Journal of Physics: Complexity

PAPER

Resilient consensus for multiagent networks with time-varying delay and mobile adversaries

Y Shang

Department of Computer and Information Sciences, Northumbria University, Newcastle NE1 8ST, United Kingdom

E-mail: yilun.shang@northumbria.ac.uk and shylmath@hotmail.com

Keywords: Multiagent system, resilient vector consensus, mobile agent, time-varying delay, Erdős–Rényi random network, power-law random network.

OPEN ACCESS

RECEIVED
28 August 2024

REVISED
7 December 2024

ACCEPTED FOR PUBLICATION
13 December 2024

PUBLISHED
30 December 2024

Abstract

Security issues in cyber-physical systems are often the Achilles heel for cooperative control and coordination. This paper addresses resilient consensus problems for multiagent systems under heterogeneous time-varying delays and mobile Byzantine adversaries. We propose two types of mobile attack models for continuous-time dynamical agents with or without infection detection capabilities. Delayed consensus protocols are presented where the agents are able to adjust their behavior during the latency period after recovery from an attack. It is shown that resilient consensus can be achieved under bounded delays and robust communication topologies. As a further contribution, resilient vector consensus has been solved within our framework. Simulations regarding the mobile patterns of adversaries indicate the irrelevance of global and local mobility in terms of convergence rate over Erdős–Rényi random networks. In contrast, local mobility of adversaries leads to longer convergence time compared to global mobility in power-law random networks due to their characteristic topology.

1. Introduction

Over the past few decades, cooperative control of multiagent systems has stirred much research attention due to its successful applications in various networked control systems such as cyber-physical systems and mobile robot networks. A multiagent system is a complex dynamical network consisting of a large number of intelligent agents interacting with each other and the environment [1]. A fundamental problem of such systems is the state consensus, which involves designing distributed communication protocols to ensure the agreement of agents on a quantity of interest. Many important progress on consensus problems can be found in e.g. [2–4].

In large-scale multiagent systems, varied units are connected through communication channels in complex environments creating vulnerabilities to potential attacks. Practical examples range from cyber security issues on the Internet and wireless networks to enemy attacks on military multi-vehicle systems [5]. A malicious agent in the network may send misleading information and even collude with other malicious agents to prevent cooperative agents reaching consensus. Resilient consensus [6] in this context refers to designing distributed control algorithms that guarantee the system performance in the presence of a set of adversaries, whose identities are often undisclosed to the cooperative agents. An early study initiated in [6] proposes a discrete-time strategy, where a cooperative agent ignores a certain number of extremal values received from its neighbors at each step. This update scheme belongs to a class of mean-subsequence reduced (MSR) algorithms [7], which facilitate consensus against malicious and Byzantine agents in networks satisfying certain connectivity condition called graph robustness. Along this line, MSR-type algorithms have been extended to investigate resilient consensus for agents with continuous-time dynamics [8], higher-order dynamics [9, 10], asynchronous networks [11], systems with communication delays [12], and constrained dynamics [13]. In view of the essential role of the communication topology in sustaining resiliency, trusted agents that are insusceptible to attacks have been utilized in [14] to weaken certain topological robustness condition. A different method to reduce network connectivity requirement for resilient consensus in both

discrete and continuous time is realized in [15] by separating optimistic and pessimistic agents in the framework of multiplex networks.

It is noted that all of the above works only consider a static partition of cooperative and malicious nodes of the network. However, in reality, adversaries may change their targets and move across the network. When an adversary leaves a node, it may recover and become functional again giving rise to a time-dependent evolution of status profile in the system. In fact, resilient consensus in the presence of mobile adversaries has been investigated deeply for distributed and parallel algorithms in computer science [16–20]. For example, resilient agreement has been examined in [18] in a synchronous message-passing system composed of n fully connected processes, where malicious adversaries can move from one process to another between rounds. In the work [16], a fault-tolerant consensus algorithm is presented for a network of processors in which a recovered agent has a certain ability to detect its infection when the adversary moves away. This type of result is developed under a common assumption that the network is a complete graph. In these works, computers sharing workloads tend to be connected by wires and the fully connected topology helps the implementation of techniques to reduce attacks and faults. Recently, MSR-type algorithms have been adopted in [21] to realize resilient consensus among a set of discrete-time agents under general topologies satisfying robustness conditions. The adversaries in the models are allowed to hop between nodes and broadcast distorted information to their neighbors. Mobile attackers in higher-order spaces over random dynamic networks are investigated in [22] via martingale theory.

Another related aspect of consensus problems is time delays, which are unavoidable usually. As different agents communicate through a shared network to perform some complicated tasks, both the information transmitted from neighbors and that processed by themselves may be delayed. There has been extensive research regarding delayed consensus [2, 11, 12, 23]. Nevertheless, designing a resilient control strategy that is resilient to both time delays and mobile adversaries is still unsolved.

Inspired by the above observations, this paper studies resilient consensus in continuous-time multiagent systems with time delay and mobile adversaries. The contributions are as follows: (1) We introduce two novel mobile attack models characterizing mobile adversaries in networks with and without infection detection capability. The agents follow continuous-time dynamics and allow the worst case, i.e. Byzantine, attacks compared to the milder malicious agents in [21]. As the agent dynamics are continuous in time, the iterative mechanism to update states therein is no longer applicable. (2) We present distributed MSR-type consensus protocols which guarantee resilience to heterogeneous time-varying delays originated from information transmission and infection detection latency of a cooperative agent. Our method is different from the previous delayed resilient consensus works such as [11, 12], where delays are only induced by information transmission through discrete-time sampling. (3) We introduce a new vector-valued resilience consensus protocol as a further generalization. Designing MSR-type resilient vector consensus protocols is particularly challenging due to the difficulty in ordering vector states appropriately. The few existing works (see e.g. [28–30]) often rely on identity-specific transformations converting vectors to scalars, which are no longer applicable here as the identities of the Byzantine agents may change as a function of time in the current scenario.

The rest of the paper is organized as follows. Section 2 contains the problem formulation and model description. Section 3 presents the delayed resilient consensus algorithms. The consensus analysis is performed in section 4. The results are extended to vector-valued agents in section 5. Simulations are shown in section 6. Section 7 concludes the paper.

2. Preliminaries and model setup

2.1. Graph theory

The communication topology of the network is characterized by a time-dependent network with directed information flow. Specifically, consider a directed graph $\mathcal{G}(t)=(\mathcal{V},\mathcal{E}(t),\mathcal{A}(t))$ with $\mathcal{V}=\{1,2,\cdots,n\}$ being the set of nodes or agents and $t\in\mathbb{R}$ representing time. An edge in $\mathcal{E}(t)\subseteq\mathcal{V}\times\mathcal{V}$ is an order pair $(j,i)\in\mathcal{E}(t)$ meaning that agent i may receive information from agent j at time t. The adjacency matrix $\mathcal{A}(t)=(a_{ij}(t))\in\mathbb{R}^{n\times n}$ describes the coupling strength of edges and we set $a_{ij}(t)>0$ if $(j,i)\in\mathcal{E}(t)$ and $a_{ij}(t)=0$ otherwise. The neighborhood of agent i, signified by $\mathcal{N}_i(t)=\{j\in\mathcal{V}:(j,i)\in\mathcal{E}(t)\}$, is composed of all in-neighbors of i at time t. Similarly, if i can send information to j, namely, $(i,j)\in\mathcal{E}(t)$, we call j an out-neighbor of i in $\mathcal{G}(t)$. A path from i_1 to i_ℓ is a sequence of edges $(i_k,i_{k+1})\in\mathcal{E}(t)$ for $k=1,2,\cdots,\ell-1$. A directed spanning tree in $\mathcal{G}(t)$ is formed by a bunch of paths originating from the same node called root, to all other nodes in $\mathcal{V}$ [1]. We will suppress the dependence on t if the dependence is not essential in our notation.

The concept of graph robustness has been found fruitful in formulating system resilience [6, 9]. Given $r\in\mathbb{N}$ and $\mathcal{S}\subseteq\mathcal{V}$, we say the set $\mathcal{S}$ is r-reachable if there is an agent $i\in\mathcal{S}$ such that $|\mathcal{N}_i\backslash\mathcal{S}|\geqslant r$, where $|\cdot|$ means the size of a set. A graph $\mathcal{G}$ is r-robust if for any two disjoint nonempty subsets $\mathcal{S}_1,\mathcal{S}_2\subseteq\mathcal{V}$, one of

them will be r-reachable. It is known that 1-robustness is equivalent to having a directed spanning tree. The higher r is, the stronger connectivity the graph $\mathcal{G}$ has.

2.2. System formulation

The node set $\mathcal{V}$ in our network $\mathcal{G}(t)$ is composed of cooperative agents and Byzantine agents. There are some attackers (i.e. adversaries) in the network and if they attack a cooperative agent, the agent will become Byzantine. Using the language of epidemic modelling, we can say that the node is infected. Byzantine agents are recognized as the worst-case scenario as they have complete information on the network and are able to transmit arbitrary messages to their neighbors [13, 14, 16, 18]. Denote by $\mathcal{C}(t)$ and $\mathcal{B}(t)$ the sets of cooperative agents and Byzantine agents, respectively. We have $\mathcal{V} = \mathcal{C}(t) \cup \mathcal{B}(t)$ and $|\mathcal{C}(t)| + |\mathcal{B}(t)| = n$ at any time t. The Byzantine agents neither disclose their number nor identity to cooperative agents nor do they comply with the predefined control laws for cooperative agents. On the other hand, each cooperative agent $i \in \mathcal{C}(t)$ has the following dynamics

$$\dot{x}_i(t) = u_i(t), \qquad t \geqslant 0, \tag{1}$$

where $x_i(t) \in \mathbb{R}$ is the state value and $u_i(t) \in \mathbb{R}$ is the control input of agent i to be designed.

Let $f\colon (a_1, a_2) \to \mathbb{R}$ be a continuous function on the interval (a_1, a_2). The Dini upper right derivative of f at t is defined as

$$d^+ f(t) = \limsup_{s \to 0^+} \frac{1}{s} (f(t+s) - f(t)). \tag{2}$$

The function f is non-increasing if and only if $d^+ f(t) \leqslant 0$ for $t \in (a_1, a_2)$. A useful property for Dini derivative is the following [24].

Lemma 1. *For $i = 1, 2, \cdots, n$, let $\theta_i(t,x) : (a_1, a_2) \times \mathbb{R}^m \to \mathbb{R}$ be a continuously differentiable function and $\theta(t,x) = \max_{i \in \mathcal{V}} \theta_i(t,x)$, where $\mathcal{V} = \{1, 2, \cdots, n\}$. If $x(t) \in \mathbb{R}^m$ is absolutely continuous on the interval (a_1, a_2), then $d^+\theta(t, x(t)) = \max_{i \in \mathcal{I}(t)} \dot{\theta}_i(t, x(t))$ for $t \in (a_1, a_2)$, where the set $\mathcal{I}(t) := \{i \in \mathcal{V} : \theta_i(t, x(t)) = \theta(t, x(t))\}$.*

2.3. Models for mobile adversaries

Let $r \in \mathbb{N}$ be an upper bound for the number of adversaries in the network. We consider the following two mobile attack models, where adversaries move across the network and change the identities, i.e. $\mathcal{C}(t)$ and $\mathcal{B}(t)$, of agents accordingly.

M1. *Mobile adversaries without detection capability.*
At $t \geqslant 0$, every cooperative agent $i \in \mathcal{C}(t)$ broadcasts its current state to its out-neighbors in $\mathcal{G}(t)$, collects delayed state information from its in-neighbors and updates its own state following (1). Suppose $i \in \mathcal{B}(t)$ and an adversary moves away from i to another agent j at time t. Then agent i recovers immediately and agent j becomes infected accordingly. In other words, $i \in \mathcal{C}(t')$ and $j \in \mathcal{B}(t')$ for $t' \to t^+$. We assume that agent i has a latency period of length $\sigma_i(t) \geqslant 0$ following its recovery at t. During this period, i may still send out faulty information but collects and updates its state normally (if not infected again). An illustration of the identity switching is shown in figure 1(a).

M2. *Mobile adversaries with detection capability.*
This model obeys the similar rules as the above model including the latency period following the recovery at time t. However, a recovering agent, say i, is assumed to have the ability to detect its infection as soon as the adversary moves away. Agent i will raise a flag over the latency period $(t, t + \sigma_i(t)]$ and put down the flag afterward. The flag status may help a cooperative agent decide whether to keep silent to avoid spreading misinformation and how to update its state wisely. We show a scenario of the flag status in figure 1(b).

The attack model M1 is more harmful compared to M2 due to a lack of detection capability. The capability of detection can be realized for example by a reset of the component or a malware scanning [5, 16]. An assumption is made for the mobility of the adversaries.

Assumption 1. There exists a constant $\rho > 0$ such that any adversary cannot move consecutively within a period of ρ. We assume $\sigma_i(t) \leqslant \rho$ for $i \in \mathcal{V}$ and $t \geqslant 0$.

This assumption specifies a restriction on the frequency of movement for adversaries. Different from discrete-time systems, if adversaries move arbitrarily fast, the system may become unstable due to fast switching [25]. Note that in the mobile attack models, any node in $\mathcal{G}(t)$ can become malicious or cooperative at some point. Despite the hop frequency restricted by Assumption 1, the adversaries can move from agent to agent

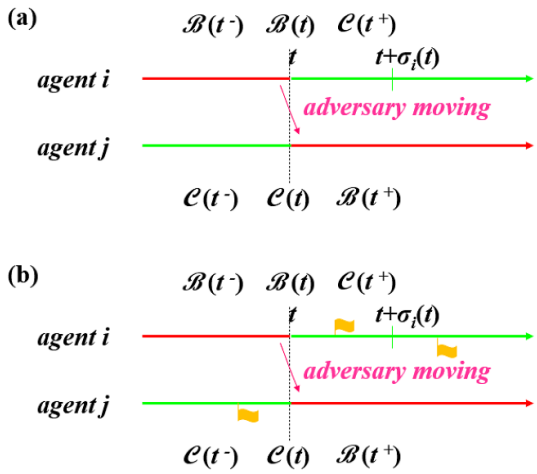

Figure 1. Schematic of a single identity switching at time t for the mobile attack models (a) M1 and (b) M2. Red and green indicate malicious and cooperative identities, respectively.

freely. Even if there is no communication link in $\mathcal{E}(t)$ between two nodes, an adversary is still allowed to move between them.

3. Delayed resilient consensus algorithms

This section introduces the delayed resilient consensus strategies for the two mobile attack models M1 and M2. Due to the mobility of the adversaries and the time delays involved, a cooperative agent may receive more faulty values than in the static and non-delay scenario in which the number is usually bounded by the number of adversaries [6, 13, 14].

When infection detection is not possible, we describe the following algorithm A1 for the model M1 with the parameter r. At $t \geqslant 0$, a cooperative agent $i \in \mathcal{C}(t)$ broadcasts its current state and receives the delayed state information from its in-neighbors. Agent i sorts the values $\{x_i(t), x_j(t-\tau_{ij}(t)), j \in \mathcal{N}_i(t)\}$ in a decreasing order. Here, $\tau_{ij}(t) \geqslant 0$ for $i \neq j$ is the communication delay from agent j to agent i. We assume the delays are bounded by a constant $\tau > 0$, namely, $\tau_{ij}(t) \leqslant \tau$ for all $i, j \in \mathcal{V}$. The first $2r$ and the last $2r$ values in this list are deleted. The set of agents remaining in the list is signified by $\mathcal{R}_i(t) \subseteq \{i\} \cup \mathcal{N}_i(t)$. The control input in (1) is given as follows

$$u_i(t) = \sum_{j \in \mathcal{R}_i(t)} a_{ij}(t)\, \varphi_{ij}\left(x_j\left(t-\tau_{ij}(t)\right), x_i(t)\right), \tag{3}$$

where $\varphi_{ij} : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ satisfies the following condition.

Assumption 2. For $i, j \in \mathcal{V}$ $(i \neq j)$, φ_{ij} is a continuous and locally Lipschitz function satisfying $\varphi_{ij}(z_1, z_2) = 0$ if and only if $z_1 = z_2$, and $\varphi_{ij}(z_1, z_2)(z_1 - z_2) > 0$ for $z_1 \neq z_2$. Moreover, there exist two constants $\alpha_2 > \alpha_1 > 0$ such that $\alpha_1 \leqslant a_{ij}(t) \leqslant \alpha_2$ for $(j, i) \in \mathcal{E}(t)$, $t \geqslant 0$.

The condition for the system functions $\varphi_{ij}(z_1, z_2)$ accommodates commonly chosen linear and nonlinear functions such as $z_1 - z_2$ and $\mathrm{sgn}(z_1 - z_2)|z_1 - z_2|^{\alpha}$ for some $\alpha > 0$ [3, 4], where $\mathrm{sgn}(\cdot)$ is the signum function. It is worth noting that in the above algorithm A1, the self-state $x_i(t)$ is involved in the computation. This is to mitigate the risk of a faulty value when agent i sends its value within the latency period. The complexity of A1 is comparable to that of an ordinary MSR-type protocol, where the sorting component dominates the computation load [13]. Specifically, the total running time of the algorithm consists of three parts. The first part is the time κ_1 for sorting the neighbors for all cooperative agents, which is bounded by $O(nq \ln q)$. The second part is the time $\kappa_2 = O(1)$ for removing the first q_1 values and the third part is the time $\kappa_3 = O(1)$ for removing the last q_1 values. Hence, the average running time of the resilient consensus algorithm is bounded by $O(n^2 \ln n)$.

When agents in the network possess the infection detection capability, we modify the above A1 and adopt the protocol A2 as below. Given the parameter r and the model M2. At $t \geqslant 0$, a cooperative agent $i \in \mathcal{C}(t)$ first checks its flag status. If the flag is down, it broadcasts its current state; Otherwise, it remains quiet without sending any value. The delayed state information is received by agent i. If the flag is down, it sorts the values $\{x_j(t-\tau_{ij}(t)), j \in \mathcal{N}_i(t)\}$ in a decreasing order; Otherwise, it sorts the values $\{x_i(t), x_j(t-\tau_{ij}(t)), j \in \mathcal{N}_i(t)\}$ in

Algorithm A1. for a cooperative agent $i \in \mathcal{C}(t)$.

Input: $x_i(t)$, $\{x_j(t-\tau_{ij}(t))\}_{j\in\mathcal{N}_i(t)}$
Output: $\mathcal{R}_i(t)$
set $q = |\mathcal{N}_i(t)| + 1$
write $\{x_i(t), x_j(t-\tau_{ij}(t)), j \in \mathcal{N}_i(t)\}$ decreasingly as $x_{i_1} \geqslant x_{i_2} \geqslant \cdots \geqslant x_{i_q}$
set $\mathcal{R}_i(t) = \{i\} \cup \mathcal{N}_i(t)$ and $q_1 = q$
if $q_1 > 2r$
  $q_1 = 2r$
end if
for $k = 1$ till $k = q_1$
  delete i_k from $\mathcal{R}_i(t)$
end for
for $k = \max\{q - q_1 + 1, q_1 + 1\}$ till $k = q$
  delete i_k from $\mathcal{R}_i(t)$
end for

Algorithm A2. for a cooperative agent $i \in \mathcal{C}(t)$.

Input: $x_i(t)$, $\mathrm{flag}_i(t)$, $\{x_j(t-\tau_{ij}(t))\}_{j\in\mathcal{N}_i(t)}$
Output: $\mathcal{R}_i(t)$
if $\mathrm{flag}_i(t) = 1$
  set $q = |\mathcal{N}_i(t)| + 1$
  write $\{x_i(t), x_j(t-\tau_{ij}(t)), j \in \mathcal{N}_i(t)\}$decreasingly as $x_{i_1} \geqslant x_{i_2} \geqslant \cdots \geqslant x_{i_q}$
  set $\mathcal{R}_i(t) = \{i\} \cup \mathcal{N}_i(t)$ and $q_1 = q$
else
  set $q = |\mathcal{N}_i(t)|$
  write $\{x_j(t-\tau_{ij}(t)), j \in \mathcal{N}_i(t)\}$decreasingly as $x_{i_1} \geqslant x_{i_2} \geqslant \cdots \geqslant x_{i_q}$
  set $\mathcal{R}_i(t) = \mathcal{N}_i(t)$ and $q_1 = q$
end if
if $q_1 > r$
  $q_1 = r$
end if
for $k = 1$ till $k = q_1$
  delete i_k from $\mathcal{R}_i(t)$
end for
for $k = \max\{q - q_1 + 1, q_1 + 1\}$ till $k = q$
  delete i_k from $\mathcal{R}_i(t)$
end for
if $\mathrm{flag}_i(t) = 0$
  insert i to $\mathcal{R}_i(t)$
end if

a decreasing order. Then the first r and the last r values in this list are deleted. The set of agents remaining in the list is denoted by $\mathcal{R}_i(t) \subseteq \{i\} \cup \mathcal{N}_i(t)$. The control input in (1) is again given by the expression (3).

Taking advantage of the extra information offered by the flag, fewer values are removed in the protocol A2 compared to A1. This is in line with our analysis that M2 is less harmful than M1. Nevertheless, the network connectivity may still be weakened since recovered agents will be literally invisible during the latency period. Similarly as before, the complexity in this scenario is bounded by $O(n^2 \ln n)$.

Remark 1. Given the time delays involved in the multiagent system (1), we can formulate it formally by using the functional differential equation framework; e.g. [26]. Let $C = C([-\tau, 0]; \mathbb{R}^n)$ be a Banach space containing continuous functions from $[-\tau, 0]$ to $\mathbb{R}^n$ with norm $\|f\| = \sup_{t\in[-\tau,0]} |f(t)|$ for $f \in C$. Given $x \in C$, for any $t \geqslant 0$, denote by $x_t(s) := x(t+s)$, $s \in [-\tau, 0]$. Given an initial function $f \in C$, the multiagent system (1) with (3) can be rewritten as

$$\dot{x}(t) = \varphi(x_t), \qquad t \geqslant 0, \tag{4}$$

where $\varphi : C \to \mathbb{R}^n$ is a continuous functional and $x(t) = (x_1(t), x_2(t), \cdots, x_n(t)) \in \mathbb{R}^n$. Here, the index i in the component $x_i(t)$ should not be confused with the time subscript t defined above. A solution of (4) with the initial condition f, denoted by $x(f)(t)$, is defined on $[-\tau, \infty)$, which satisfies the equation (4) over $t \in [0, \infty)$ and $x(f)(0) = f$. This solution is unique.

To develop the resilient consensus results, we assume a fixed underlying communication network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$. Since some edges will be removed in our protocols A1 and A2, we are essentially working with a time-dependent network $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t), \mathcal{A}(t))$ as described in section 2. The switching of the network topology satisfies the following condition.

Assumption 3. Suppose that the network $\mathcal{G}(t)$ switches at the following time points $0 < t_1 < t_2 < \cdots < t_k < t_{k+1} < \cdots$. There exists a constant $\alpha_3 > 0$ such that $t_{k+1} - t_k \geqslant \alpha_3$ for all $k \in \mathbb{N}$.

We stress here that the time-dependency of the topology $\mathcal{G}(t)$ is due to the filtering mechanism in our algorithms. The mobile adversaries do not directly contribute to the change of topology but rather give rise to the switching of identities of agents. This unique feature has not been addressed in the previous resilient consensus protocols for time-varying topologies; see e.g. [6, 9, 28].

With the above assumptions and protocols, we aim to show that the cooperative agents in the network will reach state consensus in spite of the interference of mobile adversaries, i.e. (a) $\{x_i(t)\}_{i \in \mathcal{C}(t), t \geqslant -\tau}$ remains bounded; and (b) $\lim_{t \to \infty} x_i(t) - x_j(t) = 0$ for any $i, j \in \mathcal{C}(t)$ and any initial configuration $f(t) = (f_1(t), f_2(t), \cdots, f_n(t)) \in C$, i.e. $x_i(f)(0) = f_i$ for $i \in \mathcal{V}$.

4. Main results

The delayed resilient consensus result for model M1 is the following.

Theorem 1. *Consider the attack model M1 over the network topology $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ having at most r adversaries. Suppose that Assumptions 1, 2, and 3 hold. Under the protocol A1, if $\mathcal{G}$ is $(4r+1)$-robust, the system (1) and (3) reaches resilient consensus.*

Proof. (Boundedness) Let $\overline{\theta} := \max_{i \in \mathcal{C}(0), t \in [-\tau, 0]} f_i(t)$ and $\underline{\theta} := \min_{i \in \mathcal{C}(0), t \in [-\tau, 0]} f_i(t)$. Clearly, we have $x_i(t) \in [\underline{\theta}, \overline{\theta}]$ for $-\tau \leqslant t \leqslant 0$ and $i \in \mathcal{C}(0)$. Suppose that at some time $\hat{t} > 0$, the upper bound $\overline{\theta}$ is about to break up for the first time. This means there exists an agent $i_0 \in \mathcal{C}(\hat{t})$ such that $x_{i_0}(\hat{t}) = \overline{\theta}$ and $\dot{x}_{i_0}(\hat{t}) > 0$. Moreover, $x_i(t) \leqslant \overline{\theta}$ for any $i \in \mathcal{C}(t)$ and $t \leqslant \hat{t}$. It follows from (1) and (3) that

$$\dot{x}_{i_0}(\hat{t}) = \sum_{j \in \mathcal{R}_{i_0}(\hat{t})} a_{i_0 j}(\hat{t})\, \varphi_{i_0 j}\left(x_j\left(\hat{t} - \tau_{i_0 j}(\hat{t})\right), x_{i_0}(\hat{t})\right), \tag{5}$$

where the left-hand side is positive but the right-hand side is non-positive due to Assumption 2 and $x_j(\hat{t} - \tau_{i_0 j}(\hat{t})) \leqslant x_{i_0}(\hat{t}) = \overline{\theta}$ for $j \in \mathcal{R}_{i_0}(\hat{t})$. The contradiction indicates that the upper bound $x_i(t) \leqslant \overline{\theta}$ holds for all $t \geqslant -\tau$. The same argument can be applied to show the lower bound $x_i(t) \geqslant \underline{\theta}$ holds for all $t \geqslant -\tau$.

(Agreement) Along the solution of the multiagent system (1) and (3), we introduce the continuous Lyapunov–Krasovskii functional $\theta(x_t) = M(x_t) - m(x_t)$, where $t \geqslant 0$, $M(x_t) := \max_{s \in [-\tau, 0], i \in \mathcal{C}(t)} x_i(t+s)$ and $m(x_t) := \min_{s \in [-\tau, 0], i \in \mathcal{C}(t)} x_i(t+s)$. By using lemma 1, we obtain

$$\begin{aligned} d^+ M(x_t) &= \max_{i \in \mathcal{I}_1(t)} \dot{x}_i(t+s_1) := \dot{x}_{i_1}(t+s_1) \\ &= \sum_{j \in \mathcal{R}_{i_1}(t+s_1)} a_{i_1 j}(t+s_1) \cdot \varphi_{i_1 j}\left(x_j\left(t+s_1 - \tau_{i_1 j}(t+s_1)\right), x_{i_1}(t+s_1)\right) \end{aligned} \tag{6}$$

and

$$\begin{aligned} d^+ m(x_t) &= \min_{i \in \mathcal{I}_2(t)} \dot{x}_i(t+s_2) := \dot{x}_{i_2}(t+s_2) \\ &= \sum_{j \in \mathcal{R}_{i_2}(t+s_2)} a_{i_2 j}(t+s_2) \cdot \varphi_{i_2 j}\left(x_j\left(t+s_2 - \tau_{i_2 j}(t+s_2)\right), x_{i_2}(t+s_2)\right), \end{aligned} \tag{7}$$

where $s_1, s_2 \in [-\tau, 0]$, $\mathcal{I}_1(t) = \{j \in \mathcal{C}(t) : x_j(t) = \max_{i \in \mathcal{C}(t)} x_i(t)\}$ and $\mathcal{I}_2(t) = \{j \in \mathcal{C}(t) : x_j(t) = \min_{i \in \mathcal{C}(t)} x_i(t)\}$. The sign of $d^+ M(x_t)$ in (6) can be analyzed as follows. Let $\hat{t}$ be the largest time instant within $t - \tau \leqslant \hat{t} \leqslant t$ such that $x_{i_1}(\hat{t}) = \max_{s \in [-\tau, 0], i \in \mathcal{C}(t)} x_i(t+s)$. If $\hat{t} = t - \tau$, then $t - \tau$ is the unique point within the interval $[t - \tau, t]$ attaining the maximum $\max_{s \in [-\tau, 0], i \in \mathcal{C}(t)} x_i(t+s)$ and $d^+ M(x_t) < 0$. If $t - \tau < \hat{t} < t$, then $d^+ M(x_t) = 0$. Finally, if $\hat{t} = t$, we have $x_{i_1}(\hat{t}) = x_{i_1}(t+s_1) \geqslant x_j(t+s_1 - \tau_{i_1 j}(t+s_1))$ for $j \in \mathcal{R}_{i_1}(t+s_1)$ in (6). By Assumption 2, $\varphi_{i_1 j}(x_j(t+s_1 - \tau_{i_1 j}(t+s_1)), x_{i_1}(t+s_1)) \leqslant 0$ and $d^+ M(x_t) \leqslant 0$. Combining the above discussion, we arrive at

$d^+M(x_t) \leqslant 0$ for any $t \geqslant 0$. We can show $d^+m(x_t) \geqslant 0$ for all t likewise. Therefore, by (6) and (7) we obtain $d^+\theta(x_t) \leqslant 0$ for all $t \geqslant 0$.

Next, we show that $d^+\theta(x_t) \to 0$ as t tends to infinity. Suppose this is not the case, i.e. $\liminf_{t\to\infty} d^+\theta(x_t) < 0$. Hence, there exists an $\varepsilon < 0$ such that for any time $\hat{t} > 0$ there is $t \geqslant \hat{t}$ satisfying $d^+\theta(x_t) < 2\varepsilon$. We choose an infinite sequence $\hat{t}_1 < \hat{t}_2 < \cdots < \hat{t}_l < \hat{t}_{l+1} < \cdots$ such that $d^+\theta(x_{t_l}) < 2\varepsilon$ and $\hat{t}_{l+1} - \hat{t}_l > \beta_1$ for all $l \geqslant 1$, where $\beta_1 > 0$ is a constant. For any time interval $\mathcal{T}$ such that $\mathcal{T} \cap \{t_k\}_{k\geqslant 1} = \emptyset$, where $\{t_k\}_{k\geqslant 1}$ is the set determined in Assumption 3, the solution $x_i(t)$ is always bounded and $d^+\theta(x_t)$ is uniformly continuous over $\mathcal{T}$ by using Assumption 2. Hence, there exists $\beta_2 > 0$ such that $|d^+\theta(x_{t'}) - d^+\theta(x_{t''})| \leqslant -\varepsilon$ for any $t', t'' \in \mathcal{T}$ and $|t' - t''| \leqslant \beta_2$. For any $t \in [\hat{t}_l - \beta_2, \hat{t}_l + \beta_2]$ with $l \geqslant 1$ and $[\hat{t}_l - \beta_2, \hat{t}_l + \beta_2] \cap \{t_k\}_{k\geqslant 1} = \emptyset$, we have

$$\begin{aligned} d^+\theta(x_t) &= -|d^+\theta(x_{\hat{t}_l}) - d^+\theta(x_{\hat{t}_l}) + d^+\theta(x_t)| \\ &\leqslant -|d^+\theta(x_{\hat{t}_l})| + |d^+\theta(x_{\hat{t}_l}) - d^+\theta(x_t)| \leqslant 2\varepsilon - \varepsilon = \varepsilon, \end{aligned} \tag{8}$$

where we recall that ε is a negative constant. In view of Assumption 3, there exists a constant β_3 with $0 < \beta_3 < \alpha_3$ such that (i) the intervals $\{[\hat{t}_l - \beta_3, \hat{t}_l + \beta_3]\}_{l\geqslant 1}$ are pairwise disjoint and (ii) the upper bound $d^+\theta(x_t) \leqslant \varepsilon$ in (8) holds for any $t \in [\hat{t}_l - \beta_3, \hat{t}_l + \beta_3]$ with $l \geqslant 1$. Integrating $d^+\theta(x_t)$ gives rise to the following

$$\int_0^\infty d^+\theta(x_t)\,\mathrm{d}t \leqslant \lim_{\ell\to\infty} \sum_{l=1}^{\ell} \int_{\hat{t}_l-\beta_3}^{\hat{t}_l+\beta_3} d^+\theta(x_t)\,\mathrm{d}t \leqslant \lim_{\ell\to\infty} \sum_{l=1}^{\ell} 2\varepsilon\beta_3 = \lim_{\ell\to\infty} 2\ell\varepsilon\beta_3, \tag{9}$$

which tends to negative infinity. Since $d^+\theta(x_t) \leqslant 0$ for all $t \geqslant 0$, (9) indicates that $\theta(x_t)$ will keep decreasing and cannot be held by any lower bound, say $\theta(x_t) \geqslant c$ for some $c \in \mathbb{R}$. This contradicts with the fact that $\theta(x_t) = M(x_t) - m(x_t) \geqslant 0$ for $t \geqslant 0$. Therefore, we have shown the claim $\lim_{t\to\infty} d^+\theta(x_t) = 0$.

We have $\lim_{t\to\infty} d^+M(x_t) = \lim_{t\to\infty} d^+m(x_t) = 0$. It follows from (6) and (7) that $\lim_{t\to\infty} x_{i_1}(t) = \overline{\rho}$ and $\lim_{t\to\infty} x_{i_2}(t) = \underline{\rho}$ for two constants $\overline{\rho} \geqslant \underline{\rho}$. What remains to show is $\overline{\rho} = \underline{\rho}$. Since there exist no more than r adversaries in $\mathcal{G}$, there are no more than $2r$ faulty values in the network at any time instant t by Assumption 1 and the protocol A1. As $\mathcal{G}$ is $(4r+1)$-robust, by the protocol A1 any node removes at most $4r$ values at any instant. Hence, the resulting network is 1-robust, meaning that it contains a directed spanning tree. It follows from (6) that $x_{i_1}(t) - x_j(t) \to 0$ for all $j \in \mathcal{R}_{i_1}(t)$ as $t \to \infty$. Repeating this argument, we know that the root node of the spanning tree holds the value $\overline{\rho}$ asymptotically. However, we can analogously show that the root node will eventually hold the value $\underline{\rho}$ by using (7). This means $\overline{\rho} = \underline{\rho}$ as expected. The consensus part is proved. □

Remark 2. In the above proof, we have shown that $x_i(t) \in [\underline{\theta}, \overline{\theta}]$ for any $t \geqslant -\tau$. This means the initial values of the cooperative agents form a safe region, which is stronger than the boundedness requirement (a) presented at the end of section 3. Furthermore, we have proved the convergence of states of cooperative agents. This is also stronger than the consensus requirement (b) stated at the end of section 3.

If there is no latency period, namely, $\sigma_i(t) \equiv 0$ for any agent $i \in \mathcal{V}$ and time t, an agent will recover immediately once the adversary leaves from it. This effectively halves the maximal number of potential faulty values spreading in the network. In this case, we modify the algorithm A1 by replacing $2r$, the number of scrapped values, with r. We call the modified algorithm A3. We can similarly prove the following result, which shows the connectivity of the network can be reduced to $(2r+1)$-robustness.

Corollary 1. *Consider the attack model M1 over the network topology $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ having at most r adversaries and $\sigma_i(t) \equiv 0$ for all $i \in \mathcal{V}$. Suppose that Assumptions 1, 2, and 3 hold. Under the protocol A3, if $\mathcal{G}$ is $(2r+1)$-robust, the system (1) and (3) reaches resilient consensus.*

For the attack model M2 with infection detection capability, we have the following result.

Theorem 2. *Consider the attack model M2 over the network topology $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ with at most r adversaries. Suppose that Assumptions 1, 2, and 3 hold. Under the protocol A2, if $\mathcal{G}$ is $(3r+1)$-robust, the system (1) and (3) reaches resilient consensus.*

Proof. The result can be shown following the same line as in theorem 1. We can show that $x_i(t) \in [\underline{\theta}, \overline{\theta}]$ for all $t \geqslant -\tau$ and that $d^+\theta(x_t) \to 0$ as t tends to infinity, which means $\lim_{t\to\infty} d^+M(x_t) = \lim_{t\to\infty} d^+m(x_t) = 0$. We similarly have $\lim_{t\to\infty} x_{i_1}(t) = \overline{\rho}$ and $\lim_{t\to\infty} x_{i_2}(t) = \underline{\rho}$ for two constants $\overline{\rho} \geqslant \underline{\rho}$.

As there exist at most r adversaries in $\mathcal{G}$ and any recovered agent will raise the flag during the latency period, there are at most r faulty values and at most r missing values in the network at any time instant t by Assumption 1 and the protocol A2. Since the topology $\mathcal{G}$ is $(3r+1)$-robust, the resulting network discounting the potential missing values is $(2r+1)$-robust. Applying the protocol A2, a cooperative agent i removes at most $2r$ neighbors when the flag is down; whereas it removes at most $2r$ values (potentially including itself) when the flag is up. In either case, the resulting network is 1-robust, suggesting that it contains a directed spanning tree. By (6), we

obtain that $x_{i_1}(t) - x_j(t) \to 0$ for all $j \in \mathcal{R}_{i_1}(t)$ as $t \to \infty$. Repeating the argument, we have that the root node of the spanning tree holds the value $\overline{\rho}$ asymptotically. An analogous argument shows that the root node will eventually hold the value $\underline{\rho}$ by involving (7). This shows $\overline{\rho} = \underline{\rho}$, which concludes the proof. □

Remark 3. When $\sigma_i(t) \equiv 0$ for any agent $i \in \mathcal{V}$ and any time $t \geqslant 0$, the two models M1 and M2 coincide. The proposed algorithms A3 and A2 also become equivalent. Hence, when latency periods disappear, theorem 2 readily reduces to corollary 1.

Remark 4. The network connectivity condition in theorem 1 requires $(4r+1)$-robustness, which is essential. Suppose $\mathcal{G}$ contains a node i_0 with 4 neighbors $\{i_j\}_{j=1}^4$. Let $r = 1$. By definition, $\mathcal{G}$ is not $(4r+1)$-robust. Suppose $x_{i_0}(t) = 0$, $x_{i_1}(t) = x_{i_2}(t) = 1$, and $x_{i_3}(t) = x_{i_4}(t) = -1$ at some time t. By A1, all the four neighbors will be filtered by agent i_0, which makes i_0 still. This situation can happen when the adversary moves from i_1 to i_2 at time t^- (or analogously from i_3 to i_4), leaving both of them sending false information during the latency period. On the other hand, this condition is comparable to the static adversary scenario (see Corollary 1), where $(2r+1)$-robustness is shown to be necessary [6, 8, 9]. The nearly double robustness is arguably due to the latency period induced by mobility. Theorem 2 requires a weaker robustness condition by equipping agents with infection detection capability. When a recovering agent is aware of its infection, it temporarily stops sending information, which effectively alleviates the filtering workload of other cooperative agents. This condition again is necessary if i_0 has 3 neighbors $\{i_j\}_{j=1}^3$ with $x_{i_0}(t) = 0$, $x_{i_1}(t) = 1$ and $x_{i_3}(t) = -1$ at time t. The adversary may jump from i_2 to i_1 (or analogously from i_2 to i_3) at time t^-, leaving i_2 silent under A2. Hence, none of the three neighbors can influence agent i_0.

Remark 5. MSR-type strategies, as we have performed here, remove neighbors taking extreme values to ensure secure consensus as they do not have functionality to differentiate a normal agent from a malicious one, and are particularly suitable and simple in distributed implementation with low computation resources. An explicit characterization of network robustness is emblematic for these algorithms (cf Remark 4). Another direction in the literature to tackle this problem is the so-called fault-tolerant control (FTC) methods [31], which rely on observer-based fault estimation to compensate for the effect of faults. In the FTC framework, normal agents act as detectors, and various control protocols have been designed for different types of faults such as actuator faults and DoS attacks. This specificity typically leads to a weaker network connectivity requirement. However, that often comes with a cost in more restrictive network topology (e.g. undirected graphs) [31, 32], stability or controllability conditions in agent dynamics [33], and quantized communications [34] etc.

5. An extension to vector-valued resilient consensus

In this section, we consider a generalization of the above results in the higher-dimensional state space. Different from the few existing vector-valued resilient consensus algorithms [28–30], our new strategies here are able to cope with time-varying delays and mobile adversaries.

Let $x_i(t) = (x_{i1}(t), x_{i2}(t), \cdots, x_{im}(t))^{\mathrm{T}} \in \mathbb{R}^m$ be the state vector of agent $i \in \mathcal{V}$, where $m \in \mathbb{N}$ and T means transpose. Every cooperative agent $i \in \mathcal{C}(t)$ follows the same dynamics (1) as before, and $u_i(t) \in \mathbb{R}^m$ here is the control input vector. We consider the case of no infection detection as an example. The algorithm A1 can be modified as below by taking an entry-based approach, which we call A1′.

Specifically, given the parameter r, a cooperative agent $i \in \mathcal{C}(t)$ broadcasts its current state and receives the delayed state information from its in-neighbors. Agent i sorts the state vectors $\{x_i(t), x_j(t - \tau_{ij}(t)), j \in \mathcal{N}_i(t)\}$ in a decreasing order entry-wise. Namely, for each $1 \leqslant k \leqslant m$, the values in $\{x_{ik}(t), x_{jk}(t - \tau_{ij}(t)), j \in \mathcal{N}_i(t)\}$ are sorted. For each k, the first $2r$ and the last $2r$ values in the corresponding list are deleted. The set of agents remaining in the list is denoted by $\mathcal{R}_{ik}(t) \subseteq \{i\} \cup \mathcal{N}_i(t)$. The control input in (1) is then designed as

$$u_i(t) = \sum_{k=1}^{m} e_k e_k^{\mathrm{T}} \cdot \sum_{j \in \mathcal{R}_{ik}(t)} a_{ij}(t)\, \varphi_{ij}\left(x_j\left(t - \tau_{ij}(t)\right), x_i(t)\right), \tag{10}$$

where $e_k \in \mathbb{R}^m$ is the kth unit vector and $\varphi_{ij} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ satisfies the following condition, replacing the counterpart in Assumption 2 by

Assumption 2′. For $i, j \in \mathcal{V}$ $(i \neq j)$, φ_{ij} is a continuous and locally Lipschitz function satisfying $\varphi_{ij}(z_1, z_2) = 0$ if and only if $z_1 = z_2$; and $(z_{1k} - z_{2k}) e_k^{\mathrm{T}} \varphi_{ij}(z_1, z_2) > 0$ for each $1 \leqslant k \leqslant m$, where $z_1 = (z_{11}, \cdots, z_{1m})^{\mathrm{T}}$, $z_2 = (z_{21}, \cdots, z_{2m})^{\mathrm{T}}$ and $z_1 \neq z_2$.

The system (1) with (10) is said to achieve resilient vector consensus if (a) $\{x_i(t)\}_{i \in \mathcal{C}(t), t \geqslant -\tau}$ remains bounded; and (b) $\lim_{t \to \infty} \|x_i(t) - x_j(t)\| = 0$ for any $i, j \in \mathcal{C}(t)$ and any initial configuration $\{f_i(t) = (f_{i1}(t), \cdots, f_{im}(t))\}_{i \in \mathcal{V}}$, where $\|\cdot\|$ is the Euclidean norm in $\mathbb{R}^m$.

Theorem 3. *Consider the attack model M1 over the network topology $\mathcal{G}=(\mathcal{V},\mathcal{E},\mathcal{A})$ having at most r adversaries. Suppose that Assumptions 1, 2′, and 3 hold. Under the protocol A1′, if $\mathcal{G}$ is $(4r+1)$-robust, the system (1) and (10) reaches resilient vector consensus.*

Proof. The theorem can be proved following the same line in theorem 1. It is sketched as follows.

To show the boundedness (a), fix $1\leqslant k\leqslant m$, and define $\overline{\theta}_k:=\max_{i\in\mathcal{C}(0),t\in[-\tau,0]}f_{ik}(t)$ and $\underline{\theta}_k:=\min_{i\in\mathcal{C}(0),t\in[-\tau,0]}f_{ik}(t)$. We can use essentially the same proof of contradiction as in theorem 1, where (5) is replaced by

$$\dot{x}_{i_0}\left(\hat{t}\right)=\sum_{j\in\mathcal{R}_{i_0k}\left(\hat{t}\right)}a_{i_0j}\left(\hat{t}\right)e_k^{\mathrm{T}}\cdot\varphi_{i_0j}\left(x_j\left(\hat{t}-\tau_{i_0j}\left(\hat{t}\right)\right),x_{i_0}\left(\hat{t}\right)\right), \tag{11}$$

invoking (1), (10) and Assumption 2′.

To show the agreement (b), we similarly define the entry-wise version of the Lyapunov–Krasovskii functional $\theta_k(x_t)=M_k(x_t)-m_k(x_t)$, where $t\geqslant 0$, $M_k(x_t):=\max_{s\in[-\tau,0],i\in\mathcal{C}(t)}x_{ik}(t+s)$ and $m_k(x_t):=\min_{s\in[-\tau,0],i\in\mathcal{C}(t)}x_{ik}(t+s)$. Employing lemma 1, we have □

$$\begin{aligned}d^+M_k\left(x_t\right)&=\max_{i\in\mathcal{I}_{1k}(t)}\dot{x}_{ik}\left(t+s_1\right):=\dot{x}_{i_1k}\left(t+s_1\right)\\&=e_k^{\mathrm{T}}\sum_{j\in\mathcal{R}_{i_1k}(t+s_1)}a_{i_1j}\left(t+s_1\right)\cdot\varphi_{i_1j}\left(x_j\left(t+s_1-\tau_{i_1j}\left(t+s_1\right)\right),x_{i_1}\left(t+s_1\right)\right)\end{aligned} \tag{12}$$

and

$$\begin{aligned}d^+m_k\left(x_t\right)&=\min_{i\in\mathcal{I}_{2k}(t)}\dot{x}_{ik}\left(t+s_2\right):=\dot{x}_{i_{2k}}\left(t+s_2\right)\\&=e_k^{\mathrm{T}}\sum_{j\in\mathcal{R}_{i_2k}(t+s_2)}a_{i_2j}\left(t+s_2\right)\cdot\varphi_{i_2j}\left(x_j\left(t+s_2-\tau_{i_2j}\left(t+s_2\right)\right),x_{i_2}\left(t+s_2\right)\right),\end{aligned} \tag{13}$$

where $s_1,s_2\in[-\tau,0]$, $\mathcal{I}_{1k}(t)=\{j\in\mathcal{C}(t):x_j(t)=\max_{i\in\mathcal{C}(t)}x_{ik}(t)\}$ and $\mathcal{I}_{2k}(t)=\{j\in\mathcal{C}(t):x_j(t)=\min_{i\in\mathcal{C}(t)}x_{ik}(t)\}$. By analyzing the signs of (12) and (13), we can analogously show that $d^+\theta_k(x_t)\leqslant 0$ and $d^+\theta_k(x_t)\to 0$ as $t\to\infty$. The rest of the proof follows similarly from theorem 1 by considering each individual k.

□

For the attack model M2, we can modify the algorithm A2 analogously and the resilient vector consensus versions for corollary 1 and theorem 2 hold similarly.

6. Numerical study

Here, we consider two types of moving attackers in our framework, random-walking and edge-walking adversaries. The random-walking adversaries will hop between two nodes arbitrarily ('global mobility') in $\mathcal{G}(t)$ whereas the edge-walking adversaries can only move following the explicit edges ('local mobility') in the network. We assume each adversary follows an independent Poisson process with rate $\lambda=10$. Two different number adversaries $r=5$ and $r=20$ are considered. To illustrate the consensus rate for our protocols, we adopt two network scenarios:

(a) Erdős–Rényi random networks with edge probabilities $p=0.1,0.2,0.5$, respectively, and number of nodes $n=600$. The graph is assumed to have a binary adjacency matrix. According to the robustness threshold [27, Theorem 3], these graphs almost surely satisfy the robustness requirements specified in our theorem 1 and theorem 2. Assume $\varphi_{ij}(z_1,z_2)=z_1-z_2$ for $z_1,z_2\in\mathbb{R}$, $\tau_{ij}(t)\equiv\sigma_i(t)\equiv\rho/2$ for all $i,j\in\mathcal{V}$ and t. The initial conditions for the multiagent system are chosen as $f_i(t)=(t+10)\cdot\xi_i$ for $t\in[-10,0]$, where $\{\xi_i\}_{i\in\mathcal{V}}$ are independent uniform random variables taking values in the interval $[0,1]$.

(b) Power-law random networks, where a sequence of weights $w_1,w_2,\cdots,w_n$ is assigned to the nodes in $\mathcal{V}$ and the edge probability between nodes i and j follows $p_{ij}=w_iw_j/(\sum_{l=1}^n w_l)$. As in [35], we take $w_l=\gamma/l^{1/2}$ for $l_0\leqslant l\leqslant n+l_0$, $\gamma=(d/2)n^{1/2}$, $l_0=n(d/800)^2$, where d is the average degree. Let the number of nodes be $n=600$ and the adjacency matrix be binary. The power-law networks satisfy the robustness requirements in theorem 1 and theorem 2 [36]. All the other parameters are set as in (a).

We measure the consensus time by requiring the difference among the cooperative agents by less than 10^{-4}. The consensus time for random-walking and edge-walking adversaries are denoted by t_{random} and t_{edge},

Table 1. Consensus time t_{random} for random-walking adversaries and t_{edge} for edge-walking adversaries in models M1(A1) and M2(A2) over Erdős–Rényi random networks. For each combination of edge probability p, mobility frequency bound ρ and number of adversaries r, the times are averaged over 100 instances and recorded in the unit of 1000 steps. The standard errors are less than 0.4 in all cases.

p	t_{random}(A1/A2)	t_{edge}(A1/A2)	p	t_{random}(A1/A2)	t_{edge}(A1/A2)
$\rho = 1$					
$r = 5$			$r = 20$		
0.1	4.03/3.99	4.01/3.95	0.1	5.21/5.17	5.25/5.20
0.2	3.77/3.71	3.84/3.76	0.2	5.10/5.03	5.14/5.09
0.5	3.52/3.48	3.42/3.31	0.5	4.98/4.86	4.82/4.78
$\rho = 10$					
$r = 5$			$r = 20$		
0.1	4.26/4.20	4.29/4.27	0.1	5.39/5.33	5.33/5.28
0.2	3.95/3.89	3.92/3.86	0.2	5.27/5.24	5.25/5.20
0.5	3.68/3.61	3.73/3.64	0.5	5.05/5.00	5.12/5.06

Table 2. Consensus time t_{random} for random-walking adversaries and t_{edge} for edge-walking adversaries in models M1(A1) and M2(A2) over power-law random networks. For each combination of average degree d, mobility frequency bound ρ and number of adversaries r, the times are averaged over 100 instances and recorded in the unit of 1000 steps. The standard errors are less than 0.4 in all cases.

d	t_{random}(A1/A2)	t_{edge}(A1/A2)	d	t_{random}(A1/A2)	t_{edge}(A1/A2)
$\rho = 1$					
$r = 5$			$r = 20$		
50	4.54/4.48	5.13/5.09	50	5.72/5.68	6.39/6.33
100	4.28/4.23	4.71/4.65	100	5.46/5.43	6.07/6.04
200	3.86/3.82	4.20/4.14	200	5.18/5.12	5.76/5.72
$\rho = 10$					
$r = 5$			$r = 20$		
50	4.78/4.69	5.37/5.31	50	5.96/5.92	6.63/6.58
100	4.51/4.45	4.94/4.89	100	5.69/5.63	6.31/6.25
200	4.10/4.06	4.47/4.40	200	5.43/5.37	6.02/5.94

respectively. It is not difficult to verify that all requirements in our theorems are satisfied. We present the results for simulations over the average of 100 independent instances in table 1 for (a) Erős–Rényi random networks and in table 2 for (b) power-law random networks.

The following observations are in order. (i) It is found that in each pair of consensus time, $t_{\text{random}}(\text{A1}) > t_{\text{random}}(\text{A2})$ and $t_{\text{edge}}(\text{A1}) > t_{\text{edge}}(\text{A2})$ under all circumstances. This agrees with our analysis that more edges in M1 is removed, which depreciates the connectivity of the communication topology. (ii) A denser network with larger p or d facilitates resilient consensus for both models in all scenarios. (iii) The consensus time is affected by the number r of adversaries. A larger r leads to a slower consensus as one would expect. (iv) The time delays play a clear role in both models: Longer time delay and latency (e.g. $\rho = 10$) hinder the consensus process. (v) It is interesting to note that the two different moving patterns for adversaries seem not to affect consensus time in a consistent pattern for Erdős–Rényi random networks. This is presumably due to the random graph setting here as the edges are connected independently at random, which mitigates the influence of tracking a particular edge. However, contrary to the Erdős–Rényi random networks, edge-walking adversaries prominently impede the consensus process in power-law networks. We contend that the slow consensus may find its origins in the existence of hubs in such networks, which are against consensus once compromised.

We mention that in the above experiments, the wiring probability in the Erdős-Rényi networks for example is high, which leads to dense networks. This ensures that the robustness topological conditions in our main results, theorem 1 and theorem 2, remain valid even in the presence of a large number of adversaries ($r = 5$ and $r = 10$). In the literature, a common approach to achieve resilient consensus in sparser networks is the introduction of trusted agents (e.g. [12]). To further investigate, we now consider a sparse network scenario with $r = 1$. We consider the same Erdős-Rényi networks with $n = 600$ but a much smaller wiring

Table 3. Consensus time t with models A1 and A2 and the resilient consensus protocols in [6] and [14] over Erdős–Rényi random networks with $n = 600$ and $p = 1/60$ for different adversary mobility. The times are averaged over 100 instances and recorded in the unit of 1000 steps. The standard errors are shown in the brackets.

	t(A1)	t(A2)	t([6])	t([14])
Edge-walking	26.4(0.8)	25.2(0.8)	∞	∞
Random-walking	27.0(0.9)	25.7(0.8)	∞	∞
Static	25.8(0.8)	24.6(0.8)	16.1(0.7)	14.3(0.7)

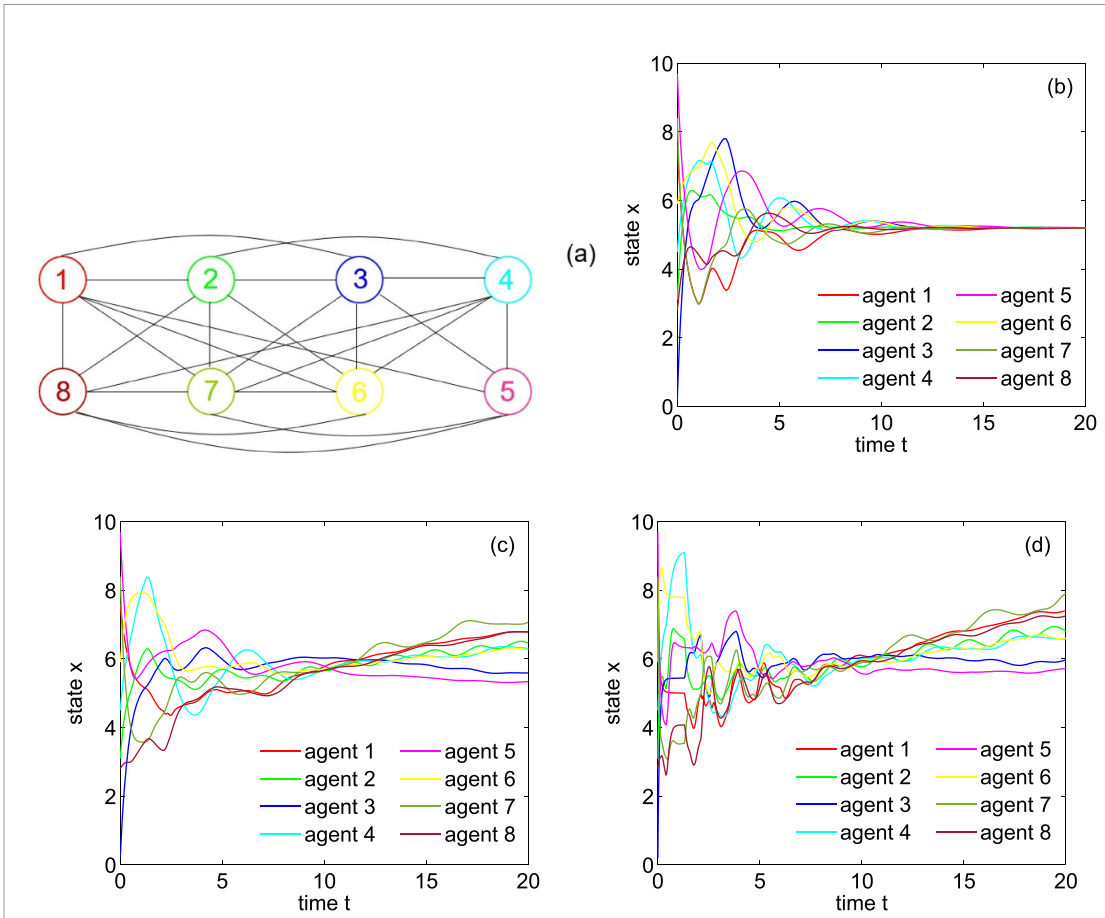

Figure 2. (a) 5-robust communication topology; (b) agent trajectories for $r = 1$ adversary; (c) agent trajectories for $r = 2$ adversaries; (d) agent trajectories for $r = 2$ adversaries and coupling weight $a_{ij} = 5$ for all edges.

probability $p = 1/60$. In table 3 below we show the consensus time for our algorithms A1, and A2 along with the classical resilient consensus protocols proposed in [6] and [14] under different mobility conditions of the adversary. We observe that when the adversary is moving, the previous static resilient consensus protocols are not able to achieve consensus as they typically do not filter enough neighboring states. However, for the same reason, when the adversary is static, the previous static protocols offer better convergence rates.

Finally, we illustrate the trajectories of agents for a subgraph of $n = 8$ agents in figure 2(a) with binary weights. The graph topology is 5-robust. As per theorem 1, the network topology is able to sustain resilient consensus against $r = 1$ adversary; see figure 2(b). If there are two adversaries moving across the graph, consensus may not be reached. This is the situation shown in figure 2(c). In this case, increasing coupling strength can not help recover consensus as shown in figure 2(d), where we have applied a five times weight, namely $a_{ij} = 5$ for each edge (j, i).

7. Conclusion

In the context of cooperative coordination of multiagent systems, this paper has studied the resilient consensus over mobile Byzantine adversaries under time-varying delays. We propose two classes of mobile attack models accommodating situations with or without infection detection capabilities. Recovered agents may decide their behavior during the latency period to contain further infection or damage to other parts of

the network. Distributed consensus protocols have been presented to effectively achieve vector consensus when the total number of adversaries is bounded in the system. The frameworks presented in this paper are general and flexible. It would be interesting to refine the attack modes and the adversary moving rules. Practical control issues such as communication noise, asynchronous algorithms, and event-triggered control are worth investigating. Also, note that the status consensus considered here is asymptotic as time tends to infinity. More efficient fixed-time and prescribed-time consensus have also been investigated in the literature; see the survey [37].

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Conflict interests

The author declare that they have no conflict of interest.

ORCID iD

Y Shang https://orcid.org/0000-0002-2817-3400

References

[1] Mesbahi M and Egerstedt M 2010 *Graph Theoretic Methods in Multiagent Networks* (Princeton University Press)
[2] Olfati-Saber R and Murray R M 2004 Consensus problems in networks of agents with switching topology and time-delays *IEEE Trans. Autom. Contr.* **49** 1520–33
[3] Qin J, Ma Q, Shi Y and Wang L 2017 Recent advances in consensus of multi-agent systems: a brief survey *IEEE Trans. Ind. Electron.* **64** 4972–83
[4] Shi P and Yan B 2021 A survey on intelligent control for multiagent systems *IEEE Trans. Syst. Man Cybern. Syst.* **51** 161–75
[5] Zhang H, Liu B and Wu H 2021 Smart grid cyber-physical attack and defense: a review *IEEE Access* **9** 29641–59
[6] LeBlanc H J, Zhang H, Koutsoukos X and Sundaram S 2013 Resilient asymptotic consensus in robust networks *IEEE J. Select. Areas Commun.* **31** 766–81
[7] Kieckhafer R M and Azadmanesh M H 1993 Low cost approximate agreement in partially connected networks *J. Comput. Inf.* **3** 53–85
[8] Shang Y 2018 Resilient consensus of switched multi-agent systems *Syst. Contr. Lett* **122** 12–18
[9] LeBlanc H J and Koutsoukos X 2018 Resilient first-order consensus and weakly stable, higher order synchronization of continuous-time networked multiagent systems *IEEE Trans. Contr. Netw. Syst.* **5** 1219–31
[10] Zhao D, Lv Y, Yu X, Wen G and Chen G 2022 Resilient consensus of higher-order multi-agent networks: an attack-isolation-based approach *IEEE Trans. Autom. Contr.* **67** 1001–7
[11] Dibaji S M and Ishii H 2017 Resilient consensus of second-order agent networks: asynchronous update rules with delays *Automatica* **81** 123–32
[12] Zhai Y, Liu Z-W, Guan Z-H and Gao Z 2022 Resilient delayed impulsive control for consensus of multiagent networks subject to malicious agents *IEEE Trans. Cybern.* **52** 7196–205
[13] Shang Y 2020 Resilient consensus in multi-agent systems with state constraints *Automatica* **122** 109288
[14] Abbas W, Laszka A and Koutsoukos X 2018 Improving network connectivity and robustness using trusted nodes with application to resilient consensus *IEEE Trans. Contr. Netw. Syst.* **5** 2036–48
[15] Shang Y 2021 Resilient consensus for robust multiplex networks with asymmetric confidence intervals *IEEE Trans. Netw. Sci. Eng.* **8** 65–74
[16] Garay J A 1994 Reaching (and maintaining) agreement in the presence of mobile faults *Proc. 8th Int. Workshop Distrib. Algorithms* (*Netherlands*) pp 253–64
[17] Buhrman S, Garay J A and Hoepman J H 1995 Optimal resiliency against mobile faults *Proc. 25th Int. Symp. Fault-Tolerant Comput. Pasadena (California)* pp 83–88
[18] Bonnet F, Défago X, Nguyen T D and Potop-Butucaru M 2016 Tight bound on mobile Byzantine agreement *Theor. Comput. Sci.* **609** 361–73
[19] Sakavalas D and Tseng L 2018 Delivery delay and mobile faults *Proc. IEEE 17th Int. Symp. Netw. Comput. Appl.* (*Cambridge MA*) pp 1–8
[20] Bonomi S, Pozzo A D, Potop-Butucaru M and Tixeuil S 2019 Approximate agreement under mobile Byzantine faults *Theor. Comput. Sci.* **758** 17–29
[21] Wang Y, Ishii H, Bonnet F and Défago X 2022 Resilient real-valued consensus in spite of mobile malicious agents on directed graphs *IEEE Trans. Parall. Distr. Syst.* **33** 586–603
[22] Shang Y 2024 Resilient vector consensus over random dynamic networks under mobile malicious attacks *The Comput. J.* **67** 1076–86
[23] Qi T, Lu R and Chen J 2021 Consensus of continuous-time multiagent systems via delayed output feedback: delay versus connectivity *IEEE Trans. Autom. Contr.* **66** 1329–36
[24] Danskin J M 1966 The theory of max-min, with applications *SIAM J. Appl. Math.* **14** 641–64
[25] Yang H, Jiang B and Cocquempot V 2014 A survey of results and perspectives on stabilization of switched nonlinear systems with unstable modes *Nonlin. Anal. Hybrid Syst* **13** 45–60
[26] Haddock J R, Krisztin T, Terjéki J and Wu J H 1994 An invariance principle of Lyapunov-Razumikhin type for neutral functional differential equations *J. Differ. Equat.* **107** 395–417
[27] Zhang H, Fata E and Sundaram S 2015 A notion of robustness in complex networks *IEEE Trans. Contr. Netw. Syst.* **2** 310–20

[28] Rezaee H, Parisini T and Polycarpou M M 2021 Almost sure resilient consensus under stochastic interaction: links failure and noisy channels *IEEE Trans. Autom. Contr.* **66** 5727–41
[29] Abbas W, Shabbir M, Li J and Koutsoukos X 2022 Resilient distributed vector consensus using centerpoint *Automatica* **136** 110046
[30] Shang Y 2022 Median-based resilient consensus over time-varying random networks *IEEE Trans. Circuits Syst. Express Briefs* **69** 1203–7
[31] Yang H, Han Q-L, Ge X, Ding L, Xu Y, Jiang B and Zhou D 2020 Fault-tolerant cooperative control of multiagent systems: a survey of trends and methodologies *IEEE Trans. Ind. Inf.* **16** 4–17
[32] Yu Z, Zhou R, Sun P, Zhang Y, Jiang B and Su C-Y 2024 Hierarchical distributed adaptive fault-tolerant control of nonlinear fractional-order multiagent systems with faults and periodic disturbances using event-triggered communication *IEEE Trans. Cybern.* **54** 5231–43
[33] Zhang L, Zhang H, Sun J and Yue X 2024 ADP-based fault-tolerant control for multiagent systems with semi-Markovian jump parameters *IEEE Trans. Cybern.* **54** 5952–62
[34] Guo X, Wang C and Liu L 2024 Adaptive fault-tolerant control for a class of nonlinear multi-agent systems with multiple unknown time-varying control directions *Automatica* **167** 111802
[35] Chung F, Lu L and Vu V 2003 Eigenvalues of random power law graphs *Ann. Combin.* **7** 21–33
[36] Shang Y 2023 On connectivity and robustness of random graphs with inhomogeneity *J. Appl. Prob.* **60** 284–94
[37] Ning B, Han Q-L, Zuo Z, Ding L, Lu Q and Ge X 2023 Fixed-time and presccribed-time consensus control of multiagent systems and its applications: a survey of recent trends and methodologies *IEEE Trans. Ind. Inf.* **19** 1121–35