Distributed Topology-preserving Collaboration Algorithm against Inference Attack

Zitong Wang, Yushan Li, Chongrong Fang and Jianping He

***Abstract*—Interaction topology through which agents achieve intelligent collaboration in multi-agent systems is of fundamental importance. Recently, many efforts have been devoted to the problem of topology inference, e.g., the trajectory information of mobile agents is utilized to regress the topology. In this paper, we develop a distributed topology-preserving collaboration algorithm for multi-agent systems against the topology inference attacks, aiming to achieve the optimal trade-off between the crypticity and the precision of collaboration. The novelties lie in that: i) the proposed algorithm can be applied for both undirected and directed networks, largely degenerating the attackers' topology inference ability. We prove the convergence and analyze the performance of the algorithm. ii) To characterize the properties of the algorithm, we propose the indicator of the algorithm's cost and evaluate the balance between maximizing the inference errors and minimizing the extra cost of modifying the original paths. iii) Lastly, we provide numerical examples to verify the effectiveness of the presented algorithm, and the flatness and flexibility of the algorithm are confirmed.**

I. Introduction

Multi-agent system (MAS) has been widely used in applications such as distributed computing [1], sensor networks [2], multi-robot systems [3], [4], and data aggregation [5]. The interaction topology of MAS, which characterizes the ability of agents to interact with others, is essential for agents to achieve efficient consensus-based collaboration. Besides, the topology will affect the autonomy, adaptation, scalability, and efficiency of the MAS [6]. Due to its significance, the research on topology has received great attention from researchers in various areas, including computer science, communication, and control theory [7], among which topology inference is a major concern in recent years.

Currently, the research on topology inference has achieved fruitful results. For instance, [8] computes the states of the agents by an iterative root-searching method driven by a maximum likelihood function, and [9] focuses on inferring the directed network topology under unmeasurable latent inputs. In addition, optimization methods are also widely used in topology inference, which aim to calculate the topology quickly and accurately. In this context, the outside observers can obtain the topology by collecting a set of observation data and then solving a well-formulated regression problem.

Unfortunately, all these methods can also be employed by malicious adversaries to infer the topology within the system (we call it a topology inference attack). Once the topology is accurately inferred, the adversary can launch further premeditated attacks on a certain critical agent in the MAS to drive the system into a state of paralysis. Taking multiple mobile agents as an example [10], an outside attacker can observe the moving trajectory of the agents and infer their internal topology structure by the aforementioned methods. Then, the attacker obtains critical guidance to incapacitate the target agent and largely degenerate the collaboration performance of the MAS. Therefore, it is necessary for MASs to design the topology-preserving collaboration algorithms against topology inference attacks.

To counter such kinds of attacks, researchers have dug deeply to study this problem, and propose a series of defense mechanisms, where the consensus algorithm is the fundamental tool to ensure their collaboration performance. The methods can be divided into two main categories: dynamic topology and noise-adding algorithms. In the former kind of studies, the interaction topology of the agents changes from time to time, which notably increases the difficulty for the outside attackers to infer an accurate and stable topology. In the pursuit of the consensus in MAS under these topologies, the key is to design appropriate protocols that can guarantee that the group of agents will reach the consensus point on the shared information with limited and unreliable information exchange and switching topologies [11]–[13]. According to [14], if the interaction topology changes frequently enough as the system evolves, the consensus can be achieved in the end. The disadvantage of these methods is that they potentially high resource consumption due to the frequent change of topologies and the strong dependence on the system. Additionally, the size and connectivity impose critical restrictions on implementation.

The noise-adding based methods are also commonly used to secure the internal information of MASs [15]–[18]. The main idea is to impose additional noisy signals on agents' states during the collaboration period, thereby hiding the true information from the attackers. Specifically, [15] proposes a differential privacy scheme based on Laplace noise and preserves the privacy of the agent state in the consensus process. In addition, [16] investigates the performance of additive injected noises for MASs and presents a theoretical analysis for the consensus error. [18] brings up a noise-injection-based algorithm to preserve the topology information, while achieving average consensus in the sense of mean square convergence. The analytical form for Gaussian noise is also derived by the maximum likelihood estimation. Nevertheless, most noise-adding algorithms focus more on data privacy of agents rather than the privacy of the topology of the system.

The authors are with the Department of Automation, Shanghai Jiao Tong University, and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai, China. E-mail address: {wangzitong, yushan_li, crfang and jphe}@sjtu.edu.cn.

Motivated by above observations, in this paper, we propose a distributed topology-preserving collaboration algorithm (DTCA). It is effective in concealing the actual topology structure from the outside inference attack, while guarantees the accurate collaboration convergence. The challenges lie in how to design the noise-adding mechanism in a distributed way while pursue the maximum inference attack degradation. The main contributions of our work are listed as follows:

- We investigate the collaboration algorithm design that secures the interaction topology while guaranteeing the collaboration performance. Specifically, the proposed distributed topology-preserving collaboration algorithm works for both undirected and directed networks to degenerate the performance of inference attacks.
- By exploiting the sufficient and necessary conditions of accurate collaboration convergence, we design a distributed two-fold noise-adding algorithm, which is adaptively bounded, asymptotically decaying, and does not rely on randomized implementation.
- Based on the self-compensating characteristic of the DTCA algorithm, we prove it has the same exponential convergence rate as the original one without any noise involved. Representative simulation examples demonstrate the effectiveness of the DTCA algorithm.

The rest of the paper is organized as follows: Section II provides some preliminary knowledge and formulates the problem. The proposed algorithm and its performance analysis are in Section III. Section IV shows the simulation and comparison results. Finally, Section V concludes the work.

II. PRELIMINARIES

Let $G = (V, E)$ be a directed graph to model the topology information within the multi-agent system, where $V = \{1, \dots, N\}$ is the set of agents and $E \subseteq V \times V$ denotes the set of edges. Each agent represents an agent, and each weighted edge represents an information transformation channel. The adjacency matrix $A = [a_{ij}]_{N\times N}$ of a graph G with N agents specifies the interconnection topology of the system, where $a_{ij} > 0$ iff $(i, j) \in E$, else $a_{ij} = 0$. Let $N_i = \{j \in V : a_{ij} \neq 0\}$ be the neighbor set of agent i. Define Laplacian matrix L of G as $L = D - A$, where D is the diagonal matrix of in-degrees. Let $\mathbf{1} = [1, 1, \dots, 1]^T \in R^N$, then clearly $L \cdot \mathbf{1} = 0$. Denote the set of positive integers as $\mathbb{N}^+$. Define the infinite norm of the matrix as $\|X\|_\infty$ and the Frobenius norm as $\|X\|_F$.

A. Consensus-based collaboration Algorithm

The consensus algorithm is the most widely used algorithm in multi-agent systems for collaboration. Thus, in this section, we will first introduce the consensus-based collaboration algorithm. Consider a multi-agent system of N agents moving in DTC (Discrete-Time Control) case. The initial state of agent i is denoted by $x_i(0)$, and the state at iteration k by $x_i(k)$. The dynamics of the state of agent i under a consensus-based collaboration algorithm is described as:

$$x_i(k+1) = x_i(k) + u_i(k), \tag{1}$$

where

$$u_i(k) = \sum_{j\in N_i} w_{ij}(x_j(k) - x_i(k)).$$

If set $w_{ij} > 0$ for $j \in N_i$ and $w_{ii} = 1 - \sum_{j\in N_i} w_{ij} > 0$, then all agents' states in the MAS will converge by the consensus algorithm in [5]:

$$\lim_{k\to\infty} x_i(k) = x_c, \forall i \in V,$$

where x_c is a constant. Furthermore, if W is a row stochastic matrix, we have $x_c = \bar{x} = \frac{\sum_{i\in V} x_i(0)}{N}$, i.e., an average consensus is achieved. If W is a doubly-stochastic matrix, we have $x_c = p_1^{\mathsf{T}} x(0)$ where p_1 is the first normalized left eigenvector, i.e., a weighted average consensus is achieved.

Agents seek to achieve a global consensus by using their local neighbors' information. This process is tightly coupled with the weights setting among the neighboring agents. It means that the consensus algorithm mainly depends on the weights, which will directly reveal the topology information of the MAS.

Then, we rewrite the above consensus algorithm in matrix form as follow:

$$x(k+1) = Wx(k), \tag{2}$$

where W is a row stochastic matrix. $W \cdot \mathbf{1} = \mathbf{1}$ can ensure the weighted average consensus.

B. Topology Inference Mechanism

Suppose the information the attacker receives is given by

$$\tilde{x}(k) = x(k) + \delta(k),$$

where the observation noise $\delta(k) \sim \mathcal{N}(0, \sigma^2 I)$ follows a Gaussian distribution. Consider attackers who can collect the agents' trajectories information

$$\mathcal{I}_a^b = [\tilde{x}(a), \cdots, \tilde{x}(b)],$$

from iteration a to iteration b, and infer the topology of the system, which will help predicting the most possible future positions and formulating a series of attacks.

After collecting the set of information, $\mathcal{I}_a^b$, the attackers will select a method to infer the topology. Suppose the attackers use some widely-adopted optimization methods, such as Least Square Estimation [19] [9], to make the inference faster and more accurate. The Least Square Estimation works on topology inference by establishing a mapping of the location information of the agents in a specific time window and minimizing the residuals' quadratic sum.

To approximate the topology of the system, $\mathcal{I}_a^b$ should include more than $N + 1$ groups of data, i.e., $b - a \geq N$. Separate data matrix Y and observation matrix Z where each column vector in Z is the mapping for Y's vector in the next iteration. The equation about the unknown W can be written as $\hat{W}Y = Z$ whose normal form is $Y^{\mathsf{T}}\hat{W}^{\mathsf{T}} = Z^{\mathsf{T}}$.

Then, the least square optimization problem to estimate W is formulated as follow:

$$\min_{\hat{W}} \|Y^{\mathsf{T}}\hat{W}^{\mathsf{T}} - Z^{\mathsf{T}}\|_F^2. \tag{3}$$

If matrix Y^T is specific and column full rank, the minimum variance unbiased estimator of the over-determined system is $W^* = ZY^\mathsf{T}(YY^\mathsf{T})^{-1}$, which is the optimal approximation of the topology. Given $Z^\mathsf{T} = Y^\mathsf{T}\hat{W}^\mathsf{T} + \varepsilon$, in general, $\varepsilon \in R^{k\times N}$ is a random variable with statistical property $\varepsilon \sim \mathcal{N}(0, I_{k\times N})$ thus the expectation of $\hat{W}$ is $E[\hat{W}] = W$, and we will have

$$E[(\hat{W} - W)(\hat{W} - W)^\mathsf{T}] = (YY^\mathsf{T})^{-1}. \tag{4}$$

C. Problem Formulation

In this paper, we mainly consider how to conceal the actual topology of the system by adding noise to the states of the agents. Instead of adding random noises with a fixed distribution, we add well-designed noises to the agents, which are more controllable and can ensure convergence. Then, we change the regular consensus algorithm (1) to

$$x_i(k+1) = w_{ii}x_i(k) + \sum_{j\in N_i} w_{ij}x_j(k) + \theta_i(k), \tag{5}$$

and rewrite formula (2) to update global states

$$x(k+1) = Wx(k) + \theta(k), \tag{6}$$

where $\theta(k) \in R^{N\times 1}$ is the vector of designed noises at iteration k, and W here is also a row stochastic matrix. Thus, it is easy to see that the dynamics of agents' states will not only be affected by the weights (indicate the topology information) but also depend on the noises. Intuitively, the higher irregularity of the noise $\theta(k)$ is, the more complex the regression of the topology based on the observations is. Meanwhile, the noise cannot be arbitrary since it needs to ensure the final convergence, especially when the final convergence should be the exact average consensus.

Therefore, the objective of this paper is to design a privacy-protecting mechanism that is effective and convergent to interfere with the topology inference from attackers while maintaining the property of consensus. The core problem is to design the noise-adding scheme to achieve a trade-off between the regression error requirements and the extra costs. Hence, we formulate the following optimization problem:

$$\begin{aligned} &\max_{\theta} \min_{\hat{W}} \|Y^\mathsf{T}\hat{W}^\mathsf{T} - Z^\mathsf{T}\|_F^2 \\ &\text{s.t.} \lim_{k\to\infty} x_i(k) = \bar{x}, \forall i \in V. \end{aligned} \tag{7}$$

For this problem, the primary concern is to design optimal inputs to maximize the inference error from the attackers and guarantee exact convergence, i.e., the weighted average consensus is ensured.

III. MAIN RESULTS

A. Algorithm Design

To fulfill the requirements of the formulated problem, we design the algorithm DTCA with the concerns in two aspects: alternatively add two different kinds of noises, thus enlarging the regression errors while ensuring the convergence, and set bounds of the noises to improves the flatness of the iteration progress.

Practically, we define a finite set $K = [k_0, k_1, \cdots, k_{K-1}]$, add additive noise $\theta_i^+(k)$ to $x_i(k)$ when $k \in K$ to increase the irregularity of the states, and balance the effect of the additive noise on convergence by imposing compensating noise $\theta_i^-(k+m)$ after m iterations. In general, the expression of the extra input to the original state $x_i(k)$ will be

$$\theta_i(k) = \begin{cases} \theta_i^+(k), & k \in K, \\ \theta_i^-(k), & (k-m) \in K, \\ 0, & \text{otherwise}, \end{cases} \tag{8}$$

where $m \in \mathbb{N}^+$ indicates the compensation gap.

As mentioned in section II, an agent's state in the next iteration depends on its neighbors' states. Denote the state of agent i in the regular iteration k as $x_i^r(k)$, where the regular iteration process follows (1). For simplicity, we denote $x_{N_i}(k)$ as $x_j(k), \forall j \in N_i$. Since W is a row stochastic matrix, in the regular iteration, we will have

$$\min\{x_{N_i}(k)\} \le x_i^r(k+1) \le \max\{x_{N_i}(k)\},$$

which implicates $x_i^r(k+1)$ will not exceed the extreme values of its neighbors. On the contrary, the inputs designed intentionally to break through the limitations will enlarge the inference errors. Designing the range of the additive noise with an extra parameter α, the upper boundary becomes

$$\beta^+(k) = \alpha \times (\max\{x_{N_i}(k)\} - x_i^r(k+1)), \tag{9}$$

and the lower boundary becomes

$$\beta^-(k) = \alpha \times (\min\{x_{N_i}(k)\} - x_i^r(k+1)). \tag{10}$$

In this regard, $\alpha = 1$ indicates that the bounds $x_i(k+1)$ can reach are the supposed boundaries of its neighbors' states, with $\alpha > 1$ beyond and $\alpha < 1$ within. The compensating noise is formulated as

$$\theta^-(k+m) = -W^m \times \theta^+(k). \tag{11}$$

Algorithm 1 DTCA

Input:
$x(0)$, w_{ij}, α;
Output:
Agents' states update following (5) and obtain $x(k+1)$;

for iteration k **do**
  if $k \in K$ **then**
    for agent i **do**
      Calculate the upper bound $\beta^+(k)$ and the lower bound $\beta^-(k)$ as in (9) and (10);
      Choose the additive noise $\theta_i^+(k)$ between the bounds, i.e., $\theta_i^+(k) \in [\beta^-(k), \beta^+(k)]$;
    end for
    Compute the compensating noise using (11);
  end if
  Update the states by (6)
  return $x(k+1)$.
end for

Remark 1: This algorithm can be used in both undirected and directed networks. The W of a directed network is a row stochastic matrix with $\theta^-(k+m) = -W^m \times \theta^+(k))$. In contrast, the W of an undirected network is a doubly-stochastic matrix with $\theta^-(k+m) = -W^s \times \theta^+(k)$ where s is not necessarily equal to m.

Remark 2: The compensation gap m also suggests that it is an m-hop neighborhood network. When $m = 1$, the system is distributed since each agent can acquire information from its direct in-degree neighbors. When $m = 2$ or higher, each agent must acquire more messages about the whole system, as it needs to know the information of neighbors' neighbors.

Besides, the value of m is enormously influential to the overall performance of the algorithm. The smaller m is, the shorter the compensation gap is, indicating a more remarkable change in the trajectory.

B. Convergence Analysis

When the DTCA is applied to the system, the added noises to the agents will confuse not only the attackers but also the agents in the neighborhood. In order to ensure the regular operation of the system, convergence to the original consensus point is the first thing to consider.

As is mentioned in [20], for the regular iteration, the asymptotic convergence factor of W is:

$$r_{asym}(W) = \sup_{x(0)\neq x_c} \lim_{k\to\infty} \left(\frac{\|x(k) - x_c\|_2}{\|x(0) - x_c\|_2} \right)^{\frac{1}{k}} \tag{12}$$

We arrange all eigenvalues of W in the descending order as $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$. The essential spectral radius of W is equal to the asymptotic convergence factor,

$$\rho_{ess} = r_{asym}(W) = \max\{|\lambda_2|^2, \cdots, |\lambda_n|^2\},$$

suggesting that the weighted average consensus is achieved exponentially.

Theorem 3.1: Given any $x(0)$, an exact average consensus is achieved exponentially using the DTCA, i.e., $\lim_{k\to\infty} x_i(k) = x_c$, where the convergence speed is ρ_{ess}.

Proof: From (6), it is not difficult to infer

$$\begin{aligned} x(k) &= Wx(k-1) + \theta(k-1) \\ &= W^k x(0) + \sum_{t=0}^{k-1} W^t \theta(k-t-1). \end{aligned} \tag{13}$$

Since W is a row stochastic matrix, it follows from [20] that the consensus can be achieved exponentially, and we have

$$\lim_{k\to\infty} x_i^r(k) = x_c.$$

Thus, one infers that

$$\lim_{k\to\infty} W^k x(0) = x_c$$

and the convergence speed is ρ_{ess}. Then, consider the second item in (13), if

$$\sum_{t=0}^{k} W^t \theta(k-t) = 0 \tag{14}$$

holds true, $\lim_{k\to\infty} x_i(k) = x_c$ is true under (6), ensuring the weighted average consensus.

Note that in the DTCA, each additive input $\theta^+(k)$ has its compensation $\theta^-(k+m) = -W^m \times \theta^+(k)$. Therefore,

$$W^m \theta^+(k) + \theta^-(k+m) = 0,$$

which means that $\sum_{t=0}^{t=k-1} W^t \theta(k-t-1) = 0$, verifying the convergence of the DTCA. Meanwhile, the noise adding process is run in a finite time due to $k_{K-1} \in K$ is finite and $\theta(k) = 0$ holds for $\forall k > k_{K-1}$. One thus conclude that the convergence speed is ρ_{ess}. The proof is completed. □

Furthermore, we define $f_s(x)$ as the sum of the elements in the column vector x, e.g., $f_s(x(0)) = \sum_{i=1}^{N} x_i(0)$. If W is a doubly-stochastic matrix,

$$f_s(W\theta(k)) = f_s(\theta(k)), \forall k \in \mathbb{N}. \tag{15}$$

According to the analysis in [5],

$$\sum_{k=0}^{\infty} \sum_{i=1}^{N} \theta_i(k) = 0 \tag{16}$$

is the necessary condition for an undirected network system which means that the input $\theta(k)$ is zero-sum in the timer shaft.

Remark 3: If W is doubly-stochastic, the condition of $\theta^-(k+m)$ can be relaxed as those in Remark 1. This relaxation will only change the convergence rate, but not the convergence.

The DTCA can be used in both directed and undirected systems while the method in [18] applies for only undirected systems. Therefore, in undirected systems, the convergence rate of the algorithms can be compared.

For the doubly-stochastic matrix under the DTCA, we have

$$\rho_{ess} = \max\{|\lambda_2|^2, |\lambda_n|^2\}.$$

Meanwhile, the convergence rate in [18] is

$$\rho_{ess} = \max\{\varphi^2, |\lambda_2|^2, |\lambda_n|^2\},$$

which means the convergence is not exact and its convergence rate is partially dependent on the attenuation coefficient of the added noises.

C. Inference Error Analysis

To keep the attackers from inferring the actual topology, we need to enlarge the approximation error:

$$\|\hat{W} - W\|_F. \tag{17}$$

Without losing generality, we suppose the information attackers received is $\mathcal{I}_0^k$. To maximize the utilization of the information, we decompose $\mathcal{I}_0^k$ into the data matrix $Y(k) = \mathcal{I}_0^{k-1}$ and the observation matrix $Z(k) = \mathcal{I}_1^k$. From Eq. (6), we have $Z(k) = WY(k) + \Theta(k)$ where $\Theta(k) = [\theta(0), \cdots, \theta(k-1)]$.

As we know from section II, the best estimation of the topology will be

$$\hat{W} = Z(k)Y(k)^\mathsf{T}(Y(k)Y(k)^\mathsf{T})^{-1},$$

leading to $\hat{W} - W = \Theta(k)Y(k)^\mathsf{T}(Y(k)Y(k)^\mathsf{T})^{-1}$.

The problem can be formulated as a constrained optimization problem targeting computing the optimal noise to maximize the inference error:

$$\begin{aligned} \text{P1:} \quad & \max_{\theta(k-1)} \|\Theta(k)Y(k)^{\mathsf{T}}(Y(k)Y(k)^{\mathsf{T}})^{-1}\|_F \\ & \text{s.t.} \quad \beta^-(k) \le \theta_i(k) \le \beta^+(k). \end{aligned} \tag{18}$$

Theorem 3.2: The optimal solution of Problem $\boldsymbol{P1}$ in (18) is equal to either $\beta^-(k)$ or $\beta^+(k)$.

Proof: Define

$$\Upsilon(k) = Y(k)^{\mathsf{T}}(Y(k)Y(k)^{\mathsf{T}})^{-1}$$

and split it into $\Upsilon_A(k)$ whose size is $(k-1)\times N$ and $\Upsilon_B(k)$ whose size is $1\times N$. $\Theta(k)$ is split into

$$\Theta_A(k) = [\theta(0), \cdots, \theta(k-2)],$$

and $\Theta_B(k) = [\theta(k-1)]$ for the same reason. Hence the error matrix can be written in a block matrix multiplication form:

$$\begin{aligned} & [\Theta_A(k) \mid \Theta_B(k)] \begin{bmatrix} \Upsilon_A(k) \\ \hline \Upsilon_B(k) \end{bmatrix} \\ = & \left[\begin{array}{c|c} \Theta_{A_1}(k) & \Theta_{B_1(k)} \\ \Theta_{A_2}(k) & \Theta_{B_2(k)} \\ \vdots & \vdots \\ \Theta_{A_N}(k) & \Theta_{B_N}(k) \end{array}\right] \begin{bmatrix} \Upsilon_A(k) \\ \hline \Upsilon_B(k) \end{bmatrix} \end{aligned} \tag{19}$$

The optimized objective function of (18) is thus broken down as

$$\begin{aligned} & \| [\Theta_A(k) \mid \Theta_B(k)] \begin{bmatrix} \Upsilon_A(k) \\ \hline \Upsilon_B(k) \end{bmatrix} \|_F \\ = & \sqrt{\sum_{i=1}^{N} \| \left(\Theta_{A_i}(k)\Upsilon_A(k) + \Theta_{B_i}(k)\Upsilon_B(k)\right) \|_F^2}. \end{aligned} \tag{20}$$

Hence, this optimization problem is decomposed into N independent sub-optimization problems, i.e., each row's F-norm optimization problems, which is given by

$$\begin{aligned} \text{P2:} \quad & \max_{\theta_i(k-1)} \| \left(\Theta_{A_i}(k)\Upsilon_A(k) + \Theta_{B_i}(k)\Upsilon_B(k)\right) \|_F \\ & \text{s.t.} \quad \beta^-(k) \le \theta_i(k) \le \beta^+(k), \end{aligned} \tag{21}$$

where the objective function can be expanded as

$$\sum_{j=1}^{N} \left(\Theta_{A_i}(k)\Upsilon_{A_j}(k) + \Theta_{B_i}(k)\Upsilon_{B_j}(k)\right)^2. \tag{22}$$

Since the historical information determines $\Theta_A(k)$ and $\Upsilon(k)$, the only variable of this formula is $\Theta_{B_i}(k)$, a 1×1 element. This formula is a convex quadratic function, resulting in a specific error bound and the property that the function is maximized when $\theta_i(k)$ equals one of the restrictions. Furthermore, as each row's F-norm is positive and restricted, the overall target of (18) is accomplished if and only if each independent optimization problem in (21) is maximized. □

To characterize the cost of the topology-preserving algorithm, we construct a function to measure the distance of one agent traveling as they seek the consensus:

$$g_i(X) = \sum_{k=1}^{T} |x_i(k) - x_i(k-1)|,$$

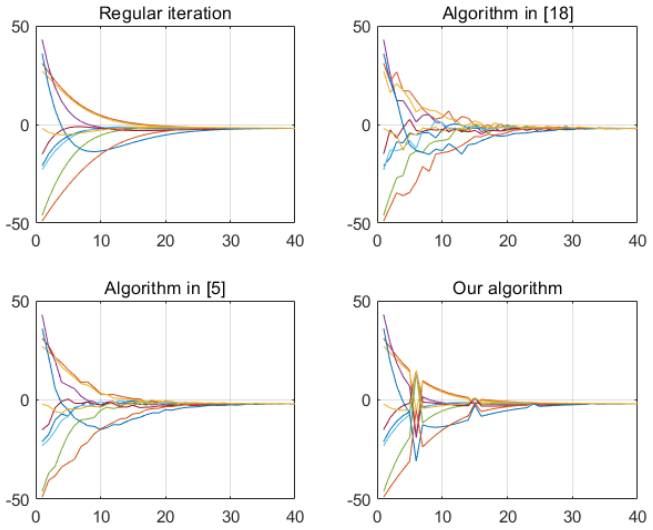

Fig. 1. The trajectories of a random system under different algorithms

resulting in the total distance of the agents be $\sum_{i\in V} g_i(X)$. This indicator shows the cost of the algorithm and can serve to reflect the cruising ability of mobile agents, which is crucial in a natural environment. Since inference errors and cost are both significant, tending to decrease one another, the trade-off between them should be made.

Besides, 2-norm $\rho = \sqrt{\lambda_{\max}(W^{\mathsf{T}}W)}$ can also be a meaningful indicator, as it calculates the maximum singular value of the matrix and represents the upper bound of the length change of the vector through the matrix. In our application scenario, the 2-norm of W can represent the convergence of the system. Consequently, the 2-norm errors can measure the correctness of the estimation to the system's convergence.

IV. SIMULATION

A. Simulation Setting

In this section, the simulation is used to present the effectiveness of the DTCA. A directed graph with ten agents is randomly constructed, reflecting the system's interaction topology. Assign all the agents with a random initial state and interaction links, and start the iteration under the DTCA and two privacy-preserving algorithms as in [18] and in [5]. During the iteration process, we widely set the main parameters in the three algorithms, namely α, σ_1 and σ_2, to evaluate the performance of the algorithms. To assess the balance between the cost and the performance, we take the length of the path as the standard of describing the cost and compare the data distribution under different algorithms.

B. Results and Analysis

Fig.1 reveals the trajectories of the system's convergence progress under different algorithms. All the topology-preserving algorithms will cause the trajectories to fluctuate around the ideal position, while the DTCA provides a more smooth curve as the system stabilizes.

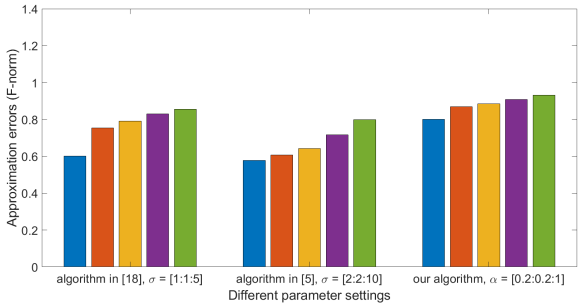

Fig. 2. The error comparisons under different parameter settings

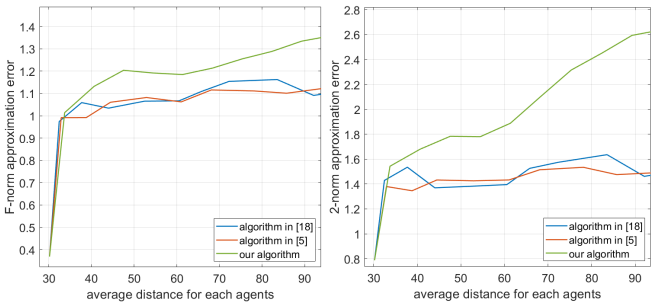

Fig. 3. F-norm errors and 2-norm errors in all time slots observation

Fig. 2 presents each algorithm's performance under five different parameter settings. The parameters play significant roles in the algorithms as they affect the trajectory of agents' motion and the estimation error of topology inference, illustrating the flexibility of the algorithms.

All Time Slots Observation. Without loss of generality, we will first verify the proposed algorithm's performance when the attackers will collect the information all the time.

Fig.3 illustrates the inferred topology's approximation error, where the horizontal axis shows the average distance for each agent to travel under this algorithm (i.e., $\sum_{i\in V} g_i(X)/N$), and the vertical axis shows the approximation error of the topology inference. As we can see, the overall trend of the estimations is the same, which is the higher the cost is, the more errors the inference will have. In particular, when there is no disturbance, the topology inference using Least Square Estimation can be highly accurate as all the lines start with 0. The most noticeable difference between applying our algorithm and applying others is that our algorithm has a better trade-off, having more significant inference errors while enjoying a minor cost than others. The objective of maximizing the inference errors while minimizing the paths' extra cost has been achieved.

Random Time Slots Observation. In practice, the observation time can be limited as the attackers will need time to verify their results and plan the subsequent attacks, and they may not be able to keep tracking the system from the beginning to the end. In this way, in the simulation part, we also randomly select some iteration slots as the information source of the attackers.

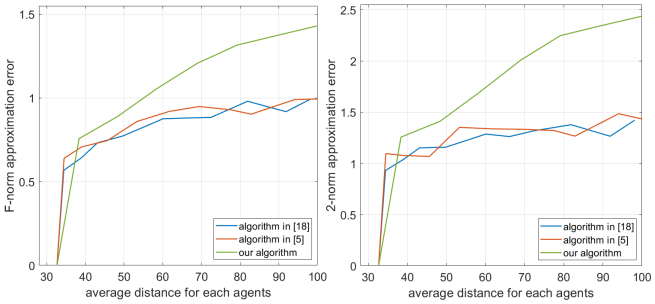

Fig. 4. F-norm errors and 2-norm errors in random time slots observation

Fig. 4 shows the approximation errors of the inferred topology with the time limits. The start point of the lines is not 0 because the limited time causes the loss of information. As we can see, our algorithm, together with others, is effective with time limits, but our algorithm outperforms others.

This conclusion can prove that our topology-preserving algorithm works well and performs better than algorithms in [18] and [5] in terms of protecting the topology information.

V. Conclusion

In this work, we investigate the distributed topology-preserving collaboration algorithm design against topology inference attack. Inspired by common noise-adding collaboration mechanisms, we propose a DTCA algorithm, where designed inputs are periodically added into the system while being eliminated with each other in the sense of guaranteeing the accuracy collaboration convergence. Specifically, the algorithm makes a trade-off between the crypticity of the topology and the precision of collaboration. Extensive simulations verify its effectiveness in achieving the balance between degenerating the attackers' topology inference ability and improving the cost, and shows its outperformance compared with other representative privacy-preserving collaboration algorithms. Future directions include extending the algorithm to a more general dynamical network and exploring the topology-preserving performance bound. The relations between cost and inference error need to be investigated further.

References

[1] A. D. Kshemkalyani and M. Singhal, *Distributed computing: Principles, algorithms, and systems*. Cambridge University Press, 2011.

[2] J. He, L. Duan, F. Hou, P. Cheng, and J. Chen, "Multiperiod scheduling for wireless sensor networks: A distributed consensus approach," *IEEE Transactions on Signal Processing*, vol. 63, no. 7, pp. 1651–1663, 2015.

[3] J. Alonso-Mora, S. Baker, and D. Rus, "Multi-robot formation control and object transport in dynamic environments via constrained optimization," *The International Journal of Robotics Research*, vol. 36, pp. 1000–1021, 2017.

[4] J. Wu, S. Yuan, S. Ji, G. Zhou, Y. Wang, and Z. Wang, "Multi-agent system design and evaluation for collaborative wireless sensor network in large structure health monitoring," *Expert Systems with Applications*, vol. 37, no. 3, pp. 2028–2036, 2010.

[5] J. He, L. Cai, P. Cheng, J. Pan, and L. Shi, "Consensus-based data-privacy preserving data aggregation," *IEEE Transactions on Automatic Control*, vol. 64, no. 12, pp. 5222–5229, 2019.

[6] Q. Zhu, "Topologies of agents interactions in knowledge intensive multi-agent systems for networked information services," *Advanced Engineering Informatics*, vol. 20, no. 1, pp. 31–45, 2006.

[7] Y. Emam, S. Mayya, G. Notomista, A. Bohannon, and M. Egerstedt, "Adaptive task allocation for heterogeneous multi-robot teams with evolving and unknown robot capabilities," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 7719–7725.

[8] D. Spinello and D. J. Stilwell, "Nonlinear estimation with state-dependent gaussian observation noise," *IEEE Transactions on Automatic Control*, vol. 55, no. 6, pp. 1358–1366, 2010.

[9] Q. Jiao, Y. Li, J. He, and L. Shi, "Topology inference for multi-agent cooperation under unmeasurable latent input," *arXiv preprint arXiv:2011.03964*, 2020.

[10] J. Li, J. He, Y. Li, and X. Guan, "Unpredictable trajectory design for mobile agents," in *American Control Conference (ACC)*, 2020, pp. 1471–1476.

[11] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on automatic control*, vol. 50, no. 5, pp. 655–661, 2005.

[12] W. Ni and D. Cheng, "Leader-following consensus of multi-agent systems under fixed and switching topologies," *Systems & control letters*, vol. 59, no. 3-4, pp. 209–217, 2010.

[13] G. Wen, G. Hu, W. Yu, J. Cao, and G. Chen, "Consensus tracking for higher-order multi-agent systems with switching directed topologies and occasionally missing control inputs," *Systems & Control Letters*, vol. 62, no. 12, pp. 1151–1158, 2013.

[14] A. Jadbabaie, Jie 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.

[15] T. Dong, X. Bu, and W. Hu, "Distributed differentially private average consensus for multi-agent networks by additive functional laplace noise," *Journal of the Franklin Institute*, vol. 357, no. 6, pp. 3565–3584, 2020.

[16] H. Sun, Z. Wang, J. Xu, and H. Zhang, "Exact consensus error for multi-agent systems with additive noises," *Journal of Systems Science and Complexity*, vol. 33, no. 3, pp. 640–651, 2020.

[17] V. Katewa, A. Chakrabortty, and V. Gupta, "Protecting privacy of topology in consensus networks," in *American Control Conference (ACC)*, 2015, pp. 2476–2481.

[18] Y. Mo and R. M. Murray, "Privacy preserving average consensus," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 753–765, 2016.

[19] C. Liu, J. He, S. Zhu, and C. Chen, "Dynamic topology inference via external observation for multi-robot formation control," in *2019 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, 2019, pp. 1–6.

[20] F. Bullo, *Lectures on Network Systems*. Kindle Direct Publishing, 2020.