x^2$ are $(\delta, \epsilon)$-*distinguishable* for $\epsilon > 0$ if

$$D_{KL}\left(p_{\tilde{X}}(\,\cdot\,|X = x^1) || p_{\tilde{X}}(\,\cdot\,|X = x^2)\right) \le \epsilon\delta^2$$

*Remark 3.* From Definition 3, we get an upper bound on the expected weight of evidence for $\{X = x^1\}$ over $\{X = x^2\}$ for each sample of $\tilde{X}$ generated from $x^1$ in step (4) of the privacy mechanism in Subsection 3.1. Further, smaller $\epsilon$ means smaller chances of an eavesdropper correctly guessing the agents’ inputs for a given $\tilde{x}$ and hence, better privacy.

*Theorem 4.* (case (i)) *Consider the privacy mechanism described in Subsection 3.1 with assumptions (A1)-(A4). There exists $\sigma_\epsilon > 0$ for every $\epsilon > 0$ such that for every $\sigma \ge \sigma_\epsilon$ any two $\delta$-companion input vectors are $(\delta, \epsilon)$-distinguishable.*

**Proof.** (By construction)
Consider two arbitrary $\delta$-*neighboring* input vectors $x^1$ and $x^2$. From (6) and (8),

$$\frac{p_{\tilde{X}}(\tilde{x}|X = x^1)}{p_{\tilde{X}}(\tilde{x}|X = x^2)} = e^{\frac{1}{2}\left\{(\tilde{x}-x^2)^T \Sigma_A^\dagger (\tilde{x}-x^2) - (\tilde{x}-x^1)^T \Sigma_A^\dagger (\tilde{x}-x^1)\right\}}$$

Let $a^1 = \tilde{x} - x^1$ and $a^2 = \tilde{x} - x^2$ be the respective *masks.* Now,

$$(a^2)^T \Sigma_A^\dagger (a^2) - (a^1)^T \Sigma_A^\dagger (a^1) = (a^2 - a^1)^T \Sigma_A^\dagger (a^2 + a^1)$$

This implies,

$$\log \frac{p_{\tilde{X}}(\tilde{x}|X = x^1)}{p_{\tilde{X}}(\tilde{x}|X = x^2)} = \frac{1}{2}(x^1 - x^2)^T \Sigma_A^\dagger (2\tilde{x} - x^1 - x^2)$$

for all values of $\tilde{x}$ generated from $x^1$ in step (4) of the privacy mechanism in Subsection 3.1. Now using (5) and (7), we get

$$\begin{aligned} &D_{KL}\left(p_{\tilde{X}}(\,\cdot\,|X = x^1) || p_{\tilde{X}}(\,\cdot\,|X = x^2)\right) \\ &= \frac{1}{2} \int_{a^1 \in \mathscr{R}(L)} (x^1 - x^2)^T \Sigma_A^\dagger (2a^1 + x^1 - x^2) P_A(a^1) da^1 \\ &= (x^1 - x^2)^T \Sigma_A^\dagger E(A) + \frac{1}{2}(x^1 - x^2)^T \Sigma_A^\dagger (x^1 - x^2) \\ &= \frac{1}{2}(x^1 - x^2)^T \Sigma_A^\dagger (x^1 - x^2) \le \frac{\delta^2}{2\mu_{n-c}\sigma^2} \end{aligned}$$

Hence, $\sigma_\epsilon = 1/\sqrt{2\mu_{n-c}\epsilon}$ and rest follows directly. ■

From Theorem 4, we deduce that larger $\sigma$ gives better privacy in the sense of Definition 3.

[7] $\mathscr{N}(\nabla^T) = \mathscr{N}(L)$

[8] restricting Lebesgue measure to rank($L$)-affine subspace of $\mathbb{R}^n$.

[9] Definition 3 is adopted from the expected *discrimination information* interpretation of the KL-divergence in Kullback (1987); Kullback and Leibler (1951); Press et al. (2007)).

*Claim 5.* An eavesdropper can determine $x_i$ if and only if $|N_i| = 0$. [10]

### *4.2 Integer Valued Inputs*

In this subsection we deal with case (ii), where the agents' inputs take values in $\{0, \ldots, q-1\}$. The input vector can be modeled as discrete random vector $X = [X_1, \ldots, X_n]^T$ consisting of $n$ independent discrete random variables $X_i$ with probability mass function

$$P_X(x) = \begin{cases} > 0 & , \ x \in \{0, \ldots, q-1\} \\ 0 & , \ x \in \{q, \ldots, p-1\} \end{cases}$$

*Remark 4.* The number of input vectors resulting to an average $z$ is not uniform [11]. For instance if $0 \le z < q$, then number of *z-permissible* vectors are $N[z] = {}^{n+z-1}C_{n-1}$. Obviously if $z = 0$ or $z = n(q-1)$ then the input vector can be uniquely identified as $0_n$ or $(q-1)1_n$, respectively.

The *masked input* vector can also be modeled as a discrete random vector $\tilde{X} = X + A$, where $A$ is a discrete random vector with each element $A_i$ being the discrete random variable associated with *mask* $a_i$ of agent $i \in \mathcal{V}$. $A$ can be written in form of the following linear combination (refer to Step (3) of the privacy mechanism in Subsection 3.2).

$$A = \left( \sum_{\{i,j\} \in \mathcal{E}} \nabla_{\{i,j\}} B_{\{i,j\}} \right) \bmod p = (\nabla B) \bmod p \qquad (9)$$

where $B_{\{i,j\}}$ is just the discrete counterpart of $B_{\{i,j\}}$ in Subsection 4.1. The discrete random variables $R_{ij}$ and $R_{ji}$ are identical and independent with probability mass function $P_R(r) = \mathcal{U}\{0, \ldots, p-1\}$ (refer to Step (1) of the privacy mechanism in Subsection 3.2).

*Lemma 6.* Let $\Theta$ and $\Gamma$ be two independent discrete random variables taking values in $\{0, \ldots, p-1\}$. Let $\Theta \sim \mathcal{U}\{0, \ldots, p-1\}$ and $\Gamma \sim P_\Gamma$. Then the discrete random variable $\Psi = (\Theta + \Gamma) \bmod p$ is uniformly distributed in $\{0, \ldots, p-1\}$.

From Lemma 6, we get

$$B_{\{i,j\}} \sim \mathcal{U}\{0, \ldots, p-1\}, \forall \{i,j\} \in \mathcal{E} \qquad (10)$$

*Lemma 7.*

$$P_A(a) = 1/|L(\nabla)| = 1/p^{n-c}, \ \forall a \in L(\nabla)$$

where $L(\nabla) = \{(\nabla b) \bmod p \,|\, b \in \{0, \ldots, p-1\}^{|\mathcal{E}|}\}$.

**Proof.** (Using mathematical induction) Let $\mathcal{E}_k \subseteq \mathcal{E}$ denote an arbitrary set of $k \in [1, |\mathcal{E}|]$ undirected edges of graph $\mathcal{G}$, and $\mathcal{G}_k = (\mathcal{V}, \mathcal{E}_k)$. $\nabla_k$, $B_k$ and $A_k$ be analogous to $\nabla$, $B$ and $A$ for the subgraph $\mathcal{G}_k$, respectively. For an arbitrary $\mathcal{E}_1$ from (10), we get

$$P_{A_1}(a) = 1/|L(\nabla_1)| = 1/p, \ \forall a \in L(\nabla_1) \qquad (11)$$

Now, let us assume that

$$P_{A_1}(a) = 1/|L(\nabla_k)| = 1/p^{\mathrm{rank}(\nabla_k)}, \ \forall a \in L(\nabla_k)$$

is true for any arbitrary $\mathcal{E}_k \subset \mathcal{E}$ $(1 < k < |\mathcal{E}|)$. Considering any arbitrary edge $\{i,j\} \in \mathcal{E} \setminus \mathcal{E}_k$, we get

$$A_{k+1} = (\nabla_{\{i,j\}} B_{\{i,j\}} + A_k) \bmod p$$

From Lemma 6 and (11) $A_{k+1}$ assumes all the values in

$$\begin{aligned} L(\nabla_{k+1}) &= \{(\nabla_{\{i,j\}} b + L(\nabla_k)) \bmod p | b \in \{0, \ldots, p-1\}\} \\ &= \{(\nabla_{k+1} b) \bmod p | \, b \in \{0, \ldots, p-1\}^{|\mathcal{E}_{k+1}|}\} \end{aligned}$$

with equal probability [12]. Hence,

$$P_{A_{k+1}}(a) = 1/|L(\nabla_{k+1})| = 1/p^{\mathrm{rank}(\nabla_{k+1})}, \ \forall a \in L(\nabla_{k+1})$$

Rest follows from the principle of mathematical induction. ■

*Definition 4.* (*companions*) Two input vectors $x^1$ and $x^2$ in $\{0, \ldots, q-1\}^n$ are *companions* if $P_{\tilde{X}}(\tilde{x}|X = x^1) > 0 \Leftrightarrow P_{\tilde{X}}(\tilde{x}|X = x^2) > 0$.

As $X$ and $A$ are independent, implies

$$P_{\tilde{X}}(\tilde{x}|X = x) = P_A(\tilde{x} - x) \qquad (12)$$

Thus, if $x^1$ and $x^2$ are *companions*, then $1_n^T x^1 = 1_n^T x^2$.

*Definition 5.* (*$\epsilon$-distinguishable*) Two *companion* input vectors $x^1$ and $x^2$ are *$(z, \epsilon)$-distinguishable* $(\epsilon > 0)$ if

$$D_{KL}\left(P_{\tilde{X}}(\,\cdot\,|X = x^1) || P_{\tilde{X}}(\,\cdot\,|X = x^2)\right) \le \epsilon \qquad (13)$$

*Remark 5.* From Definition 5, we get an upper bound on the expected weight of evidence for $\{X = x^1\}$ over $\{X = x^2\}$ for each sample of $\tilde{X}$ generated from $x^1$ in step (4) of the privacy mechanism in Subsection 3.1. Further, smaller $\epsilon$ means smaller chances of an eavesdropper correctly guessing the agents' inputs for a given $\tilde{x}$ and hence, better privacy.

*Theorem 8.* (case (ii)) Consider the privacy mechanism described in Subsection 3.2 with assumptions (A1)-(A4). Any two *companion* input vectors are 0-*distinguishable.*

**Proof.** Consider two *companion* input vectors $x^1$ and $x^2$ in $\{0, \ldots, q-1\}^n$. Because $X$ and $A$ are independent to each other, implies

$$P_{\tilde{X}}(\tilde{x}|X = x^l) = \frac{P_{X,\tilde{X}}(x^l, \tilde{x})}{P_X(x^l)} = P_A(\tilde{x} - x^l), \ l = 1, 2$$

Thus, from Lemma 7 we get

$$P_{\tilde{X}}(\tilde{x}|X = x^1) = P_{\tilde{X}}(\tilde{x}|X = x^2) > 0$$

for all the values of $\tilde{X}$ generated from $x^1$ in Step (4) of the privacy mechanism in Subsection 3.2. Rest of the proof follows directly from the definition of KL-divergence. ■

*Remark 6.* (Follows from Theorem 8) For a given $\tilde{X} = \tilde{x}$, regardless of its computational capability an eavesdropper can not discriminate between any two input vectors that could have generated $\tilde{x}$ in Step (4) of the privacy mechanism in Subsection 3.2.

*Claim 9.* For $z \notin \{0, |\mathcal{V}|(q-1)\}$, an eavesdropper can determine $x_i$ if and only if $|N_i| = 0$ [13].

## 5. PRIVACY ANALYSIS (AGAINST SEMI-HONEST AGENTS)

In this section we utilize the privacy analysis against an eavesdropper to analyze the privacy of inputs against $0 \le t < n$ number of semi-honest agents that are colluding to determine the rest of the agents' inputs without

[10] (*If*) Is obvious. (*Only if*) Let $j \in N_i$. Consider an input vector $x^1$, then a change of $\delta/2$ or a lesser value in $b_{\{i,j\}}$ gives a *$\delta$-companion* vector; $x^2$ of $x^1$. Contradiction follows directly from Theorem 4.

[11] It is equivalent to arranging $z$ number of 1s and $n-1$ number of 0s in a row such that at most $q-1$ number of 1s are together.

[12] If $\nabla_{\{i,j\}}$ and columns in $\nabla_k$ are linearly dependent (when $i$ and $j$ are already connected by edges in $\mathcal{E}_k$), then invoke Lemma 6, otherwise it is obvious.

[13] Similar to Claim 5

disrupting the privacy mechanism. For convenience, let $\mathcal{V}_t = \{i_1, \dots, i_t\} \subset \mathcal{V}$ be the set of $t$ semi-honest collaboration agents. Throughout the rest of this paper, we address $\mathcal{V}_t$ as a single entity.

Let $\mathcal{G}(\mathcal{V}_t) \subset \mathcal{G}$ be the subgraph that consists of all the agents in $\mathcal{V}_t$ and undirected edges incident to the agents in $\mathcal{V}_t$. Let $\mathcal{G}^c(\mathcal{V}_t) = \mathcal{G} \setminus \mathcal{G}(\mathcal{V}_t)$ and $\mathcal{V}_t^c = \mathcal{V} \setminus \mathcal{V}_t$. Additionally, $x(\mathcal{V}_t)$, $\tilde{x}(\mathcal{V}_t)$ and $z(\mathcal{V}_t)$ denote the input vector, *masked inputs* vector and sum of the inputs of agents of $\mathcal{V}_t^c$, respectively. Assume the following,
**A5**: $\mathcal{V}_t$ knows *masked input* vector $\tilde{x}$[14].

From (4) and (9), we know that the *mask* $a_i$ of any agent $i$ is just summation of secret random numbers $r_{ji} - r_{ji}$ associated with undirected edges $\{i, j\}$, $\forall j \in N_i$. Thus, every agent in $\mathcal{V}_t$ basically knows all the secret random numbers associated with the undirected edges incident to the agents in $\mathcal{V}_t$. Hence, if $\mathcal{V}_t$ is the vertex cover of $\mathcal{G}$, then $\mathcal{V}_t$ can uniquely determine all the agents' inputs.

*Claim 10.* Assume **(A5)** and $t <$ size of the minimum vertex cover of $\mathcal{G}$. Then results of Section 4 hold if $\mathcal{G}$ is replaced by $\mathcal{G}^c(\mathcal{V}_t)$.

*Remark 7.* It follows from *claims* 10, 5 and 9 that in case $\mathcal{V}_t$ is not a vertex cut of $\mathcal{G}$[15], privacy of inputs in $\mathcal{V}_t^c$ can be preserved by the proposed distributed privacy mechanism.

Unlike MPC protocols, the privacy against $\mathcal{V}_t$ does not just depend on $|\mathcal{V}_t|$, but on the topology of $\mathcal{G}(\mathcal{V}_t)$.

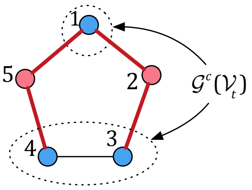

Fig. 1. (An illustration) Consider a simple cycle graph of 5 agents with $\mathcal{V}_t = \{2, 5\}$ (shown in red). In this case $\mathcal{G}^c(\mathcal{V}_t) = (\{1, 3, 4\}, \{\{3, 4\}\})$. $x(\mathcal{V}_t) = [x_1, x_3, x_4]^T$. Consider $\mathcal{G}^c(\mathcal{V}_t)$; for case (i) from Theorem 4, any two $\delta$-*companions* $x^1$ and $x^2$ are $(\delta, \epsilon)$-*distinguishable*. For case (ii) from Theorem 8 for $z(\mathcal{V}_t) \in \{1, \dots, 3q-2\}$ any two *companion* input vectors $x^1$ and $x^2$ are 0-*distinguishable*. But, $\mathcal{V}_t$ can perfectly determine $x_1$ in both the cases (Claim 5 and 9).

## 6. DISCUSSION

The paper proposes a novel distributed privacy mechanism to preserve privacy of agents' inputs against a group of $t$ number of semi-honest agents colluding to determine inputs of rest of the agents. We consider two cases; case i) inputs are real-valued, and case ii) inputs are bounded integers. For both the cases the privacy is quantified using estimated discrimination information given by KL-divergence. The proposed distributed privacy mechanism is designed in such a way that it does not alter the output of any distributed average consensus protocol, hence it does not protect privacy of agents' inputs against the concomitant leakage due to the knowledge of the average of inputs as discussed in Remarks 2 and 4. Further, the proposed privacy mechanism suffer from limitations mentioned in Claim 5 and 9.

## REFERENCES

Ben-Israel, A. and Greville, T.N. (2003). *Generalized inverses: theory and applications*, volume 15. Springer Science & Business Media.

Ben-Or, M., Goldwasser, S., and Wigderson, A. (1988). Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, 1–10. ACM.

Chaum, D., Crépeau, C., and Damgard, I. (1988). Multiparty unconditionally secure protocols. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, 11–19. ACM.

Dwork, C., Roth, A., et al. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4), 211–407.

Goldreich, O., Micali, S., and Wigderson, A. (1987). How to play any mental game. In *Proceedings of the nineteenth annual ACM symposium on Theory of computing*, 218–229. ACM.

Gutman, I. and Xiao, W. (2004). Generalized inverse of the laplacian matrix and some applications. *Bulletin de l'Academie Serbe des Sciences at des Arts (Cl. Math. Natur.)*, 129, 15–23.

Huang, Z., Mitra, S., and Dullerud, G. (2012). Differentially private iterative synchronous consensus. In *Proceedings of the 2012 ACM workshop on Privacy in the electronic society*, 81–90. ACM.

Kullback, S. (1987). Letter to the editor: The kullback-leibler distance.

Kullback, S. and Leibler, R.A. (1951). On information and sufficiency. *The annals of mathematical statistics*, 22(1), 79–86.

Lindell, Y. and Pinkas, B. (2009). Secure multiparty computation for privacy-preserving data mining. *Journal of Privacy and Confidentiality*, 1(1), 5.

Manitara, N.E. and Hadjicostis, C.N. (2013). Privacy-preserving asymptotic average consensus. In *Control Conference (ECC), 2013 European*, 760–765. IEEE.

Mo, Y. and Murray, R.M. (2014). Privacy preserving average consensus. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, 2154–2159. IEEE.

Nozari, E., Tallapragada, P., and Cortés, J. (2015). Differentially private average consensus with optimal noise selection. *IFAC-PapersOnLine*, 48(22), 203–208.

Pequito, S., Kar, S., Sundaram, S., and Aguiar, A.P. (2014). Design of communication networks for distributed computation with privacy guarantees. In *53rd IEEE Conference on Decision and Control*, 1370–1376. IEEE.

Press, W., Teukolsky, S., Vetterling, W., and Flannery, B. (2007). Section 14.7. 2. kullback–leibler distance. *Numerical Recipes: The Art of Scientific Computing*.

[14] Also, for certain iterative distributed average consensus algorithms it is feasible for a subset of agents to determine the inputs of all other remaining agents (Pequito et al. (2014)).

[15] Implies, $\mathcal{G}^c(\mathcal{V}_t)$ is connected