2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

On the trade-offs between accuracy, privacy, and resilience in average consensus algorithms

Guilherme Ramos, André M. H. Teixeira, and Sérgio Pequito

Abstract— **There can be none. In this paper, we address the problem of a set of discrete-time networked agents reaching average consensus privately and resiliently in the presence of a subset of attacked agents. Existing approaches to the problem rely on trade-offs between accuracy, privacy, and resilience, sacrificing one for the others. We show that a separation-like principle for privacy-preserving and resilient discrete-time average consensus is possible. Specifically, we propose a scheme that combines strategies from resilient average consensus and private average consensus, which yields both desired properties. The proposed scheme has polynomial time-complexity on the number of agents and the maximum number of attacked agents. In other words, each agent that is not under attack is able to detect and discard the values of the attacked agents, reaching the average consensus of non-attacked agents while keeping each agent's initial state private. Finally, we demonstrate the effectiveness of the proposed method with numerical results.**

I. Introduction

The consensus problem is a prominent circumstance in numerous networked multi-agent systems. Therefore, this problem emerges in a multitude of applications. For instance, it is a central problem in optimization [1], [2], motion coordination tasks [3], [4], rendezvous problems [5]–[7], resource allocation in computer networks [8], and healthcare and medical applications [9]–[11].

The problem challenge is to design an iterative algorithm allowing a set of agents to agree upon a value via local interactions in a communication network. That is, the solution of a consensus problem is the design of a distributed procedure, where each agent has low computational power and its communication with other agents is limited by the network topology [12].

Due to its ubiquity in diverse applications, there is the need to ensure beyond-accuracy properties in consensus methods, such as *resilience* and *privacy*. Specifically, a resilient consensus algorithm should enable each agent to effectively and efficiently identify neighbors disseminating erroneous information. Thus, by screening out incorrect state values, the non-attacked agents seek to converge to a common value, which would ideally be the correct one. Additionally, the agents' state should only be accessible to the agent itself, i.e., it should be private. In this context, the consensus protocol should guarantee the privacy of all agents so that they may safeguard their initial values as confidential, but still reach the desired consensus. Both resilient and private discrete-time consensus protocols are active research subjects as we overview next.

Resilient average consensus: The work in [13], [14] tackles the general problem of reaching discrete-time resilient consensus in the presence of faulty agents. The authors devised a general strategy that requires, as input, a consensus algorithm and the resilience parameter f. The trustworthy agents identify the attacked ones and rectify the consensus value by discarding erroneous information, reaching accurately the consensus value, but having additional computation and communication costs. The attackers may even determine and choose the set of agents to tamper with and to achieve a desired goal [15].

In [16], the authors present a resilient leader-follower consensus to arbitrary reference values, where an agent ignores a number of the largest and the smallest received values. This work guarantees a consensus value in the convex hull of initial agents' states, scarifying the accuracy for resilience. Following the same line, [17] creates a resilient consensus method for time-varying networks of dynamic agents.

The authors of [18] propose a reputation-based switching mechanism to select the network topology that prevents attacked agents from communicating, where the non-attacked agents converge to a value close to the original steady-state. Again, it is achieved by sacrificing accuracy for resiliency.

Private average consensus: Privacy also plays a key role in consensus methods [19]–[21]. The typical approaches aiming to achieve privacy in consensus methods can be classified into the following classes: homomorphic encryption-based (*HE-based*); differential privacy-based (*DP-based*); and observability-based (*O-based*.)

Briefly, HE-based average consensus methods require costly computations and communications, yielding a potentially prohibitive limitation in real applications with restricted computation and communication power [22]–[25]. DP-based approaches aim to attain privacy by introducing uncertainty via noise addition to shared information [26]–[31]. In this case, the consensus is guaranteed in expected value, which may not be suitable for accurate decision-making. Moreover, noise generation is commonly performed via a pseudo-random generator relying on the initial seed. Thus, we need to use a secret seed or expensive true random number generator devices [32].

The O-based strategies focus on curious agents trying to recover other agents' states via the dynamics evolution. In other words, when agents estimate states considered to be private. Hence, *observability* (in dynamical systems) renders necessary and sufficient conditions to get an estimator able

G. Ramos (guilherme.ramos@tecnico.ulisboa.pt) is with Dept. of Computer Science and Engineering, Instituto Superior Técnico, University of Lisbon, Portugal and Instituto de Telecomunicações, 1049-001 Lisbon, Portugal. A. M. H. Teixeira and S. Pequito are with the Division of Systems and Control, Department of Information Technology, Uppsala University, Sweden.

to retrieve agents' initial states, meant to be private [20], [33], [34]. A possible way to attain the O-based privacy is by doing a network augmentation [34], where each agent augments its state, $x_i(k) \in \mathbb{R}$, with a minimum additional states (in this case 3), $\tilde{x}_i(k) \in \mathbb{R}^4$, to distribute its initial value and achieve privacy.

Main contributions: It has been established that the use of differential privacy is not compatible with resilience guarantees in average consensus [35], [36]. Surprisingly, hereafter, we show it is possible to devise a novel average consensus method with accuracy, resilience and privacy guarantees, without trade-offs among these properties.

Specifically, we show how to combine two independent protocols for resilient average consensus and privacy in discrete-time consensus. To the best of the authors knowledge, it is the first time that such an approach is proposed. Thus, it is possible to attain a sort of privacy and resilience "separation-principle", by merging two approaches that individually ensure each of the properties – with the necessary changes.

Remarkably, we show that few additional computational resources are necessary at each agent, and each agent broadcasts more information (a vector state instead of a scalar one.) Hence, it entails a computational efficient protocol, contrasting with encryption methods that may not be viable in several engineering contexts. Further, the proposed scheme does not sacrifice accuracy, in contrast with previously proposed approaches to ensure privacy and resilient.

II. Preliminaries and Notation

We denote by $\mathbb{N}$ the set of positive integers and by $\mathbb{N}_0$ the set of non-negative ones. Further, we denote the set with the first n positive integers as $[n]$, where $[n] = \{1, \dots, n\}$. Next, we revise graph theory concepts [37]. A *digraph* $\mathcal{G}$ is a pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of $n > 1$ *nodes*, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is a set of *edges*. Edges are ordered pairs representing an accessibility relationship between nodes. If $u, v \in \mathcal{V}$ and $(u, v) \in \mathcal{E}$, then node v directly accesses information shared by node u. We also call the digraph by *network* and the nodes by *agents*. A digraph is a *complete digraph* when each agent can directly access information of all the other agents. Let $v \in \mathcal{V}$, we define the *neighbors* of v as $\mathcal{N}_v = \{v\} \cup \{u : (u, v) \in \mathcal{E}\}$, and they are the set of agents from which v can directly access information. Given a digraph $\mathcal{G}$, we define a *path* as a sequence of agents $(v_1, v_2, \dots v_k)$ with $(v_i, v_{i+1}) \in \mathcal{E}$, for all $i \in [k-1]$. A digraph $\mathcal{G}$ is *strongly connected* if for any agents $u, v \in \mathcal{V}$ there is a path from u to v. A helpful way to describe a digraph is by its adjacency matrix, $A \in \mathbb{R}^{n \times n}$. For a digraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $A_{u,v} = 1$ if $(u, v) \in \mathcal{E}$, and $A_{u,v} = 0$, otherwise. A *subgraph*, $\mathcal{H} = (\mathcal{V}', \mathcal{E}')$, of $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a digraph with $\mathcal{V}' \subset \mathcal{V}$ and $\mathcal{E}' \subset \mathcal{E}$. Let $\mathcal{A} \subset \mathcal{V}$, to ease notation, we define $\mathcal{H} = \mathcal{G} \setminus \mathcal{A}$ as a subgraph of $\mathcal{G}$, with $\mathcal{H} = (\mathcal{V} \setminus \mathcal{A}, \mathcal{E}')$, and $\mathcal{E}' = \{(u, v) \in \mathcal{E} : u, v \notin \mathcal{A}\}$.

Given a square matrix $A \in \mathbb{R}^{n \times n}$ and a set $\mathcal{U} \subset [n]$, we define $minor(A, \mathcal{U})$ as the square matrix $A' \in \mathbb{R}^{(n-|\mathcal{U}|) \times (n-|\mathcal{U}|)}$ that consists of dropping the rows and columns of A with indices in $\mathcal{U}$. Additionally, we define $\widehat{minor}(A, \mathcal{U})$ as the matrix obtain from $minor(A, \mathcal{U})$ when we normalize each of its rows to sum up to 1. We denote by $\mathbf{I}_n$ the $n \times n$ identity matrix and by $\mathbf{0}_{n \times m}$ the $n \times m$ zero matrix. We denote by $span(A)$ the row space of the matrix A. We use vectors as column vectors, we denote by $\mathbf{1}_n$ the n-dimensional vector of ones, and we denote by e_N^i the i-th canonical N-dimensional column vector.

From this point on, we use the discrete-time variable $k \in \mathbb{N}_0$. Given a sequence of values $\{s^{(k)}\}_{k \in \mathbb{N}_0}$ or a function $f : \mathbb{R} \to \mathbb{R}$, if the sequence or the function has limit, i.e., $\lim_{k \to \infty} s^{(k)} = a$ or $\lim_{k \to \infty} f(k) = b$, then we write compactly that $s^{(k)} \to a$ or $f(k) \to b$. For a set $\mathcal{S} \subset \mathbb{N}$, we define its subsets by $\wp(\mathcal{S})$. For example, if $\mathcal{S} = \{1, 2, 3\}$, then $\wp(\mathcal{S}) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$. We denote the subsets of $\mathcal{S}$ of size $i \leq |\mathcal{S}|$ by $\wp(\mathcal{S}, i) = \{w \in \wp(\mathcal{S}) : |w| = i\}$. For $v \in \mathbb{R}^n$, we denote its i-th entry by $v[i]$ for $i \in [n]$. When useful, we index vector positions by keys (as to define dictionaries in computer science), e.g. for $v \in \mathbb{R}^8$ and for the above $\wp(\mathcal{S})$, we may index the entries of v by the elements of $\wp(\mathcal{S})$, e.g. $v[\emptyset]$ or $v[\{2, 3\}]$.

Finally, we use the standard universal (i.e., $\forall$) and existential quantifier (i.e., $\exists$.) Also, we use the "exists one and only one" quantifier $\exists!_x.\varphi(x) \equiv \exists_x.\varphi(x) \wedge \forall_{y \neq x}.\neg\varphi(y)$.

III. Problem Statement

In this work, we are interested in designing an average consensus method with accuracy, resilience and privacy guarantees, without trade-offs among these properties.

Towards attaining the resilience aspect, we consider the case where an attacker (malicious entity) has a particular goal (not just preventing consensus convergence.) To this end, an attacker wants to deviate the consensus of a network to a specific value a that may be harmful to the system.

Let the unknown set of attacked agents of network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be denoted by $\mathcal{A}$, with $\mathcal{A} \subset \mathcal{V}$. The resilience goal is to create an average consensus protocol of the form $x^{(k+1)} = Ax^{(k)}$, with $x^{(0)} = x_0$, and with $x^{(k)}$ a vector that collects the agents' states, as a result of a design mechanism that consists of creating an algorithm that receives a network of agents, $\mathcal{G}$, a dynamics matrix A, and the maximum number of attacked agents, f, and allows the non-attacked agents to identify the attacked agents, and subsequently, ignore their values in the final average consensus.

To attain privacy, we aim to develop an average consensus method where each agent cannot recover the initial state of any other agent. That is, the goal is to keep the agents' initial states private and to prevent the values shared during the consensus method execution from leaking information that allows an agent to recover the initial state of another agent.

Assumption 1: Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a digraph, $\mathcal{A}$ a set of attacked agents and $x^{(N)}$ be the consensus value of agents $\mathcal{V} \setminus \mathcal{A}$ resulting from applying the consensus method C for $N \in \mathbb{N}$ time steps. Let $\varepsilon > 0$ denote the precision utilized for computations. For all $v \in \mathcal{V} \setminus \mathcal{A}$ and for all $u \in \mathcal{V}$ it follows that $\lim_{k \to \infty} \left| \left(\widehat{minor}(A, \{u\}) \right)^k minor(x^{(0)}, \{u\}) - x_v^{(k)} \right| > \varepsilon$, where A is the consensus dynamics matrix, and $x^{(0)}$ is the initial state vector of all the agents. $\diamond$

Simply speaking, Assumption 1 only requires that no agent has a state equal to the consensus of the subgraph excluding

that agent, which is required for the well-posedeness of the resilience property.

Assumption 2: Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a digraph and $\mathcal{A} \subset \mathcal{V}$ a set of attacked agents. Each subgraph $\mathcal{H}$ of $\mathcal{G}$ with $\mathcal{H} = \mathcal{G} \setminus \mathcal{V}'$, where $|\mathcal{V}'| \leq |\mathcal{A}|$ is a network that reaches consensus. $\diamond$

In contrast, Assumption 2 is a more general assumption. Essentially, we require the network without attacked agents to be connected in order to reach consensus.

Overall, we can formally state the problem we aim to address as follows.

P Given N agents with a communication digraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a maximum number of attacked agents f, *is there* a distributed algorithm of the form

– Dynamics –

$$x_i^{(k+1)} = \mathbf{m}(\mathcal{G}, x_i^{(k)}), \tag{1}$$

such that the following holds:

– Specifications –

Privacy: Agent i cannot recover the initial state, $x_j^{(0)}$, of any agent, $j \neq i$. (2a)

Resilient average consensus: If an unknown set of agents are attacked, $\mathcal{A}$, and $|\mathcal{A}| \leq f$ then, for $i \in \mathcal{V} \setminus \mathcal{A}$

$$\lim_{k\to\infty} x_i^{(k)} = \frac{1}{|\mathcal{V} \setminus \mathcal{A}|} \sum_{j \in \mathcal{V}\setminus\mathcal{A}} x_j^{(0)}. \tag{2b}$$

IV. Design privacy-preserving and resilient deterministic discrete-time average consensus

In this section, we build up on two existing consensus protocols (one ensuring resilience and the other privacy) to achieve the desired objective. In Section IV-A, we revisit the resilient consensus method proposed in [13], [14]. In Section IV-B, we overview an average consensus method with privacy guarantees [34]. Finally, in Section IV-C, we show how the two previous approaches can be combined with some necessary changes to build an average consensus method with both resilience and privacy guarantees.

In particular, we are going to consider an augmentation of the state space,

$$\tilde{x}_i^{(0)} = \mathbf{g}(x_i^{(0)}) \;\text{ and }\; \tilde{x}_i^{(k+1)} = \mathbf{h}(W_i)\tilde{x}_i^{(k)}, \tag{3}$$

where we are going to ensure both resilience and privacy, Sections IV-A and IV-B, respectively. This strategy is then followed by a projection on the original state space dimension (i.e., the agents' state.) Thus, fulfilling the resiliency and privacy specifications.

A. Resilient average consensus

We consider attacked agents broadcasting values to drive the final consensus to a desired state, where their states converge accordingly, i.e., $\lim_{k\to\infty} x_i^{(k)} = \mathbf{1}_{d_i+1}\left(\frac{1}{|\mathcal{V}\setminus\mathcal{A}|}\sum_{j\in\mathcal{V}\setminus\mathcal{A}} \mathbf{p}(\tilde{x}_j^{(0)})\right)$.

Intuitively, each agent scalar state is going to be augmented by a vector with $|\wp([f])|$ entries, where f is the maximum number of allowed attacked agents. Each entry corresponds to the scalar state of the agent where the consensus protocol consists on a normalization of the interactions of the agents except those in the set $\wp([f])[j]$, with $j = 0, \ldots, |\wp([f])|$. In particular, the first entry corresponds to the case where the consensus protocol runs without attacked agents. The remaining ones correspond to different combinations of possible attacked agents, and the last entries correspond to the worst case where f agents are discarded in the consensus protocol. At each time, each agent verifies if there is an entry corresponding to the smallest possible set of agents, $\mathcal{V}' \subset \mathcal{V}$, with a value different from the entries corresponding to sets with size less or equal to $\mathcal{V}'$.

The first result concerns the identification of attacked agents and the convergence to the correct consensus value.

Theorem 1 ([13]): Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a digraph with n agents, C be a consensus algorithm, and $\mathcal{A} = \{v_1, \ldots, v_s\} \subset \mathcal{V}$ be a set of s agents attacked by a malicious entity which makes these agents share values converging to a ($x_{v_i}^{(k)} \to a$, for $i \in [s]$.) Let $\varepsilon > 0$ be the precision utilized to do comparisons between values. In this scenario, Algorithm 2 of [13] with robustness $f \geq s$ identifies, after a number of time steps, the attacked agents in $\mathcal{A}$, and the agents $v \in \mathcal{V}\setminus\mathcal{A}$ converge to the consensus value of $\mathcal{G} \setminus \mathcal{A}$ from the input consensus algorithm C. $\circ$

By Theorem 1, any non-attacked agent identifies and corrects its state. The next results shows that the detection cannot yield false positives.

Proposition 1 ([13]): Consider the digraph of n agents $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a consensus algorithm C. By using Algorithm 2 of [13] with $f \geq s = |\mathcal{A}|$ if, after some time steps, an agent $v \in \mathcal{V}$ finds $s \leq f$ attacked agents, then there exist s attacked agents. $\circ$

Finally, the next result details the computational complexity of the method proposed in Algorithm 2 of [13].

Proposition 2 ([13]): If the computational complexity of the consensus algorithm C to run for $T \in \mathbb{N}$ time steps is $C(T)$, then Algorithm 2 of [13] has time complexity of $\mathcal{O}(n^f C(T))$. $\circ$

The results above demonstrate that the resilience assurances remain unaffected by the choice of the privacy consensus protocol, provided that the protocol is deterministic.

B. Private average consensus

We consider an O-based approach to achieve privacy in average consensus, as defined in property (2b) of **P**. Specifically, we consider the state augmentation approach using the architecture in Fig. 1, and described in Algorithm 1 [34].

The intuition behind this approach is to augment each agent's dynamics with a local network ($\tilde{W}_i$), and distribute the agent's initial state across the augmented network nodes ($\tilde{x}_i^{(0)}$). By doing so with an appropriate augmentation and initial state distribution, the original initial state cannot be observed by other agents. From now on, we denote $\tilde{W}$ by A^{P} and $\tilde{x}$ by x^{P} to emphasize privacy.

The augmentation in Algorithm 1 implements the function $\mathbf{h}$ of (3) and, for a suitable function $\mathbf{g}$, (3) with the dynamics

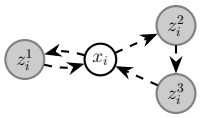


Fig. 1: The original agent is agent x_i (in white) and the remaining agents and edges define the augmentation (gray agents and dashed edges.)

Algorithm 1 Privacy dynamics matrix A^{P} for consensus

1: **input**: dynamics matrix $A \in \mathbb{R}^{N\times N}$
2: **output**: dynamics matrix $A^{\mathsf{P}} \in \mathbb{R}^{4N\times 4N}$
3: **set** $A \in \mathbb{R}^{N\times N}$ as the adjacency matrix of $\mathcal{G}$
4: **fill** the entries of A^{P} with zeros
5: **set** ▷ Copy the matrix A to the first n rows and n columns

$$A^{\mathsf{P}}_{ij} = A_{ij}, \forall i,j \in \{1,\dots,N\} \text{ and } i \neq j$$

6: **for** $i = 1,\dots,N$ ▷ Set additional entries values

$$A^{\mathsf{P}}_{i,N+3i-2} = 1, \quad A^{\mathsf{P}}_{N+3i-2,i} = 2, \quad A^{\mathsf{P}}_{i,N+3i-1} = 1,$$
$$A^{\mathsf{P}}_{N+3i-1,N+3i-0} = 1, \quad A^{\mathsf{P}}_{N+3i-0,i} = 1$$

7: **normalize** the rows of A^{P} dividing by their sum

matrix from Algorithm 1 achieves consensus with privacy, as stated in the next theorem.

***Theorem** 2 ([34]):* Let $A \in \mathbb{R}^{N\times N}$ be a matrix such that $\mathcal{G}(A)$ is a strongly connected graph. Then, the system $\tilde{x}^{\mathsf{P}(k+1)} = A^{\mathsf{P}}\tilde{x}^{\mathsf{P}(k)}$ and $y[k] = C\tilde{x}^{\mathsf{P}(k)}$, where the matrix $A^{\mathsf{P}} \in \mathbb{R}^{4N\times 4N}$ results from Algorithm 1 and $C = \left[\begin{array}{c|c} \mathbf{I}_N & \mathbf{0}_{N\times 3N} \\ \mathbf{0}_{3\times N} & H_i \end{array}\right]$, where $[\mathbf{I}_N \mid \mathbf{0}_{N\times 3N}]\tilde{x}^{\mathsf{P}(k)} = x_i^{(k)}$ and $[\mathbf{0}_{3\times N} \mid H_i]\tilde{x}^{\mathsf{P}(k)} = z_i^{(k)}$ (i.e., the internal states of agent i.) So, w.l.o.g., assume the states are permuted so that $H_i = [\mathbf{I}_3 \mid \mathbf{0}_{3\times(N-3)}]$ and $C = [\mathbf{I}_{N+3} \mid \mathbf{0}_{(N+3)\times(3N-3)}]$. Then, the next properties hold:

1) it is not observable; and
2) $\left(e^j_{4N} + \sum_{k=N+3j-2}^{N+3j} e^k_{4N}\right) \notin span\left(P^{\lambda}_{O(A^{\mathsf{P}},C)}\right)$, for all $j \neq i$, where $P^{\lambda}_{O(A,C)} \equiv \left[\begin{smallmatrix} C \\ \lambda\mathbf{I}_{N^2} - A \end{smallmatrix}\right]$, $\forall \lambda \in \mathbb{C}$.

Moreover, if $\left(\frac{1}{(v_L)_j}e^j_{4N} + \sum_{k=N+3j-2}^{N+3j}\frac{1}{(v_L)_k}e^k_{4N}\right) \notin span\left(P^{\lambda}_{O(A^{\mathsf{P}},C)}\right)$, for all $j \neq i$, where v_L is the left-eigenvector of A^{P} obtained with Algorithm 1, associated with the eigenvalue 1, then the initial state of agent j (distributed among its augmented states agents) is private as per property (2a). Since i is an arbitrary node, we conclude that j is private w.r.t. all other nodes. ∘

In the next subsection, we build up on the two previous approaches to design an average consensus method that is both resilience and privacy while preserving accuracy.

C. Private and resilient average consensus

Intuitively, the aforementioned approaches that deal with resilience and privacy independently can be combined with some adaptations, such that the resilience property can be achieved on top of the privacy property.

First, we are going to consider the augmented network for the resilient case presented in Section IV-A. Without loss of generality, when an agent is attacked, it means the original agent. Moreover, if any of the virtual agents is attacked, then it would be seen as only the original agent is under attack. Therefore, we consider the augmented state where we use subgraphs of removing agents together with their virtual agents. Finally, we need to guarantee that regardless of the subgraphs considered for the augmented vector dynamics, which entries discard the possibly attacked agents, the consensus protocol is private. Additionally, the privacy is ensured under the assumption that the network without attacked agents has strictly more than 2 agents.

In Algorithm 2, we detail how to design the initial setup, before executing the consensus protocol, resorting to Algorithm 1. Observe that Algorithm 2 implements the functions

Algorithm 2 Private and resilient average consensus method initialization

1: **input**: dynamics matrix $A \in \mathbb{R}^{N\times N}$, with N agents, and initial states $x^{(0)} \in \mathbb{R}^N$
2: **output**: $\left\{A^{\mathcal{F}[i]}\right\}_{i=1}^{|\mathcal{F}|}$ and initial states $\left\{\left(\tilde{x}^{\mathcal{F}[i]}\right)^{(0)}\right\}_{i=1}^{|\mathcal{F}|}$
3: **set** $\mathcal{F} = \bigcup_{i=0}^{f}\wp(\mathcal{V},i)$ as the set of all subsets of agents with sizes from 0 to f
4: **for** $i = 1,\dots,f$ **do**
5: $A^{\mathcal{F}[i]} = \widehat{A^{\mathsf{P}}}$, where A^{P} is the output of Algorithm 1 with input $\widehat{minor}(A,\mathcal{F}[i])$
6: **compute** the left-eigenvector v^0 of $A^{\mathcal{F}[i]}$ associated with the eigenvalue 1
7: **for** $u \in (\mathcal{V}\setminus\mathcal{F}[i])\setminus\mathcal{A}$ **do**
8: ▷ distribute the initial condition of each agent across its augmented states (setting as 0 for the original state), the distribution can be tailored by each agent, and scaling it to make the new average of initial states equal to the original one
9: **select** $\alpha_u, \beta_u, \gamma_u \neq 0$ such that $\alpha_u + \beta_u + \gamma_u \neq 0$
10: **set**

$$\left[\left(\tilde{x}^{\mathcal{F}[i]}_u\right)^{(0)} \left(\tilde{x}^{\mathcal{F}[i]}_{N+3u-2}\right)^{(0)} \left(\tilde{x}^{\mathcal{F}[i]}_{N+3u-1}\right)^{(0)} \left(\tilde{x}^{\mathcal{F}[i]}_{N+3u}\right)^{(0)}\right] = \frac{4\left(x^{\mathcal{F}[i]}_u\right)^{(0)}}{\alpha_u+\beta_u+\gamma_u}\left[0\ \alpha_u\ \beta_u\ \gamma_u\right]$$

11: **end for**
12: **for** $u \in (\mathcal{V}\setminus\mathcal{F}[i])\setminus\mathcal{A}$ **do**
13: **for** $l \in \{u, N+3u-2, N+3u-1, N+3u\}$ **do**
14: **set** $\left(\tilde{x}^{\mathcal{F}[i]}_l\right)^{(0)} = \frac{\left(x^{\mathcal{F}[i]}_l\right)^{(0)}}{4(N-|\mathcal{F}[i]|)v^0_l}$.
15: **end for**
16: **end for**
17: **end for**

$\mathbf{g}$ and $\mathbf{p}$ (steps 10–14) of (3), whereas $\mathbf{h}$ is implemented by Algorithm 1. Now that we have detailed how the initial setup of the consensus method must be designed, we have the ingredients to present the main algorithm (Algorithm 3.)

Next, we show that Algorithm 3 reaches resilient average consensus whenever $|\mathcal{A}| \leq f$, i.e., the property (2b) of $\mathbf{P}$.

***Theorem** 3:* Let $\mathcal{G} = (\mathcal{V},\mathcal{E})$ be a digraph with n agents, and $\mathcal{A} = \{v_1,\dots,v_k\} \subset \mathcal{V}$ be a set of k agents attacked by a malicious entity which makes these agents share values converging to a. Let $\varepsilon > 0$ be the precision utilized to do comparisons between values. In this scenario, Algorithm 3 with robustness $f \geq k$ identifies, after a number of time steps, the attacked agents in $\mathcal{A}$, and the agents $v \in \mathcal{V}\setminus\mathcal{A}$ converge to the average of their initial states. ∘

Proof: From Theorem 1 and Proposition 1, the non-attacked agents correctly identify and correct their states. Also, the non-attacked agents follow the dynamics of the

Algorithm 3 Private and resilient average consensus method

input: dynamics matrix $A \in \mathbb{R}^{N\times N}$, corresponding to the network $\mathcal{G} = (\mathcal{V} = [N], \mathcal{E})$ of N agents, initial states $x^{(0)} \in \mathbb{R}^N$, number of iterations T, the resilience parameter $f \in \mathbb{N}_0$ and the precision parameter $\varepsilon > 0$
output: final consensus $x^{(T)} \in \mathbb{R}^N$
set $\mathcal{F} = \bigcup_{i=0}^{f} \wp(\mathcal{V}, i)$ as the set of all subsets of agents with sizes from 0 to f
compute $\left\{A^{\mathcal{F}[i]}\right\}_{i=1}^{|\mathcal{F}|}$ and initial states $\left\{\left(\tilde{x}^{\mathcal{F}[i]}\right)^{(0)}\right\}_{i=1}^{|\mathcal{F}|}$ with Algorithm 2
for $u \in [4N] \setminus \mathcal{A}$ **do**
▷ the attacked agents may not follow the protocol
  for $i = 1, \ldots, |\mathcal{F}|$ **do**
    set $c_u^{(0)}[\mathcal{F}[i]] = \left(\tilde{x}_u^{\mathcal{F}[i]}\right)^{(0)}$
  end for
end for
compute $\tilde{x}^{(T)}$ as the output of Algorithm 2 of [13] using the dynamics of $\left\{A^{\mathcal{F}[i]}\right\}_{i=1}^{|\mathcal{F}|}$, the computed vector c, the set of subsets of agents $\mathcal{F}$, the number of iterations T and precision parameter ε
set $x^{(T)}$ as the first N values of $\tilde{x}^{(T)}$

subgraph without attacked agents, which by design (Algorithm 2) converges to $\sum_{k=1}^{|v^0|} v_k^0 \left(\tilde{x}_k^{\mathcal{F}[i]}\right)^{(0)}$, where $\mathcal{F}[i] = \mathcal{A}$ is the correct attacked agents' set and v^0 is the left-eigenvector of $A^{\mathcal{F}[i]}$ associated with eigenvalue 1. Now, let $\mathcal{N} = \mathcal{V} \setminus \mathcal{A}$ and note that $|\mathcal{V} \setminus \mathcal{A}| = N - |\mathcal{F}[i]|$, we have that

$$\sum_{k=1}^{|v^0|} v_k^0 \left(\tilde{x}_k^{\mathcal{F}[i]}\right)^{(0)} = \sum_{k=1}^{|v^0|} v_k^0 \frac{\left(x_k^{\mathcal{F}[i]}\right)^{(0)}}{4\left(N - |\mathcal{F}[i]|\right) v_k^0} = \frac{\sum_{k=1}^{|v^0|} \left(x_k^{\mathcal{F}[i]}\right)^{(0)}}{4|\mathcal{N}|}$$

$= \sum_{j\in\mathcal{N}} 4x_j^{(0)}/(4|\mathcal{N}|) = \sum_{j\in\mathcal{N}} x_j^{(0)}/|\mathcal{N}|$. ■

Finally, in the next result, we show that Algorithm 3 achieves privacy, entailing the property (2a) of **P**.

Theorem 4: Under the setting of Theorem 3, the non-attacked agents' initial states, agents in $\mathcal{V} \setminus \mathcal{A}$, are kept private, if more than 2 agents are non-attacked. ○

Proof: By considering the construction for the resilient component in Algorithm 1 for each of the sugraphs, and initial setup of Algorithm 2, then we can ensure privacy in each subgraph, under the assumption that there are more than 2 non-attacked agents. Thus, we use Theorem 2 in each subgraph of agents excluding at most f agents. ■

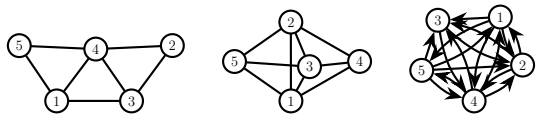
Proposition 3: Algorithm 3 has polynomial time complexity of $\mathcal{O}(\max\{N^{f+2}, N^{f+1}T\})$. ○

Proof: The proof follows from using Proposition 2 and replacing the term $C(T)$ by the cost of an agent: (i) computing the initial steps 3–10, with cost $\mathcal{O}(N^f \times N^2) = \mathcal{O}(N^{f+2})$ to compute the dynamics matrix of each subset in $\mathcal{F}$; (ii) running step 12 with cost $\mathcal{O}(N^f \times NT) = \mathcal{O}(N^{f+1}T)$ to update T times each entry of its vector state. Thus, the total cost is $\mathcal{O}(\max\{N^{f+2}, N^{f+1}T\})$. ■

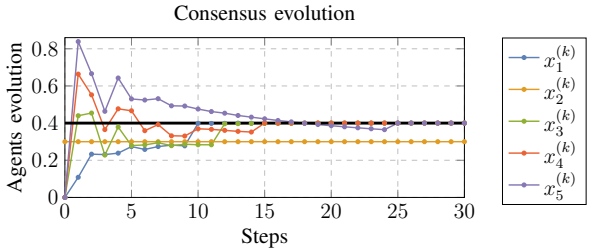
V. Illustrative Examples

In this section, we illustrate the proposed average consensus method that has resilience and privacy guarantees with examples. In the plots, we omit the augmented agents evolution to ease the relevant agents' evolution (which initial states are 0 by the augmentation design.)

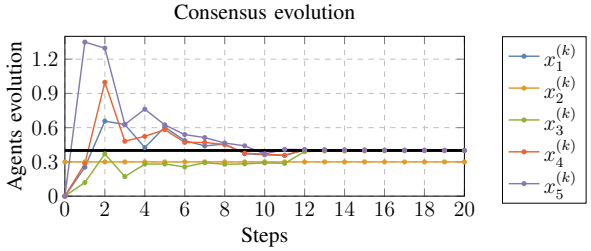
Consider five agents, $\mathcal{V} = [5]$, with initial states $x_0 = x^{(0)} = [0.1\ 0.3\ 0.35\ 0.6\ 0.55]$. In the first example, we consider the network of agents $\mathcal{G}_1$, depicted in Fig. 2 (a), and the set of attacked agents $\mathcal{A}_1 = \{2\}$. In this case, the attacked agent behaves as a stubborn agents that always shares the value 0.3. In Fig. 2 (d), we present the agent's state evolution using Algorithm 3. We can see that the non-attacked agents, agents in $\mathcal{V} \setminus \mathcal{A}_1$, identify correctly the attacked agent and reach the average of the non-attacked agents' initial state.



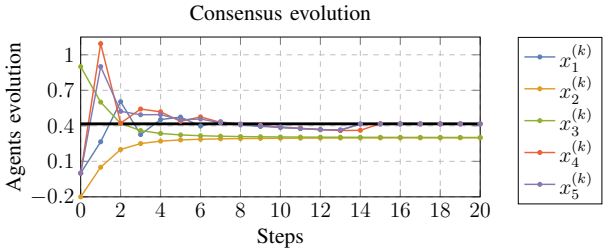
(a) Network $\mathcal{G}_1$. (b) Network $\mathcal{G}_2$. (c) Network $\mathcal{G}_3$.



(d) Agents' consensus evolution with network $\mathcal{G}_1$, initial state x_0, and set of attacked agents $\mathcal{A}_1$, using Algorithm 3.



(e) Agents' consensus evolution with network $\mathcal{G}_3$, initial state x_0, and set of attacked agents $\mathcal{A}_3$, using Algorithm 3.



(f) Agents' consensus evolution with network $\mathcal{G}_2$, initial state x_0, and set of attacked agents $\mathcal{A}_2$, using Algorithm 3.

Fig. 2: Numerical results.

Next, consider the same setup as the first example, but now the network of agents in $\mathcal{G}_2$, depicted in Fig. 2 (b), and the set of attacked agents is $\mathcal{A}_2 = \{2, 3\}$. This time, agent 2 shares values according to the function $f_2(k) = 0.3 - \frac{0.5}{1+k^2}$ and agent 3 according to the function $f_3(k) = 0.3 + \frac{0.6}{1+k^2}$. In Fig. 2 (e), we show the agent's state evolution using Algorithm 3. We can see that the non-attacked agents, agents in $\mathcal{V} \setminus \mathcal{A}_2$, identify correctly the attacked agent and reach the average of the non-attacked agents' initial state.

Finally, consider the same setup as the first example, but now the network of agents is directed, $\mathcal{G}_3$, depicted in Fig. 2 (c), and the set of attacked agents is $\mathcal{A}_3 = \mathcal{A}_1 = \{2\}$. In Fig. 2 (f), we present the agent's state evolution using Algorithm 3. We can see that the non-attacked agents, agents in $\mathcal{V} \setminus \mathcal{A}_3$, identify correctly the attacked agent and reach the

average of the non-attacked agents' initial states.

VI. Conclusions & Future Research

In this paper, we addressed the problem of a set of network agents reaching resilient and private average consensus in the presence of a subset of attacked agents. The results we proposed demonstrate that the resilience assurances remain unaffected by the choice of the privacy consensus protocol, provided that the protocol is deterministic. Additionally, we proposed a privacy protocol that relies on state-space augmentation and with compromising the resilience specification. The method has polynomial time complexity on the number of agents and the maximum number of attacked agents. The proposed method enables each non-attacked agent to detect and discard the values of the attacked agents, reaching the average consensus of non-attacked agents while keeping each agent initial state private.

Future research includes exploring if it is possible to consider resilience protocols with lower computational complexity which can be intertwined with privacy protocols.

References

[1] J. Tsitsiklis, D. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Transactions on Automatic Control*, vol. 31, no. 9, pp. 803–812, September 1986.

[2] B. Johansson, T. Keviczky, M. Johansson, and K. H. Johansson, "Subgradient methods and consensus algorithms for solving convex optimization problems," in *47th IEEE Conference on Decision and Control (CDC)*, Dec 2008, pp. 4185–4190.

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

[4] A. Alessandretti and A. P. Aguiar, "An optimization-based cooperative path-following framework for multiple robotic vehicles," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 2, pp. 1002–1014, 2019.

[5] J. Cortés, S. Martínez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions," *IEEE Transactions on Automatic Control*, vol. 51, no. 8, pp. 1289–1298, 2006.

[6] R. Ribeiro, D. Silvestre, and C. Silvestre, "A rendezvous algorithm for multi-agent systems in disconnected network topologies," in *2020 28th Mediterranean Conference on Control and Automation (MED)*, 2020, pp. 592–597.

[7] ——, "Decentralized control for multi-agent missions based on flocking rules," in *CONTROLO 2020*, J. A. Gonçalves, M. Braz-César, and J. P. Coelho, Eds. Cham: Springer International Publishing, 2021, pp. 445–454.

[8] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, 2007.

[9] B. Camajori Tedeschini, S. Savazzi, R. Stoklasa, L. Barbieri, I. Stathopoulos, M. Nicoli, and L. Serio, "Decentralized federated learning for healthcare networks: A case study on tumor segmentation," *IEEE Access*, vol. 10, pp. 8693–8708, 2022.

[10] C. Pedroso, Y. U. de Moraes, M. Nogueira, and A. Santos, "Relational consensus-based cooperative task allocation management for IIoT-Health networks," in *2021 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2021, pp. 579–585.

[11] J. Brogan, I. Baskaran, and N. Ramachandran, "Authenticating health activity data using distributed ledger technologies," *Computational and Structural Biotechnology Journal*, vol. 16, pp. 257–266, 2018.

[12] W. Ren, R. Beard, and E. Atkins, "A survey of consensus problems in multi-agent coordination," in *Proceedings of the 2005, American Control Conference, 2005.*, 2005, pp. 1859–1864 vol. 3.

[13] G. Ramos, D. Silvestre, and C. Silvestre, "A general discrete-time method to achieve resilience in consensus algorithms," in *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 2702–2707.

[14] ——, "General resilient consensus algorithms," *International Journal of Control*, vol. 95, no. 6, pp. 1482–1496, 2022.

[15] ——, "Node and network resistance to bribery in multi-agent systems," *Systems & Control Letters*, vol. 147, p. 104842, 2021.

[16] J. Usevitch and D. Panagou, "Resilient leader-follower consensus to arbitrary reference values in time-varying graphs," *IEEE Transactions on Automatic Control*, vol. 65, no. 4, pp. 1755–1762, 2019.

[17] D. Saldana, A. Prorok, S. Sundaram, M. F. M. Campos, and V. Kumar, "Resilient consensus for time-varying networks of dynamic agents," in *American Control Conference (ACC)*, May 2017, pp. 252–258.

[18] G. Ramos, D. Silvestre, and A. P. Aguiar, "A resilient continuous-time consensus method using a switching topology," *Systems & Control Letters*, vol. 169, p. 105381, 2022.

[19] J. M. Such, A. Espinosa, and A. García-Fornes, "A survey of privacy in multi-agent systems," *The Knowledge Engineering Review*, vol. 29, no. 3, pp. 314–344, 2014.

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

[21] N. Gupta, J. Katz, and N. Chopra, "Privacy in distributed average consensus," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 9515–9520, 2017, proceedings of the 20th IFAC World Congress.

[22] M. Kishida, "Encrypted average consensus with quantized control law," in *Proceedings of the IEEE Conference on Decision and Control*. IEEE, 2018, pp. 5850–5856.

[23] T. Yin, Y. Lv, and W. Yu, "Accurate privacy preserving average consensus," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 4, pp. 690–694, 2019.

[24] M. Ruan, H. Gao, and Y. Wang, "Secure and privacy-preserving consensus," *IEEE Transactions on Automatic Control*, vol. 64, no. 10, pp. 4035–4049, 2019.

[25] C. N. Hadjicostis and A. D. Domínguez-García, "Privacy-preserving distributed averaging via homomorphically encrypted ratio consensus," *IEEE Transactions on Automatic Control*, vol. 65, no. 9, pp. 3887–3894, 2020.

[26] Z. Huang, S. Mitra, and G. Dullerud, "Differentially private iterative synchronous consensus," in *Proceedings of the ACM workshop on Privacy in the electronic society*, 2012, pp. 81–90.

[27] E. Nozari, P. Tallapragada, and J. Cortés, "Differentially private average consensus: Obstructions, trade-offs, and optimal algorithm design," *Automatica*, vol. 81, pp. 221–231, 2017.

[28] X. Wang, J. He, P. Cheng, and J. Chen, "Privacy preserving average consensus with different privacy guarantee," in *Proceedings of the Annual American Control Conference*. IEEE, 2018, pp. 5189–5194.

[29] L. Gao, S. Deng, and W. Ren, "Differentially private consensus with an event-triggered mechanism," *IEEE Transactions on Control of Network Systems*, vol. 6, no. 1, pp. 60–71, 2018.

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

[31] 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.

[32] F. Yu, L. Li, Q. Tang, S. Cai, Y. Song, and Q. Xu, "A survey on true random number generators based on chaos," *Discrete Dynamics in Nature and Society*, vol. 2019, 2019.

[33] D. Boutat and G. Zheng, "Observability and observer for dynamical systems," in *Observer Design for Nonlinear Dynamical Systems*. Springer, 2021, pp. 1–29.

[34] G. Ramos, A. P. Aguiar, S. Kar, and S. Pequito, "Distributed design of deterministic discrete-time privacy preserving average consensus for multi-agent systems through network augmentation," *arXiv preprint arXiv:2112.09914*, 2021.

[35] J. Giraldo, A. A. Cardenas, and M. Kantarcioglu, "Security vs. privacy: How integrity attacks can be masked by the noise of differential privacy," in *2017 American Control Conference (ACC)*, 2017, pp. 1679–1684.

[36] V. Katewa, R. Anguluri, and F. Pasqualetti, "On a security vs privacy trade-off in interconnected dynamical systems," *Automatica*, vol. 125, p. 109426, 2021.

[37] B. Bollobás, *Modern graph theory*. Springer Science & Business Media, 2013, vol. 184.