A Secure High-dimension Consensus Mechanism against Adversaries

Xiaoyu Luo, Chengcheng Zhao, and Jianping He

Abstract— **This paper investigates the problem of high-dimension consensus against adversaries for multi-agent systems. Most of the existing works focus on each-dimension consensus separately. Differently, we introduce the incremental norm, i.e., the norm of local state vector's deviation for two consecutive iterations, to characterize the property of high-dimension consensus. Then, inspired by the existing two-hop-based secure one-dimension consensus, an adversary defense mechanism (ADM) against adversaries is proposed, where two-hop neighboring information is used to constrain the effect of adversaries on the performance of high-dimension consensus in a tolerable range. Specifically, if and only if the incremental norm of adversaries' state is bounded by their neighbors' maximum and minimum incremental norms, their normal neighboring agents use the transmitted state for local state update. We also show that under ADM, when adversaries do not collaborate/neighbor with each other, high-dimension consensus can be achieved and the final state is bounded by initial states of all normal agents. Finally, extensive simulations are conducted to demonstrate the effectiveness of the proposed defense mechanism.**

I. INTRODUCTION

In the past decades, high-dimension consensus has sparked considerable research interest in many areas including formation control [1], time synchronization [2], current sharing [3], [4]. The core of high-dimension consensus is to seek a distributed algorithm to drive the agents to reach an agreement state [5], where the state is a vector instead of a scalar. It is worth noting that adversaries are unavoidable and widely existing in the consensus progress [6]–[8]. It is because open and wireless networks are usually vulnerable to malicious attacks so that the transmitted state can be manipulated by the attacker to produce counterfeit data [9]. As a result, high-dimension consensus is unable to be achieved or the final state of consensus is manipulated by the attacker arbitrarily. Hence, it is practical and important to design effective defense mechanisms to ensure the security of high-dimension consensus.

Numerous efforts have been devoted to secure consensus problems in an adverse environment, which mainly focus on the fundamental theories and defense scheme design of one-dimension systems, such as the condition of reaching consensus under attacks, and resilient countermeasures design. We categorize the existing works into three aspects. The first one is based on the Mean-Subsequence Reduced (MSR) technique, where the number of tolerable adversaries is strictly limited by the network connectivity[1]. For example, Leblanc *et al.* proposed a resilient consensus scheme based on network robustness against malicious attacks, where each agent only uses local network information to eliminate extreme values instead of the whole network connectivity [10]. Dibaji *et al.* [11] developed a resilient algorithm where each agent ignores the states of its neighbors with the largest (smallest) position values to ensure secure consensus and the number of ignored largest (smallest) states is equal to the maximum tolerable number of attacks. The second and third ones are the multi-hop-based defense algorithm and the mobility-based scheme, respectively. For example, He *et al.* exploited two-hop neighboring information to constrain the used information bounded by the neighboring maximum state and minimum state to achieve time synchronization [2]. To break the rule on the number of tolerable malicious attacks and the assumption on attack collaborations, Zhao *et al.* designed a resilient consensus algorithm with mobile detectors to identify malicious agents, which is not limited by the network connectivity [12].

However, the above mechanisms and analysis are not applicable to high-dimension consensus. This is because in high-dimension consensus, the comparison between different high-dimension states is more complex than the scalar case. Moreover, states in different dimensions are coupled via complex local dynamics and local interactions, which renders the design and analysis hard to tract. Therefore, the design of a defense mechanism which guarantees high-dimension consensus against adversaries is challenging and prominent. Some attention has been paid to the security of high-dimension consensus against adversaries, which mainly generalizes the Mean-Subsequence Reduced (MSR) method to high-dimension cases [11]. Cui proposed a two-dimension MSR-based algorithm where the MSR method is applied for each dimension of state separately [13]. Yan *et al.* developed a resilient consensus algorithm where each normal agent sorts its received states for different dimensions, computed two ”middle points” based on the sorted value and moved its state toward these middle points such that the agreement of benign agents is reached in the convex hull of all nodes' initial state [14].

Note that if we need to sort each dimension of the received state separately, the computation complexity can be very large when the dimension of the local state grows.

Xiaoyu Luo and Jianping He are with the Department of Automation, Shanghai Jiao Tong University, Key Laboratory of System Control and Information Processing, Ministry of Education of China, and Shanghai Engineering Research Center of Intelligent Control and Management, Shanghai 200240, China. E-mail: xyl.sjtu@sjtu.edu.cn, jphe@sjtu.edu.cn.

Chengcheng Zhao is with The State Key Laboratory of Industrial Control Technology and Institute of Cyberspace Research, Zhejiang University, China, and the Department of Electrical and Computer Engineering, University of Victoria, BC, Canada. E-mail: zccsq90@gmail.com.

[1] Network connectivity is the minimum number of agents that can be cut from the network to make the rest network disconnected.

Moreover, these existing works still only work when the number of tolerable attacks is known by each normal agent and strictly limited by the network connectivity. Different from applying the MSR technique for each dimension, we introduce the incremental norm that refers to the state deviation norm at adjacent iterations to characterize the consensus property for high-dimension cases. Since the multi-hop information can be easily obtained by distributed networks, it is interesting to investigate how to use multi-hop information to limit the effect of adversaries on the high-dimension consensus. Then, inspired by the work in [2], we propose an adversary defense mechanism (ADM) to constrain the effect of adversaries in a tolerable range, where each normal agent only uses neighboring agents' states that have the incremental norm bounded by the minimum and maximum incremental norm of their neighbors. The main contributions are summarized as follows.

- We investigate the secure high-dimension consensus problem against adversaries, where adversaries do not collaborate or neighbor with each other and the number of adversaries is arbitrary.
- We design ADM to constrain the states used by normal agents from agents compromised by adversaries, which exploits the neighboring largest and smallest incremental norm based on state information from two consecutive iterations to achieve consensus.
- We obtain sufficient conditions to ensure the stability of the overall system composed by the normal agents under ADM. Simulation results show the effectiveness of ADM and the convergence rate is affected by initial states and self-feedback matrix while not influenced by the adversary.

The rest of this paper is organized as follows. Section II introduces the system model and adversary model briefly. In Section III, the ADM is proposed and theoretical analysis is given. Simulation results are presented in Section IV. Finally, we summarize our work in Section V.

Notations. Let $\mathbb{C}$, $\mathbb{Z}$, and $\mathbb{Z}^+$ denote the set of complex numbers, the set of non-negative integers, and the set of positive integers, respectively. For a vector $\mathbf{p} \in \mathbb{R}^n$, we let $\|\mathbf{p}\|$ denote its l_2-norm, $\|\mathbf{p}\|_\infty$ denote its l_∞-norm, and $\mathbf{p}^{\mathrm{T}}$ denote its transpose. Denote I_n and $\mathbf{1}_n$ as the n dimensional diagonal unit matrix and column vector with all elements 1, respectively. For matrix $P \in \mathbb{R}^{n\times n}$, we use $\sigma_{\max}(P)$ to denote its maximum singular value, $\mathrm{Ker}(P)$ to denote its nuclear space, $\rho(P)$ to denote its spectral radius, and $\mathrm{rank}(P)$ to denote its rank. The symbol $\otimes$ denotes the Kronecker product, $\mathrm{diag}(\cdot)$ denotes the diagonal matrix, $\min\{\cdot\}$ denotes the minimum value and $\max\{\cdot\}$ denotes the maximum value.

II. PROBLEM FORMULATION

A. Network Model

Consider a system composed by $n \in \mathbb{Z}^+$ autonomous agents with the identical dynamics and each agent has a unique ID number denoted by $i = 1, 2, \cdots, n$. A strongly undirected connected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is used to model the communication topology among n agents, where $\mathcal{V} = \{1, 2, \cdots, n\}$ is the set of agents and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set. The neighbor set of agent i is denoted by $\mathcal{N}_i = \{j | (i,j) \in \mathcal{E}, \forall j \in \mathcal{V}\}$ with its cardinality $d_i = |\mathcal{N}_i|$, where $(i,j) \in \mathcal{E}$ illustrates that agent i can receive information from agent j. The degree matrix is a diagonal matrix defined as $D = \mathrm{diag}\{D_{ij}\} \in \mathbb{R}^{n\times n}$ with $D_{ii} = d_i$ for all $i \in \mathcal{V}$. The weighted adjacency matrix is represented by $W = [w_{ij}] \in \mathbb{R}^{n\times n}$, with $w_{ij} = 1$ if $(i,j) \in \mathcal{E}$, and $w_{ij} = 0$ otherwise. We do not consider self-loops here, i.e., $w_{ii} = 0$. Then, the Laplacian matrix is written as $L = D - W$. We let $\mathcal{V}_s \subseteq \mathcal{V}$ represent the set of normal agents, which are not compromised by the adversary and behave normally, and $\mathcal{V}\backslash\mathcal{V}_s = \{i \in \mathcal{V} : i \notin \mathcal{V}_s\}$ denotes the set of agents compromised by the adversaries.

B. System Dynamic Model

Each agent has a discrete time-invariant linear system model and the dynamics of the i-th agent for $\forall i \in \mathcal{V}$ are

$$\mathbf{x}_i(k+1) = A\mathbf{x}_i(k) + B\mathbf{u}_i(k), \tag{1}$$

where $A \in \mathbb{R}^{m\times m}$ and $B \in \mathbb{R}^{m\times p}$ are constant system matrices, $\mathbf{x}_i(k) \in \mathbb{R}^m$ and $\mathbf{u}_i(k) \in \mathbb{R}^p$ denote the state and the control input of agent i at iteration k with $m, p \in \mathbb{Z}^+$. We consider a widely used distributed controller [15], i.e.,

$$\mathbf{u}_i(k) = K\mathbf{x}_i(k) + P \sum_{j\in\mathcal{N}_i} (\mathbf{x}_j(k) - \mathbf{x}_i(k)), \tag{2}$$

where $K \in \mathbb{R}^{p\times m}$ and $P \in \mathbb{R}^{p\times m}$ are control gain matrices corresponding to the local state and the deviation between the local state and the neighboring one, respectively. Thus, for $\forall i \in \mathcal{V}$, we have

$$\mathbf{x}_i(k+1) = (\tilde{A} - d_i\tilde{B})\mathbf{x}_i(k) + \tilde{B} \sum_{j\in\mathcal{N}_i} \mathbf{x}_j(k), \tag{3}$$

where $\tilde{A} = A + BK \in \mathbb{R}^{m\times m}$ and $\tilde{B} = BP \in \mathbb{R}^{m\times m}$. Let $\mathbf{x}(k) = [\mathbf{x}_1(k)^{\mathrm{T}}, \cdots, \mathbf{x}_n(k)^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{mn}$ be the global state of the system, whose dynamics can be written in a compact form

$$\mathbf{x}(k+1) = M\mathbf{x}(k), \tag{4}$$

where $M = (I_n \otimes \tilde{A} - L \otimes \tilde{B}) \in \mathbb{R}^{mn\times mn}$ is the system matrix. Here, we provide the following assumption to guarantee the system stability for the tractable analysis.

Assumption 1: The dynamic system (A, B) is controllable. There exist the control gain K and P such that the global system in (4) is asymptotically stable.

C. Adversary Model

We consider that the objective of the adversary is to compromise agents and manipulate their states arbitrarily such that high-dimension consensus process is destroyed. The capability of the adversary is described below.

- The adversary can only read the transmitted information and cannot manipulate it, which is easily realized by information encryption and authentication [16].

- Any two adversaries never neighbor/collaborate with each other. This is reasonable when the network is sparse and the number of adversaries is smaller than the size of the network [2].

Let $\mathbf{x}_j^+(k)$ denote the state of agent j suffering from the adversary at iteration k, which is described by

$$\mathbf{x}_j^+(k) = \mathbf{x}_j(k) + \boldsymbol{\theta}_j(k), k > 0, j \in \mathcal{V} \backslash \mathcal{V}_s, \tag{5}$$

where $\boldsymbol{\theta}_j(k) \in \mathbb{R}^m \neq \mathbf{0}_m$, $\exists k \in \mathbb{Z}^+$ denotes the false data injected by the adversary arbitrarily. Based on the adversary model, the original controller is unable to defend against the adversary. We will use agents compromised by the adversary and adversaries interchangeably.

D. Problem Formulation

Different from one-dimension consensus under the adversary, it is hard to design the defense mechanism to ensure high-dimension consensus under the adversary. On the one hand, the local system matrices A and B make consensus analysis more complicated and also render different dimensions of the local state coupled. On the other hand, how to compare different high-dimension states from different neighbors is more challenging than the one-dimension case. The idea of exploiting multi-hop information is beneficial for constraining the effect of adversaries. Specifically, we aim to solve the following problems.

- The first one is how to use multi-hop information to ensure secure high-dimension consensus against adversaries, i.e.,

$$\lim_{k \to \infty} \|\mathbf{x}_i(k) - \bar{\mathbf{c}}\| = 0, \forall i \in \mathcal{V}_s, \tag{6}$$

  where $\bar{\mathbf{c}}$ is called final state.
- The second one is how to analyze the performance of the designed defense mechanism, i.e., the stability of the whole system composed by normal agents, the effect of the adversary on the convergence rate and the deviation of the final state from the final state for the case without any adversary.

III. Main Results

In this section, we first provide the necessary condition for achieving high-dimension consensus under the adversary. Then, a critical definition, i.e., incremental norm, is introduced to set the tolerable secure range. Later, ADM is proposed to limit the effect of adversaries in a tolerable range to ensure high-dimension consensus. Finally, the performance of the proposed mechanism is thoroughly analyzed.

A. Necessary Condition Analysis

Suppose that only agent j is compromised by the adversary and it injects false data $\boldsymbol{\theta}_j(k)$ to the true state. Under (3) with the considered adversary, we obtain

$$\mathbf{x}_i(k+1) = \begin{cases} (\tilde{A} - d_i\tilde{B})\mathbf{x}_i(k) \\ \quad + \tilde{B} \sum_{\ell \in \mathcal{N}_i} \mathbf{x}_\ell(k) + \tilde{B}\boldsymbol{\theta}_j(k), j \in \mathcal{N}_i \cup i, \\ (\tilde{A} - d_i\tilde{B})\mathbf{x}_i(k) + \tilde{B} \sum_{\ell \in \mathcal{N}_i} \mathbf{x}_\ell(k), j \notin \mathcal{N}_i \cup i. \end{cases} \tag{7}$$

We provide one lemma below to show the necessary condition for ensuring high-dimension consensus by referring to one-dimension case [2].

Lemma 1: Under (7), if (6) is achieved, then we have

$$\lim_{k \to \infty} \boldsymbol{\theta}_j(k) = \mathbf{0}_m. \tag{8}$$

Remark 1: Lemma 1 means that the necessary condition for achieving high-dimension consensus is that the injected false data vector goes to zero as $k \to \infty$. It should be pointed out that the necessary condition for the case with only one adversary in Lemma 1 can be easily extended to the case with more than one adversary.

B. Important Definition

Before introducing ADM, we define the incremental norm to describe the performance of high-dimension consensus, which is used to set the tolerable secure range.

Definition 1: (Incremental Norm) The incremental norm of agent j is defined as the norm of local states' deviation at two consecutive iterations, i.e.,

$$\delta_j(k) = \|\mathbf{x}_j(k) - \mathbf{x}_j(k-1)\|, \forall j \in \mathcal{V}, k \geq 1. \tag{9}$$

Let $\delta_{j_m}(k) = \min\{\delta_l(k)|\forall l \in \mathcal{N}_j \cup j\}$ be the minimum incremental norm among the neighbors of agent $j \in \mathcal{V}$ and itself at iteration k, where j and $j_m(k)$ are ID number. Similarly, let $\delta_{j_M}(k) = \max\{\delta_l(k)|\forall l \in \mathcal{N}_j \cup j\}$ be the maximum incremental norm. Then, we denote the maximum incremental norm information set as $\mathcal{I}_{j_M}(k) = \{j_M(k), \mathbf{x}_{j_M}(k), \mathbf{x}_{j_M}(k-1)\}$ at iteration $k \geq 1$, which includes ID number and state information from agent $j_M(k)$. Similarly, we denote the minimum incremental norm information set as $\mathcal{I}_{j_m}(k) = \{j_m(k), \mathbf{x}_{j_m}(k), \mathbf{x}_{j_m}(k-1)\}$ at iteration $k \geq 1$.

C. Defense Mechanism Design

The critical idea of ADM is to utilize the two-hop information to constrain the effect of adversaries in the tolerable secure range bounded by the neighbors' minimum and maximum incremental norm. Taking the update of agent i as an example, the information set $\mathcal{I}_j^i(k)$ that its neighboring agent j sends to its own agent i is defined as

$$\begin{cases} \mathcal{I}_j^i(0) = \{\mathcal{I}_j(0)\}, \quad k = 1 \\ \mathcal{I}_j^i(1) = \{\mathcal{I}_j(1), j_m(0), \mathbf{x}_{j_m}(0), j_M(0), \mathbf{x}_{j_M}(0)\}, \quad k = 2 \\ \mathcal{I}_j^i(k-1) = \{\mathcal{I}_j(k-1), \mathcal{I}_{j_m}(k-2), \mathcal{I}_{j_M}(k-2)\}, k \geq 3 \end{cases}$$

where $\mathcal{I}_j(0) = \{j, \mathbf{x}_j(0)\}$, $\mathcal{I}_j(k-1) = \{j, \mathbf{x}_j(k-1), \mathbf{x}_j(k-2)\}$ is the state from itself at iteration $k \geq 2$.

We divide the iteration process into two stages based on whether the two-hop neighboring information is available, shown in Algorithm 1.

In stage 1, agent i only receives its neighbors' information set $\mathcal{I}_j^i(0)$, stores the information set $\mathcal{I}_j(0)$ and updates its local state based on update rule (3) at iteration $k = 1$.

In stage 2, there are two cases, i.e., $k = 2$ and $k \geqslant 3$. **For case 1**, i.e., iteration $k = 2$, agent i receives its neighboring agent j's information set $\mathcal{I}_j^i(1)$ and stores the information set $\mathcal{I}_j(1)$. If $j_m(1) = j$ ($j_M(1) = j$), agent i

checks whether $\mathbf{x}_{j_m}(0)$ ($\mathbf{x}_{j_M}(0)$) is legal by comparing it with that in the stored information $\mathcal{I}_j(0)$. If it is not legal, the information will not be used. Otherwise, agent i will check whether $\mathbf{x}_j(1)$ is legal by calculating $\delta_j(1)$, $\|\mathbf{x}_{j_m}(0)\|$ and $\|\mathbf{x}_{j_M}(0)\|$. If $\delta_j(1)$ satisfies the following secure range

$$\delta_j(1) \in [\|\mathbf{x}_{j_m}(0)\|, \|\mathbf{x}_{j_M}(0)\|], \tag{10}$$

agent i will render $\mathbf{x}_j(1)$ as legal information that can be used for update at iteration $k = 2$. Otherwise, $\mathbf{x}_j(1)$ will be discarded.

For case 2, i.e., iteration $k \geq 3$, agent i receives its neighboring agent j's information set $\mathcal{I}_j^i(k-1)$ at iteration k and stores the information set $\mathcal{I}_j(k-1)$. If $j_m(k-1) = j$ ($j_M(k-1) = j$), agent i checks whether $\mathbf{x}_{j_m}(k-2)$ ($\mathbf{x}_{j_M}(k-2)$) is legal by comparing it with that in the stored information $\mathcal{I}_j(k-1)$. If it is not legal, the information will not be used. Otherwise, agent i will check whether $\mathbf{x}_j(k-1)$ is legal by computing $\delta_j(k-1)$, $\delta_{j_m}(k-2)$ and $\delta_{j_M}(k-2)$. If $\delta_j(k-1)$ satisfies the following tolerable secure range

$$\delta_j(k-1) \in [\delta_{j_m}(k-2), \delta_{j_M}(k-2)], \tag{11}$$

agent i will view $\mathbf{x}_j(k-1)$ as legal information that can be used for update at iteration k. The set of neighbors with legal information is denoted by $\mathcal{N}_i^l(k-1)$ and then the update rule can be written as

$$\mathbf{x}_i(k) = (\tilde{A} - d_i\tilde{B})\mathbf{x}_i(k-1) + \tilde{B} \sum_{j \in \mathcal{N}_i^l(k-1)} \mathbf{x}_j(k-1), \tag{12}$$

where $\mathbf{x}_j(k-1) = \mathbf{x}_j^{+}(k-1)$ if $j \in \mathcal{V} \backslash \mathcal{V}_s$. Once $\delta_j(k-1)$ violates (11), we will not use $\mathbf{x}_j(k-1)$.

Remark 2: Note that the storage space required by the mechanism will increase with the dimension of the state, which can be viewed as the extra cost to enhance the security of the system. Since only two-hop information is required to be stored, which is the least number of hops when we consider using multiple-hop information, our proposed mechanism is efficient. When the dimension of the system state is moderate, the proposed mechanism still works.

D. Performance Analysis

Here, we provide the sufficient condition to ensure high-dimension consensus under ADM by analyzing the variation of the incremental norm with iterations. For simplicity, the maximum singular value of $\tilde{A} - d_i\tilde{B}$ for $i \in \mathcal{V}$ is denoted by α_i, and β is the maximum singular value of $\tilde{B}$.

Theorem 1: Consider system (4) with adversaries (5). Under ADM, if we have $\alpha_i < 1$, $\beta < \frac{1-\alpha_i}{d_i}$ for $\forall i \in \mathcal{V}$, then the high-dimension consensus is achieved, i.e., (6) holds.

Proof: The proof process is divided into two parts. The first part is to show that each agent can be stable and the second part is to ensure the final state of each agent is the same by contradiction. Combining the two parts, the system will achieve high-dimension consensus exponentially. Due to the limited space, we omit the proof. ■

Remark 3: Theorem 1 shows the sufficient condition to ensure high-dimension consensus under ADM if and only if adversaries never neighbor/collaborate with each other. And the result of Theorem 1 can be easily extended to the case that multiple adversaries send random states to their neighbors since the incremental norm is still bounded under multiple adversaries.

Algorithm 1: ADM

```
Input: $A$, $B$, $K$, $P$, $L$ and prescribed error threshold $\varepsilon$
Output: $\mathbf{x}_i(T), \forall i \in \mathcal{V}_s$
Initialization: Each agent $i \in \mathcal{V}_s$ initializes its information
set $\mathcal{I}_j^i(0)$ and $\mathbf{x}_i(0)$;
Iteration:
for each normal agent $i$ do
    Stage 1 ($k = 1$):
    Receive its neighbors' information set $\mathcal{I}_j^i(0)$, update its
    state at iteration $k = 1$ based on (3), store $\mathbf{x}_i(1)$ and
    $\mathcal{I}_j^i(0)$;
    Stage 2 ($k \geq 2$):
    if $k = 2$ then
        Receive its neighboring information set $\mathcal{I}_j^i(1)$, store
        $\mathcal{I}_j(1)$, calculate $\delta_j(1)$, $\|\mathbf{x}_{j_m}(0)\|$, and $\|\mathbf{x}_{j_M}(0)\|$;
        for each neighboring agent $j$ do
            if $j_m(1) = j$ ( $j_M(1) = j$), $\mathbf{x}_{j_m}(0)$ ($\mathbf{x}_{j_M}(0)$) is
            legal by comparing it with the stored
            information $\mathcal{I}_j(0)$ , or $j_m(1) \neq j$ and
            $j_M(1) \neq j$. then
                if (10) is satisfied then
                    $\mathbf{x}_j(1)$ is legal, store $j(1)$ and $\mathbf{x}_j(1)$;
                else
                    $\mathbf{x}_j(1)$ is illegal, discard $\mathbf{x}_j(1)$;
        Update its state at iteration $k = 2$ based on (12), and
        update $\mathcal{I}_i^j(1)$;
    else if $k \geq 3$ then
        Receive its neighboring information set $\mathcal{I}_j^i(k-1)$,
        store $\mathcal{I}_j(k-1)$, calculate $\delta_j(k-1)$, $\delta_{j_m}(k-2)$ and
        $\delta_{j_M}(k-2)$;
        for each neighboring agent $j$, $k' = k-1$ do
            if $j_m(k') = j$ ( $j_M(k') = j$), $\mathbf{x}_{j_m}(k')$
            ($\mathbf{x}_{j_M}(k')$) is legal by comparing it with the
            stored information $\mathcal{I}_j(k')$ , or $j_m(k') \neq j$ and
            $j_M(k') \neq j$. then
                if (11) is satisfied then
                    $\mathbf{x}_j(k')$ is legal, store $j(k')$ and $\mathbf{x}_j(k')$;
                else
                    $\mathbf{x}_j(k')$ is illegal, discard $\mathbf{x}_j(k')$;
        Update its state based on (12), store $\mathbf{x}_i(k)$, and
        update $\mathcal{I}_j^i(k-1)$;
If $\|\mathbf{x}_i(k) - \mathbf{x}_j(k)\| < \varepsilon, \forall i, j \in \mathcal{V}_s, j \in \mathcal{N}_i$, then let $T = k$
and the iteration terminates.
```

E. Discussions

In this part, we discuss the final state and the convergence rate under the adversary and no adversary. In addition, the relevant affecting factors are analyzed to further research.

1) Final state deviation: Without loss of generality, the overall system will achieve the pre-set high-dimension consensus without the adversary according to gain matrices K, initial states, and interaction topology $\mathcal{G}$. Under ADM and the adversary (5), we constrain the incremental norm of each agent within the tolerable secure range. Satisfying the necessary condition of achieving consensus does not mean that the final state of the whole system composed of all normal agents is not disturbed and invariant to that without the adversary. Hence, it is critical and meaningful to analyze what elements will affect the final state and design a more efficient defense mechanism under the adversary. From Theorem 1, we know the final state of system (4) under (5) is bounded by initial states of all normal agents through ADM. Besides the above elements under no adversary, the system parameter P is potential to influence the final state due to its connection to the adversary. However, how these elements affect the final state, and whether there exist other elements influencing the final state are worth further consideration in the future from the perspective of theoretical analysis.

2) Convergence rate: Consider system (4) without any adversary (5), we obtain

$$\begin{aligned}\|\mathbf{x}(k)-\mathbf{x}^*\| =&\|M^k\mathbf{x}(0)-\mathbf{x}^*\| = \|M^k(\mathbf{x}(0)-\mathbf{x}^*)\| \\ \le&\|M\|^k\|\mathbf{x}(0)-\mathbf{x}^*\|.\end{aligned}$$

According to the definition of convergence rate r, we have

$$r = \sup \lim_{k\to\infty} (\frac{\|\mathbf{x}(k)-\mathbf{x}^*\|}{\|\mathbf{x}(0)-\mathbf{x}^*\|})^{\frac{1}{k}} = \sigma_{\max}(M). \tag{13}$$

It shows that the convergence rate is determined by the maximum singular value of system matrix M when they are no adversaries. Once there exists the adversary, we will choose whether to use the state from the adversary for update through ADM. It can be viewed as the case with a switching network topology and the analysis of convergence rate is complex due to the complicated system dynamics.

IV. Simulation Results

In this section, we evaluate the performance of ADM for the cases without/with adversaries.

We consider a formation control scenario with six agents connected. The second order double-integrator dynamics of each agent are given by [11], where the relative position and velocity are two-dimension state variables. The system matrix and input matrix of each agent are set as follows:

$$A = \begin{bmatrix} 0.5 & T \\ 0 & 0.5 \end{bmatrix}, B = \begin{bmatrix} \frac{T^2}{2} \\ T \end{bmatrix},$$

where the sampling period is set as $T = 0.1$. It can be easily validated that the considered system is controllable. The set of initial states and the set of system parameters are shown in Table I and Table II, respectively. We set initial state 2 and system parameter 2 as a set of baseline conditions. Then, we have $\alpha_i < 1$ for $\forall i \in \mathcal{V}$ and $\beta = 0.0751$, which satisfy the condition in Theorem 1.

We use the consensus error $\epsilon(k)$ to quantify the performance, which is the deviation from the final state without adversaries, i.e.,

$$\epsilon(k) = \sum_{i\in\mathcal{V}_s} \|\mathbf{x}_i(k) - \bar{\mathbf{c}}\|. \tag{14}$$

TABLE I
THE SETTING OF INITIAL STATES

Initial States	$\mathbf{x}(0)$
1	$[-7,-3,-2,-2,0,1,0,-1,2,3,6,2]^{\mathrm{T}}$
2	$[5,-3,2,-2,10,1,70,-1,4,3,6,2]^{\mathrm{T}}$
3	$[5,-3,2,-2,10,1,10,-1,4,3,6,2]^{\mathrm{T}}$

TABLE II
THE SETTING OF SYSTEM PARAMETERS

System Parameters	K and P
1	$K = [-0.25,-0.05], P = [1,1]$
2	$K = [-0.25,-0.05], P = [0.02,0.75]$
3	$K = [1,1], P = [0.02,0.75]$

A. The Performance of ADM without the adversary

In this part, we investigate how the proposed mechanism performs when there are no adversaries. Based on the baseline conditions, Fig. 1 shows that the convergence of the two-dimension state under ADM is effectively achieved, and the final states of relative position and velocity approach to zero shown in Fig. 1(a) and 1(b).

Then, we investigate the effect of different initial states, and system parameters K and P on the performance of ADM. The initial states are set as shown in Table I and other baseline conditions remain. As we can see from Fig. 2(a), different initial state has a certain effect on the convergence rate. Then, we consider different system parameters set in Table II and remain other baseline conditions. Fig. 2(b) demonstrates that the convergence rate is related to the system parameter. Moreover, the influence of self-feedback gain matrix K is greater than that of gain matrix P on the convergence rate.

B. The Defense Performance of ADM with Adversaries

Here, we investigate how ADM performs under adversaries for the same scenario settings. Consider that agent 3 is manipulated by the adversary which injects false data $\boldsymbol{\theta}_3(k)$ at iteration $k \in \mathbb{Z}^+$ generated randomly from the interval $[0, 10]$ for each dimension of state. Based on a set of baseline conditions, Fig. 3 demonstrates that the two-dimension consensus is still achieved by all normal agents under ADM, which shows the effectiveness of ADM against adversaries. Note that the state of agent 3 still fluctuates and the magnitude of the change in state gradually decays, which accord with the result in Lemma 1 since the state is compromised by the adversary and its update is constrained by the rule of ADM.

Likewise, we investigate the effect of different initial states and system parameters K and P on the defense and convergence performance of ADM under the adversary. As we can see in Fig. 4(a), in the presence of the adversary, the convergence rate is almost the same as that in the absence

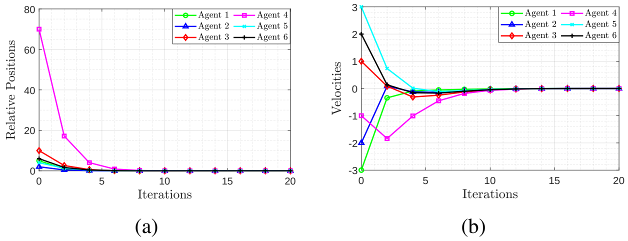

Fig. 1. Performance of ADM without adversaries. (a) Relative Position. (b) Velocities.

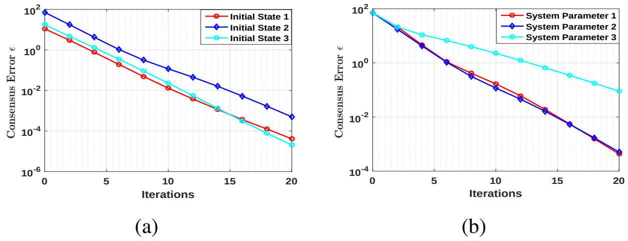

Fig. 2. The performance of ADM without adversaries. (a) Different initial states. (b) Different system parameters.

of adversaries on the whole. From Fig. 4(b), we observe that there exists a certain small fluctuation under the condition of system parameter 1 or 2 with the adversary than that without adversaries, which illustrates the efficient defense performance of ADM when there exist adversaries.

V. Conclusions

In this paper, we investigated the problem of high-dimension consensus against adversaries for multi-agent systems. Instead of focusing on each-dimension consensus separately, we introduced the incremental norm that refers to the norm of state deviation at any two consecutive iterations, to characterize the convergence of high-dimension consensus. Then considering non-cooperative/non-neighboring adversaries with an incubation period, ADM is designed by using two-hop information to limit the used information from neighboring agents within a tolerable range such that it prevents the false information from disrupting the consensus process. When there are no adversaries, the performance of ADM is the same as that under the traditional consensus protocol. In addition, we showed that secure consensus can always be achieved by ADM under adversaries, where the final state is limited by the initial states of all agents in the system. For future works, we will further investigate how the specific adversary model affects the performance of high-dimension consensus, how to further decrease the extra storage and computation cost, and the design of the defense mechanism against collaborated/neighboring adversaries.

References

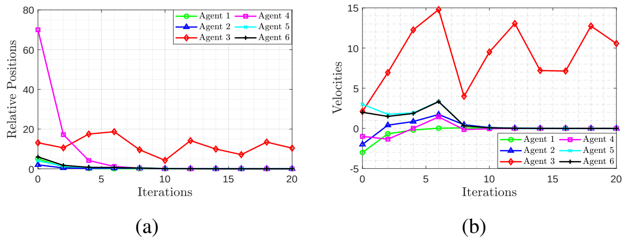

Fig. 3. Performance of ADM against adversaries. (a) Relative Position. (b) Velocities.

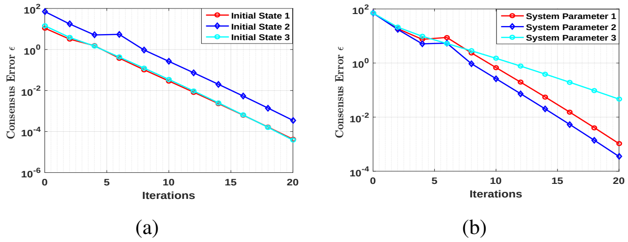

Fig. 4. Defense performance of ADM under adversaries. (a) Different initial states. (b) Different system parameter.

[1] W. Wang, J. Huang, C. Wen, and H. Fan, "Distributed adaptive control for consensus tracking with application to formation control of nonholonomic mobile robots," *Automatica*, vol. 50, no. 4, pp. 1254–1263, 2014.

[2] J. He, P. Cheng, L. Shi, and J. Chen, "Sats: Secure average-consensus-based time synchronization in wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 61, no. 24, pp. 6387–6400, 2013.

[3] C. Jiang, H. Du, and G. Wen, "Current sharing control for parallel dc-dc buck converters based on consensus theory," in *IEEE ICCA*, 2017, pp. 536–540.

[4] X. Luo, J. He, and S. Zhu, "On-board supercapacitors cooperative charging algorithm: Stability analysis and weight optimization," in *IEEE ACC*, 2020, pp. 4975–4980.

[5] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[6] T. Li and J. F. Zhang, "Consensus conditions of multi-agent systems with time-varying topologies and stochastic communication noises," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2043–2057, 2010.

[7] ——, "Mean square average-consensus under measurement noises and fixed topologies: Necessary and sufficient conditions," *Automatica*, vol. 45, no. 8, pp. 1929–1936, 2009.

[8] Y. Xu, M. Fang, Z. G. Wu, Y. J. Pan, M. Chadli, and T. Huang, "Input-based event-triggering consensus of multiagent systems under denial-of-service attacks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 4, pp. 1455–1464, 2020.

[9] X. M. Li, Q. Zhou, P. Li, H. Li, and R. Lu, "Event-triggered consensus control for multi-agent systems against false data-injection attacks," *IEEE Transactions on Cybernetics*, vol. 50, no. 5, pp. 1856–1866, 2020.

[10] H. J. Leblanc, H. Zhang, X. Koutsoukos, and S. Sundaram, "Resilient asymptotic consensus in robust networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 4, pp. 766–781, 2013.

[11] S. M. Dibaji and H. Ishii, "Resilient consensus of double-integrator multi-agent systems, acc 2014 portland," in *American Control Conference*, 2014, pp. 5239–5144.

[12] C. Zhao, J. He, and J. Chen, "Resilient consensus with mobile detectors against malicious attacks," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 4, no. 1, pp. 60–69, 2018.

[13] L. Cui, "On the two-dimensional resilient consensus," in *IEEE ICCSNT*, 2019, pp. 451–455.

[14] J. Yan, X. Li, Y. Mo, and C. Wen, "Resilient multi-dimensional consensus and optimization in adversarial environment," *arXiv preprint arXiv:2001.00937*, 2020.

[15] G. Zhang, J. Xu, J. Zeng, J. Xi, and W. Tang, "Consensus of high-order discrete-time linear networked multi-agent systems with switching topology and time delays," *Transactions of the Institute of Measurement and Control*, vol. 39, no. 8, pp. 1182–1194, 2016.

[16] R. Lu, X. Lin, H. Zhu, X. Liang, and X. Shen, "Becan: A bandwidth-efficient cooperative authentication scheme for filtering injected false data in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 1, pp. 32–43, 2012.