2015 American Control Conference
Palmer House Hilton
July 1-3, 2015. Chicago, IL, USA

Protecting Privacy of Topology in Consensus Networks

Vaibhav Katewa*, Aranya Chakrabortty† and Vijay Gupta*

Abstract— **Consider a set of agents implementing the discrete time consensus algorithm. At each time step, all agents also transmit their states to a central estimator that wishes to identify the underlying topology and eigenvalues of the network. It does so by using a nonlinear least squares (NLS) algorithm to identify the state evolution matrix used in the consensus algorithm. We present a mechanism to protect the differential privacy of this topology from an eavesdropper who may have unauthorized access to the estimator. In this mechanism, every agent purposely adds noise to its measurements before transmission to the estimator. The noise is designed to ensure that the eavesdropper cannot uniquely identify the topology with a specified confidence level. Numerical results are presented to describe the corresponding trade-off in estimation accuracy as a function of the level of differential privacy achieved.**

I. Introduction

A distributed linear dynamical system consists of multiple agents that are coupled with each other. The coupling may be present due to communication among the agents or due to the dynamics being interdependent. In either case, the coupling induces a weighted graph/network with the agents as nodes and links representing the coupling between agents [3]. Examples of such networks include consensus networks [1], power grid networks [2], formation control networks and so on. In all these cases, the network topology and weights represent how one agent's state affects the states of other agents and thus, it is equivalent to the state evolution matrix of the overall distributed linear system. The network topology may change over time due to variations in the system operating conditions such as the number of agents, number of links, proximity of the nodes, network traffic, loads, etc. Thus, a system administrator monitoring the network needs to keep a continuous track of the topology. One possible way to perform this system monitoring is to collect the measured outputs of the agents at a central control center and then estimate the system topology using the outputs. For example, this type of architecture is used in a power grid network, where multiple geographically distributed sensors such as Phasor Measurement Units (PMUs) transmit their measurements to a central estimator [17]. The central estimator then estimates the topology of the network model using suitable system identification methods [12].

A malicious eavesdropper who hacks into the central estimator and gets access to measurements of all agents can determine the network topology, and thereby gain access to critical system-level information. This information can be used to plan an attack on the network. Protecting the privacy of the topology, therefore, is a crucial task for network operators. In addition to estimating the topology, another important objective of the system administrator is to track the eigenvalues of the state matrix of the distributed system using the measurements. This information is important since eigenvalues dictate the stability and convergence rates of the system modes. The objective of the eavesdropper, however, is always to estimate the state matrix from which it can infer the network topology. In this work, we consider the privacy protection of the topology of a discrete time linear consensus network using the notion of Differential Privacy (DP) proposed by Dwork [4]. The main advantages provided by DP is that it abstracts away from the potential side information that the eavesdropper might have and provides a mathematical definition of level of privacy [11], [4]. As is standard in the mechanisms that guarantee DP, each agent adds a synthetic noise to its state measurements before sending them to the central estimator. The noise is designed in such a way that the eavesdropper cannot identify the topology uniquely from the noisy measurements. We analytically characterize the noise properties that ensures a specified level of DP.

We borrow the DP framework developed for discrete time dynamical system in [5] and apply it to the linear consensus problem. There are some other recent works that present DP mechanism for dynamical systems. In [6], the authors propose a DP mechanism for the consensus problem and generalize it for a general distributed control system in [7]. The authors in [9] improve on the results of [7] by careful noise addition and removal to achieve exact consensus. In [8] and [10], the authors present noisy update algorithms to ensure DP in optimization problems. However, all these works aim to protect the privacy of the initial conditions, inputs, reference trajectories or cost functions of different users. In contrast, our work aims to protect the network topology which is represented by the state evolution matrix. Note that the mapping from initial condition, input or reference trajectory to the output is linear whereas the mapping from the state matrix to the output is non-linear, which poses additional challenges. We derive bounds on the sensitivity of this non-linear mapping.

The contributions of the paper are as follows. We present a noise adding DP mechanism to protect the privacy of the topology of a consensus network. We analytically characterize the level of noise required to ensure DP to a specified level. Using numerical simulations, we illustrate that the noise degrades the performance of the topology estimation and eigenvalue estimation procedures. We show that although

* The authors are with Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556. (Email:<vkatewa@nd.edu, vijay.gupta.21@nd.edu>).

† The author is with Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695. (Email:<achakra2@ncsu.edu>).

978-1-4799-8684-2/$31.00 ©2015 AACC

the estimation accuracy for both the administrator and the eavesdropper may suffer equally for the topology identification problem, the administrator can have significant benefit over the eavesdropper if its goal is to estimate only the eigenvalues.

II. Problem Formulation

In this section, we present the consensus algorithm and discuss the identification methods to estimate the network topology and eigenvalues. We then provide the definition of differential privacy and present the DP mechanism.

A. Mathematical Notation

$\|.\|_p$ denotes the p-norm of a vector or the induced p-norm of a matrix with $p \in [1, \infty]$. Further, $\|.\|_F$ denotes the Frobenius norm of a matrix. If $\{y(k)\}_{k\geq 0}$ is a signal, then the truncated version of y upto time T is donated by $y[0:T]$. For a square matrix A, let $\rho(A)$ denote the spectral radius of A. I_N denotes the $N \times N$ identity matrix and $\mathbf{1}_N = [1, 1, \cdots, 1]^T \in \mathbb{R}^N$. $\delta(k)$ denotes the Kronecker Delta function. $Lap(0, b)^N$ denotes a N dimensional Laplace distribution with $i.i.d.$ components, each with probability density function $f(x) = \frac{1}{2b} e^{\frac{-|x|}{b}}$.

B. Consensus Algorithm

Consider a consensus network with N agents and let $\mathcal{N}$ denote the set of agent indices, i.e. $\mathcal{N} = \{1, 2, \cdots, N\}$. The agents are dynamically coupled and $p_{ij}, i \neq j$ denotes the edge weight of the edge between agent i and j. Further, the agents perfectly communicate their measurements to a central estimator. By convention, we adopt that $p_{ij} = 0$ if there is no edge between agents i and j. The agents implement the standard consensus algorithm [1] in which the i^{th} agent performs the following iteration

$$x_i(k+1) = \sum_{j=1}^{N} p_{ij} x_j(k). \tag{1}$$

where $x_i \in \mathbb{R}$ denotes the state of agent i. We assume that the initial condition $x_i(0) = 0$ for every agent.

For identification, the system needs to evolve from an initial condition that is not the origin. We model this as an impulse disturbance occurring at time $k = 0$. Thus, the evolution of the entire system can be described as

$$x(k+1) = Px(k) + Bd(k) \tag{2}$$

where $x(k) = [x_1(k), x_2(k), \cdots, x_N(k)]^T$, $[P]_{i,j} = p_{ij}$ and $d(k) = \delta(k)$ is the disturbance. The state evolution matrix P contains all the link weights p_{ij}. Thus, the topology of the network can be completely determined by the state matrix P. By a slight abuse of notation, we will sometimes denote both the state matrix and the network topology by P.

Assumptions A1: We make the following assumptions:

(i) The non diagonal weights of matrix P are non-negative, i.e. $p_{ij} \geq 0$ for all $i \in \mathcal{N}, j \in \mathcal{N}, i \neq j$. Further, the diagonal weights are positive, i.e. $p_{ii} > 0$ for all $i \in \mathcal{N}$.

(ii) The matrix P is row stochastic, i.e. $\sum_{j=1}^{N} p_{ij} = 1$ for all $i \in \mathcal{N}$.

(iii) The matrix P is symmetric. Combined with (ii), it implies that P is a doubly stochastic matrix.

(iv) The matrix P is irreducible. The implies that the graph associated with P is strongly connected.
Assumptions (i)-(iv) are standard assumptions made in the consensus problem [1].

(v) The disturbance input is an impulse and it acts at a single agent i_0 of the system. Without loss of generality we assume $i_0 = 1$. Thus, $B = [1, 0, \cdots, 0]^T$.

Since the input is an impulse function, the state evolution in (2) can be equivalently represented as the state evolving with the initial condition $x(1)$ and zero input

$$x(k+1) = Px(k), \qquad x(1) = B, k \geq 1 \tag{3}$$

It is well known [1] that under assumptions A1.(i)-(iv),

- All eigenvalues of P are inside the unit circle except the eigenvalue 1, which has algebraic multiplicity one.
- The consensus algorithm achieves average consensus

$$\lim_{k\to\infty} x_i(k) = \bar{x}_i \triangleq \frac{1}{N} \sum_{j=1}^{N} x_j(1). \tag{4}$$

C. Topology Identification Using NLS Estimation

For the topology identification purposes, each agent in the consensus network produces a measurement/output as $y_i(k) = c_i x_i(k)$ where $y_i \in \mathbb{R}$. All agents transmit their outputs to the central estimator and the received outputs can be collectively represented as

$$y(k) = Cx(k) \tag{5}$$

where, $y(k) = [y_1(k), y_2(k), \cdots, y_N(k)]^T \in \mathbb{R}^N$, and $C = diag(c_1, c_2, \cdots, c_N)$.

The central estimator (or a potential eavesdropper) uses these outputs to estimate the network topology. Since the outputs y are a non-linear function of the topology P, the estimate is obtained by solving the following Non-linear Least Squares (NLS) optimization problem

$$\min_{\tilde{P}} \left\| \sum_{k=1}^{T} y(k) - C\tilde{P}^{k-1} x(1) \right\|_2^2 \tag{6}$$

where $\tilde{P}$ and y follow from (3) and (5). Following [13], we assume $C = I_N$ and the pair (P, B) in (2) to be controllable so that despite the non-convexity of the NLS problem, the topology can be uniquely and accurately identified. We also assume that the central estimator knows the order N of the system.

D. Eigenvalue Identification Using LLS Estimation

In addition to the topology identification, the central estimator is often interested in knowing the eigenvalues of the system. For eigenvalue identification, it estimates the coefficients of the characteristic polynomial (referred to as parameters hereinafter) of the system from which the eigenvalues can be evaluated. For ease of notation and without loss of generality, we assume that the parameter estimation is performed using the output of a single agent (say i^{th} agent). The results extend to the case when the central estimator uses outputs of all agents. The parameters are related through the input-output equation

$$\begin{aligned} y_i(k)+a_1y_i(k-1)+\cdots+a_Ny_i(k-N) \\ = b_1u(k-1)+\cdots+b_{m_i}u(k-m_i) \end{aligned} \tag{7}$$

Let $\theta = [a_1, \cdots, a_N, b_1, \cdots, b_{m_i}]$ denote the parameter vector that needs to be estimated and let $\varphi(k) = [-y_i(k-1), \cdots, -y_i(k-N), u(k-1), \cdots, u(k-m_i)]$ denote the regression vector. Then, the outputs are a linear function of the parameters and can be represented as $y_i(k) = \varphi^T(k)\theta$. Thus, we can use the Linear Least Squares (LLS) estimation [12] to obtain the parameter estimate as

$$\hat{\theta}_T = \left(\frac{1}{T}\sum_{k=1}^{T}\varphi(k)\varphi(k)^T\right)^{-1}\frac{1}{T}\sum_{k=1}^{T}\varphi(k)y_i(k) \tag{8}$$

If the outputs are generated by actual parameters θ_0 as $y_i(k) = \varphi^T(k)\theta_0$, then we have $\hat{\theta}_T = \theta_0$ and thus, the LLS method provides perfect estimation and the eigenvalues of the system can be identified accurately.

E. Differential Privacy for Topology Identification

In an adverse scenario, an eavesdropper that has unauthorized access to the central estimator can use the outputs (5) to uniquely identify the topology, thus causing a privacy breach. To prevent this, we present a Differential Privacy (DP) mechanism that can be intuitively explained as follows. Suppose the mechanism ensures the following property: if a topology is *"changed "*, then the corresponding outputs are "*statistically not very different*". Then, by observing the outputs, the eavesdropper will not be able to distinguish between the two topologies with high confidence level and privacy of the topology will be preserved. Of course for the outputs not to be *very different*, the topology change should also be *bounded*. To make the outputs look statistically same, the DP mechanism adds a synthetic noise to the actual outputs. The noise level depends on the sensitivity between the topology and its output, as we will see in the next section. Next, we formally define what "*change within a specified bound*" and "*statistically not very different*" mean.

Definition 2.1: Adjacency: Two topologies P and $P^{'}$ are β-adjacent(denoted by $adj(\beta)$) if for some $\beta \geq 0$ we have

$$\left\|P - P^{'}\right\|_2 \leq \beta. \tag{9}$$

Remark 2.2: In the DP definition for static databases [4] and for dynamical systems [5], adjacency is defined w.r.t. the change of data/input of *one* agent while keeping the data/inputs of other agents unchanged. In contrast, our definition of adjacency allows changes in the topology that can possibly affect the weights of links between multiple agent pairs, including creation and deletion of links.

As mentioned before, the agents add noise to the outputs according to the following DP mechanism

$$\mathcal{M}: \qquad \tilde{y}(k) = y(k) + n(k), \tag{10}$$

where $n(k) = [n_1(k), \cdots, n_N(k)]^T \in \mathbb{R}^N$ is the noise. We will specify the properties of the noise in the next section.

Let $\tilde{y}_P$ denote the noisy output of the topology P. Note that $\tilde{y}_P[0:T] \in \mathbb{R}^{N(T+1)}$. Next, we give the definition of differential privacy.

Definition 2.3: Differential Privacy: The mechanism $\mathcal{M}$ in (10) is ϵ-differentially private upto time T if for any two β-adjacent topologies P and $P^{'}$ and for all $S \in \mathbb{R}^{N(T+1)}$

$$\mathbb{P}[\tilde{y}_P[0:T] \in S] \leq e^{\epsilon}\mathbb{P}[\tilde{y}_{P'}[0:T] \in S], \tag{11}$$

where $\epsilon > 0$ is the privacy parameter. The definition says that if the topology changes from P to $P^{'}$ that is β-adjacent to P, then the corresponding output statistics change only within a factor of e^{ϵ}.

F. Effect of DP Mechanism on Topology and Eigenvalue Identification

In case an eavesdropper gets an unauthorized access to the central estimator, it estimates the topology from the noisy outputs $\tilde{y}_P$ generated by topology P. The estimate is obtained by solving the NLS problem presented in (6) and can be represented as

$$\hat{P} = \arg\min_{\tilde{P}} \left\|\sum_{k=1}^{T}\tilde{y}_P(k) - C\tilde{P}^{k-1}x(1)\right\|_2^2 \tag{12}$$

The DP mechanism will ensure that the estimate $\hat{P}$ is almost equally likely for two cases when the outputs are generated by P and a β-adjacent topology $P^{'}$. Thus, it ensures indistinguishability among the set of β-adjacent topologies and results in the following topology estimation error for the eavesdropper

$$E = \mathbb{E}\left[\left\|\hat{P} - P\right\|_F\right], \tag{13}$$

where the expectation is taken $w.r.t$ the noise. We will present numerical simulation results on this estimation error in section IV and show that it increases with the increase in noise level.

We next characterize the effect of noise on parameter estimate $\hat{\theta}$. Assuming that the noisy outputs are generated by the actual parameter θ_0, using (7) and (10) we get

$$\tilde{y}_i(k) = \tilde{\varphi}^T(k)\theta_0 + v(k) \tag{14}$$

$$\begin{aligned} \text{where} \quad & v(k) \triangleq n_i(k) + a_1n_i(k-1) + \cdots + a_Nn_i(k-N) \\ \text{and} \quad & \tilde{\varphi}(k) = [-\tilde{y}_i(k-1), \cdots, -\tilde{y}_i(k-N), u(k-1), \cdots \\ & \cdots, u(k-m_i)]. \end{aligned}$$

With the above characterization, it can be shown [12] that if $v(k)$ and $\varphi(k)$ are quasi stationary and the system is stable, then as the number of observations T becomes asymptotically large

$$(i) \qquad b = \lim_{T\to\infty}\hat{\theta}_T - \theta_0 = (R^*)^{-1}f^* \tag{15}$$

$$\text{where} \quad R^* \triangleq \lim_{T\to\infty}\frac{1}{T}\sum_{k=1}^{T}\mathbb{E}\left[\varphi(k)\varphi(k)^T\right] \quad \text{and}$$

$$f^* \triangleq \lim_{T\to\infty}\frac{1}{T}\sum_{k=1}^{T}\mathbb{E}\left[\varphi(k)v(k)\right]$$

$$(ii) \qquad \lim_{T\to\infty}Cov(\hat{\theta}_T) \sim \frac{1}{T}I_{\theta}, \tag{16}$$

where $Cov(.)$ denotes the covariance and the expression of I_θ is complicated in general and is omitted here. Note that the noise $v(k)$ as defined in (14) is not white and thus it introduces a bias in the asymptotic estimate. In section IV, we will show the effect of noise on the estimate via numerical simulations.

Remark 2.4: The above mentioned LLS eigenvalue identification method results in a biased estimate. An alternate method to obtain an unbiased estimate is through the Instrumental Variable (IV) method [12], in which the estimate is calculated as a solution of the following equation

$$\frac{1}{N}\sum_{k=1}^{T}\zeta(k)\left[\tilde{y}(k)-\tilde{\varphi}(k)^T\theta\right]=0$$

where $\zeta(k)$ is called instruments and they are chosen such that $\zeta(k)$ and $v(k)$ are independent. It can be shown that this method leads to an unbiased asymptotic estimate with zero covariance as number of measurements becomes large. Thus, the IV method ensures that there is no degradation in the eigenvalue estimation performance while preserving the privacy of the network topology. We plan to propose and analyze this identification method in future extensions of this work.

Remark 2.5: We would like to emphasize that both the central estimator and eavesdropper will incur the same performance loss as given by (13) and (15) for topology and eigenvalue estimation, respectively. However, in many cases the eavesdropper is solely interested in determining the network topology whereas the central estimator is required to monitor only the eigenvalues of the system [18]. In such cases where the objectives of the eavesdropper and the central estimator are different, our study provides an interesting trade-off between the two.

Problem Objective: Given the above system setup, the goal of this paper is to design the noise $n(k)$ that ensures that the DP definition is satisfied for any given privacy parameter ϵ and to characterize the resulting trade-off between the level of privacy and the performance degradation as defined by (13) and (15) . We present the design criteria in the next section.

III. Differential Privacy Mechanism

In this section, we present a mechanism to add noise for protecting the differential privacy of the network topology.

A. The Noise Mechanism

A standard way to implement DP mechanism is to add Laplacian noise to the outputs, with the noise level depending on the sensitivity of the system [4], [16]. In our privacy problem, the goal is to protect the topology from an eavesdropper that has access to the system outputs. Thus, we consider the sensitivity from the topology P to the outputs y. If the sensitivity is low, then for two different topologies, the change in the corresponding outputs will not be large. Thus, the level of noise required to make the two outputs "statistically not very different" will also be small. On the other hand, if sensitivity is large, a large level of noise is needed to ensure DP. Thus, sensitivity plays a crucial role in noise design.

Definition 3.1: The system sensitivity upto time T is defined as

$$\Delta(T)=\sup_{P,P':adj(\beta)}\left\|y_P[0:T]-y_{P'}[0:T]\right\|_1. \quad (17)$$

Sensitivity characterizes the maximum possible difference in the outputs for any two β-adjacent topologies. It depends on the system parameters and the adjacency bound β. We will provide the characterization of sensitivity in terms of these parameters later in this section. The next theorem shows that sensitivity provides a sufficient condition for noise design.

Theorem 3.2: The mechanism $\mathcal{M}$ in (10) is ϵ-differentially private upto time T if $n(k)$ is white Laplacian noise with the distribution $n(k)\sim Lap(0,c)^N$ and $c\geq\frac{\Delta(T)}{\epsilon}$.

Proof: See [5], Theorem 2. ■

Remark 3.3: The noise parameter c is directly proportional to the sensitivity as explained intuitively at the start of the section. Further, it is inversely proportional to the privacy parameter ϵ. Observe from the DP definition (11) that privacy increases as ϵ decreases and vice versa. Thus, for small values of ϵ, higher noise level is required to ensure higher level of privacy.

It is apparent from the preceding theorem that characterization of sensitivity of the system is required to design the DP mechanism. However, it is infeasible to obtain an exact analytical expression for sensitivity. Thus, we obtain an upper bound on the sensitivity. The upper bound also provides a sufficient condition for the noise level that can ensure DP.

B. Characterizing Privacy Level

We start by defining the error between the state and the final consensus value as $e(k)=x(k)-\bar{x}$, where $\bar{x}=[\bar{x}_1,\bar{x}_2,\cdots,\bar{x}_N]^T$. We have the following facts:

1) The final consensus value can also be written as $\bar{x}=\frac{\mathbf{1}_N\mathbf{1}_N^T}{N}x(1)$, where $\mathbf{1}_N=[1,1,\cdots,1]^T\in\mathbb{R}^N$.
2) Since P is stochastic, $P\mathbf{1}_N=\mathbf{1}_N$ and $\mathbf{1}_N^TP=\mathbf{1}_N^T$.

Next, we present the dynamics of the error evolution.

Lemma 3.4: The evolution of the error can be stated as

$$e(k+1)=\tilde{P}e(k) \quad \text{where} \quad \tilde{P} \triangleq \left(P-\frac{\mathbf{1}_N\mathbf{1}_N^T}{N}\right). \quad (18)$$

Proof: The proof can be derived using the preceding facts and is ommitted due to lack of space. ■

Being consistent with (3), we assume that the error evolution starts at time $k=1$ with $e(1)=\left(I_N-\frac{\mathbf{1}_N\mathbf{1}_N^T}{N}\right)x(1)$. It can be shown [14] that all eigenvalues of $\tilde{P}$ lie inside the open unit circle. The output of the system can be written as

$$y(k)=C(e(k)+\bar{x}). \quad (19)$$

Further, let ρ_{max} denote the largest possible spectral radius of all $\tilde{P}$, i.e. $\rho_{max}=\sup_{P}\ \rho(\tilde{P})$. Since $\rho(\tilde{P})<1$ for all $\tilde{P}$, we get $\rho_{max}<1$.

We use the following lemma from [15] to derive the sensitivity bound.

Lemma 3.5: For two symmetric matrices P and Q

$$\|P^k - Q^k\|_F \le k\left(\max\{\rho(P), \rho(Q)\}\right)^{k-1}\|P - Q\|_F \quad (20)$$

Proof: The proof follows from Theorem 7 in [15]. ■

We now present the sensitivity bound.

Theorem 3.6: The sensitivity $\Delta(T)$ is upper bounded by

$$\bar{\Delta}(T) \triangleq 2\|C\|_1\|x(1)\|_1(N-1)\beta S_{\rho_{max}}(T-1)$$

where $\displaystyle S_r(T) = \sum_{k=1}^{T} k r^{k-1} = \frac{1 - r^T}{(1-r)^2} - \frac{T r^T}{1-r}.$

Proof: For measurements y_P and $y_{P'}$ produced by two β-adjacent topologies P and $P^{'}$, we have

$$\begin{aligned}
\|y_P(k) - y_{P'}(k)\|_1 &\overset{(a)}{=} \|Ce_P(k) - Ce_{P'}(k)\|_1 \\
&\overset{(b)}{\le} \|C\|_1\|e(1)\|_1\|\tilde{P}^{k-1} - (\tilde{P}^{'})^{k-1}\|_1 \\
&\overset{(c)}{\le} \|C\|_1\|e(1)\|_1\sqrt{N}\|\tilde{P}^{k-1} - (\tilde{P}^{'})^{k-1}\|_F \\
&\overset{(d)}{\le} \|C\|_1\|e(1)\|_1\sqrt{N}(k-1)\rho_{max}^{k-2}\|\tilde{P} - \tilde{P}^{'}\|_F \\
&\overset{(e)}{\le} \|C\|_1\|e(1)\|_1 N(k-1)\rho_{max}^{k-2}\beta
\end{aligned}$$

where, (a) follows from (19), (b) follows from Lemma 3.4 and submiltiplicative property of norm, (c) follows from the matrix norm property $\frac{||.||_1}{\sqrt{N}} \le ||.||_2 \le ||.||_F$, (d) follows from Lemma 3.5 and definition of ρ_{max} and (e) follows from the matrix norm inequality $||.||_F \le \sqrt{N}||.||_2$ and since P and $P^{'}$ are adjacent, $\|\tilde{P} - \tilde{P}^{'}\|_2 = \|P - P^{'}\|_2 \le \beta$. Thus,

$$\begin{aligned}
\left\|y_P[0:T] - y_{P'}[0:T]\right\|_1 &= \sum_{k=2}^{T}\|y_P(k) - y_{P'}(k)\|_1 \\
&\le \|C\|_1\|e(1)\|_1 N\beta S_{\rho_{max}}(T-1)
\end{aligned}$$

Moreover,

$$\|e(1)\|_1 = \left\|\left(I_N - \frac{\mathbf{1}_N\mathbf{1}_N^T}{N}\right)x(1)\right\|_1 \le 2\left(\frac{N-1}{N}\right)\|x(1)\|_1$$

The theorem then follows from the above inequalities. ■

Remark 3.7:

- Using the sensitivity bound, the noise $n(k)$ in Theorem 3.2 can be generated by setting

$$c = \frac{\bar{\Delta}(T)}{\epsilon} = 2\|C\|_1\|x(1)\|_1(N-1)S_{\rho_{max}}(T-1)\frac{\beta}{\epsilon} \quad (21)$$

  Note that in (21), β and ϵ are the privacy design parameters and the ratio $\gamma \triangleq \frac{\beta}{\epsilon}$ represents the privacy level. If the adjacency parameter β increases with a fixed ϵ, it signifies that the DP is ensured for a larger set of topologies. Similarly, if ϵ decreases, it also signifies an increase in privacy level as explained in remark 3.3.
- As T becomes very large, $S_{\rho_{max}}(T)$ converges to $\frac{1}{(1-r)^2}$. Thus, the noise level is bounded for all T.

IV. Simulation Results

We consider a dummy consensus network with 4 agents for the simulations. The network is fully connected and the topology is given by

$$P = \begin{bmatrix} 0.1 & 0.3 & 0.2 & 0.4 \\ 0.3 & 0.3 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.4 & 0.2 \\ 0.4 & 0.2 & 0.2 & 0.2 \end{bmatrix}$$

Further, as stated in the assumptions $B = \begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}^T$, $C = I_4$, and $\|x(1)\|_1 = \|B\|_1 = 1$. The noise is added according to the DP mechanism $\mathcal{M}$ in (10). To generate the Laplacian noise, we use the upper bound to the sensitivity obtained in Theorem 3.6. We assume $\rho_{max} = 0.7$ and simulate the system till $T = 100$ time steps. Thus, $S_{0.7}(T-1) \sim 11.1$, and the noise level is given by

$$c = \frac{\bar{\Delta}(T)}{\epsilon} = 66.6\frac{\beta}{\epsilon} = 66.6\gamma \quad (22)$$

Figure 1 shows the noisy outputs for various values of γ. In Figure 1a, noise is absent and we can observe that all outputs (=states) converge and average consensus is achieved. In Figures 1b and 1c, the noise perturbs the outputs from the average value.

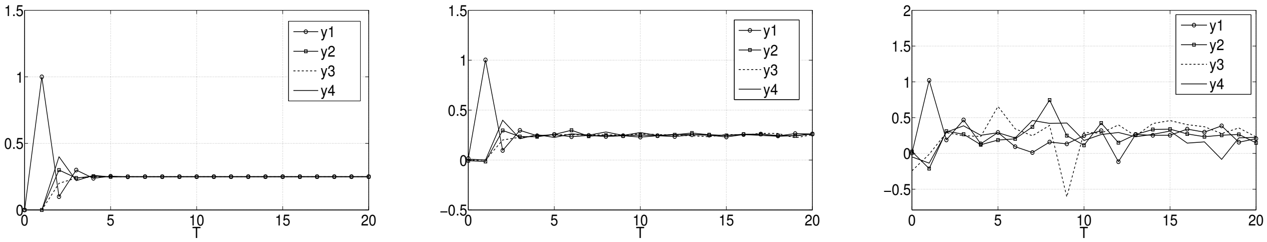
A. Topology Estimation

The eavesdropper obtains the topology estimate $\hat{P}$ by solving the NLS optimization problem given in (12). We use the MATLAB `fmincon` function to obtain the NLS estimate. We approximate the expected topology error E in (13) by running the simulation for multiple noise realizations and then take the sample mean. Figure 2 shows the topology estimation error as a function γ. As expected, an increase in the privacy level requires an increase in the noise level which degrades the estimation performance. Also, notice that for $\gamma = 0$, no privacy is guaranteed. Thus, the estimation error is zero and the eavesdropper obtains the exact topology. Further, Table I shows the topology estimates $\hat{P}$ for two different values of γ. It can be observed that the weights of the estimated topologies are different from the actual topology P due to the DP-induced noise. For example, the noise misleads the eavesdropper to estimate that there are no links between agents $2-4$ and $3-4$.

B. Eigenvalue Estimation

The central estimator identifies the eigenvalues of the system by estimating the parameter θ in (7) using the LLS estimation. For calculating the eigenvalues of the system, only parameters $\underline{\theta} = [a_1, \cdots, a_N]$ are required. We perform the estimation using the MATLAB `arx` function. In accordance with (16), the covariance of $\hat{\underline{\theta}}$ becomes zero asymptotically and hence, the estimates converge to a steady state value given by $\lim_{T\to\infty}\hat{\underline{\theta}}_T = [-0.39, -0.41, -0.17, -0.19]$.

Comparing with the actual parameter value of the system $\underline{\theta}_0 = [-1, -0.06, 0.064, -0.004]$, we see that the estimate is biased according to (15). Figure 3 shows the norm of the bias $\|b\|_2^2$ as a function of privacy level γ. Observe that as the privacy level increases, the bias degrades due to the increasing noise required to maintain privacy. Thus, there is a trade-off between the desired privacy level and the parameter estimation error. Further, Table II shows the actual eigenvalues of the system and the estimated eigenvalues for two different values of γ.



(a) $\gamma = 0$

(b) $\gamma = 1.5 * 10^{-4}$

(c) $\gamma = 1.5 * 10^{-3}$

Fig. 1: Output Evolution.

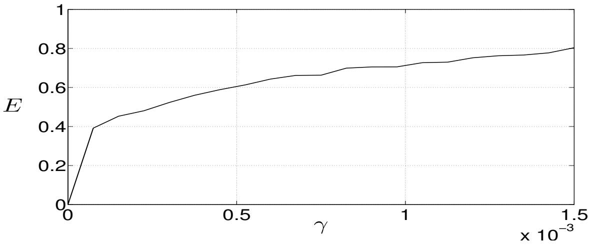


Fig. 2: Topology Estimation Error

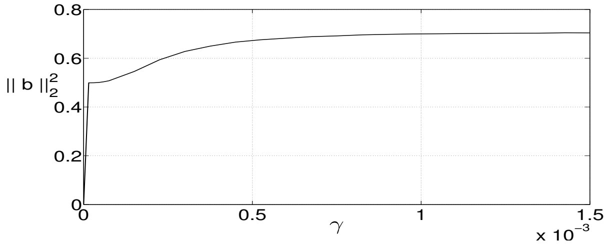


Fig. 3: Parameter Estimation Error

$\gamma = 1.5 * 10^{-4}$	$\gamma = 1.5 * 10^{-3}$
$\begin{bmatrix} 0.12 & 0.30 & 0.20 & 0.38 \\ 0.30 & 0.57 & 0.07 & 0.06 \\ 0.20 & 0.07 & 0.44 & 0.29 \\ 0.38 & 0.06 & 0.29 & 0.27 \end{bmatrix}$	$\begin{bmatrix} 0.07 & 0.21 & 0.29 & 0.43 \\ 0.21 & 0.19 & 0.60 & 0 \\ 0.29 & 0.60 & 0.11 & 0 \\ 0.43 & 0 & 0 & 0.57 \end{bmatrix}$

TABLE I: Estimated Topology

Actual Eigenvalues	Estimated Eigenvalues	
($\gamma = 0$)	$\gamma = 1.5 * 10^{-4}$	$\gamma = 1.5 * 10^{-3}$
1.00	1.00	1.00
-0.27	-0.22 +i 0.26	-0.24 + i0.50
0.20	-0.22 - i0.26	-0.24 - i0.50
0.07	-0.16	-0.23

TABLE II: Estimated Eigenvalues

V. Conclusion

In this paper, we presented a differential privacy mechanism for protecting the topology of a consensus network. The mechanism adds noise to the outputs of the system and prevents an eavesdropper from correctly identifying the topology from the outputs. We derived an analytical bound for the sensitivity for designing the noise. The simulations verify that the DP induced noise degrades the topology estimation performance of the eavesdropper, thus protecting the privacy of the topology. Further, the noise also degrades the eigenvalue estimation performance of the system administrator, thereby showing a trade-off between level of privacy and estimation error. We plan to extend the DP framework developed in this paper for topology identification in a more general dynamical network such as power grid.

References

[1] R. Olfati-Saber, J.A. Fax and R. M. Murray, "Consensus and Cooperation in Networked Multi-Agent Systems," *Proceedings of the IEEE*, Vol. 95, No. 2, pp. 215-233, 2007.

[2] G.A. Pagani and M. Aiello, "The Power Grid as a Complex Network: A Survey," *Physica A*, Vol. 392, No. 11, pp. 2688 - 2700, 2013.

[3] M. Mesbahi and M. Egerstedt, "Graph Theoretic Methods in Multiagent Networks," *Princeton University Press*, 2010.

[4] C. Dwork, "Differential Privacy," *Proceedings of ICALP*, Vol. 4052, pp. 1-12, 2006.

[5] J. Le Ny and G. J. Pappas, "Differentially Private Filtering," *IEEE Transactions on Automatic Control,* Vol. 59, No. 2, pp. 341-354, 2014.

[6] Z. Huang, S. Mitra and G. Dullerud, "Differentially Private Iterative Synchronous Consensus," *In Proceedings of the ACM Workshop on Privacy in the Electronic Society, WPES*, pp. 81 - 90, 2012.

[7] Z. Huang, Y. Wang, S. Mitra, G. Dullerud, "On the Cost of Differential Privacy in Distributed Control Systems," *In The 3rd ACM International Conference on High Confidence Networked Systems (HiCoNS)*, 2014.

[8] Z. Huang, S. Mitra and N. Vaidya, "Differentially Private Distributed Optimization," *arXiv Preprint arXiv:1401.2596*, 2014.

[9] Y. Mo and R. M. Murray, "Privacy Preserving Average Consensus," Submitted to *IEEE Conference on Decision and Control*, 2014.

[10] S. Song, K. Chaudhuri and A. D. Sarwate, "Stochastic Gradient Descent with Differentially Private Updates," *IEEE Global Conference on Signal and Information Processing,* pp. 245-248, 2013.

[11] C. Dwork, "A Firm Foundation for Private Data Analysis," *Communications of the ACM,* Vol. 54, No. 1, pp. 86-95, 2011.

[12] L. Ljung, "System Identification: Theory for the User," *Prentice Hall*, 1999.

[13] R. Bellman and K. J. Astrom, "On Structural Identifiability, *Mathematical Biosciences*, Vol. 7, pp. 329 - 339, 1970.

[14] L. Xiao and S. Boyd, "Fast Linear Iterations for Distributed Averaging," *Systems and Control Letters*, Vol. 53, pp. 65-78, 2004.

[15] E. Jarlebring and E. H. Rubensson, "On the Condition Number and Perturbation of Matrix Functions for Hermitian Matrices," *arXiv Preprint arXiv:1206.1762*, 2012.

[16] C. Dwork et. al., "Calibrating Noise to Sensitivity in Private Data Analysis," *Theory of Cryptography Conference*, pp. 265-284, 2006.

[17] S. Nabavi and A. Chakrabortty, "Topology Identification for Dynamic Equivalent Models of Large Power System Networks," *American Control Conference*, pp. 1138 - 1143, 2013.

[18] Y. Liu, P. Ning and M. K. Reiter, "False Data Injection Attacks Against State Estimation in Electric Power Grids, *ACM Conference on Computer and Communications Security*, pp. 2132, 2009.