SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

September 2026, Vol. 69, Iss. 9, 192201:1–192201:18
https://doi.org/10.1007/s11432-024-4459-5

# Privacy-preserving leaderless consensus control of nonlinear multi-agent systems under attacks: improved Liu cryptosystem

Yang YANG[1*], Fanming HUANG[1] & Wenbin YUE[2]

[1]*College of Automation, Nanjing University of Posts and Telecommunications, Nanjing 210023, China*
[2]*China North Vehicle Research Institute, Beijing 100072, China*

Received 19 December 2024/Revised 11 February 2025/Accepted 3 April 2025/Published online 13 May 2026

**Abstract** This paper investigates a privacy-preserving leaderless consensus problem for a class of nonlinear multiagent systems (MASs) under attacks. For false data injection attacks, by a finite-time attack detector for nonlinear MASs with privacy-preserving signals, a distributed detection-isolation algorithm is developed to detect and isolate compromised agents within nonlinear dynamics. Via an amplification technique for signals in an improved Liu cryptosystem, the sensitivity of the information is preserved, and decryption errors between plaintext encryption and decryption are mitigated, ensuring both privacy preservation and satisfactory recovery of plaintext information. With privacy-preserving and attack-free information, a privacy-preserving leaderless consensus control strategy is then developed via backstepping and reinforcement learning (RL) techniques. This privacy-preserving RL-based consensus control strategy compensates for unknown dynamics and errors between true signals and decrypted ones with fewer learning parameters. With graph theory and Lyapunov stability theory, it is also proven that the output consensus error and all signals are ultimately bounded. Simulation examples are provided to validate the effectiveness of this control strategy.

**Keywords** privacy-preserving control, Liu cryptosystem, attack detector, consensus

**Citation** Yang Y, Huang F M, Yue W B. Privacy-preserving leaderless consensus control of nonlinear multi-agent systems under attacks: improved Liu cryptosystem. Sci China Inf Sci, 2026, 69(9): 192201, https://doi.org/10.1007/s11432-024-4459-5

## 1 Introduction

Over the past decades, consensus control of multiagent systems (MASs) has developed rapidly and consistently [1,2]. Its wide-ranging applications span military and civilian fields, including underwater vehicles, mobile robots, multi-vehicle systems, and power systems [3, 4]. Despite the convenience of networks, the risk of MASs facing threats from cyberattacks and eavesdroppers is increasing due to the openness of communication as well as insufficient protection. Therefore, security and privacy are of paramount importance for MASs.

Attack detection and isolation are critical for consensus control of MASs [5–9]. In this respect, a machine learning-based jamming detection algorithm is proposed for wireless communication attacks [6]. This algorithm can classify known attacks that are used during training and detect unknown attacks that are not part of the training process. For covert attacks and collusions among them [7], a two-stage fixed-time observer is designed, and, additionally, an attack isolation algorithm is developed. Gallo et al. [9] introduced a distributed monitoring scheme that provides attack detection capabilities for linear large-scale systems, where the proposed architecture relies on Luenberger observers and a set of unknown-input observers. However, these existing attack detection methods [8, 9] are not applied in real-world conditions and fail to account for the combined impact of both noise and privacy preservation.

Privacy preservation is a research hotspot in the consensus control of MASs [10, 11]. Numerous relevant technologies reportedly preserve privacy, such as state decomposition [12, 13], differential privacy [14] and Paillier encryption [15]. Differential privacy [14] involves adding noise directly into data before transmission, which impacts control precision. Meanwhile, state decomposition [12, 13] requires decomposing states into two parts, as well as two controllers. However, this decomposition is complex and only applicable to linear MASs. Different from these privacy-preserving methods [12–14], the Liu cryptosystem [16] encrypts data before transmission, and decrypted information is then provided for control schemes. Notably, these results [15] do not account for the impact of the error between encryption and decryption in an MAS.

* Corresponding author (email: yyang@njupt.edu.cn)

 

Reinforcement learning (RL) is a powerful tool for handling nonlinear dynamics in MASs. To eliminate dependence on inaccurate and incomplete network models and enhance resilience against communication, and controller failures, a deep RL consensus algorithm [17] is proposed to solve the voltage—VAR control problem. Moreover, to address the issue posed by internal coupling among agents, uncertainties, and nonlinear dynamics, with RL techniques, a sliding-mode control consensus approach [18] is developed, where RL is introduced to seek consensus control protocols for all agents. Meanwhile, a data-based distributed control algorithm [19] is proposed for optimal distributed consensus control of a discrete-time MAS with completely unknown dynamics using offline interaction datasets in an MAS. In Wen and Li [20], RL approaches overcome the difficulty in solving a Hamilton-Jacobi-Bellman equation, and then an optimized leader-following consensus control method is proposed for a second-order MAS with unknown nonlinear dynamics. This technique is also extended to high-order canonical MASs [21] with sliding-mode control approaches.

Although significant progress has been made in areas related to cyberattacks, privacy preservation using the Liu cryptosystem, and the RL technique, several technical gaps still exist. First, most studies on cyberattack observers focus on linear MASs [8, 9]. However, nonlinear MASs predominate in real-world situations, where nonlinear dynamics are unknown. In these cases, cyberattack observers for linear MASs are infeasible, requiring the construction of nonlinear dynamics. There exists a lack of relevant processing techniques for nonlinear dynamics. Also, existing attack observers [8,9] do not utilize privacy-preserving information. Thus, the construction of cyberattack detectors in the presence of potential encryption errors and noise is a challenge. Second, the existing Liu cryptosystem [16] utilizes numerous random sequences to create a ciphertext. Although these random numbers enhance privacy, they introduce decryption errors between the plaintext and the decrypted plaintext. There exists a lack of theoretical improvement for mitigating decryption errors caused by random numbers, and further technical advancements are thus necessary. Third, most existing RL-based control schemes [17–21] and cyberattack observers utilize neural networks (NNs) to approximate unknown dynamics. Along with the increase in the number of nodes in hidden layers, the number of weights increases accordingly for a single agent. More seriously, as the number of agents increases, these quantities grow dramatically. This entails numerous parameter updates, resulting in computational burdens.

To address these concerns, for a nonlinear MAS, a privacy-preserving RL-based consensus control strategy is proposed to ensure that normal agents achieve consensus. A finite-time attack detector with a privacy-preserving signal is proposed to detect and isolate compromised agents. The minimal number of learning parameters (MNLPs) technique is utilized to reduce the number of update parameters of NNs. The specific technical contributions of this paper are summarized below.

(1) A finite-time attack detector is proposed for a nonlinear MAS with privacy-preserving signals. In the exchange of privacy-preserving information, there exist errors between encryption and decryption. Simultaneously, external noises impact sensor measurements. Both these errors and noises are considered part of the detection threshold for the proposed attack detector. If the detected error significantly exceeds this threshold, it can be inferred that external attacks have paralyzed agents. Compared with attack detectors only for linear MASs [7], there exist unknown dynamics, as well as noises and errors between encryption and decryption, in the attack detector for nonlinear MASs of this paper.

(2) An improved Liu cryptosystem is proposed for privacy preservation. Signal amplification is implemented to reduce encryption/decryption errors. Unlike the existing Liu public key cryptosystem [16], an amplification operation is performed herein on the plaintext before the encryption, increasing its proportion in the ciphertext. According to the decryption algorithm, the decrypted plaintext consists of two parts: the original plaintext and a random number and public key 0. The plaintext amplification results in a reduced proportion of components consisting of the random number and public key 0. Consequently, by mitigating the impact of errors between encryption and decryption on the plaintext, plaintext information is satisfactorily recovered.

(3) The MNLP technique is applied in both the proposed attack detector and consensus control strategy. By learning the norm of NN weights, this technique reduces the number of learning parameters of NN for approximating unknown dynamics in the attack detector and RL-based control strategy and results in significantly fewer learning parameters compared with existing methods [7, 17–21].

**Notations.** For a set $A$, its elements may be values, structures, or agents. $|A|$ is the cardinality of the set. For a real value $B$, $|B|$ is the absolute value of $B$. For a matrix $C$, $||C||$ denotes its two-norm. For two sets $X$ and $Y$, $X\backslash Y$ represents that the difference set between $X$ and $Y$. For a set $X$ and an element $y$, $X\backslash y$ indicates that the set $X$ removes the element $y$.

## 2 Problem formulation and preliminaries

### 2.1 Graph theory

A graph is introduced to describe the information flow among agents. Consider a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$, where $\mathcal{V} = \{1, \ldots, N\}$ is a nonempty set consisting of $N$ agents, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, and $\mathcal{A} = \left[a_{i,j}\right] \in \mathbb{R}^{N\times N}$ is an adjacency matrix. Edge $(j, i) \in \mathcal{E}$ stands for an edge from Agent $j$ to Agent $i$. If Agent $j$ can receive information from $i$, $a_{i,j} > 0$; otherwise, $a_{i,j} = 0$. A self-loop is not considered. That is, $a_{i,i} = 0$. $\mathcal{D} = \mathrm{diag}\{d_1, \ldots, d_N\}$ is a diagonal matrix, where $d_i = \sum_{j=1}^{N} a_{i,j}$. The Laplacian matrix of $\mathcal{G}$ is $\mathcal{L} = \mathcal{D} - \mathcal{A}$. For Agent $i$, the set of all its neighbors is defined as $\mathcal{N}_i = \{\mathcal{N}_{i^1}, \mathcal{N}_{i^2}, \ldots, \mathcal{N}_{i^j}\}$. $\mathcal{N}_i^* = \{k \in \mathcal{V} | (i, j) \in \mathcal{E}, (j, k) \in \mathcal{E}\}$ represents a set of two-hop neighbors of Agent $i$. $\mathcal{J}_i = \mathcal{N}_i \cap \{i\}$ is a set of direct neighbors of Agent $i$ and Agent $i$ itself. $\mathcal{J}_i^* = \mathcal{N}_i \cup \mathcal{N}_i^*$ represents a set comprising both direct neighbors and two-hop neighbors of Agent $i$. Here, $\mathcal{G}_i(\mathcal{J}_i, \mathcal{E}_i)$ with $\mathcal{E}_i = \{(i, j) \in \mathcal{E} | i, j \in \mathcal{J}_i\}$ is the subgraph induced by $\mathcal{J}_i$, and $\mathcal{G}_i^*(\mathcal{J}_i^*, \mathcal{E}_i^*)$ with $\mathcal{E}_i^* = \{(i, j) \in \mathcal{E}, (j, k) \in \mathcal{E} | i, j, k \in \mathcal{J}_i^*\}$ is the one induced by $\mathcal{J}_i^*$. A graph $\mathcal{G}$ is connected if there is a path between every pair of agents. In a graph $\mathcal{G}$, a circle is a path that starts and ends at the same agent. A ring graph $\mathcal{G}$ contains a single circle through all agents.

Intuitively, a graph plays a key role in counteracting influence from attack agents, such as isolating paralyzed ones. To seek a sufficient condition for isolation, a graph-theoretic definition for isolability is introduced.

**Definition 1** ([8]). (Agent isolability) In a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$, Agent $i$ is deemed isolable if $\mathcal{J}_i \cap \mathcal{J}_{i^1} \cap \mathcal{J}_{i^2} \cap \cdots \cap \mathcal{J}_{i^{\mathcal{N}_i}} = i$.

A graph $\mathcal{G}$ is said to be 1-isolable if all agents contained in $\mathcal{G}$ satisfy Definition 1.

### 2.2 Problem formulation

Consider an MAS consisting of $N$ agents with the $n_i$-order form. The dynamics of the $i$th agent are

$$\begin{cases} \dot{x}_{i,1} = f_{i,1}(\bar{\boldsymbol{x}}_{i,n_i}) + g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i}) x_{i,2}, \\ \dot{x}_{i,2} = f_{i,2}(\bar{\boldsymbol{x}}_{i,n_i}) + g_{i,2}(\bar{\boldsymbol{x}}_{i,n_i}) x_{i,3}, \\ \qquad\qquad \vdots \\ \dot{x}_{i,n_i} = f_{i,n_i}(\bar{\boldsymbol{x}}_{i,n_i}) + g_{i,n_i}(\bar{\boldsymbol{x}}_{i,n_i}) u_i, \\ y_i = x_{i,1}, \quad i = 1, \ldots, N, \end{cases} \tag{1}$$

where $x_{i,l} \in \mathbb{R}$ is a state variable with $l = 1, \ldots, n_i$, $\bar{\boldsymbol{x}}_{i,n_i} = \left[x_{i,1}, x_{i,2}, \ldots, x_{i,n_i}\right]^{\mathrm{T}} \in \mathbb{R}^{n_i}$ represents a state vector, $y_i \in \mathbb{R}$ is the agent's output, $u_i \in \mathbb{R}$ is its input signal, $f_{i,l}(\bar{\boldsymbol{x}}_{i,n_i})$ is an unknown and smooth function denoting uncertain dynamics, and $g_{i,l}(\bar{\boldsymbol{x}}_{i,n_i})$ is a nonzero gain. In an actual and complex scenario, external noises often affect agents' information, and the output signal $y_i$ becomes $y_i^{s\Delta} = y_i + \Delta_i$ in absence of attacks, where $\Delta_i$ is a noise signal and $i \in \mathcal{V}$.

**Assumption 1.** The nonzero gain $g_{i,l}(\bar{\boldsymbol{x}}_{i,n_i})$ satisfies $0 < \underline{g}_{i,l} \leqslant |g_{i,l}(\bar{\boldsymbol{x}}_{i,n_i})| \leqslant \bar{g}_{i,l}$, and $|\dot{g}_{i,l}(\bar{\boldsymbol{x}}_{i,n_i})| \leqslant \breve{g}_{i,l}$, where $\underline{g}_{i,l}$, $\bar{g}_{i,l}$, and $\breve{g}_{i,l}$ are unknown positive constants.

**Remark 1.** For the continuous unknown control gain $g_{i,l}$, there exist three scenarios: (a) $g_{i,l}$ is always greater than 0; (b) $g_{i,l}$ is always less than 0; and (c) $g_{i,l}$ crosses the horizontal axis. Because $g_{i,l}$ is continuous, there exists $g_{i,l} = 0$ in (c). At this point, the control law $u_i$ is infeasible. To avoid such situations, we should exclude the singular point $g_{i,l} = 0$. Therefore, we need to assume that the unknown control gain is strictly positive or strictly negative. Many practical system models satisfy this assumption, such as robotic manipulator systems, surface vessels, aircraft, and microgrid systems. This assumption can also be found in [22].

**Assumption 2.** $\mathcal{G}$ is a ring graph.

**Remark 2.** A ring graph ensures that the remaining graph remains connected even after isolating agents [8] under attack. Ring graphs are widely used in industrial systems, such as transmission networks of telecom carriers [23] and unmanned aerial vehicles [24].

As illustrated previously [25], malicious adversaries inject false data into sensors. For clarification, we give the following definition to describe sensor attacks.

**Definition 2** ([8]). (1-weak-local attack) If at most 1 compromised agent is contained in the direct neighbors and two-hop neighbors of normal Agent $i$ in $\mathcal{F}$, (i.e., $|\mathcal{J}_i^* \cap \mathcal{F}| \leqslant 1$, $\forall i \in \mathcal{T}$), the set $\mathcal{F} \subseteq \mathcal{V}$ is 1-weak-local attacked, where $\mathcal{F}$ is the set of compromised agents and $\mathcal{T} = \mathcal{V} \backslash \mathcal{F}$ is the set of normal agents.

Consequently, messages from sensors may be paralyzed. As an agent is compromised by sensor attacks, its output signal becomes

$$y_i^s = y_i + \delta_i + \Delta_i, \quad i \in \mathcal{F}, \tag{2}$$

where $\delta_i$ is a signal from the sensor attacks.

**Assumption 3.** The noise signal $\Delta_i$ satisfies $|\Delta_i| \leqslant \bar{\Delta}_i$, and $|\dot{\Delta}_i| \leqslant \breve{\Delta}_i$, where $\bar{\Delta}_i$ and $\breve{\Delta}_i$ are unknown positive constants.

**Remark 3.** In signal processing, noise is a general term for unwanted/unknown modifications that a signal may suffer during capture, storage, transmission, processing, or conversion [26]. Gaussian white noise commonly appears in [27]. A cyberattack is a malicious and deliberate attempt by an individual or organization to breach the information system of another individual or organization. In this respect, there exist methods for compensation of attacks [28].

The exchange of sensitive information is susceptible to either attacks or theft from both malicious eavesdroppers and adversaries, thereby adversely affecting consensus performance and security. This paper primarily focuses on addressing the consensus issue in the presence of both adversaries and eavesdroppers. Agents execute consensus control on the basis of output signals from their neighbors. The output signal $y_i$ in an MAS is exposed to public networks and is susceptible to interception.

Agent $i$ may suffer from attacks and noises, and a distributed synchronization error for Agent $i$ is thus defined as

$$e_{i,1} = \sum_{j \in \mathcal{N}_i} a_{i,j}(y_i^s - y_j^s). \tag{3}$$

Subsequently, compromised agents are isolated with a distributed attack isolation algorithm. Then, Eq. (3) becomes

$$e_{i,1} = \bar{e}_{i,1} + \sum_{j \in \bar{\mathcal{N}}_i} a_{i,j}(\Delta_i - \Delta_j), \tag{4}$$

where $\bar{\mathcal{N}}_i$ is the set of neighbors after isolating the compromised agent and $\bar{e}_{i,1} = \sum_{j \in \bar{\mathcal{N}}_i} a_{i,j}(y_i - y_j)$.

Define a vector representing the output $\boldsymbol{y} = [y_1, \ldots, y_N]^{\mathrm{T}}$, and a vector output consensus errors is denoted as $\bar{\boldsymbol{e}}_{\boldsymbol{1}} = [\bar{e}_{1,1}, \ldots, \bar{e}_{N,1}]^{\mathrm{T}}$. According to the graph theory [29], one obtains the relationship between $\bar{\boldsymbol{e}}_{\boldsymbol{1}}$ and $\boldsymbol{y}$ as

$$\bar{\boldsymbol{e}}_{\boldsymbol{1}} = \mathcal{L}\boldsymbol{y}. \tag{5}$$

Furthermore, because of the existence of a privacy-preserving technology, the information of Agent $i$'s neighbors is encrypted before being transmitted over networks, and only the decrypted information $\hat{y}_j^s$ is obtained. Then, in the design of the control strategy, only $\breve{e}_{i,1} = \sum_{j \in \bar{\mathcal{N}}_i} a_{i,j}(y_i^s - \hat{y}_j^s)$ is available.

The control objective of this paper is to develop a privacy-preserving attack-isolation-based consensus control strategy such that an MAS with (1) under Assumptions 1, 2, and 1-weak-local attacks isolates compromised agents and achieves leaderless consensus, i.e., $|y_i - y_j|$ converges to a small neighborhood around the origin, and the information of the MAS over networks is privacy-preserving.

### 2.3 Improved Liu cryptosystem

*Motivation of the introduction of the Liu cryptosystem.* Although state decomposition [12, 13] ensures precise convergence to desired values while preserving privacy, it introduces decomposed substates and increases design complexity in terms of managing multiple substate components. Differential privacy [14] adds noise to transmitted information, restricting controllers to design based on the noisy messages. This approach may result in reduced control accuracy. In many other homomorphic encryption approaches [15], the construction of public and private keys requires different algorithms. In the Liu cryptosystem, public key generation is relatively simple as it is constructed using a private key method [16], consisting of four random numbers and shared with the sender and receiver. The sender generates ciphertexts for 0 and 1 by using the private key to create a public key. Then, the sender encrypts the plaintext using the public key for transmission, and the receiver decrypts the ciphertext using the private key to obtain the plaintext.

*Principle of the Liu cryptosystem.* The Liu cryptosystem often consists of two parts: encryption and decryption algorithms. A block diagram of the Liu cryptosystem is shown in Figure 1. A private key $\boldsymbol{K}(M) = \{(q_1, s_1, t_1), \ldots, (q_M, s_M, t_M)\}$ is generated with randomly generated $q_i \in \mathbb{R}$, $s_i \in \mathbb{R}$, and $t_i \in \mathbb{R}$ ($M \geqslant 3$,

$q_i \neq 0$ $(1 \leqslant i \leqslant M-1))$, $q_M + s_M + t_M \neq 0$, and $M \in \mathbb{Z}$. The sender utilizes the Liu encryption algorithm $\mathrm{Enc}(\boldsymbol{K}(M), 0)$ and $\mathrm{Enc}(\boldsymbol{K}(M), 1)$ to encrypt the plaintext 0 and 1, resulting in a public key. Then, by using the public key, a set of ciphertexts $\bar{\boldsymbol{c}}_{m,M} = \{c_1, \ldots, c_M\} = \{mc_1^1 + \sum_{j=1}^{M} c_{1,j}^0, \ldots, mc_M^1 + \sum_{j=1}^{M} c_{M,j}^0\}$ is generated.

*Improved Liu cryptosystem.* In the Liu decryption algorithm, the decrypted plaintext $\hat{m} = (1+\varrho)m + \vartheta$, where $T = \sum_{i=1}^{M-1} t_i$, $S = \frac{c_M}{q_M + t_M + s_M}$, $\vartheta = \sum_{i=1}^{n-1} \frac{\sum_{j=1}^{M} c_{i,j}^0 - Ss_i}{Tq_i}$, and $\varrho = \sum_{i=1}^{n-1} \frac{s_i r_n + q_i(r_i - r_{i-1})}{Tq_i}$. From the study of Liu [16], $\varrho = 0$, and we further obtain $\hat{m} = m + \vartheta$. We prefer to amplify the value of $m$ to improve the decryption performance. This process is summarized in Algorithm 1, and its block diagram is shown in Figure 2. In summary, compared with existing privacy-preserving results for consensus [13, 30, 31], the improved Liu cryptosystem has the following advantages: (1) better decryption performance; (2) suitability for high-precision scenarios; and (3) support for computations for many data types.

**Algorithm 1** Improved Liu cryptosystem.

**Require:** Initialization parameters $M$, $k_p$, $T = 0$, $T_1 = 0$; Plaintext $\to m$; amplify the signal: $m_A = k_p m$;
  **Generation of the private key** $\boldsymbol{K}(M)$:
  Randomly generate $r_n$, $q_n$, $t_n$, and $s_n$, $n = 1, \ldots, M$;
  **Encryption** $(\mathrm{Enc}(\boldsymbol{K}(M), m_A))$:
  Public key 0:
  **for** $j = 1$ to $M$ **do**
    **for** $i = 1$ to $M-1$ **do**
      $c_{i,j}^0 = q_i t_i 0 + s_i r_M + q_i(r_i - r_{i-1})$;
    **end for**
    $c_{M,j}^0 = (q_M s_M t_M) r_M$;
  **end for**
  Public key 1:
  **for** $i = 1$ to $M-1$ **do**
    $c_i^1 = q_i t_i 1 + s_i r_M + q_i(r_i - r_{i-1})$;
  **end for**
  $c_M^1 = (q_M s_M t_M) r_M$;
  **for** $i = 1$ to $M$ **do**
    $c_i = m_A c_i^1 + \sum_{j=1}^{M} c_{i,j}^0$;
  **end for**
  **Decryption** $(\mathrm{Dec}(\boldsymbol{K}(M), \bar{\boldsymbol{c}}_{m_A,M}))$:
  **for** $i = 2$ to $M-1$ **do**
    $T = T + t_i$;
  **end for**
  $S = c_M/(q_M + s_M + t_M)$;
  **for** $i = 2$ to $M-1$ **do**
    $T_1 = T_1 + (c_i - Ss_i)/q_i$;
  **end for**
  $\hat{m}_A = T_1/T$;
**Output:** $\hat{m} = \hat{m}_A/k_p$.

**Remark 4.** As shown in Figures 1 and 2, the improved Liu cryptosystem adds two parts, "Multiply" and "Divide", and a series of operations slightly increases the computational burden. According to the ciphertext creation by our improved Liu cryptosystem, the ciphertexts are related to random numbers $r_M$, $q_i$, $t_i$, and $s_i$. Because the generation of random numbers is not definite, the size of the ciphertext is still ambiguous with the increased plaintext. Therefore, the network transmission burden for the ciphertext on the network is also not definite in the improved Liu cryptosystem. In fact, we pay more attention to better reproducibility with privacy preservation, and the network transmission burden is omitted in this paper.

## 3 Consensus control strategy based on distributed attack isolation

In this section, the following issues are addressed.

(1) A finite-time attack detector is designed to detect distributed synchronization errors.

(2) Detecting errors is utilized to design a resilient algorithm based on distributed attack isolation to detect and isolate compromised agents.

(3) An RL-based privacy-preserving consensus control strategy is proposed to achieve consensus for normal agents.

### 3.1 Finite-time attack detector

The finite-time attack detector estimates the distributed synchronization error (3) and assesses whether it is influenced by an attack based on the estimated error. The existing attack detectors of linear MASs [7] do not apply to

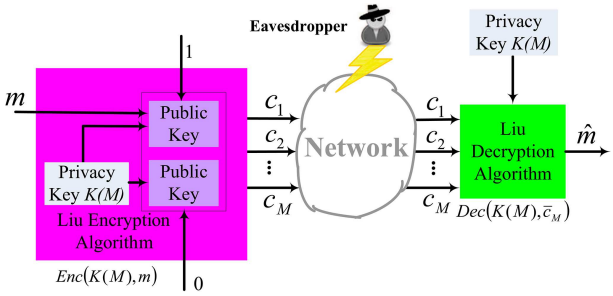


**Figure 1** (Color online) Block diagram of the Liu cryptosystem.

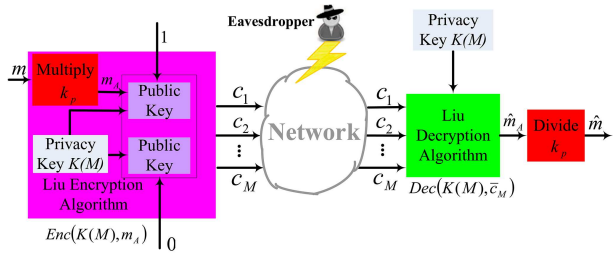


**Figure 2** (Color online) Block diagram of the improved Liu cryptosystem.

nonlinear systems because of the existence of nonlinear terms. To deal with nonlinear terms, we utilize radial basis function NNs (RBFNNs) for approximation. Moreover, in complex network environments, where agents' sensors are susceptible to external attacks, this detector accounts for two factors: external noise and the errors between encryption and decryption. Thus, we denote $\varepsilon_i = \sum_{j\in\mathcal{N}_i} a_{i,j}(y_i^s - \hat{y}_j^s) = \sum_{j\in\mathcal{N}_i} a_{i,j}(x_{i,1} + \Delta_i - x_{j,1} - \Delta_j + \hat{z}_j)$, where $\hat{y}_j^s$ is the decrypted information of Agent $j$ using the improved Liu decryption algorithm and $\hat{z}_j = y_j^s - \hat{y}_j^s$ is the error between the output information $y_j$ of the $j$th agent and a decrypted information $\hat{y}_j$. According to Subsection 2.3, the error between encryption and decryption $\vartheta = \sum_{i=1}^{n-1} \frac{\sum_{j=1}^{M} c_{i,j}^0 - Ss_i}{Tq_i}$ is considered a constant composed of randomly chosen constants. For the $i$th agent, $\hat{z}_j$ is the error between encryption and decryption of neighbors' information, and one has $\dot{\hat{z}}_j = 0$.

To precisely identify compromised agents, a finite-time attack detector using the MNLPs technique and projection operators is introduced for (1)

$$\begin{cases} \dot{\hat{\varepsilon}}_i = -\dfrac{\boldsymbol{S}_i^{\mathrm{T}}(\hbar_i)\boldsymbol{S}_i(\hbar_i)}{4\propto_i}\hat{\xi}_i\tilde{\varepsilon}_i - \hat{\bar{\varepsilon}}_i - c_i\tilde{\varepsilon}_i^{2\beta-1}, \\ \dot{\hat{\xi}}_i = \mathrm{Proj}\left(-\dfrac{\boldsymbol{S}_i^{\mathrm{T}}(\hbar_i)\boldsymbol{S}_i(\hbar_i)}{4\propto_i}\tilde{\varepsilon}_i^2 + \Gamma_i\hat{\xi}_i, \hat{\xi}_i\right), \\ \dot{\hat{\bar{\varepsilon}}}_i = \mathrm{Proj}(-\tilde{\varepsilon}_i + \bar{\sigma}_i\hat{\bar{\varepsilon}}_i, \hat{\bar{\varepsilon}}_i), \end{cases} \tag{6}$$

where $\hat{\bar{\varepsilon}}_i$ is an adaptive compensation term of NN approximation error and noise-related signal, $c_i$, $\Gamma_i$, and $\propto_i$ are positive parameters, $\mathrm{Proj}(\cdot, \hat{\xi}_i)$ and $\mathrm{Proj}(\cdot, \hat{\bar{\varepsilon}}_i)$ are projection operators [32] with

$$\mathrm{Proj}(a,b) = \begin{cases} a, & \text{if } p(b) \leqslant 0, \\ a, & \text{if } p(b) \geqslant 0 \text{ or } (\partial p/\partial b)\times a \leqslant 0, \\ a - (p(b)\partial p/\partial b \times a)/(\parallel \partial p/\partial b \parallel^2)(\partial p/\partial b)^{\mathrm{T}}, & \text{otherwise}, \end{cases}$$

where $p(b) = (b^{\mathrm{T}}b - r_{\hat{\xi}_i})/(\delta_{\hat{\xi}_i}^2 + 2\delta_{\hat{\xi}_i} r_{\hat{\xi}_i})$, $r_{\hat{\xi}_i}$ is a known radius of the closed sphere of the projection operator, and $\delta_{\hat{\xi}_i}$ is a small positive constant.

Denote $\tilde{\varepsilon}_i = \hat{\varepsilon}_i - \varepsilon_i$. From (1) and (2), we have

$$\dot{\tilde{\varepsilon}}_i = \dot{\hat{\varepsilon}}_i - F_i - \sum_{j\in\mathcal{N}_i} a_{i,j}(\dot{\Delta}_i - \dot{\Delta}_j), \tag{7}$$

where $F_i = \sum_{j\in\mathcal{N}_i} a_{i,j}(f_{i,1}(\bar{\boldsymbol{x}}_{i,n_i}) + g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})x_{i,2} - f_{j,1}(\bar{\boldsymbol{x}}_{j,n_j}) + g_{j,1}(\bar{\boldsymbol{x}}_{j,n_j})x_{j,2})$. Because $F_i$ is unattainable under ideal circumstances, we approximate $F_i = \boldsymbol{\theta}_i^{*\mathrm{T}}\boldsymbol{S}_i(\boldsymbol{Z}_i) + \epsilon_i$ using RBFNN, where $\boldsymbol{\theta}_i^* = \left[\theta_{i,1}^*, \ldots, \theta_{i,\gamma_i}^*\right]^{\mathrm{T}} \in \mathbb{R}^{\gamma_i}$ is the ideal weight vector, $\gamma_i$ is the number of NN nodes, $\boldsymbol{S}_i(\cdot)$ is the NN basis function vector, $\boldsymbol{Z}_i = \left[x_{i,1}, \ldots, x_{i,n_i}, x_{j,1}, \ldots, x_{j,n_j}, \sum_{j\in\mathcal{N}_i}\hat{y}_j\right]^{\mathrm{T}}$, and $\epsilon_i$ is an approximation error. From [33], $\|\boldsymbol{\theta}_i^*\| \leqslant \bar{\theta}_i$ and $|\epsilon_i| \leqslant \bar{\epsilon}_i$, where $\bar{\theta}_i$ and $\bar{\epsilon}_i$ are unknown positive constants.

Next, we present the stability of the attack detector in the absence of attacks and obtain the convergence performance of the detector error.

**Theorem 1.** Consider an MAS with agents in (1) and the detector (7). The detection error $\tilde{\varepsilon}_i$ of the attack detector arrives at a small neighborhood of the origin in a finite time $T_{i,r}$.

*Proof.* Choose a Lyapunov function candidate $V_{i,ob} = \frac{1}{2}\tilde{\varepsilon}_i^2 + \frac{1}{2}\tilde{\xi}_i^2 + \frac{1}{2}\tilde{\bar{\varepsilon}}_i^2$, where $\tilde{\xi}_i = \hat{\xi}_i - \xi_i$, $\tilde{\bar{\varepsilon}}_i = \hat{\bar{\varepsilon}}_i - \bar{\varepsilon}_i, \bar{\varepsilon}_i = \boldsymbol{\theta}_i^{*\mathrm{T}}\boldsymbol{S}_i(\boldsymbol{Z}_i) - \boldsymbol{\theta}_i^{*\mathrm{T}}\boldsymbol{S}_i(\boldsymbol{\hbar}_i) + \epsilon_i - \sum_{j\in\mathcal{N}_i} a_{i,j}(\dot{\Delta}_i - \dot{\Delta}_j)$, $\boldsymbol{\hbar}_i = \left[x_{i,1}, \sum_{j\in\mathcal{N}_i}\hat{y}_j\right]^{\mathrm{T}}$, and $\hat{\xi}_i$ is the estimation of $\xi_i = \|\boldsymbol{\theta}_i^*\|^2$. From (7) and $\dot{\hat{z}}_j = 0$, we have $\dot{V}_{i,ob} \leqslant -\frac{\boldsymbol{S}_i^{\mathrm{T}}\boldsymbol{S}_i}{4\propto_i}\xi_i\tilde{\varepsilon}_i^2 - c_i\varepsilon_i^{2p} + \propto_i \gamma_i - \tilde{\varepsilon}_i\bar{\varepsilon}_i + \tilde{\xi}_i\mathrm{Proj}(\cdot,\hat{\xi}_i) + \tilde{\bar{\varepsilon}}_i\mathrm{Proj}(\cdot,\hat{\bar{\varepsilon}}_i)$.

According to the properties of projection operators, there exist $\tilde{\xi}_i(\mathrm{Proj}(\cdot,\hat{\xi}_i) - \frac{\boldsymbol{S}_i^{\mathrm{T}}\boldsymbol{S}_i}{4\propto_i}\xi_i\tilde{\varepsilon}_i^2 + \Gamma_i\tilde{\xi}_i\hat{\xi}_i) \leqslant 0$ and $\tilde{\bar{\varepsilon}}_i(\mathrm{Proj}(\cdot,\hat{\bar{\varepsilon}}_i) - \tilde{\varepsilon}_i + \bar{\sigma}_i\tilde{\bar{\varepsilon}}_i\hat{\bar{\varepsilon}}_i) \leqslant 0$, and one has $\dot{V}_{i,ob} \leqslant -c_i\tilde{\varepsilon}_i^{2p} - \Gamma_i\tilde{\xi}_i\hat{\xi}_i - \bar{\sigma}_i\tilde{\bar{\varepsilon}}_i\hat{\bar{\varepsilon}}_i + \propto_i \gamma_i$. According to Young's inequality, Lemma 2 in [34], Lemma 3 in [35], and the properties of projection operators $|\hat{\xi}_i| \leqslant r_{\hat{\xi}_i} + \delta_{\hat{\xi}_i}$ and $|\hat{\bar{\varepsilon}}_i| \leqslant r_{\hat{\bar{\varepsilon}}_i} + \delta_{\hat{\bar{\varepsilon}}_i}$, we obtain $\dot{V}_{i,ob} \leqslant -2^\beta c_i\frac{1}{2}\tilde{\varepsilon}_i^{2\beta} - 2\Gamma_i(1 - \frac{a_i+2}{2})\frac{1}{2}\tilde{\xi}_i^{2\beta} - 2\bar{\sigma}_i(1 - \frac{b_i+2}{2})\frac{1}{2}\tilde{\bar{\varepsilon}}_i^{2\beta} + D_i$, where $D_i = (2\Gamma_i(1 - \frac{a_i+2}{2}) + 2\bar{\sigma}(1 - \frac{b_i+2}{2}))(1-\beta)\beta^{\frac{\beta}{1-\beta}} + \propto_i \gamma_i + \frac{1}{2a_i}\Gamma_i(r_{\hat{\xi}_i} + \delta_{\hat{\xi}_i}) + \frac{1}{2b_i}\bar{\sigma}_i(r_{\hat{\bar{\varepsilon}}_i} + \delta_{\hat{\bar{\varepsilon}}_i})$, $r_{\hat{\xi}_i}$ and $r_{\hat{\bar{\varepsilon}}_i}$ are known radii of the closed sphere of the projection operators, $\delta_{\hat{\xi}_i}$ and $\delta_{\hat{\bar{\varepsilon}}_i}$ are small positive constants, and $a_i$ and $b_i$ are positive constants. Here, $D_i$ thus consists of available and known parameters. According to the finite-time theorem [36], one has $\dot{V}_{i,ob}(\boldsymbol{\Pi}) \leqslant -\Lambda C_i V_{i,ob}^\beta - (1-\Lambda)C_i V_{i,ob}^\beta + D_i$, where $0 < \Lambda < 1$, $\boldsymbol{\Pi} = \left[\tilde{\varepsilon}_i, \tilde{\xi}_i, \tilde{\bar{\varepsilon}}_i\right]^{\mathrm{T}}$, and $C_i = \min(2^\beta c_i, 2\Gamma_i(1-\frac{a_i+2}{2}), 2\bar{\sigma}_i(1-\frac{b_i+2}{2}))$. This satisfies Lemma 4 in [36] such that $T_{i,r} = \frac{1}{(1-\beta)\Lambda C_i}\left[V_{i,ob}^{1-\beta}(\boldsymbol{\Pi}(0)) - (\frac{D_i}{(1-\Lambda)C_i})^{\frac{1-\beta}{\beta}}\right]$ is obtained. According to the definition of $V_{i,ob}$, for $t \geqslant T_{i,r}$, we have $\frac{1}{2}\tilde{\varepsilon}_i^2 \leqslant V_{i,ob} \leqslant (\frac{D_i}{(1-\Lambda)C_i})^{\frac{1}{\beta}}$ which yields that $\tilde{\varepsilon}_i$ is bounded. That is,

$$|\tilde{\varepsilon}_i| \leqslant \sqrt{2\left(\frac{D_i}{(1-\Lambda)C_i}\right)^{\frac{1}{\beta}}} \tag{8}$$

implies that the detection error converges to a small neighborhood of the origin within the finite time $T_{i,r}$. Also, the size of the error can be arbitrarily small from the definition of $C_i$, $D_i$, and $\beta$.

Theoretically, as an agent is not attacked, the detection error is $|\tilde{\varepsilon}_i| \leqslant \sqrt{2(\frac{D_i}{(1-\Lambda)C_i})^{\frac{1}{\beta}}}$. From a similar analysis, as agents' sensors are subjected to attacks, the boundary for the attack detector error is

$$|\tilde{\varepsilon}_i^a| \leqslant \sqrt{2\left(\frac{D_i^a}{(1-\Lambda)C_i}\right)^{\frac{1}{\beta}}}, \tag{9}$$

where $D_i^a = D_i + \frac{1}{2}\sum_{j\in\mathcal{N}_i}(\delta_i + \delta_j)^2$ and $\delta_i$ is the sensor attack. The only difference between the two detection errors is $D_i$ (without attacks) and $D_i^a$ (with an attack). Because of the existence of external attacks, $D_i^a \gg D_i$ often holds. Therefore, the attack-free detector error $\sqrt{2(\frac{D_i}{(1-\Lambda)C_i})^{\frac{1}{\beta}}}$ is set as a threshold for attack detection. These differences enable an accurate determination of whether the sensors of an agent have been attacked. The following corollary comes.

**Corollary 1.** With Assumptions 1 and 2, for an MAS in (1), there exist compromised agents if $\exists t \geqslant \tau$ s.t. $|\tilde{\varepsilon}_i(t)| > \sqrt{2(\frac{D_i}{(1-\Lambda)C_i})^{\frac{1}{\beta}}}$ with the distributed detector (9).

*Proof.* This corollary is directly proven from (8) and (9). The detailed process is omitted here.

This establishes the feasibility of the attack detection algorithm from theoretical analysis. With the detector (9), a resilient consensus algorithm is then proposed based on distributed attack isolation in the following subsection.

**Remark 5.** If the attack detector is not designed within a finite time, then the bound of the detection error $\tilde{\varepsilon}_i$ can be obtained only as $t \to \infty$. From the theoretical analysis, the task of isolating compromised agents is only completed as $t \to \infty$. This isolation method is obviously not impractical. The detection error $\tilde{\varepsilon}_i$ of the attack detector arrives at a small neighborhood of the origin within a finite time $T_{i,r}$ [36]. That is, as $t \geqslant T_{i,r}$, there exists $\frac{1}{2}\tilde{\varepsilon}_i^2 \leqslant V_{i,ob} \leqslant (\frac{D_i}{(1-\Lambda)C_i})^{\frac{1}{\beta}}$. Furthermore, a resilient consensus algorithm based on distributed attack isolation (DAI-RC) can be used to detect and correctly isolate compromised agents based on the detector error $\tilde{\varepsilon}_i$.

### 3.2 Resilient consensus algorithm based on distributed attack isolation

From the design and analysis of the attack detector in the previous subsection, the DAI-RC algorithm is proposed. Some symbols are introduced here. The state index is indicated by $\mathcal{Q}_i$, where $\mathcal{Q}_i = 1$ means that there exist compromised agents in the subnetwork $\mathcal{G}_i(\mathcal{J}_i, \mathcal{E}_i)$, whereas $\mathcal{Q}_i = 0$ means agents are compromise-free. The counter $\mathcal{W}_i$ is the sum of $\mathcal{Q}_j = 1$ with $j \in \mathcal{J}_i$. The index $\mathcal{H}_i = 1$ indicates that Agent $i$ is paralyzed, whereas $\mathcal{H}_i = 0$ represents that $i$ is attack-free. The isolation index $\mathcal{P}_i = 1$ means that Agent $i$ is isolated.

The DAI-RC algorithm is summarized in Algorithm 2, where necessary and sufficient conditions are provided for isolating compromised agents within the 1-weak-local attack.

**Algorithm 2** DAI-RC algorithm.

**Require:** Initialization parameters $\mathcal{Q}_i = 0$, $\mathcal{W}_i = 0$, $\mathcal{H}_i = 0$, $\mathcal{P}_i = 0$;
  **for** time step < simulation time step **do**
    **if** $|\tilde{\varepsilon}_i| > \sqrt{2(\frac{D_i}{(1-\Lambda)C_i})^{\frac{1}{\beta}}}$ **then**
      $\mathcal{Q}_i = 1$;
      $\mathcal{W}_i = 1$;
    **end if**
    $i$ shares $\mathcal{Q}_i$ with all its neighbors in $\mathcal{N}_i$;
    **for** $j = \mathcal{N}_{i1} : \mathcal{N}_{i|\mathcal{N}_i|}$ **do**
      **if** $\mathcal{Q}_j = 1$ **then**
        $\mathcal{W}_i = \mathcal{W}_i + 1$;
      **end if**
    **end for**
    **if** $\mathcal{W}_i = |\mathcal{J}_i|$ **then**
      $\mathcal{H}_i = 1$;
      $i$ is isolated;
      $i$ shares $\mathcal{H}_i$ with all its neighbors within $\mathcal{N}_i$;
    **end if**
    **for** $j = \mathcal{N}_{i1} : \mathcal{N}_{i|\mathcal{N}_i|}$ **do**
      **if** $\mathcal{H}_j = 1$ **then**
        $a_{i,j} = 0$;
        $\bar{\mathcal{N}}_i = \mathcal{N}_i \backslash j$;
      **end if**
      **if** $a_{i,j} = 0$ **then**
        $\mathcal{P}_j = 1$;
      **end if**
    **end for**
  **end for**
  Using $u_i$ as in Subsection 3.3.

**Theorem 2.** An MAS, described by (1) with communication topology $\mathcal{G}$, achieves attack isolation and resilient consensus under the DAI-RC algorithm. The following conditions are sufficient and necessary.

(1) $\exists t \geqslant \tau$ s.t. $|\tilde{\varepsilon}_k(t)| > \sqrt{2(\frac{D_k}{(1-\Lambda)C_k})^{\frac{1}{\beta}}}$ for all $i \in \mathcal{F}$, $k \in \mathcal{J}_i$.

(2) The graph $\mathcal{G}$ is 1-isolable.

(3) In the subgraph $\mathcal{G}'$, where $\mathcal{G}'$ represents a subgraph after the compromised agent has been isolated, all normal agents are connected after all compromised agents are removed.

*Proof.* For the 1-weak-local attack, with the distributed finite-time detector, it follows from and Corollary 1 that there exists one compromised agent in $\mathcal{G}_i^*$ if and only if $\exists t \geqslant \tau$ s.t. $|\tilde{\varepsilon}_i(t)| > \sqrt{2(\frac{D_i}{(1-\Lambda)C_i})^{\frac{1}{\beta}}}$.

(Sufficiency) Consider two cases for Conditions (1) and (2). In Case (a), at least one compromised agent remains connected to others. In Case (b), all compromised agents are isolated, but at least one innocent agent is mistakenly isolated as being attacked.

Case (a). If the compromised Agent $i$ is not in isolation, there exists $\exists t \geqslant \tau$ s.t. $|\tilde{\varepsilon}_k(t)| \leqslant \sqrt{2(\frac{D_k}{(1-\Lambda)C_k})^{\frac{1}{\beta}}}$, $k \in \mathcal{J}_i$, which goes against Condition (1).

Case (b). In the scenario where both the compromised Agent $i$ and the normal Agent $m$ are isolated, they have both been attacked within the same subnetwork $\mathcal{G}_i^*(\mathcal{J}_i^*, \mathcal{E}_i^*)$. Then, $\exists t \geqslant \tau$ s.t. $|\tilde{\varepsilon}_k(t)| > \sqrt{2(\frac{D_k}{(1-\Lambda)C_k})^{\frac{1}{\beta}}}$ for all $k \in \mathcal{J}_i \cup \mathcal{J}_m$. According to Definition 2, in $\mathcal{G}_m^*(\mathcal{J}_m^*, \mathcal{E}_m^*)$, there exists at most one compromised agent. Consequently, all neighbors of $m$ are neighbors of $i$. This goes in conflict with the definition of agent isolability.

(Necessity) Case (a). Condition (1) is not satisfied. At least one Agent $m \in \mathcal{J}_i, i \in \mathcal{F}$ satisfies $|\tilde{\varepsilon}_m(t)| > \sqrt{2(\frac{D_m}{(1-\Lambda)C_m})^{\frac{1}{\beta}}}$ for $t \geqslant \tau$, indicating that the DAI-RC algorithm fails to isolate the attacked Agent $i$.

Case (b). As $\mathcal{J}_m \subseteq \mathcal{J}_i$, denote $i$ as the compromised agent and $m$ as a normal agent. Clearly, $\exists t \geqslant \tau$ s.t. $|\tilde{\varepsilon}_k(t)| \leqslant \sqrt{2(\frac{D_k}{(1-\Lambda)C_k})^{\frac{1}{\beta}}}$, $\forall k \in \mathcal{J}_m$, which means that the DAI-RC algorithm erroneously isolates the normal Agent $m$.

Conditions (1) and (2) guarantee the feasibility of achieving attack isolation, whereas Condition (3) ensures resilient consensus among the normal agents following the removal of the compromised agents. Given Assumption 2, the subgraph $\mathcal{G}'$ after isolating the compromised agent remains connected.

**Remark 6.** The DAI-RC algorithm identifies compromised agents by assessing the judgment condition related to the error $\tilde{\varepsilon}_k, \forall k \in \mathcal{J}_i$, of the current Agent $i$ and its neighbors. To ensure the correctness of the algorithm, two conditions are provided: Condition (1) provides the basis for judgment, and Condition (2) ensures that false

judgments are avoided. If both conditions are satisfied, $|\tilde{\varepsilon}_k(t)| > \sqrt{2(\frac{D_k}{(1-\Lambda)C_k})^{\frac{1}{\beta}}}$ holds for compromised agents, as well as their neighbors, $\forall k \in \mathcal{J}_i, i \in \mathcal{F}$, thereby identifying the compromised agent. If the conditions are not simultaneously satisfied, normal agents and their neighbors may also violate the threshold, leading to misjudgment.

**Remark 7.** To ensure that the subgraph formed by normal agents remains connected after isolating suspected agents, there must be at most one compromised agent in this ring graph. For a ring graph $\mathcal{G}$ composed of $N$ agents, if one Agent $l$, $l \in \{1, 2, \ldots, N\}$ is isolated, all eigenvalues of $\mathcal{L}_{N-1}$ can also be rearranged as $0 = \lambda_1(\mathcal{L}_{N-1}) < \lambda_2(\mathcal{L}_{N-1}) \leqslant \cdots \leqslant \lambda_{N-1}(\mathcal{L}_{N-1})$. This indicates that the graph $\mathcal{G}$ is still connected after isolating Agent $l$. This analysis shows that an MAS connected via a ring graph is still connected after isolating one agent.

As Agent $i$ detects the safety index $\mathcal{H}_i$ of its neighboring agents, if it finds that the safety index $\mathcal{H}_j = 1$, then a variable $a_{i,j} = 0$, indicating that Agent $i$ takes measures to isolate itself from the compromised Agent $j$. $\bar{\mathcal{N}}_i = \mathcal{N}_i \backslash j$ represents a set of normal agents that isolate the compromised Agent $j$. Thus, the isolation index $\mathcal{P}_j = 1$. Then, normal agents utilize secure information after isolation and design $u_i$ in the following subsection.

### 3.3 RL-based privacy-preserving consensus control strategy

A DAI-RC algorithm is formulated to ensure the resilient consensus of normal agents, leveraging the concept of attack isolation. With the finite-time attack detector (9), we present an RL-based privacy-preserving consensus control strategy. Compared with existing RL control strategies [17–21], our strategy introduces a privacy-preserving mechanism to safeguard sensitive information under the premise of attack detection and isolation of the attacked agents. Also, to alleviate the complexity of NN computation, we incorporate the MNLPs technique.

For an agent in information reception, an encrypted signal must be decrypted to obtain the true value. The distributed synchronization error (4) is thus represented as $\breve{e}_{i,1}(t) = \sum_{j\in\bar{\mathcal{N}}_i} a_{i,j}(y_i^s(t) - \hat{y}_j^s(t))$. According to [37], the long-term performance index can be prescribed by $J_i(t) = \int_t^\infty \gamma^{\frac{-\zeta+t}{T}} p(\breve{e}_{i,1}(\zeta))d\zeta$, where $\gamma \in (0,1)$ represents a discount factor that decreases the significance of the cost incurred over longer time horizons, $T > 0$ stands for an integral reinforcement interval,

$$p_i(\breve{e}_{i,1}(\zeta)) = \begin{cases} 0, & \text{if } \breve{e}_{i,1}(t)^2 \leqslant c_{p_i}, \\ 1, & \text{if } \breve{e}_{i,1}(t)^2 > c_{p_i}, \end{cases}$$

and $c_{p_i}$ is a small constant related to performance. $J_i(t)$ is currently unavailable, and a critic NN is introduced to approximate $J_i(t) = \boldsymbol{W}_{i,c}^{*\mathrm{T}} \boldsymbol{\phi}_{i,c}(\boldsymbol{Z}_{i,c}(t)) + \varepsilon(t)$, where $\boldsymbol{W}_{i,c}^{*\mathrm{T}} \in \mathbb{R}^{\iota_{i,c}}$ is the ideal weight, $\iota_{i,c}$ is the number of NN nodes, $x_{i,c}$ is the input vector of the critic NN, $\boldsymbol{\phi}_{i,c}(\boldsymbol{Z}_{i,c})$ is the NN basis function vector, and $\varepsilon_{i,c}$ is the approximate error. From [33], $\|\boldsymbol{W}_{i,c}^*\| \leqslant b_{\boldsymbol{W}_{i,c}}$, $\|\boldsymbol{\phi}_{i,c}(\boldsymbol{Z}_{i,c})\| \leqslant b_{\phi_{i,c}}$ and $|\varepsilon_{i,c}| \leqslant b_{\varepsilon_{i,c}}$, where $b_{\boldsymbol{W}_{i,c}}$, $b_{\boldsymbol{\phi}_{i,c}}$ and $b_{\varepsilon_{i,c}}$ are unknown positive constants.

The following properties are necessary for the design of the control strategy. Because the ideal weight $\boldsymbol{W}_{i,c}^*$ is an $\iota_{i,c}$-dimensional vector, its high dimension results in heavy computational burdens. To address this problem, an MNLPs technique is introduced. In the MNLPs technique, an unknown parameter $\lambda_{i,c}$ is introduced, which is specified as $\lambda_{i,c}^* = \max_{l=1,\ldots,\iota_{i,c}}\{\|\boldsymbol{W}_{i,c,l}^*\|\}$. Consequently, an MNLPs-based long-term strategic utility function is given as $J_i(t) = \lambda_{i,c}^* \|\boldsymbol{\phi}_{i,c}(\boldsymbol{Z}_{i,c}(t))\| + \varepsilon(t)$.

In general, the ideal weight $\lambda_{i,c}^*$ is unknown; then, the current weight is used to approximate $J_i(t)$ in real time, $\hat{J}_i(t) = \hat{\lambda}_{i,c}\|\boldsymbol{\phi}_{i,c}(\boldsymbol{Z}_{i,c}(t))\|$, and $J_i(t-T)$ is approximated by $\hat{J}_i(t-T) = \hat{\lambda}_{i,c}\|\boldsymbol{\phi}_{i,c}(\boldsymbol{Z}_{i,c}(t-T))\|$. Then, the temporal difference error is denoted as

$$E_{i,c} = \hat{J}_i(t) - \gamma\hat{J}_i(t-T) + p_{i,c} = \tilde{\lambda}_{i,c}\|\Delta\boldsymbol{\phi}_{i,c}(t)\| + p_{i,c} + \lambda_{i,c}^*\|\Delta\boldsymbol{\phi}_{i,c}(t)\|, \tag{10}$$

where $\Delta\boldsymbol{\phi}_{i,c}(t) = \boldsymbol{\phi}_{i,c}(\boldsymbol{Z}_{i,c}(t)) - \gamma\boldsymbol{\phi}_{i,c}(\boldsymbol{Z}_{i,c}(t-T))$ and $\tilde{\lambda}_{i,c} = \hat{\lambda}_{i,c} - \lambda_{i,c}^*$. Clearly, $\|\Delta\boldsymbol{\phi}_{i,c}(t)\| \leqslant (1+\gamma)b_{\boldsymbol{\phi}_{i,c}}$. Then, $\hat{\lambda}_{i,c}$ is updated according to

$$\dot{\hat{\lambda}}_{i,c} = -\Gamma_{i,c}\|\Delta\boldsymbol{\phi}_{i,c}(t)\|[\hat{\lambda}_{i,c}\|\Delta\boldsymbol{\phi}_{i,c}(t)\| + p_{i,c}] - \sigma_{i,c}\Gamma_{i,c}\hat{\lambda}_{i,c}, \tag{11}$$

where $\Gamma_{i,c} > 0$, $\sigma_{i,c} > 0$,

$$p_{i,c} = \int_{t-T}^{t} \gamma^{\frac{-\zeta+t}{T}} p_i(\breve{e}_{i,1}(\zeta))d\zeta = \begin{cases} 0, & \text{if } \breve{e}_{i,1}(t)^2 \leqslant c_{p_i}, \\ \dfrac{T}{\ln\gamma}(\gamma-1), & \text{if } \breve{e}_{i,1}(t)^2 > c_{p_i}, \end{cases}$$

$|p_{i,c}| \leqslant b_{p_{i,c}}$, and $b_{p_{i,c}}$ is a positive constant.

In the following, an RL-based privacy-preserving consensus control strategy is given for (1).

**Step 1.** From the definition of $e_{i,1}$ in (3) and the dynamics (1), we obtain

$$\dot{e}_{i,1}=\sum_{j\in\bar{\mathcal{N}}_i}a_{i,j}g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})\left[\frac{f_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})}{g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})}+\frac{\dot{\Delta}_i}{g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})}-\frac{\dot{y}_j^s}{g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})}+y_j^s-y_j^s+x_{i,2}\right]. \tag{12}$$

According to the results in Subsection 2.3, for better information security, transmitted signals are encrypted using the improved Liu encryption algorithm. Upon receiving the ciphertext, the signal is decrypted using the improved Liu decryption algorithm to obtain the desired message. There exists $\hat{z}_j$ between the true signal and the decrypted one. In fact, $\hat{z}_j$ is unknown and unpredictable for Agent $i$. These errors are treated as unknown variables and are composed into unknown functions. Denote

$$\psi_{i,1}=\frac{f_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})}{g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})}-\frac{\dot{y}_j^s}{g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})}+y_j^s. \tag{13}$$

According to (12), Eq. (13) is written as $\psi_{i,1}=\frac{f_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})}{g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})}-\frac{\dot{\hat{y}}_j^s+\dot{\hat{z}}_j}{g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})}+\hat{y}_j^s+\hat{z}_j$.

The unknown function $\psi_{i,1}$ is approximated by an actor NN $\psi_{i,1}=\lambda_{i,1}^*\|\boldsymbol{\phi}_{i,1}(\boldsymbol{Z}_{i,1})\|+\varepsilon_{i,1}$, where $\lambda_{i,1}=\max_{l=1,\ldots,\iota_{i,1}}\{\|\boldsymbol{W}_{i,1,l}^*\|\}$ is the ideal weight, $\iota_{i,1}$ is the number of NN nodes, $\boldsymbol{\phi}_{i,1}(\cdot)$ is the basis function vector, $\varepsilon_{i,1}$ is the approximate error, and $\boldsymbol{Z}_{i,1}=\left[x_{i,1},\ldots,x_{i,n_i},\sum_{j\in\bar{\mathcal{N}}_i}\hat{y}_j^s\right]^{\mathrm{T}}$. Then, the error (12) becomes

$$\begin{aligned}\dot{e}_{i,1}=\sum_{j\in\bar{\mathcal{N}}_i}a_{i,j}g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})\Bigg[&\lambda_{i,1}^*\|\boldsymbol{\phi}_{i,1}(\boldsymbol{Z}_{i,1})\|+\lambda_{i,1}^*\|\boldsymbol{\phi}_{i,1}(\hbar_{i,1})\|+\varepsilon_{i,1}-\lambda_{i,1}^*\|\boldsymbol{\phi}_{i,1}(\hbar_{i,1})\|-\hat{y}_j^s\\&+\hat{z}_j+x_{i,2}+\frac{\dot{\Delta}_i}{g_{i,1}(\bar{\boldsymbol{x}}_{i,n_i})}\Bigg],\end{aligned} \tag{14}$$

where $\hbar_{i,1}=\left[x_{i,1},\sum_{j\in\bar{\mathcal{N}}_i}\hat{y}_j^s\right]^{\mathrm{T}}$.

Design the virtual control law for this step

$$\alpha_{i,2}=-\hat{\lambda}_{i,1}\|\boldsymbol{\phi}_{i,1}(\hbar_{i,1})\|+\hat{y}_j^s-k_{i,1}\breve{e}_{i,1} \tag{15}$$

with an adaptive update law

$$\dot{\hat{\lambda}}_{i,1}=-\Gamma_{i,1}\|\boldsymbol{\phi}_{i,1}\|[\hat{\lambda}_{i,c}\|\boldsymbol{\phi}_{i,c}\|+\breve{e}_{i,1}]-\sigma_{i,1}\Gamma_{i,1}\hat{\lambda}_{i,1}, \tag{16}$$

where $k_{i,1}$ is a positive gain, $\Gamma_{i,1}>0$ is a learning rate coefficient, $\sigma_{i,1}>0$ is a small coefficient to be prescribed, and $\hat{\lambda}_{i,1}$ is the estimation of $\lambda_{i,1}^*$, with $\tilde{\lambda}_{i,1}=\hat{\lambda}_{i,1}-\lambda_{i,1}^*$. Introduce a new state variable $\kappa_{i,2}$, and let $\alpha_{i,2}$ pass through a first-order filter with a time constant $\epsilon_{i,2}$ to obtain $\kappa_{i,2}$:

$$\epsilon_{i,2}\dot{\kappa}_{i,2}+\kappa_{i,2}=\alpha_{i,2},\quad \kappa_{i,2}(0)=\alpha_{i,2}(0). \tag{17}$$

**Remark 8.** Without considering privacy-preserving scenarios, the control law $\alpha_{i,2}$ becomes $\alpha_{i,2}=-\hat{\lambda}_{i,1}\|\boldsymbol{\phi}_{i,1}(\hbar_{i,1})\|+y_j^s-k_{i,1}e_{i,1}$ with the adaptive update law $\dot{\hat{\lambda}}_{i,1}=-\Gamma_{i,1}\|\boldsymbol{\phi}_{i,1}\|[\hat{\lambda}_{i,c}\|\boldsymbol{\phi}_{i,c}\|+e_{i,1}]-\sigma_{i,1}\Gamma_{i,1}\hat{\lambda}_{i,1}$. For the control law, Eq. (15) utilizes decrypted information $\hat{y}_j^s$ and $\breve{e}_{i,1}$, whereas Eq. (16) utilizes decrypted information $\breve{e}_{i,1}$ for the adaptive update law. Privacy-preserving methods ensure the confidentiality and integrity of control inputs, thereby preventing unauthorized access and tampering. In contrast, the design of the control law and adaptive update law based on real information may be more susceptible to the leakage of sensitive information.

**Step** $l$ $(2\leqslant l\leqslant n_i-1)$**.** Define the $l$th error as $e_{i,l}=x_{i,l}-\kappa_{i,l}$, where $\kappa_{i,l}$ is the state variable at Step $l-1$. With the dynamics of the agent in (1), similarly denote $\psi_{i,l}=\frac{f_{i,l}(\bar{\boldsymbol{x}}_{i,n_i})}{g_{i,l}(\bar{\boldsymbol{x}}_{i,n_i})}-\frac{\dot{\kappa}_{i,l}}{g_{i,l}(\bar{\boldsymbol{x}}_{i,n_i})}+\dot{\kappa}_{i,l}$. We employ the NN approximation technique and, in contrast to [33], substitute the neighbor signals with the information generated by the improved Liu cryptosystem. Then, the $l$th error yields

$$\dot{e}_{i,l}=g_{i,l}(\bar{\boldsymbol{x}}_{i,n_i})\left[\lambda_{i,l}^*\|\boldsymbol{\phi}_{i,l}(\boldsymbol{Z}_{i,l})\|+\lambda_{i,l}^*\|\boldsymbol{\phi}_{i,l}(\hbar_{i,l})\|-\lambda_{i,l}^*\|\boldsymbol{\phi}_{i,l}(\hbar_{i,l})\|+\varepsilon_{i,l}+x_{i,l+1}-\dot{\kappa}_{i,l}\right], \tag{18}$$

where $\boldsymbol{\hbar}_{i,l} = \left[x_{i,1}, \ldots, x_{i,l}, \sum_{j\in\mathcal{N}_i} \hat{y}_j^s\right]^{\mathrm{T}}$ and $\boldsymbol{Z}_{i,l} = \left[x_{i,1}, \ldots, x_{i,n_i}, \sum_{j\in\mathcal{N}_i} \hat{y}_j\right]^{\mathrm{T}}$. To stabilize $e_{i,l}$, we design a virtual control law at this step

$$\alpha_{i,l+1} = -\hat{\lambda}_{i,l}\|\boldsymbol{\phi}_{i,l}(\boldsymbol{\hbar}_{i,l})\| + \dot{\kappa}_{i,l} - k_{i,l}e_{i,l} \tag{19}$$

with an adaptive update law $\dot{\hat{\lambda}}_{i,l} = -\Gamma_{i,l}\|\boldsymbol{\phi}_{i,l}\|[\hat{\lambda}_{i,c}\|\boldsymbol{\phi}_{i,c}\| + e_{i,l}] - \sigma_{i,l}\Gamma_{i,l}\hat{\lambda}_{i,l}$, where $k_{i,l}$ is a positive gain, $\Gamma_{i,l} > 0$ is a learning rate coefficient to be designed, $\sigma_{i,l} > 0$ is a small coefficient to be prescribed, $\hat{\lambda}_{i,l}$ is the estimation of $\lambda_{i,l}^*$ with $\tilde{\lambda}_{i,l} = \hat{\lambda}_{i,l} - \lambda_{i,l}^*$, and $\kappa_{i,l+1}$ is from a first-order filter $\epsilon_{i,l+1}\dot{\kappa}_{i,l+1} + \kappa_{i,l+1} = \alpha_{i,l+1}$ with a time constant $\epsilon_{i,l+1}$ and $\kappa_{i,l+1}(0) = \alpha_{i,l+1}(0)$.

**Step $n_i$.** Define the $n_i$th error variable $e_{i,n_i} = x_{i,n_i} - \kappa_{i,n_i}$. Then, this error becomes

$$\dot{e}_{i,n_i} = g_{i,n_i}(\bar{\boldsymbol{x}}_{i,n_i})\left[\frac{f_{i,n_i}(\bar{\boldsymbol{x}}_{i,n_i})}{g_{i,n_i}(\bar{\boldsymbol{x}}_{i,n_i})} - \frac{\dot{\kappa}_{i,n_i}}{g_{i,n_i}(\bar{\boldsymbol{x}}_{i,n_i})} + \dot{\kappa}_{i,n_i} + u_i - \dot{\kappa}_{i,n_i}\right], \tag{20}$$

where $\kappa_{i,n_i}$ is the state variable at Step $n_i - 1$.

In (20), denote an unknown function $\psi_{i,n_i} = \frac{f_{i,n_i}(\bar{\boldsymbol{x}}_{i,n_i})}{g_{i,n_i}(\bar{\boldsymbol{x}}_{i,n_i})} - \frac{\dot{\kappa}_{i,n_i}}{g_{i,n_i}(\bar{\boldsymbol{x}}_{i,n_i})} + \dot{\kappa}_{i,n_i}$. An actor NN is employed to approximate this unknown function $\psi_{i,n_i} = \lambda_{i,n_i}^*\|\boldsymbol{\phi}_{i,n_i}(\boldsymbol{Z}_{i,n_i})\| + \varepsilon_{i,n_i}$, where $\boldsymbol{Z}_{i,n_i} = \left[x_{i,1}, \ldots, x_{i,n_i}, \sum_{j\in\bar{\mathcal{N}}_i} \hat{y}_j^s\right]^{\mathrm{T}}$. Then, the final control law is designed

$$u_i = -\hat{\lambda}_{i,n_i}\|\boldsymbol{\phi}_{i,n_i}(\boldsymbol{\hbar}_{i,n_i})\| + \dot{\kappa}_{i,n_i} - k_{i,n_i}e_{i,n_i} \tag{21}$$

with an adaptive update law

$$\dot{\hat{\lambda}}_{i,n_i} = -\Gamma_{i,n_i}\|\boldsymbol{\phi}_{i,n_i}\|[\hat{\lambda}_{i,c}\|\boldsymbol{\phi}_{i,c}\| + e_{i,n_i}] - \sigma_{i,n_i}\Gamma_{i,n_i}\hat{\lambda}_{i,n_i}, \tag{22}$$

where $k_{i,n_i}$ is a positive gain, $\Gamma_{i,n_i} > 0$ is a learning rate coefficient, $\sigma_{i,n_i} > 0$ is a small coefficient to be prescribed, and $\hat{\lambda}_{i,n_i}$ is the estimation of $\lambda_{i,n_i}^*$ with $\tilde{\lambda}_{i,n_i} = \hat{\lambda}_{i,n_i} - \lambda_{i,n_i}^*$.

The block diagram of the overall MAS is shown in Figure 3. Meanwhile, the diagram of the proposed strategy for the $i$th agent is shown in Figure 4. It consists of encryption/decryption, attack detection and isolation, critic NNs, and actor NNs.

## 4 Stability analysis

In this section, a stability analysis is presented. The main result of this paper is summarized in Theorem 3.

**Theorem 3.** For a nonlinear MAS that isolates attacked agents using the DAI-RC algorithm under Assumptions 1 and 2, the RL-based adaptive consensus control strategy with the privacy-preserving method consists of virtual control laws (15) and (19), update laws (11), (16), and (22), and control law (21). If the design parameters satisfy $\Gamma_{i,c} > 0$, $\sigma_{i,c} > 0$, $\Gamma_{i,l} > 0$, $\sigma_{i,l} > b_{\boldsymbol{\phi}_{i,l}}b_{\boldsymbol{\phi}_{i,c}}$, $k_{i,l} > 0$, $0 < l_{i,c} < l_{i,v}\sigma_{i,c} - \sum_{l=1}^{n_i}\frac{b_{\boldsymbol{\phi}_{i,l}}b_{\boldsymbol{\phi}_{i,c}}}{2}$, and $0 < \varphi_{i,l} < \frac{\sigma_{i,l}}{2} - \frac{b_{\boldsymbol{\phi}_{i,l}}b_{\boldsymbol{\phi}_{i,c}}}{2}, 1 \leqslant l \leqslant n_i$, the RL-based control strategy with the privacy-preserving method ensures that all signals in the closed-loop system are ultimately bounded and that the output consensus error of the MAS converges to a neighborhood around the origin.

*Proof.* Choose a Lyapunov function

$$V_i = V_{i,1} + V_{i,2} + V_{i,3} + V_{i,4}, \tag{23}$$

where $V_{i,1} = \frac{l_{i,v}\Gamma_{i,c}^{-1}}{2}\tilde{\lambda}_{i,c}^2$, $V_{i,2} = \frac{1}{2\sum_{j\in\mathcal{N}_i}a_{i,j}g_{i,1}}e_{i,1}^2 + \frac{\Gamma_{i,1}^{-1}}{2}\tilde{\lambda}_{i,1}^2 + \frac{1}{2}\chi_{i,2}^2, \chi_{i,2} = \kappa_{i,2} - \alpha_{i,2}$, $V_{i,3} = \frac{1}{2g_{i,l}}\sum_{l=2}^{n_i-1}e_{i,l}^2 + \frac{\Gamma_{i,1}^{-1}}{2}\sum_{l=2}^{n_i-1}\tilde{\lambda}_{i,l}^2 + \sum_{l=2}^{n_i-1}\frac{1}{2}\chi_{i,l+1}^2, \chi_{i,l+1} = \kappa_{i,l+1} - \alpha_{i,l+1}$ $(2 \leqslant l \leqslant n_i - 1)$, and $V_{i,4} = \frac{1}{2g_{i,n_i}}e_{i,n_i}^2 + \frac{\Gamma_{i,n_i}^{-1}}{2}\tilde{\lambda}_{i,n_i}^2$.

With Young's inequality and the inequality $0 < \boldsymbol{\phi}_{i,l}^{\mathrm{T}}(\cdot)\boldsymbol{\phi}_{i,l}(\cdot) \leqslant \iota_{i,l}$ from the property of RBFNNs, we obtain $\boldsymbol{W}_{i,l}^{\mathrm{T}}\boldsymbol{\phi}_{i,l}(Z_{i,l}) - \lambda_{i,l}\|\boldsymbol{\phi}_{i,l}(\hbar_{i,l})\| \leqslant b_{\boldsymbol{W}_{i,l}}^2 + \iota_{i,l}$, where $l = 1, \ldots, n_i$. From (11), the derivative of $V_{i,1}$ yields

$$\dot{V}_{i,1} \leqslant -l_{i,v}\sigma_{i,c}\tilde{\lambda}_{i,c}^2 + b_{V_{i,1}}|\tilde{\lambda}_{i,c}|, \tag{24}$$

where $b_{V_{i,1}} = (1+\gamma_i)l_{i,v}b_{\boldsymbol{\phi}_{i,c}}b_{p_{i,c}} + (1+\gamma_i)^2 l_{i,v}b_{\boldsymbol{\phi}_{i,c}}^2 b_{\boldsymbol{W}_{i,c}} + \sigma_{i,c}l_{i,v}b_{\boldsymbol{W}_{i,c}}$.

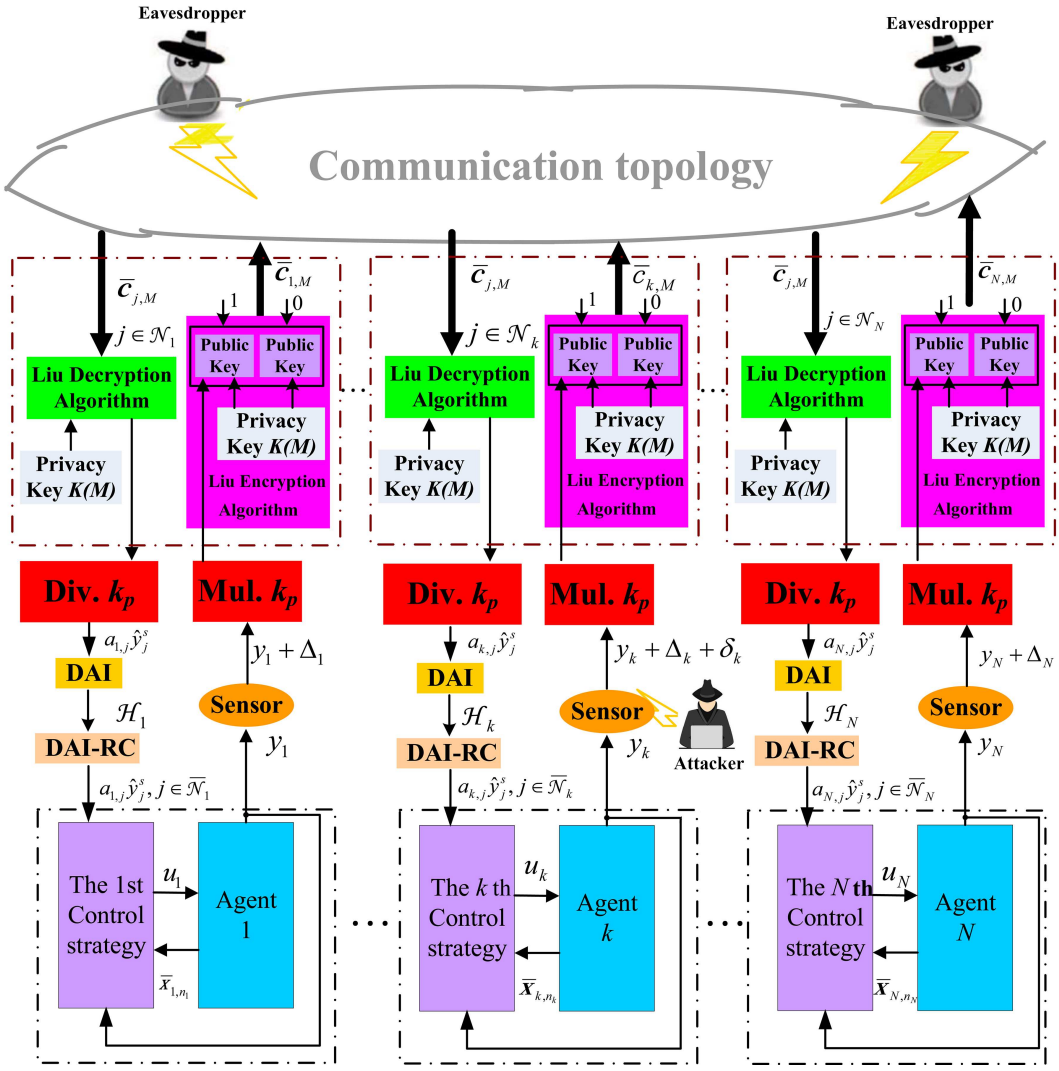


**Figure 3** (Color online) Block diagram of the MAS.

From (14)–(16), Young's inequality and $-\varphi_{i,1}\tilde{\lambda}_{i,1}^2 + b_{\phi_{i,1}} b_{\phi_{i,c}} b_{\boldsymbol{W}_{i,c}} |\tilde{\lambda}_{i,1}| \leqslant \frac{b_{\phi_{i,1}}^2 b_{\phi_{i,c}}^2 b_{\boldsymbol{W}_{i,c}}^2}{4\varphi_{i,1}}$, the derivative of $V_{i,2}$ becomes

$$\begin{aligned}\dot{V}_{i,2} \leqslant & -\left(k_{i,1} + \frac{\breve{g}_{i,1}}{2\sum_{j\in\mathcal{N}_i} a_{i,j}\underline{g}_{i,1}^2} - 4\right) e_{i,1}^2 + \left(\frac{1}{4} - \frac{1}{\epsilon_{i,2}}\right)\chi_{i,2}^2 - \left(\frac{\sigma_{i,1}}{2} - 1 - \frac{b_{\phi_{i,1}} b_{\phi_{i,c}}}{2} - \varphi_{i,1}\right)\tilde{\lambda}_{i,1}^2 + \frac{b_{\phi_{i,1}} b_{\phi_{i,c}}}{2}\tilde{\lambda}_{i,c}^2 \\ & + \frac{b_{\phi_{i,1}}^2 b_{\phi_{i,c}}^2 b_{\boldsymbol{W}_{i,c}}^2}{4\varphi_{i,1}} + \frac{\sigma_{i,1}}{2} b_{\boldsymbol{W}_{i,1}}^2 + |\chi_{i,2} B_{i,2}| + \frac{1}{4} b_{i,1}^2 + \frac{1}{4} e_{i,2}^2 + \left(\frac{k_{i,1}^2}{4} + \frac{b_{\phi_{i,1}}^2}{2}\right)\sum_{j\in\bar{\mathcal{N}}_i} a_{i,j}^2 \hat{z}_j^2, \end{aligned} \tag{25}$$

where $b_{i,1} = b_{\boldsymbol{W}_{i,1}}^2 + \iota_{i,1} + b_{\varepsilon_{i,1}} + \frac{\breve{\Delta}_i}{\breve{g}_{i,1}}$ and $B_{i,2}$ is from [38]. We choose $k_{i,1} = 4 - \frac{\breve{g}_{i,1}}{2\sum_{j\in\mathcal{N}_i} a_{i,j}\underline{g}_{i,1}^2} + \alpha_{i,0}$, $|B_{i,2}| < M_{i,2}$, $|\chi_{i,2}B_{i,2}| \leqslant \frac{\chi_{i,2}^2 B_{i,2}^2}{2\tau_i} + \frac{\tau_i}{2}$, $\frac{1}{\epsilon_{i,2}} = \frac{1}{4} + \frac{M_{i,2}^2}{2\tau_i} + \alpha_{i,0}$, and there exists $|\hat{z}_j| < Z_j$ due to the error between encryption and decryption in Subsection 2.3 consisting of random constants, where $\tau_i$ and $Z_j$ are positive constants. Then, Eq. (25) yields

$$\begin{aligned}\dot{V}_{i,2} \leqslant & -\alpha_{i,0} e_{i,1}^2 - \alpha_{i,0}\chi_{i,2}^2 - \left(\frac{\sigma_{i,1}}{2} - \frac{b_{\phi_{i,1}}^2 + b_{\phi_{i,1}} b_{\phi_{i,c}}}{2} - \varphi_{i,1}\right)\tilde{\lambda}_{i,1}^2 \\ & + \frac{b_{\phi_{i,1}} b_{\phi_{i,c}}}{2}\tilde{\lambda}_{i,c}^2 + \frac{1}{4} e_{i,2}^2 + c_{i,1} + \frac{\tau_i}{2}, \end{aligned} \tag{26}$$

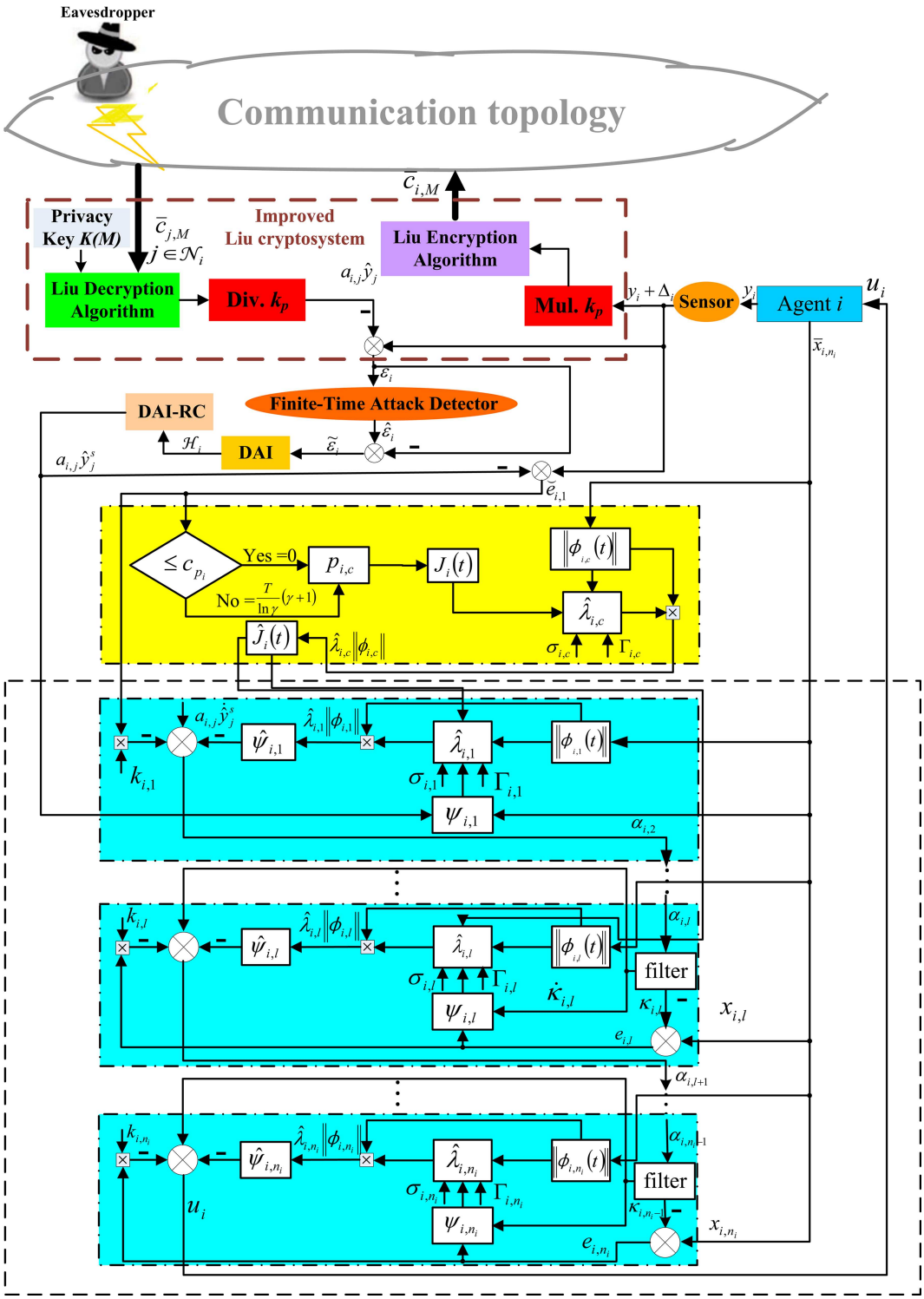


**Figure 4** (Color online) Block diagram of Agent $i$.

where $c_{i,1}=\frac{b_{\phi_{i,1}}^2 b_{\phi_{i,c}}^2 b_{\boldsymbol{W}_{i,c}}^2}{4\varphi_{i,1}}+\frac{\sigma_{i,1}}{2}b_{\boldsymbol{W}_{i,1}}^2+\frac{1}{4}b_{i,1}^2+(\frac{k_{i,1}^2}{4}+\frac{b_{\phi_{i,1}}^2}{2})\sum_{j\in\bar{\mathcal{N}}_i}a_{i,j}^2Z_j^2$.

According to (19), $(\frac{1}{4}-\frac{1}{\epsilon_{i,l}})\chi_{i,l+1}^2+|B_{i,l+1}\chi_{i,l+1}|\leqslant-\alpha_{i,0}\chi_{i,l+1}^2+\frac{\tau_i}{2}$, and $k_{i,l}=3\frac{1}{4}+\frac{\breve{g}_{i,l}}{2\underline{g}_{i,l}}+\alpha_{i,0}$, one has

$$\dot{V}_{i,3}\leqslant-\sum_{l=2}^{n_i-1}\alpha_{i,0}e_{i,l}^2-\sum_{l=2}^{n_i-1}\left(\frac{\sigma_{i,l}}{2}-\frac{b_{\phi_{i,l}}b_{\phi_{i,c}}}{2}-\varphi_{i,l}\right)\tilde{\lambda}_{i,l}^2+\sum_{l=2}^{n_i-1}\frac{b_{\phi_{i,l}}b_{\phi_{i,c}}}{2}\tilde{\lambda}_{i,c}^2$$

$$-\sum_{l=2}^{n_i-1}\alpha_{i,0}\chi_{i,l+1}^2+c_{i,l}+\frac{n_i-2}{2}\tau_i+\sum_{l=2}^{n_i-1}\frac{1}{4}(e_{i,l+1}^2-e_{i,l}^2),\tag{27}$$

where $c_{i,l}=(n_i-2)(\frac{b_{\boldsymbol{\phi}_{i,l}}^2 b_{\boldsymbol{\phi}_{i,c}}^2 b_{\boldsymbol{W}_{i,c}}^2}{4\varphi_{i,l}}+\frac{\sigma_{i,l}}{2}b_{\boldsymbol{W}_{i,l}}^2+\frac{1}{4}b_{i,l}^2)$ and $b_{i,l}=b_{\boldsymbol{W}_{i,l}}^2+\iota_{i,l}+b_{\varepsilon_{i,l}}$.

From (21), (22), and $k_{i,n_i}=\frac{1}{4}+\frac{\breve{g}_{i,n_i}}{2\underline{g}_{i,n_i}}+\alpha_{i,0}$, the derivative of $V_{i,4}$ yields

$$\dot{V}_{i,4}\leqslant-\alpha_{i,0}e_{i,n_i}^2-\frac{1}{4}e_{i,n_i}^2-\left(\frac{\sigma_{i,n_i}}{2}-\frac{b_{\boldsymbol{\phi}_{i,n_i}}b_{\boldsymbol{\phi}_{i,c}}}{2}-\varphi_{i,n_i}\right)\tilde{\lambda}_{i,n_i}^2+\frac{b_{\boldsymbol{\phi}_{i,n_i}}b_{\boldsymbol{\phi}_{i,c}}}{2}\tilde{\lambda}_{i,c}^2+c_{i,n_i},\tag{28}$$

where $c_{i,n_i}=\frac{b_{\boldsymbol{\phi}_{i,n_i}}^2 b_{\boldsymbol{\phi}_{i,c}}^2 b_{\boldsymbol{W}_{i,c}}^2}{4\varphi_{i,n_i}}+\frac{\sigma_{i,n_i}}{2}b_{\boldsymbol{W}_{i,n_i}}^2+\frac{1}{4}b_{i,n_i}^2$ and $b_{i,n_i}=b_{\boldsymbol{W}_{i,n_i}}^2+\iota_{i,n_i}+b_{\varepsilon_{i,n_i}}$. With $-l_{i,c}\tilde{\lambda}_{i,c}^2+b_{V_{i,1}}|\tilde{\lambda}_{i,c}|\leqslant\frac{b_{V_{i,1}}^2}{2l_{i,c}}$, (23), (24), and (27), the derivative of $V_i$ yields

$$\dot{V}_i\leqslant-\wp_i V_i+\mho_i,\tag{29}$$

where $\mho_i=\sum_{l=1}^{n_i}c_{i,l}+\frac{b_{V_{i,1}}^2}{2l_{i,c}}+\frac{n-1}{2}\tau_i$ and $\wp_i=\min\{2\alpha_{i,0},\sum_{l=1}^{n_i}(\sigma_{i,l}-b_{\boldsymbol{\phi}_{i,l}}b_{\boldsymbol{\phi}_{i,c}}-2\varphi_{i,l}),2l_{i,v}\sigma_{i,c}-\sum_{l=1}^{n_i}b_{\boldsymbol{\phi}_{i,l}}b_{\boldsymbol{\phi}_{i,c}}-2l_{i,c}\}$. Multiplying (29) by $\mathrm{e}^{\wp_i t}$ and integrating it over $[0,t]$, we have

$$V_i(t)\leqslant\rho_i+\big[V_i(0)-\rho_i\big]\mathrm{e}^{-\wp_i t},\tag{30}$$

where $\rho_i=\frac{\mho_i}{\wp_i}$.

From (23) and (30), as $t\to\infty$, one gets $|e_{i,1}|\leqslant\sqrt{2\sum_{j\in\bar{\mathcal{N}}_i}a_{i,j}\bar{g}_{i,1}\rho_i}$, $|e_{i,l}|\leqslant\sqrt{2\bar{g}_{i,l}\rho_i}$, $2\leqslant l\leqslant n_i$, $|\tilde{\lambda}_{i,l}|\leqslant\sqrt{2\rho_i}$, $1\leqslant l\leqslant n_i$, $|\tilde{\lambda}_{i,c}|\leqslant\sqrt{2\rho_i}$ and $|\chi_{i,l+1}|\leqslant\sqrt{2\rho_i}$, $2\leqslant l\leqslant n_i-1$. Apparently, the distributed synchronization error $e_{i,1}$ and all signals of the closed-loop MAS are bounded. According to (4) and Assumption 3, we obtain $|\bar{e}_{i,1}|\leqslant\sqrt{2\sum_{j\in\bar{\mathcal{N}}_i}a_{i,j}\bar{g}_{i,1}\rho_i}+\widehat{\Delta}_i$, where $\widehat{\Delta}_i=\sum_{j\in\bar{\mathcal{N}}_i}a_{i,j}(\bar{\Delta}_i+\bar{\Delta}_j)$. Thus, $|\bar{e}_{i,1}|$ converges to a neighborhood around the origin. According to (5) with the graph theory [29] and the properties of inequalities, we obtain $y_i\to y_j$. Then, the absolute value $|y_i-y_j|$ converges to a small neighborhood around the origin. The errors in the MAS and other signals in the closed-loop system are ultimately bounded. This concludes the proof.

**Remark 9.** From the stability analysis, we have $V_i(t)\leqslant\rho_i+\big[V_i(0)-\rho_i\big]\mathrm{e}^{-\wp_i t}$. As $t\to\infty$, one gets $|e_{i,1}|\leqslant\sqrt{2\sum_{j\in\bar{\mathcal{N}}_i}a_{i,j}\bar{g}_{i,1}\rho_i}$. By decreasing the parameter $\rho_i$, the convergence boundary of the error $|e_{i,1}|$ can be reduced. To decrease $\rho_i=\frac{\mho_i}{\wp_i}$, we can enlarge the parameter $\wp_i$, where $\wp_i$ is related to control parameters, such as $k_{i,l}$ and $\sigma_{i,l}$, and $\mho_i$ is a constant composed of nonadjustable parameters. Therefore, from the analysis, scaling up $k_{i,l}$ and $\sigma_{i,l}$ is a feasible way to reduce the size of the error $|e_{i,1}|$.

## 5 Simulation results

In this section, a numerical example and an example of multiple unmanned aerial vehicles (UAVs) are given to demonstrate the effectiveness of our strategy. The MASs in the two examples consist of five agents, and there is a risk of both privacy disclosure and cyber attacks. Figure 5 shows the communication graph, where 1–5 indicate the five agents.

### 5.1 Numerical example

Consider five nonlinear agents

$$\begin{cases}\dot{x}_{i,1}=f_{i,1}(\bar{x}_{i,2})+g_{i,1}(\bar{x}_{i,2})x_{i,2},\\ \dot{x}_{i,2}=f_{i,2}(\bar{x}_{i,2})+g_{i,2}(\bar{x}_{i,2})u_i,\\ y_i=x_{i,1},\end{cases}$$

where $f_{i,1}(\bar{x}_{i,2})=-0.05x_{i,1}$, $f_{i,2}(\bar{x}_{i,2})=0.05\big[-x_{i,2}+3.1(0.4-x_{i,1})\exp(1.5x_{i,2})-4.1x_{i,2}\big]$, $g_{i,1}(\bar{x}_{i,2})=0.05\big[0.019(1.5+0.01x_{i,1}^2)\exp(4x_{i,2})\big]$, and $g_{i,2}(\bar{x}_{i,2})=0.205(1.001+\sin(x_{i,1}x_{i,2}))$.

The centers of RBF NN for $J_i$, $\psi_{i,1}$, and $\psi_{i,2}$ are uniformly spaced over $\big[-2,2\big]\times\big[-2,2\big]\times\big[-2,2\big]$, with $\eta_i=\sqrt{2}$. The index threshold is chosen as $c_{p_i}=0.0001$. The initial values of the states are selected as $x_{1,1}(0)=$

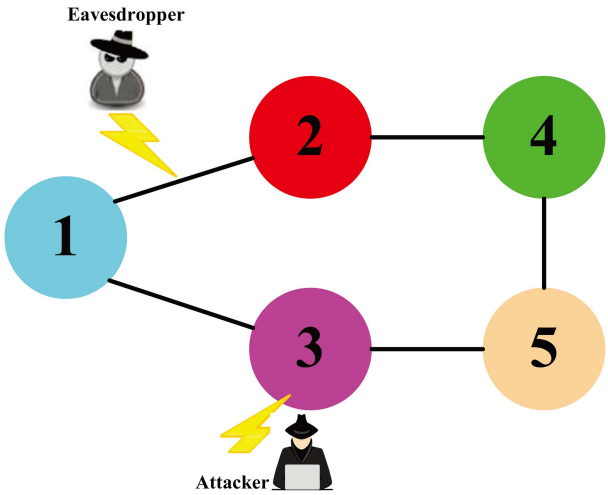


**Figure 5** (Color online) Communication graph.

**Table 1** Comparison of the number of learning parameters. $\gamma_{i,l}$ and $\gamma_{j,1}$ are the numbers parameters for the $i$th agent with $l$th-order dynamics and the 1st subsystem in the $j$th agent, respectively, $j \in \mathcal{N}_i$. Refs. [39, 40] do not hold any attack detectors. If attack detectors are added to these methods, additional learning parameters are necessary.

| | Ref. [39] | Ref. [40] | Ours |
|---|---|---|---|
| Each agent | $\sum\limits_{l=1}^{n_i}\gamma_{i,l}+\sum\limits_{j\in\mathcal{N}_i}\gamma_{j,1}$ | $\sum\limits_{l=1}^{n_i}\gamma_{i,l}$ | $2+n_i$ |
| Overall MAS | $\sum\limits_{i=1}^{N}\left(\sum\limits_{l=1}^{n_i}\gamma_{i,l}+\sum\limits_{j\in\mathcal{N}_i}\gamma_{j,1}\right)$ | $\sum\limits_{i=1}^{N}\sum\limits_{l=1}^{n_i}\gamma_{i,l}$ | $\sum\limits_{i=1}^{N}(2+n_i)$ |

$-0.05$, $x_{2,1}(0)=0.15$, $x_{3,1}(0)=-0.12$, $x_{4,1}(0)=0.2$ and $x_{i,2}(0)=0.1$, $\hat{\lambda}_{i,c}(0)=0.003$, $\hat{\lambda}_{i,1}(0)=0.0003$, and $\hat{\lambda}_{i,2}(0)=0.0003$. The control parameters are selected as $\Gamma_{i,c}=10$, $\Gamma_{i,1}=2$, $\Gamma_{i,2}=2$, $\sigma_{i,c}=0.5$, $\sigma_{i,1}=0.2$, $\sigma_{i,2}=0.2$, $\epsilon_{i,2}=0.02$, $k_{i,1}=10$, and $k_{i,2}=0.2$.

Because of the sensitivity of the output information from neighboring agents, this information must be encrypted before communication interaction among agents. Furthermore, we assume that the sensors of Agent 3 are under an attack signal $\delta_i=0.05t+0.018\sin(0.1t)$ and affected by Gaussian white noise. To address this issue, we utilize the finite-time attack detector (9) and the DAI-RC algorithm, and the compromised Agent 3 is detected successfully, thus achieving consensus. The simulation results are displayed in Figures 6–8. The curves of normal agents and the compromised Agent 3 are plotted in Figure 6. The curves of indexes are plotted in Figure 7. $\mathcal{H}_3=1$ indicates that Agent 3 is paralyzed and successfully detected. The isolation index $\mathcal{P}_3=1$ indicates successful isolation. For an intuitive comparison, we take Agent 1 as an example. As shown in Figure 8, the amplified plaintext can better restore the truth value after decryption. This shows that, with our improved Liu cryptosystem, the decryption error is smaller than the one obtained using a previous algorithm from [16].

Comparisons with the existing methods [39, 40] are shown in Table 1. The method in [39] requires NNs to approximate the current agent dynamics and those of neighboring agents. Meanwhile, the control approach in [40] processes the unknown dynamics $f_{i,k}(\cdot)$ and $g_{i,k}(\cdot)$ into a single entity $F_{i,k}(\cdot)$. Therefore, it utilizes an NN but still requires a substantial number of learning parameters. Notably, these previous methods in [39, 40] have no attack detectors. If an attack detector is added to these methods, additional learning parameters will be needed. In this paper, for the finite-time attack detector and a critic NN, the unknown dynamics in each agent must be approximated. Therefore, two additional NNs for attack detection will be added. Thus, in our strategy, the number of learning parameters is $2+n_i$ for each agent, and that of the overall system with $N$ agents is only $\sum_{i=1}^{N}(2+n_i)$.

### 5.2 Multi-UAVs

Consider multi-UAVs as another example. The topology in this example is the same as that in Example 1, and the position dynamics of UAVs [41] are

$$\begin{cases}\dot{\chi}_{i,1}=\chi_{i,2},\\ \dot{\chi}_{i,2}=u_i+F_i,\end{cases}$$

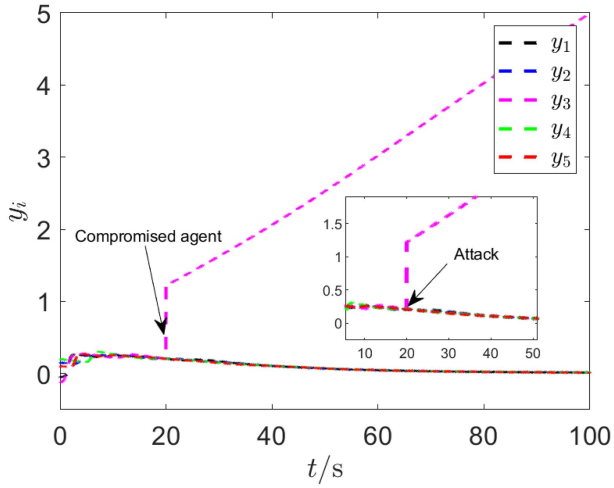


**Figure 6** (Color online) $y_i$ in Example 1, $i = 1, \ldots, 5$.

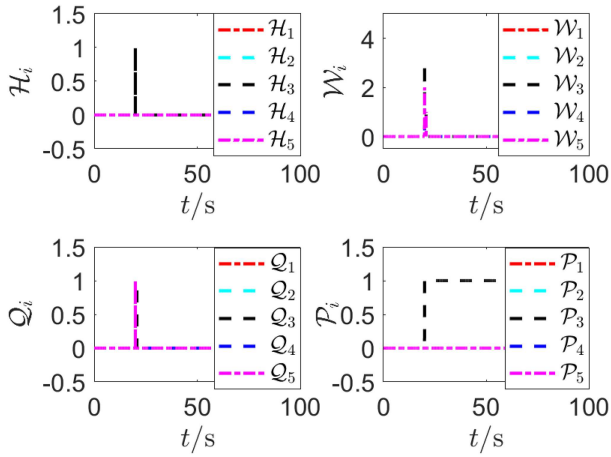


**Figure 7** (Color online) Indexes in Example 1.

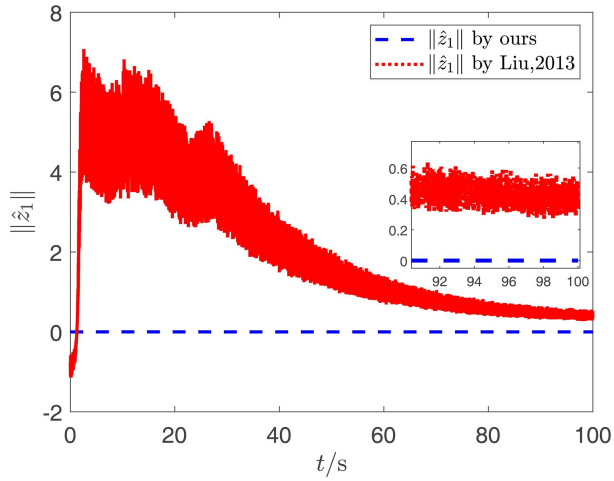


**Figure 8** (Color online) Comparison of Agent 1's decryption errors with Liu's method in Example 1.

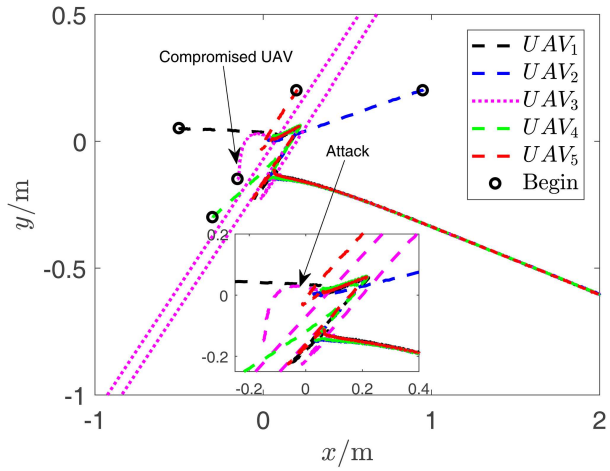


**Figure 9** (Color online) 2-D output consensus performance in Example 2.

where $\chi_{i,1} = [x_{i,1}, y_{i,1}]^{\mathrm{T}}$, $\chi_{i,2} = [x_{i,2}, y_{i,2}]^{\mathrm{T}}$, $F_i = [\xi_{x_i}\dot{x}_{i,1}/m_i + u_{x_i}/m_i - u_{x_i}, \xi_{y_i}\dot{y}_{i,1}/m_i + u_{y_i}/m_i - u_{y_i}]^{\mathrm{T}}$, and $u_i = [u_{x_i}, u_{y_i}]^{\mathrm{T}}$. The physical parameters are $m_i = 2$ kg, $\xi_{x_i} = 1.2$ N$\cdot$s/rad, and $\xi_{y_i} = 1.44$ N$\cdot$s/rad. The initial values are $\chi_{1,1}(0) = [-0.5, 0.5]^{\mathrm{T}}$, $\chi_{2,1}(0) = [0.95, 0.5]^{\mathrm{T}}$, $\chi_{3,1}(0) = [-0.15, 0.5]^{\mathrm{T}}$, $\chi_{4,1}(0) = [-0.3, 0.5]^{\mathrm{T}}$, $\chi_{5,1}(0) = [0.2, 0.52]^{\mathrm{T}}$, $\chi_{1,2}(0) = [0.05, 0.01]^{\mathrm{T}}$, $\chi_{2,2}(0) = [0.2, 0.5]^{\mathrm{T}}$, $\chi_{3,2}(0) = [-0.15, 0.5]^{\mathrm{T}}$, $\chi_{4,2}(0) = [-0.3, 0.5]^{\mathrm{T}}$, $\chi_{5,2}(0) = [0.2, 0.52]^{\mathrm{T}}$, $\hat{\lambda}_{i,c}(0) = 0.003$, $\hat{\lambda}_{i,1}(0) = 0.0003$, and $\hat{\lambda}_{i,2}(0) = 0.0003$. The control parameters are $\Gamma_{i,c} = 10$, $\Gamma_{i,1} = 2$, $\Gamma_{i,2} = 2$, $\sigma_{i,c} = 0.5$, $\sigma_{i,1} = 0.2$, $\sigma_{i,2} = 0.2$, $\epsilon_{i,2} = 0.02$, $k_{i,1} = 25$, and $k_{i,2} = 30$. The sensor in UAV 3 is under an attack signal $\delta_3 = 0.05t + 0.018\sin(0.1t) - 30$ and Gaussian white noise.

The simulation results are shown in Figures 9–11. The curves of normal UAVs and the compromised UAV 3 are plotted in Figure 9. Figure 10 shows that UAV 3 is attacked in the $x$-direction, and the proposed control strategy ensures that the normal UAVs achieve consensus. Their indexes are given in Figure 11. From these curves, $\mathcal{H}_3 = 1$ indicates that UAV 3 is attacked and is detected successfully. The isolation index $\mathcal{P}_3 = 1$ indicates successful isolation.

## 6 Conclusion

In this paper, an improved Liu cryptosystem is utilized to safeguard sensitive information, effectively mitigating the potential for external information intrusion. To deal with network attacks, a distributed detection-isolation algorithm is introduced. With privacy-preserving and normal agents information and using backstepping and RL

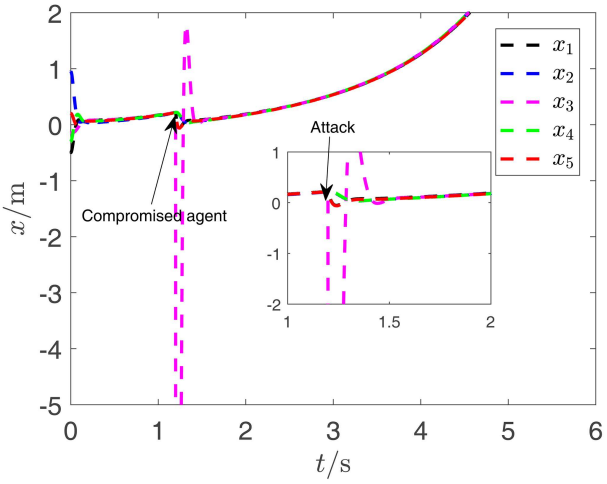


**Figure 10** (Color online) Consensus performance of $x_i$ in Example 2.

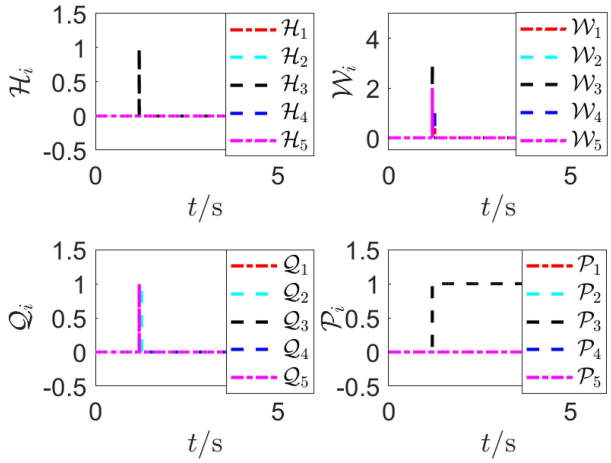


**Figure 11** (Color online) Indexes of UAVs in Example 2.

techniques, a privacy-preserving leaderless consensus control strategy is then developed. With the aid of Lyapunov functions, the proposed control strategy ensures that all signals are ultimately bounded, and the output consensus error of an MAS converges to a neighborhood around the origin.

**Acknowledgements** This work was supported by National Natural Science Foundation of China (Grant Nos. 62473204, 62473345), "Chunhui Program" Collaborative Scientific Research Project (Grant No. 202202004), Natural Science Foundation of Nanjing University of Posts and Telecommunications (Grant Nos. NY222144, NY223075), Huali Program for Excellent Talents in Nanjing University of Posts and Telecommunications, and Postgraduate Research & Practice Innovation Program of Jiangsu Province (Grant No. SJCX23_0279).

**References**

1 Meng Y H, Zhang H W, Wu A G. Leaderless output sign consensus of heterogeneous multi-agent systems over switching signed graphs. Sci China Inf Sci, 2023, 66: 190208
2 Liu Y, Wu X, Long J, et al. Event-triggered distributed adaptive leaderless consensus of uncertain heterogenous nonlinear multi-agent systems. IEEE Trans Circ Syst II, 2024, 71: 2694–2698
3 Chen B, Hu J P, Ghosh B K. Finite-time tracking control of heterogeneous multi-AUV systems with partial measurements and intermittent communication. Sci China Inf Sci, 2024, 67: 152202
4 Ning B, Han Q L, Zuo Z, et al. Fixed-time and prescribed-time consensus control of multiagent systems and its applications: a survey of recent trends and methodologies. IEEE Trans Ind Inf, 2022, 19: 1121–1135
5 An L W, Yang G H. Attack detectability and stealthiness in distributed optimal coordination of cyber-physical systems. Sci China Inf Sci, 2023, 66: 199204
6 Hong S, Kim K, Lee S H. A hybrid jamming detection algorithm for wireless communications: simultaneous classification of known attacks and detection of unknown attacks. IEEE Commun Lett, 2023, 27: 1769–1773
7 Zhao D, Lv Y, Zhou J, et al. Attack-isolation-based resilient control of large-scale systems against collusive attacks. IEEE Trans Netw Sci Eng, 2022, 9: 2857–2869
8 Zhao D, Lv Y, Yu X, et al. Resilient consensus of higher order multiagent networks: an attack isolation-based approach. IEEE Trans Automat Contr, 2021, 67: 1001–1007
9 Gallo A J, Turan M S, Boem F, et al. A distributed cyber-attack detection scheme with application to DC microgrids. IEEE Trans Automat Contr, 2020, 65: 3800–3815
10 Yang S F, Liang H J, Pan Y N, et al. Security control for air-sea heterogeneous multiagent systems with cooperative-antagonistic interactions: an intermittent privacy preservation mechanism. Sci China Tech Sci, 2025, 68: 1420402
11 Gao C, Lu J, Lou J, et al. Privacy-preserving algorithm for APPs in vehicle intelligent terminal system: a compressive method. IEEE Trans Intell Transp Syst, 2024, 25: 17352–17365
12 Wang Y. Privacy-preserving average consensus via state decomposition. IEEE Trans Automat Contr, 2019, 64: 4711–4716
13 Zhang Y, Peng Z, Wen G, et al. Privacy preserving-based resilient consensus for multiagent systems via state decomposition. IEEE Trans Control Netw Syst, 2022, 10: 1172–1183
14 Huang J, Huang Q, Mou G, et al. DPWGAN: high-quality load profiles synthesis with differential privacy guarantees. IEEE Trans Smart Grid, 2022, 14: 3283–3295
15 Yan Y, Chen Z, Varadharajan V, et al. Distributed consensus-based economic dispatch in power grids using the Paillier cryptosystem. IEEE Trans Smart Grid, 2021, 12: 3493–3502
16 Liu D. Homomorphic encryption for database querying. U.S. Patent 10,027,486, 2018
17 Gao Y, Wang W, Yu N. Consensus multi-agent reinforcement learning for volt-VAR control in power distribution networks. IEEE Trans Smart Grid, 2021, 12: 3594–3604
18 Li J, Yuan L, Chai T, et al. Consensus of nonlinear multiagent systems with uncertainties using reinforcement learning based sliding mode control. IEEE Trans Circ Syst I, 2022, 70: 424–434
19 Yang X, Zhang H, Wang Z. Data-based optimal consensus control for multiagent systems with policy gradient reinforcement learning. IEEE Trans Neural Netw Learn Syst, 2021, 33: 3872–3883
20 Wen G X, Li B. Optimized leader-follower consensus control using reinforcement learning for a class of second-order nonlinear multiagent systems. IEEE Trans Syst Man Cybern Syst, 2022, 52: 5546–5555
21 Li Z J, Song Y F, Wen G X. Reinforcement learning based optimized sliding-mode consensus control of high-order nonlinear canonical dynamic multiagent system. IEEE Syst J, 2023, 17: 1–10
22 Wang X, Niu B, Wang X. Distributed adaptive bipartite consensus tracking of high-order nonstrict-feedback nonlinear multi-agent systems. J Control Decision, 2023, 10: 393–401

23 Sung T Y, Ho T Y, Chang C P, et al. Optimal k-fault-tolerant networks for token rings. J Inf Sci Eng, 2000, 16: 381–390
24 Zuo Z, Liu C, Han Q L, et al. Unmanned aerial vehicles: control methods and future challenges. IEEE CAA J Autom Sin, 2022, 9: 601–614
25 Zuo S, Yue D. Resilient output formation containment of heterogeneous multigroup systems against unbounded attacks. IEEE Trans Cybern, 2020, 52: 1902–1910
26 Cao L, Pan Y, Liang H, et al. Event-based adaptive neural network control for large-scale systems with nonconstant control gains and unknown measurement sensitivity. IEEE Trans Syst Man Cybern Syst, 2024, 54: 7027–7038
27 Qin L M, Lu Y Z, Xu Y, et al. The calibration methods of hydrophones for underwater environmental sound measurements or biomedical ultrasound measurements: a review. Measurement, 2025, 242: 115700
28 Sargolzaei A, Allen B C, Crane C D, et al. Lyapunov-based control of a nonlinear multiagent system with a time-varying input delay under false-data-injection attacks. IEEE Trans Ind Inf, 2021, 18: 2693–2703
29 Zhang H, Jiang H, Luo Y, et al. Data-driven optimal consensus control for discrete-time multi-agent systems with unknown dynamics using reinforcement learning method. IEEE Trans Ind Electron, 2017, 64: 4091–4100
30 Chen W, Liu L, Liu G P. Privacy-preserving distributed economic dispatch of microgrids: a dynamic quantization-based consensus scheme with homomorphic encryption. IEEE Trans Smart Grid, 2023, 14: 701–713
31 Wang J, Ke J, Zhang J F. Differentially private bipartite consensus over signed networks with time-varying noises. IEEE Trans Automat Contr, 2024, 69: 5788–5803
32 Pomet J B, Praly L. Adaptive nonlinear regulation: estimation from the Lyapunov equation. IEEE Trans Automat Contr, 1992, 37: 729–740
33 Bai W, Li T, Tong S. NN reinforcement learning adaptive control for a class of nonstrict-feedback discrete-time systems. IEEE Trans Cybern, 2020, 50: 4573–4584
34 Sui S, Chen C L P, Tong S. Fuzzy adaptive finite-time control design for nontriangular stochastic nonlinear systems. IEEE Trans Fuzzy Syst, 2018, 27: 172–184
35 Wang F, Chen B, Lin C, et al. Adaptive neural network finite-time output feedback control of quantized nonlinear systems. IEEE Trans Cybern, 2017, 48: 1839–1848
36 Sun K K, Qiu J B, Karimi H R, et al. A novel finite-time control for nonstrict feedback saturated nonlinear systems with tracking error constraint. IEEE Trans Syst Man Cybern Syst, 2021, 51: 3968–3979
37 Guo X X, Yan W S, Cui R X. Integral reinforcement learning-based adaptive NN control for continuous-time nonlinear MIMO systems with unknown control directions. IEEE Trans Syst Man Cybern Syst, 2020, 50: 4068–4077
38 Wang D, Huang J. Neural network-based adaptive dynamic surface control for a class of uncertain nonlinear systems in strict-feedback form. IEEE Trans Neural Netw, 2005, 16: 195–202
39 Wang W, Wang D, Peng Z. Predictor-based adaptive dynamic surface control for consensus of uncertain nonlinear systems in strict-feedback form. Adaptive Control Signal, 2017, 31: 68–82
40 Li H, Wu Y, Chen M. Adaptive fault-tolerant tracking control for discrete-time multiagent systems via reinforcement learning algorithm. IEEE Trans Cybern, 2020, 51: 1163–1174
41 Song Y, He L, Zhang D, et al. Neuroadaptive fault-tolerant control of quadrotor UAVs: a more affordable solution. IEEE Trans Neural Netw Learn Syst, 2018, 30: 1975–1983