Journal of Network and Computer Applications 221 (2024) 103769

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

A fast coordination approach for large-scale drone swarm

Wu Chen, Jiayi Zhu, Jiajia Liu *, Hongzhi Guo

Northwestern Polytechnical University, Xi'an 710129, China

ARTICLE INFO

Keywords:
Consensus
Local interaction
Fast coordination method
Confrontational environment

ABSTRACT

With the advances in artificial intelligence, robotics, and data fusion, large numbers of drones operating in a coordinated manner will become commonplace for a wide range of commercial and military uses. At present, the application methods of drone swarms are mainly divided into fully autonomous methods and controlled methods with human participation. Because of the limited level of artificial intelligence, controlled drone swarms will be the main way for the application of large-scale drone swarms for a long time. However, there is less research on achieving global coordination in a limited time for a controlled large-scale drone swarm. Therefore, a new large-scale drone swarm framework is proposed firstly in this paper, which achieves global coordination through local interaction and reduces the impact of limited channel resources. Secondly, this paper proposes a local interaction-based fast coordination method and introduces a prediction mechanism, to ensure that large-scale drone swarms can quickly achieve coordination even in the presence of node loss. Moreover, the numerical integration method is used to update the consensus state, so that the drones can increase the iteration period, reduce the number of packets, and further reduce the channel burden. Finally, considering that large-scale drones swarm are usually composed of drone swarms launched at different locations and times, a consensus algorithm considering the merging behavior of drone swarms is also proposed. The simulation results show that the large-scale drone swarm using the proposed architecture can achieve the leader–follower consensus in a very short time and even in a confrontational environment with poor communication conditions. Besides, after the merger of multiple drone swarms, the consensus problem can still be solved in very few iteration cycles.

1. Introduction

Drones, otherwise called autonomous unmanned aerial vehicles (UAVs), have received increasing demand from civilian, commercial, and military applications in recent years. Especially in military applications, drones become an indispensable part of the future battlefield (Orfanus et al., 2016). Due to the limitations of payload and energy of a single drone, the deployment of a drone swarm has been proposed for a diverse set of applications for higher mission requirements, which is re-shaping the future applications of drones (Chmaj and Selvaraj, 2015; Chen et al., 2020b). Drone swarms perform missions collaboratively and have the potential of fast deployment and wild coverage to go beyond the individual capabilities of single drones. Currently, the quantity of simultaneous drone missions is in the dozens or hundreds. However, with the technological development of solar cells, battery capacity, and electric motors, the interest is shifting the size towards thousands, or even ten thousand drones forming swarms to carry out collaborative missions. A large-scale drone swarm is becoming a dominant practical choice for both civilian and military operations, such as environment monitoring, saturation attack, transport packages, and traffic surveillance (Oubbati et al., 2022).

In a swarm, the drone acts as a simple agent and is conjoined in a complex networking system, giving rise to collaborative or collective behaviors. To achieve this propriety, a drone swarm usually operates in a centrally controlled or self-organized manner. In a centrally controlled drone swarm, a selected drone or a ground station makes decisions and assigns instructions to the drones in the swarm. For example, the Crazyswarm are centrally controlled by a single computer (Hönig et al., 2018). In a self-organized swarm, drones achieve task goals and make collective decisions by sharing information (Coppola et al., 2020). Nägeli et al. (2014) and Basiri et al. (2014) used onboard monocular cameras and onboard audio-based localization systems to achieve distributed information sharing for drones. Independent of control manners, drone swarms work in a three-step loop of perception, decision, and action.

Perception means drones observe the situation by sensors and build a comprehensive image of the goal and environment. Timely and accurately sensed information is fundamental and crucial for accomplishing

* Corresponding author.
E-mail address: liujiajia@nwpu.edu.cn (J. Liu).

https://doi.org/10.1016/j.jnca.2023.103769
Received 16 March 2023; Received in revised form 11 August 2023; Accepted 17 October 2023
Available online 20 October 2023

the task. The sensed data is subsequently gathered by the central node in the centrally controlled swarm or exchanged by local interactions in a self-organized manner. The decision means reasoning and it is made by a centralized node or by the drone itself with full or part of sensed data. In the third step, drones take action following the decisions. The drone swarm will perceive the effectiveness of the action and step into the next loop.

In the perception–decision–action loop, communication and networking are crucial because they enable information, including sensed data and decision instructions, to be disseminated in the drone swarm. As drone swarms grow in size, ensuring timely and effective communication becomes challenging due to the contention overhead, and congestion. Consequently, a centralized control mode only guarantees timely instruction and sensed data distribution for small-scale drone swarms. A few programmed large-scale drone swarms can only complete simple behaviors like a light show, e.g., Intel's Shooting Stars.

In recent years, self-organized robot swarms involving drone swarms that use local interaction to produce global behaviors hold great promise. Some proposals have demonstrated successful implementations of self-organized robot swarms up to 1000 units (Rubenstein et al., 2014). Meanwhile, small-scale self-organized robot swarms have also received increasing scientific interest. For example, the SWARM-BOTS project presents a drone swarm with six tiny quadrotors autonomously exploring an unknown environment and back to the departure point (Brambilla et al., 2013).

However, a complex maneuver of robot swarms still relies on the centralized control unit or external station (Berlinger et al., 2021). Most studies in the field of fully decentralized methods have only focused on theoretical work. In practical application, due to the low level of intelligence of drones, the impaction will be massive when the completely autonomous drone swarm has decision errors. Therefore, some envisioned applications of large-scale drone swarms like search and rescue, surveillance, and saturation attack still rely on the intervention of the centralized base station. However, as we mentioned above, control instruction cannot be distributed timely in large-scale swarms. Therefore, it is still a great challenge to accomplish the task effectively under the control of a centralized-based station.

In this article, to address the problem that large-scale drone swarms cannot coordinate due to insufficient channel resources under controlled conditions, we proposed a semi-autonomous mode for a large-scale drone swarm. The main contributions can be summarized as follows:

- A working architecture is proposed for a large-scale drone swarm to achieve global consensus. Unlike traditional coordination methods, this architecture is based on local interaction and consensus, prioritizing the perception–decision–action cycle over reliable information transmission. The architecture can effectively address the problem of lack of channel resources that arise from an increasing number of drones.
- A fast coordination method is proposed for large-scale drone swarms in a confrontational environment with packet loss and dynamic topology. The delay problem caused by local interaction is addressed by introducing a prediction mechanism. Compared to existing works, its advantages lie in ensuring fast consensus among large-scale drone swarms even in the presence of node and link failures.
- Considering the possibility of the Zeno phenomenon in existing event-triggered consensus methods, the Runge–Kutta method is adopted to discretize the consensus algorithm for continuous systems. This approach increases the iteration period, reduces the number of packets, and further lightens the channel burden.
- A robust and time-limited algorithm is proposed to ensure consensus during the merging process of large-scale multiple drone swarms, which is a scenario that has been rarely addressed in existing works.

The remainder of this paper is organized as follows. In Section 2, we discuss several existing related works. The system architecture and problem statement are introduced in Section 3, then it follows our proposed fast coordination method for large-scale drone swarm under the confrontational environment in Section 4. Simulation results are presented to validate the performance of the proposed algorithms in Section 5. Finally, this paper is concluded in Section 6.

2. Related work

A drone swarm achieves impressive global behaviors that surpass the capabilities of individual drones by leveraging shared information among a group of drones. It is evident that efficient communication and coordination are essential prerequisites for a drone swarm to successfully execute collaborative tasks.

It is crucial for drones that function as a cohesive unit to share information, which places a high demand on communication. Therefore, communication becomes an essential aspect of a drone swarm (Cui et al., 2017). However, drone swarms with characteristics of dynamic topology usually operate on open wireless channels and work in confrontational environments, resulting in the rise of challenge to ensure efficient communications. To address this challenge, a range of problems, from networked communication to local exchanges, need to be addressed (Dorigo et al., 2013).

Networked communication technology has garnered significant attention from researchers due to its reliance on mature network architectures. Much of the research in this area has focused on enhancing the reliability and throughput of drone swarm networks, as well as improving their performance under dynamic topology and confrontational circumstances (Oubbati et al., 2021). This is achieved by establishing and maintaining the connectivity referred to as the robust network. Some solutions such as topology control and robust routing have been developed to address these challenges (Hou et al., 2021). Additionally, in recent years, there has been a growing emphasis on maintaining the connectivity of large-scale drone swarms by using maneuvering tactics or adjusting speed, as highlighted in studies such as those by Zou et al. (2021), Tahir et al. (2019), Yanmaz et al. (2018), and Tegicho et al. (2021).

However, as the size of the drone swarm increases, the data traffic within the swarm also grows exponentially. It poses a challenge caused by the shortage of spectrum resources, making the communication of the drone swarm vulnerable, especially in the face of jamming and congestion. Simply improving network reliability is not an effective solution (Wang et al., 2022a). To address this issue, various spectrum-sharing mechanisms and bandwidth-efficient algorithms have been proposed to mitigate the spectrum shortage (Zou et al., 2021; Onthoni et al., 2023; Huang et al., 2022). Additionally, local exchanging methods aim to tackle the problem at its root by reducing multi-hop data in the network (McCune and Madey, 2013). With the advancement of coordination technology, this approach is gradually becoming the primary mode of communication for drone swarms (Cui et al., 2017; Hussen et al., 2017).

When it comes to drone swarm coordination, there are two modes: centrally controlled coordination and self-organized coordination. The centrally controlled coordination is also called direct-controlled or leader–follower manner which means a special leader drone manipulated by a pilot directly and orchestrates the collective behavior of the rest of the swarm through the leader (Saffre et al., 2021). In a centrally controlled coordinated mode, coordination information is transmitted hop by hop in the network. And reliable network communication is necessary to ensure the drones reliably receive and execute the directives sent by the leader, which is the prerequisite for successful completion of missions (Tahir et al., 2019; Tian et al., 2023). As mentioned above, due to the shortage of spectrum resources, this kind of coordination manner cannot support a large-scale drone swarm. By April 2022, The US Army just tested a swarm of up to 30 drones over a desert in Utah (Saballa, 2022).

In comparison to centrally controlled coordination, self-organized coordination, also known as indirect control or leaderless manner, is preferred by large-scale drone swarms, in which coordination information is shared through the local exchange. This coordination mode does not require instructions to be reliably distributed to each drone but mainly achieves coordination through information consensus. The consensus means each drone updates the state based on the information exchanged with neighbor drones, and finally achieves the consistency of the state of whole drones (Carli et al., 2020). The characteristic of the consensus based on local interactive information enables space-division multiplexing, which can mitigate the shortage of spectrum resources. The local interaction does not require routing, which is more suitable for drones with low computing power. Furthermore, consensus can also be applied to the leader–follower model. Therefore, the consensus has received great attention recently (Pasek and Kaniewski, 2022; Doostmohammadian et al., 2021).

In recent years, research on consensus for drone swarms can be categorized into the following groups:

Firstly, achieving consensus in finite time is crucial for large-scale drone swarms due to battery limitations and mission time constraints. Recent studies have focused on finite-time consensus, including works specifically addressing drone swarms (Rikos et al., 2022; Pal, 2022). However, these existing finite-time consensus is based on the Lyapunov function, set-valued Lie derivative, or sliding mode (Pal, 2022; Liu et al., 2019). The algorithm is complicated and the convergence time relies on the initial state, leading unprocurable to be adopted by large-scale drone swarms.

Secondly, most mature consensus algorithms are designed for continuous-time systems and cannot be directly applied to drone swarms that rely on discrete communication. Although some event-triggered consensus algorithms have been proposed (Liu et al., 2022; Wang et al., 2022b), the Zeno phenomenon may occur which leads to the system cannot reach consensus.

Finally, many tasks still require human operators to control fleets of drone swarms. In such cases, leader–follower consensus algorithms are much more suitable. However, the large-scale swarm results in a prolonged convergence time because the state of the leader node is diffused hop-by-hop. Therefore, most large-scale drone swarms still rely on programmed behaviors instead of inter-drone communication (Kallenborn, 2018).

In summary, the application of large-scale drone swarms still faces some intractable problems. An efficient coordination schedule for large-scale drone swarms is urgently needed.

3. System architecture and problem statement

In this section, we present a system architecture for a large-scale drone swarm and describe the special problem faced by the large-scale drone swarm under the proposed architecture.

3.1. System architecture

During the mission execution of the drone swarm, the information transfer process is divided into bottom-up and top-down. The bottom-up approach means the drone swarm needs to observe the environment, share situation information and makes decisions based on consensus. Decision instructions are delivered in a top-down approach, enabling the drone to act effectively. The information transfer goes through the perception–decision–action loop. For this reason, we proposed a system architecture for drone swarm as illustrated in Fig. 1.

Platform Layer: The platform layer refers to the payload of software and hardware on the drone. It mainly includes a sensor block, an actuator block, a flight control block, and a transceiver block. The sensor block aims at observing the environment. The actuator block converts the command signal into a physical parameter to accomplish a task. The flight control block is responsible for maintaining the stability of flight attitude and path with or without receiving commands. The responsibility of the transceiver block is to receive and to send packets.

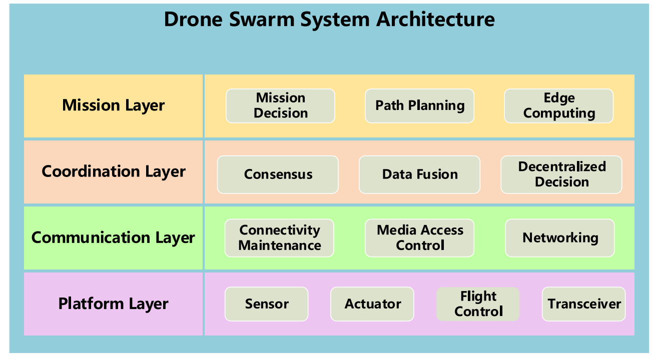


Fig. 1. Drone swarm system architecture.

Communication Layer: The communication layer is a crucial layer for a drone swarm because the prerequisite for drone swarms' consensus is to communicate between the drones. The primary responsibility of the communication layer is to ensure efficient data collection and exchange. The connectivity maintenance leveraging a robust and reliable infrastructure aims to ensure the connection between drones and the ground station. In this architecture, to reduce the number of packets in the communication network, the communication layer only achieves local interaction and does not guarantee the topology of the entire swarm. Media access control block uses protocols to deal with the problem of allocation and management of channel resources effectively. Moreover, the communication layer implements networking by sharing data.

Coordination Layer: The coordination layer is served for reasoning and decision-making based on the observations obtained by the drone, forming cooperative situation awareness to achieve team behavior based on information sharing. The consensus block guarantees drones in the swarm to share information with a consistent view of the states, which is critical to the coordination task. Moreover, sensors carried by drones are different due to limited load capacity. The data fusion block collects the data from multiple sources to build more sophisticated and precise models of the environment, providing effective data to support the correct decision-making. As the drone swarm may lose connection with the ground station or work in autonomous mode, decentralized decision-making based on the local-selected information is supported by the coordination layer.

Mission Layer: The mission layer is the principal subject for reasoning, decision-making, and organizing drones. The mission decision block allocates tasks and generates collaborative instruction to achieve global behavior, which is an essential section to guarantee the successful mission execution of a drone swarm. The path planning block is in charge of searching for an appropriate path from the starting point to the destination and provides a mechanism for collision avoidance. The edge computing block provides extra computation and storage capacity when the resources of drones are not sufficient to fulfill the cooperation requirements.

3.2. Problem statement

Fig. 1 presents a general structural model of a drone swarm in the perception–decision–action loop. This work focuses on the cooperation mission of drone swarm on a large scale. It only involves some functional blocks in Fig. 1. For the platform layer, we assume that the drone has a complete platform, by which the drone is able to normally fly, perceive the environment, and transceive signals. In the communication layer, it is assumed that large-scale drone swarms achieve cooperation through local information exchange, hence the proposed approach only involves connection maintenance and medium access control blocks. The medium access block adopts the CSMA/CA protocol, and the connection maintenance block adopts a robust networking solution proposed in the Chen et al. (2020b). The proposed approach achieves collaboration through consensus and does not concern data fusion and distributed decision-making modules in the cooperative layer. Finally, this paper focuses on the challenging problem of using a large-scale drone swarm to accomplish the task effectively under the control of a centralized base station. Therefore, we can reasonably assume that the mission planning is accomplished by the ground station, so this paper does not involve the mission layer.

Based on the architecture presented in Fig. 1, drones in a swarm are generally homogeneous and interact with each other through the communication topology. The topology can be represented by an undirected graph $G = (V, E)$, where V is a set of drones and E is a set of links between drones. Let $A = \left[a_{ij}\right]$ be the adjacency matrix associated with G which represents the connection of a drone swarm. The cell a_{ij} of the adjacent matrix equals 1 if drone i and j are connected, otherwise, the cell equals 0. In the proposed large-scale drone swarm system architecture, communication between drones is based on local interaction. A drone and its neighbors are considered as a neighborhood system, and the definition is given as follows.

Definition 1. Neighbor System: A neighborhood system for G is defined as:

$$N = \left\{N_i | \forall i \in V\right\}$$

where N_i is the set of drones neighboring drone i. Drone j can directly interact with drone i if $\forall j \in N_i$.

A large-scale drone swarm with fully autonomous decision-making only performs simple behaviors such as hovering. To complete complicated cooperative tasks, drones need to be controlled by receiving external state information. Assume that drone i maintains a state variable $x_i(t) \in \mathbb{R}$, and set $x(t) = \left[x_1(t), x_2(t), \dots, x_n(t)\right]^T \in \mathbb{R}^n$ represents the parameters related to the flight state such as speed, direction and so on. Global coordination will be achieved when all drones in the swarm converge to an external state. The definition is as follows:

Definition 2. Coordination: If the state of each drone satisfies the following formula:

$$\lim_{t\to\infty} \left\|x_i(t) - s(t)\right\| = 0 \tag{1}$$

it can be viewed as a drone swarm reaching coordination with the external state $s(t)$, where $s(t)$ represents external state information.

Due to the bandwidth constraints, it is hard to directly control each drone in the large-scale swarm. A common method is to adopt a leader–follower model, where the leader is directly controlled by external state information, and the followers achieve coordination by reaching state convergence with the leader. The leader drone's state information $x_L(t)$ is described as:

$$x_L(t) = s(t)$$

The swarm achieves global coordination when the drones in the swarm coordinate with the state of the leader. In such case, Eq. (1) can be rewritten as:

$$\lim_{t\to\infty} \left\|x_i(t) - x_L(t)\right\| = 0 \tag{2}$$

To satisfy Eq. (2), it is required to guarantee $a_{iL} \equiv 1, i = 1, \dots, n$, which means each drone in the swarm interacts with the leader directly. In a large-scale drone swarm based on local interaction, a stable connection between every drone in the swarm and the leader may be failed. Therefore, we maintain a spanning tree with the leader drone as the root node, and the drone achieves coordination with the leader by coordinating with the father drone. If drones in the swarm satisfy the following formula:

$$\lim_{t\to\infty} \left\|x_i(t) - \langle x_F(t)\rangle_i\right\| = 0 \tag{3}$$

where $\langle x_F(t)\rangle_i$ is the state of drone i's father drone. Thus, Eq. (2) can be considered to be held.

However, in order to achieve coordination between follower drones and the leader by maintaining the spanning tree, some problems should be solved:

(a) How to achieve coordination in a confrontational environment with packet loss and dynamic topology: Packet loss is inevitable for large-scale drone swarms working in confrontational environments. As we mentioned above, to accomplish complex maneuvers, a large-scale drone swarm still needs a leader controlled by an external commander. If the drone frequently fails to receive state information from the father drone, the swarm cannot achieve global coordination. Moreover, due to changes in the relative position between drones, the topology of the drone swarm changes dynamically and make the spanning tree unstable. Towards this end, a semi-autonomous coordination mode is proposed in Section 4.1.

(b) How to achieve fast coordination based on local interaction: The leader drone in a large-scale drone swarm shares its state information hop-by-hop along the spanning tree, and the information spreads from the leader to the entire swarm like ripples, which fatally brings delay and causes the large-scale drone swarm to fail to coordinately work. With the increment of the drone swarm size, the delay becomes pronounced serious. In this paper, a prediction mechanism is introduced in Section 4.2 to alleviate the delay caused by local interactions.

(c) How to achieve coordination with local interaction of long interval: Drones in large-scale swarms update state information through local interaction with interval communication. To reduce the payload in drone swarms, a relatively long interval should be adopted. However, Eq. (3) is analyzed based on a continuous-time system and cannot be directly applied in the drone swarm communication system based on transceiving packets. Therefore, numerical integration methods are adopted to discretize the coordination algorithm which is described in Section 4.3.

(d) How to achieve coordination while multi-drone swarms merging: For large-scale drone swarms, it is infrequent to take off at the same time due to the large number of drones. A common way of swarm formation is to combine multiple small-scale drone swarms into a large-scale swarm, hence the coordination problem while merging multiple drone swarms is also worth studying. In this paper, the drone swarm merging algorithm is presented in which provides a solution to this problem.

4. Fast coordination approach for large-scale drone swarm

In this section, we propose a fast coordination approach for large-scale drone swarms. To deal with the packet loss and the dynamic topology in a confrontational environment, a semi-autonomous coordination mode for large-scale drone swarm is introduced in Section 4.1. In Section 4.2, a prediction mechanism is presented to ensure large-scale drone swarm achieves coordination in a limited time. A numerical integration method is presented in Section 4.3 which provides a solution to achieve coordination with local interaction of long intervals. Aiming at the multi-swarm merging coordination problem, a drone swarm merging algorithm is adopted in Section 4.4.

4.1. Coordination method in confrontation environment

In this sub-section, to solve the problems of packets losing and to adapt the dynamic topology in a confrontational environment, we propose a semi-autonomous coordination mode for large-scale drone swarm coordination.

As mentioned above, a large-scale drone swarm performs a complex task only if under the control of an external command through a chosen leader drone. It indicates that the state of drones in the swarm must be converged with the leader drone, as shown in Eq. (2). Considering that drones in the swarm cannot ensure connection to the leader directly, a spanning tree with the leader as root is established, and every drone achieves coordination with the leader through coordinating with its father drone as shown in Eq. (3).

However, the drone may lose connection with its father node under a confrontation environment, as a result, Eq. (3) cannot hold all the time. However, it is rare for a node to lose connections to all its neighbors. Inspired by bird flocking, in which every bird could always maintain consistency with its neighbors not only depending on its father node, drones can also achieve coordination through interaction with its neighbors. From this point, Eq. (3) can be modified as follow.

$$\lim_{t\to\infty}(\left\|x_i - \langle x_F\rangle_i\right\| + \left\|x_i - x_j\right\|) = 0, \forall i \in V, \forall j \in N_i \tag{4}$$

There still is a problem of how to deal with the item of $\langle x_F(t)\rangle_i$ when a drone lost connection with its father drone. To solve the problem, we mimic a flock of birds by changing from leader–follower mode to leaderless mode when a drone temporarily loses its father drone. According to the previous research (Chen et al., 2020a), a leaderless drone swarm coordination can be expressed as the following equation:

$$\lim_{t\to\infty}\left(\left\|x_i - \langle x_R\rangle_i\right\| + \left\|x_i - x_j\right\|\right) = 0, \forall i \in V, \forall j \in N_i \tag{5}$$

where $\langle x_R\rangle_i$ is the virtual reference state which means the averaged effect on drone i by all its neighbors. It can be calculated by mean-field:

$$\langle x_R\rangle_i = \sum_{j\in N_i} x_j P\left(x_j\right) \tag{6}$$

here, $P\left(x_j\right)$ represents the probability for state information of drone j and it obeys Gibbs distribution (Li, 2009):

$$P(x) = Z^{-1} \times e^{-\frac{1}{T}E(x)}$$

where the normalized parameter is $Z = \sum_{i\in V} e^{-\frac{1}{T}E(x_i)}$ and T is a free parameter which is assumed to be 1 generally. $E(x)$ is energy function:

$$E(x) = \sum_{i\in V}\left(x_i - \langle x_R\rangle_i\right)^2 + \sum_{i\in V, j\in N_i}\left(x_i - x_j\right)^2 \tag{7}$$

The first term of the Eq. (7) means the overall energy of the drones themselves, which represents the external field acting on the drone, and the second term represents the interactions between adjacent drones.

The different term of $\left\|x_i - \langle x_R\rangle_i\right\|$ compared to Eq. (4) means that drones are capable to converge to the virtual state according to the average interaction of their neighbors with no requirement to receive the state information from the father drone. Therefore, Eq. (5) can be utilized to solve the case where $\langle x_F(t)\rangle_i$ is missing in Eq. (4).

A target state variable $\langle x_T\rangle_i$ is proposed to unify Eqs. (4) and (5), it is defined as:

$$\langle x_T\rangle_i = \begin{cases} \langle x_F\rangle_i, \text{drone } i \text{ can connect with the father node} \\ \langle x_R\rangle_i, \text{otherwise} \end{cases}$$

by using the target state, a semi-autonomous coordination model is obtained as follow.

$$\lim_{t\to\infty}\left(\left\|x_i(t) - \langle x_T(t)\rangle_i\right\| + \left\|x_i(t) - x_j(t)\right\|\right) = 0 \tag{8}$$

The main idea of Eq. (8) is that the drone is controlled by the leader drone when it can receive messages from its father drone. Otherwise, it holds coordination with its neighbor automatically. That means the drone i achieves coordination with the leader by converging with the target state $\langle x_T\rangle_i$. When the father drone has no packet loss, the target state is the state of the father drone; when packet loss occurs, the target state is generated by the average action of neighboring drones, which tends to local leaderless consensus.

By using Eq. (8), the packet loss problem between the drone and its father drone in the confrontational environment is solved effectively by switching between controlled coordination and local automation coordination.

To simplify the calculation, Eq. (8) can be replaced by an equivalent equation:

$$\lim_{t\to\infty}\left[\left(x_i - \langle x_T\rangle_i\right)^2 + \left(x_i - x_j\right)^2\right] = 0, \forall i \in V, \forall j \in N_i \tag{9}$$

We also modify the Eq. (7) as follow.

$$E(x) = \sum_{i\in V}\left(x_i - \langle x_T\rangle_i\right)^2 + \sum_{i\in V, j\in N_i}\left(x_i - x_j\right)^2 \tag{10}$$

Based on the above two equations, the large-scale drone swarm coordination is equivalent to the minimization of energy E.

We still face the problem of dynamic topology. For a large-scale drone swarm working in a confrontational environment, the relative positions of drones change frequently, and the spanning tree with the leader as the root node cannot be maintained stably.

According to Moreau (2005), it is not necessary to ensure that each node is connected for the coordination of dynamic networks, but only to ensure that the network jointly stays connected, which means that the union of network topology graphs is connected in a time period. Therefore, a spanning tree exists all the time by maintaining a dynamic topology tree with the leader as the root node, thereby ensuring that the state of the drone swarm is coordinated with the leader.

We adopt the dynamic spanning-tree algorithm which draws on the reverse path broadcasting technique (Chen et al., 2020a). Drones without father drones achieve local consensus through Eq. (8) and wait to join the spanning tree before the spanning tree is constructed for the first time. Once the spanning tree is constructed, it always exists and transmits the state information of the leader through the dynamic spanning tree algorithm.

4.2. Coordination state prediction mechanism

For the large-scale drone swarm based on local interaction, the state information of the leader is shared by a dynamic spanning tree algorithm with a delay problem inevitably. As the size of the swarm increases, the height of the spanning tree rises, and it takes a long time for state information delivery from the root to the leaf node. In this manner, it is hard to achieve coordination with the leader of the drone swarm.

According to Eq. (8), the drones in the swarm coordinate with the leader by converging with the target state $\langle x_T(t)\rangle_i$. However, drones can only interact with adjacent drones under the system work architecture proposed. At the same time, drones send and update state information in parallel, so the target state $\langle x_T(t)\rangle_i$ of drone i at t is actually the state of its father drone j at $t-1$, which means $\langle x_T(t)\rangle_i = x_j(t-1)$. Obviously, there is a delay error, and such error gets worse when the drone gets further away from the leader.

To deal with the delay caused by local interaction, a prediction mechanism is introduced. The main idea is to use the cubic spline interpolation algorithm to perform curve fitting on the state change of the father drone and predict the state of the parent drone at t with the fitted curve. The target state $\langle x_T(t)\rangle_i$ is replaced by the predicted information to reduce the impact of delay and enables drone swarms to achieve coordination rapidly.

To keep the flight stability of the drone, the change of the drone state usually has inertia, so the state of the father drone can be effectively predicted. As a kind of interpolation algorithm, the cubic spline interpolation algorithm uses multiple low-degree polynomials to fit a smooth state change curve in sections with the characteristics of a simple formula, fast calculation speed, and good stability. Hence it is suitable for drones with limited CPU power and energy.

In general, the cubic spline interpolation algorithm uses four sampling values to ensure the accuracy of the algorithm. Therefore, a state information queue is maintained for each drone, which is used to record four historical target state information x_{t-4}, x_{t-3}, x_{t-2} and x_{t-1}.

Assuming that τ is any time in the interval $[t-4, t-1]$, three endpoint continuous state functions $S_{t-i}(\tau)$ can be determined by x_{t-i} and x_{t-i+1}, $i = 2, 3, 4$. The state function $S_{t-i}(\tau)$ can be defined as:

$$S_{t-i}(\tau) = a_{t-i} + b_{t-i}\left(\tau - \tau_{t-i}\right) + c_{t-i}\left(\tau - \tau_{t-i}\right)^2 + d_{t-i}\left(\tau - \tau_{t-i}\right)^3 \tag{11}$$

Suppose the sampling step is h, then $\tau_t = \tau_{t-2} + 2h$. The predicted state information $\langle x_P(t)\rangle_i$ can be obtained by the third state function $S_{t-2}\left(\tau_{t-2} + 2h\right)$ as follow.

$$\langle x_P(t)\rangle_i = a_{t-2} + 2b_{t-2}h + 4c_{t-2}h^2 + 8d_{t-2}h^3 \tag{12}$$

According to the continuous endpoints of cubic spline interpolation, a_{t-2}, b_{t-2}, c_{t-2} and d_{t-2} can be calculated by the following formulas:

$$\begin{aligned} a_{t-2} &= x_{t-2} \\ b_{t-2} &= \frac{x_{t-1}-x_{t-2}}{h} - \frac{h}{2}m_{t-2} - \frac{h}{6}\left(m_{t-1} - m_{t-2}\right) \\ c_{t-2} &= \frac{m_{t-2}}{2} \\ d_{t-2} &= \frac{m_{t-1}-m_{t-2}}{6h} \end{aligned} \tag{13}$$

where m_{t-2} represents the second-order derivative of the state information at time $t-2$, which can be solved by the following determinant according to the boundary condition (Hussain et al., 2015):

$$\begin{bmatrix} -h & 2h & -h & 0 \\ h & 4h & h & 0 \\ 0 & h & 4h & h \\ 0 & -h & 2h & h \end{bmatrix} \begin{bmatrix} m_{t-4} \\ m_{t-3} \\ m_{t-2} \\ m_{t-1} \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{x_{t-2}-x_{t-3}}{h} - \frac{x_{t-3}-x_{t-4}}{h} \\ \frac{x_{t-1}-x_{t-2}}{h} - \frac{x_{t-2}-x_{t-3}}{h} \\ 0 \end{bmatrix} \tag{14}$$

The predicted state information $\langle x_P(t)\rangle_i$ can be obtained based on Eq. (12), then Eq. (8) can be rewritten as:

$$\lim_{t\to\infty} \left[\left(x_i(t) - \langle x_P(t)\rangle_i\right)^2 + \left(x_i(t) - x_j(t)\right)^2\right] = 0, \quad \forall i \in V, \forall j \in N_i \tag{15}$$

The target state information is determined after the drones interact with their neighbor drones. The predicted state information obtained from Eq. (12) is used to replace the delayed target state information to reduce the influence caused by the local interaction. After introducing the prediction mechanism, the complete fast coordination algorithm for large-scale drone swarm in a confrontation environment, we call it a semi-autonomous coordination algorithm with prediction, is shown in Algorithm 1.

According to the description provided in Algorithm 1, it is assumed that the drone swarm consists of n nodes. Each drone in the swarm performs parallel computation of the coordination state at a fixed interval, resulting in the outer loop iterating n times as it visits each node once. Within each iteration of the outer loop, $\langle x_R\rangle_i$ is calculated by iterating over the adjacent nodes of the current node. Although the number of adjacent nodes for each node can vary, it is assumed that on average, each node has m adjacent nodes. Therefore, the inner loop will iterate m times for each node. Overall, the total number of iterations of the inner loop can be approximated as $n \times m$. Since calculating $\langle x_R\rangle_i$ performs a constant number of operations for each iteration, the computational complexity of the inner loop is $O(1)$. Therefore, the computational complexity of the entire algorithm can be expressed as $O(n \times m)$. As m is assumed to be less than n, the maximum complexity of this algorithm is $O\left(n^2\right)$.

Algorithm 1 The semi-autonomous coordination algorithm with prediction

Input: The leader in the drone swarm.

Output: The state of each drone in the swarm.

Initialization: set the leader as root drone and its layer number equals to 0; set layer number is int_max for other drones and set itself as its father drone.
for Each drone i and every refresh cycle k in parallel **do**
  if Received state from neighbor **then**
    if the neighbor is root drone **then**
      Set the tier of drone i equals to 1.
      Set the father drone of drone i as root node.
    else
      if the neighbor's layer < the drone i's layer **then**
        Set the layer of drone i equals to the neighbor's layer + 1.
        Set the father drone of drone i as the neighbor.
      end if
    end if
    if Refresh cycle timeout **then**
      if Received consensus state from father drone **then**
        Set $\langle x_T\rangle_i = \langle x_F\rangle_i$.
      else
        Set $\langle x_R\rangle_i = \sum_{j\in N_i} x_j P\left(x_j\right)$.
        Set $\langle x_T\rangle_i = \langle x_R\rangle_i$.
      end if
      Push $\langle x_T\rangle_i$ into the state information queue.
      Calculate $\langle x_P(t)\rangle_i$ by (12)∼(14).
    end if
    Update coordination state by using Eq. (15).
  end if
  Broadcast coordination state to its one-hop neighbors.
end for

4.3. Discretization of coordination algorithm

The large-scale drone swarm mentioned in this paper is based on local interaction. The local interaction is realized by transceiving packets and has a communication interval. We assume that the drones in the swarm have the same frequency of sending packets which are represented by iteration k. Based on this, Eq. (15) can be rewritten as:

$$\lim_{k\to\infty} \left[\left(x_i(k) - \langle x_P(k)\rangle_i\right)^2 + \left(x_i(k) - x_j(k)\right)^2\right] = 0, \quad \forall i \in V, \forall j \in N_i \tag{16}$$

To solve the coordination model of Eq. (16), the method of parallel energy minimization is adopted. The energy function of state information in the drone swarm is rewritten as:

$$E[x(k)] = \sum_{i\in V} \left\{ x_i(k) - \langle x_P(k)\rangle_i \right\}^2 + \sum_{i\in V, j\in N_i} \left[x_i(k) - x_j(k)\right]^2 \tag{17}$$

According to the Ising model, the drone swarm reaches stable coordination and the gradient of the energy function becomes a zero vector, which means the following equation should be satisfied when the energy function is minimized:

$$\frac{\partial E[x(k)]}{\partial x_i(k)} = 2 \times \left\{ \sum_{i\in V} \left[x_i(k) - \langle x_P(k)\rangle_i\right] + \sum_{i\in V} \sum_{j\in N_i} \left[x_i(k) - x_j(k)\right] \right\} = 0 \tag{18}$$

To solve Eq. (18), it is required to sum the state information of each drone in the swarm, which will increase the calculation burden and the network load. If Eq. (18) is rewritten as:

$$\frac{\partial E\left[x\left(k\right)\right]}{\partial x_i\left(k\right)} = \sum_{i\in V}\left\{ 2\times\left\{ \left[x_i\left(k\right) - \left\langle x_P\left(k\right)\right\rangle_i\right] + \sum_{j\in N_i}\left[x_i\left(k\right) - x_j\left(k\right)\right]\right\}\right\} \tag{19}$$

and considering each drone in the swarm individually, the following formula is obtained:

$$\frac{\partial E\left[x_i\left(k\right)\right]}{\partial x_i\left(k\right)} = 2\times\left\{ \left[x_i\left(k\right) - \left\langle x_P\left(k\right)\right\rangle_i\right] + \sum_{j\in N_i}\left[x_i\left(k\right) - x_j\left(k\right)\right]\right\} \tag{20}$$

Eq. (19) then can be described as:

$$\frac{\partial E\left[x\left(k\right)\right]}{\partial x_i\left(k\right)} = \sum_{i\in V}\frac{\partial E\left[x_i\left(k\right)\right]}{\partial x_i\left(k\right)} \tag{21}$$

The above equation shows that the global energy function could be minimized through the energy function for each drone to obtain the minimum. According to Eq. (18), the energy function is a strictly convex function in its domain, which has only one global minimum point (Hindi, 2004). The drone swarm is capable to achieve coordination at the global minimum point. In actual applications, the following equation is used to solve the energy minimization function for each drone in the swarm:

$$\frac{x_i\left(k+1\right) - x_i\left(k\right)}{\mu} = -2\times\left\{ \left[x_i\left(k\right) - \left\langle x_P\left(k\right)\right\rangle_i\right] + \sum_{j\in N_i}\left[x_i\left(k\right) - x_j\left(k\right)\right]\right\} \tag{22}$$

where μ is a small constant depending on the experimental accuracy and numerical integral step size. As μ approaches to 0, Eq. (22) is equivalent to:

$$\dot{x}_i = -1\times\left[\left(x_i - \left\langle x_P\right\rangle_i\right) + \sum_{j\in N_i}\left(x_i - x_j\right)\right] \tag{23}$$

Assuming that x_P can accurately predict x_T. Eq. (23) can be rewritten as:

$$\dot{x}_i = -1\times\left[\left(x_i - \left\langle x_T\right\rangle_i\right) + \sum_{j\in N_i}\left(x_i - x_j\right)\right] \tag{24}$$

Substituting Eq. (6) into Eq. (24), the above equation is rewritten as:

$$\dot{x}_i = -1\times\left[\left(x_i - \sum_{j\in N_i} x_j P(x_j)\right) + \sum_{j\in N_i}\left(x_i - x_j\right)\right] \tag{25}$$

Assuming a drone swarm with n nodes is represented by the adjacency matrix $A = [a_{ij}] \in R^{n\times n}$. In the context of drone swarms, it is typical for the drones to communicate through a bidirectional link, which enables us to consider the drone swarm as an undirected graph. If nodes i and j are adjacent neighbors, the element a_{ij} set to 1. Otherwise, a_{ij} is 0. We can further transform Eq. (25) by introducing the adjacency matrix A:

$$\dot{x} = -Lx \tag{26}$$

where

$$L_{ii} = 1 + \sum_{j\in n} a_{ij} = 1 + D_i, L_{ij} = -a_{ij}\left(1 + P\left(x_j\right)\right) \tag{27}$$

where D_i is the degree of node i. It is clear that the matrix L is diagonally dominant, with all its diagonal elements being positive. As a result, the real parts of L's eigenvalues are also positive, which demonstrates the convergence of the system, as pointed out in Ren and Beard (2007).

In the drone swarm based on local interaction, μ is the communication interval between drones which cannot be ignored. Therefore, the drone swarm system is a nonlinear discrete system that needs numerical integration methods to solve Eq. (22).

The Runge–Kutta integral method is an iterative numerical integration algorithm. The function is discretized by increasing the order of the iterative formula. The prior value is applied to calculate the new value and subsequently the average value can be obtained which continuously approximates the true value of the variable. With the higher order, the accuracy of the algorithm is improved while the computation load increases. Generally, the fourth-order Runge–Kutta method is suitable in the engineering field.

According Eq. (22), the Runge–Kutta function is defined as:

$$f\left[k, x_i\left(k\right)\right] = \left\{ \left[x_i\left(k\right) - \left\langle x_P\left(k\right)\right\rangle_i\right] + \sum_{j\in N_i}\left[x_i\left(k\right) - x_j\left(k\right)\right]\right\} \tag{28}$$

The segmented slopes are calculated as follows

$$\begin{aligned} K_1 &= f\left(k, x_i\left(k\right)\right) \\ K_2 &= f\left(k + \tfrac{\mu}{2}, x_i\left(k\right) + \tfrac{\mu}{2}K_1\right) \\ K_3 &= f\left(k + \tfrac{\mu}{2}, x_i\left(k\right) + \tfrac{\mu}{2}K_2\right) \\ K_4 &= f\left(k + \mu, x_i\left(k\right) + \mu K_3\right) \end{aligned} \tag{29}$$

where K_1 is the slope at the beginning of the sampling interval k. K_2 is the slope at the midpoint of the sampling interval, and the Euler method which is an alternative numerical algorithm is adopted to determine the value of $x_i(k)$ at point $k + \frac{\mu}{2}$ by using the slope K_1. K_3 is also the slope of the midpoint while the slope K_2 is used to determine the $x_i(k)$ value. K_4 is the slope at the end of the sampling interval k, and its $x_i(k)$ value is determined by K_3.

The final slope is obtained by the weighted average of the four segmented slopes, which can be used to calculate the value of the state information at $k+1$ with higher accuracy. The calculation equation is as follows:

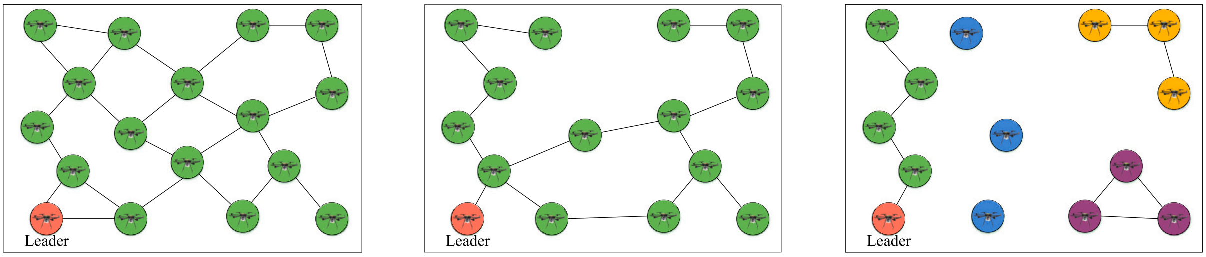
$$x_i\left(k+1\right) = x_i\left(k\right) + \frac{\left(K_1 + 2K_2 + 2K_3 + K_4\right)}{6}\mu \tag{30}$$

Through the fourth-order Runge–Kutta method, the coordination state energy function of the drone swarm based on local interaction is solved.

4.4. Multi-drone swarms merging coordination

In practical applications, it is not realistic for large-scale drone swarms to take off simultaneously in a certain place. To improve the achievement ratio of the missions like area reconnaissance or search and precision strikes on a target, multi-drone swarms usually take off and reach coordination separately. Drone swarms fly to the mission area from different directions and merge into a large-scale drone swarm over the mission area to start the mission. Different from the traditional flight formation control methods, a self-organizing algorithm to achieve merging coordination is proposed for system architecture, in which the drones do not need to receive instructions from the leader or the ground station to complete the merging behavior. To minimize the rendezvous time and energy consumption of the drone swarm, each drone makes its own decision to change the working model and choose the new father drone when the state information from other swarms is perceived. The multi-swarm merging method optimizes the energy consumption in flight, and the computing time does not increase significantly with the enlargement of the drones.

The leader drones in each swarm are directly controlled by the ground station to ensure each leader maintains a priority queue that



(a) The topology of the drone swarm (b) The topology changing with drones lost (c) The topology splitting with drones lost

Fig. 2. The key drones losing leads topology changes.

contains all leaders' state information. A drone swarm with a higher-priority leader is able to accept a lower-priority leader with its followers. If the leader in a swarm becomes a follower of one of the other swarms, the rest of the drones in this swarm subsequently change their working model to leaderless and reach local consensus with neighboring drones until receiving a new leader's state information. Note that if a leader i receives a message from a non-leader drone j in another high-priority drone swarm, the leader i will become a follower of drone j. It ensures that the merging algorithm can be triggered when the edges of two drone swarms interact with each other.

The details of the complete multi-drone swarms merging algorithm are shown in Algorithm 2. Similar to Algorithm 1, the multi-drone swarms merging algorithm also achieves a time complexity of $O\left(n^2\right)$ by performing parallel computation of the drone's state.

5. Performance evaluation

To validate the effectiveness of the fast coordination approach for large-scale drone swarms, simulations are implemented using QualNet. The effectiveness is verified by applying the proposed approach to scenarios with different numbers of drone swarms. The comparison results of proposed schemes with other traditional methods are presented. Moreover, some highly realistic simulations are conducted for the scenarios where key drones are lost and multi-drone swarms merge.

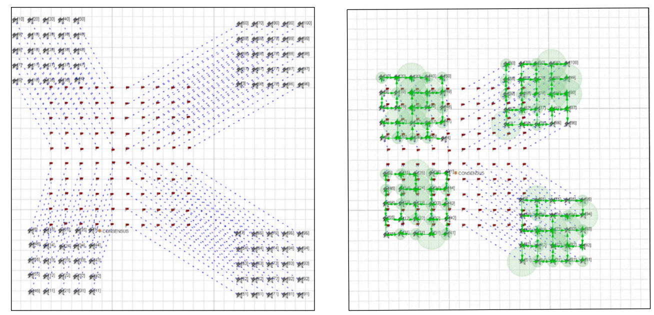
5.1. Simulation settings

In most hypothetical application scenarios, drone swarms fly to the target area in fixed formations and collaborate on tasks. Without loss of generality, a fixed formation is devised in QualNet, and a space area of 3000 m × 3000 m is set as the congregation area for the drone swarm. Considering the need for achieving consensus within a few minutes, the simulation time is specified as 300 s. Various communication technologies are available for drone networks, such as IEEE 802.11, IEEE 802.15.4, LTE, etc. Among these technologies, IEEE 802.11b offers a relatively moderate transmission rate and consumption. It has been adopted by some actual drones like AR Drone 2.0. Therefore, the physical layer and the MAC protocol of each drone are chosen as 802.11b. For further details of the parameters, please refer to Table 1.

Firstly, we deployed 50, 100, 200, and 300 drones to mimic the swarm scenario with different scales of drones. The performance of the proposed approach for tracking state information of the leader drone is evaluated. The simulation results are compared with the average consensus method, which is a widely-used consensus method (Ren and Cao, 2010). Moreover, we compare the results with the traditional way of transceiving commands from the leader.

Secondly, we deploy 100 drones to mimic the leader–follower swarm scenario to compare the performance with and without the prediction mechanism. The performance with and without the four-order Runge–Kutta method is also evaluated.

Thirdly, the topology of the drone swarm in a confrontational environment is changed as shown in Fig. 2. With the drone gradually



(a) Initial configuration of four drone swarms (b) The process of merging four drone swarms

Fig. 3. Merging simulation scenario of multiple drone swarms.

Table 1
Simulation parameters of Qualnet.

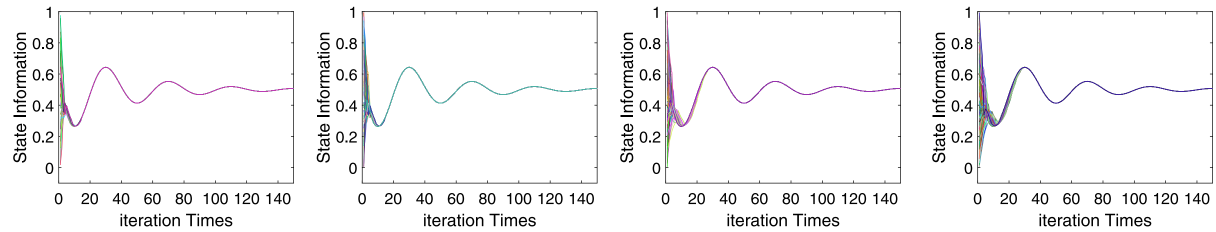
Parameter	Value
Map size (km × km)	3.0 × 3.0
Number of drones	50/100/200/300
Formation	Grid
Simulation time (s)	300
Antenna model	Omnidirectional
Physical layer	802.11b radio
Data rate (Kbps)	500
Transmission power (dbm)	15
Pathloss model	Two ray
Shadowing model	Constant
Fading model	Rayleigh
Noise factor (db)	10
MAC layer	802.11

lost, the drone swarm changes the topology from Figs. 2(a) to 2(b). As the number of losing drones increases, the drone swarm may split into multiple new sub-swarms or a single drone as Fig. 2(c). We simulate the scenario of drone loss to verify that the proposed system architecture is robust to the confrontational environment (i.e., some drones may be attacked and lose their communication abilities).

Finally, the scenario of merging multiple drone swarms is simulated in Fig. 3. At the initial configuration, four drone swarms with random formations are devised. We preset swarms merging instructions and observe the convergence of consensus information during the merging process. 4 × 25 and 4 × 50 drones are deployed to verify the effectiveness of the drone swarm merging algorithm.

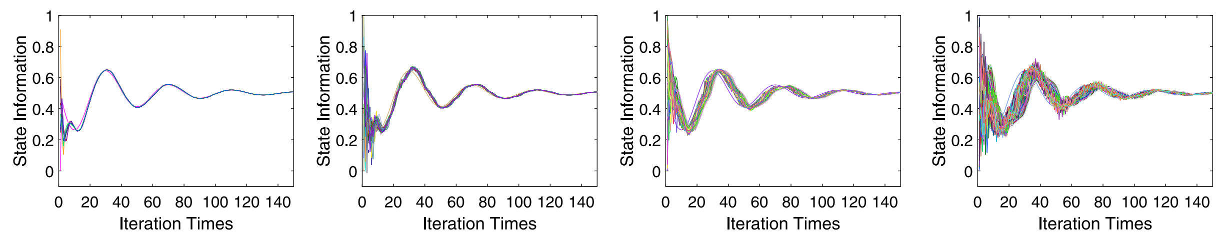
5.2. Numerical results

We evaluate the performance of the proposed semi-autonomous coordination algorithm in tracking the dynamic target state information from the leader for different scale drone swarms and the results are shown in Fig. 4. As can be seen, the state of the large-scale drone swarm can quickly reach coordination with the leader due to the delay



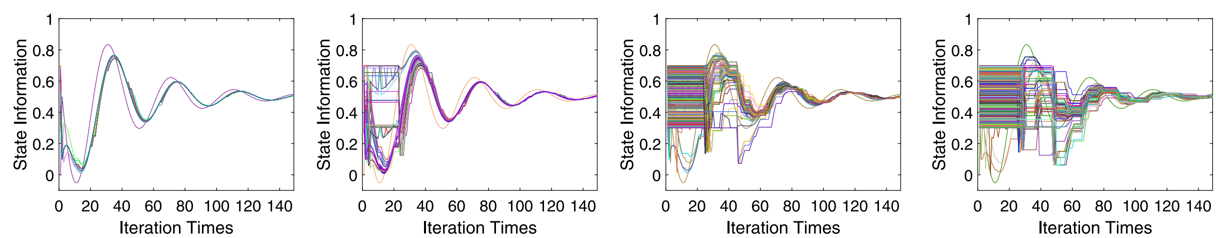
(a) Coordination model of 50 drones (b) Coordination model of 100 drones (c) Coordination model of 200 drones (d) Coordination model of 300 drones

Fig. 4. Dynamic target state coordination convergence process of fast coordination method for different scale drone swarm.



(a) Average consensus of 10 drones (b) Average consensus of 50 drones (c) Average consensus of 100 drones (d) Average consensus of 200 drones

Fig. 5. Dynamic target state coordination convergence process of average consensus for different scale drone swarm.



(a) Transceiving instructions method of 50 drones (b) Transceiving instructions method of 100 drones (c) Transceiving instructions method of 200 drones (d) Transceiving instructions method of 300 drones

Fig. 6. Dynamic target state coordination convergence process of transceiving instructions method for different scale drone swarm.

compensation based on the prediction mechanism, and the convergence time is not affected by the size of the swarm, which is kept within 20 iterations. With the growth of the scale, the convergence time is still kept within 40 iterations.

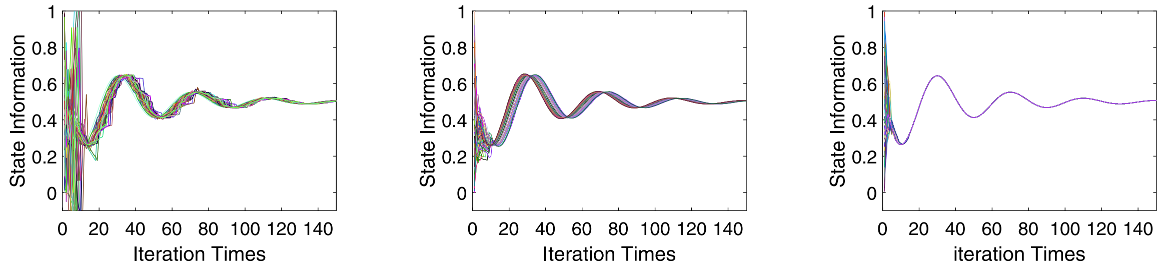
Fig. 5 shows the convergence process with average consensus for different scale drone swarms. In Fig. 5(a), a drone swarm of 10 drones can achieve coordination by using the average consensus method. With the growth of the scale, the swarm state can eventually converge with an obvious system delay error as shown in Fig. 5(b). In Figs. 5(c) and 5(d), the delay error becomes significantly larger when the number of drone swarms reaches 100, and the system cannot converge when the number reaches 200.

The process of tracking the dynamic target state of the drone swarm using traditional transceiving instructions is shown in Fig. 6, which is based on multi-hop forwarding communication and instruction distribution. The leader's movement is controlled by the ground station, while other drones receive instructions and follow the leader. Fig. 6(a) shows that when the number of drones is 50, the drone swarm can eventually converge but the time delay is also highly obvious. Because a large number of drones will compete for limited channel resources when performing multi-hop forwarding communication, resulting in serious access delays. The delay becomes serious as the number of drones increases. Some drones cannot receive information from the leader, so the initial value is maintained as a straight line in Fig. 6(b). As shown in Figs. 6(c) and 6(d), it is obvious that more drones cannot be able to maintain consensus with the leader when the formation size reaches 200. As the formation size increases to 300, a large number of drones cannot receive the leader's consensus information in time, therefore, the entire drone swarm becomes very chaotic.

Due to the high fidelity of the experimental simulations, the experimental results exhibit small fluctuations that are within an acceptable range considering the noise and packet loss in communication among drones. By comparing Figs. 4, 5, and 6, it is evident that as the scale of the drone swarm increases, the superiority of the proposed method becomes more significant, ensuring rapid coordination state convergence in large-scale drone swarms. When comparing drone swarms of the same scale using different coordination methods, the fast coordination method also demonstrates significant advantages in terms of fast convergence speed and effective coordination.

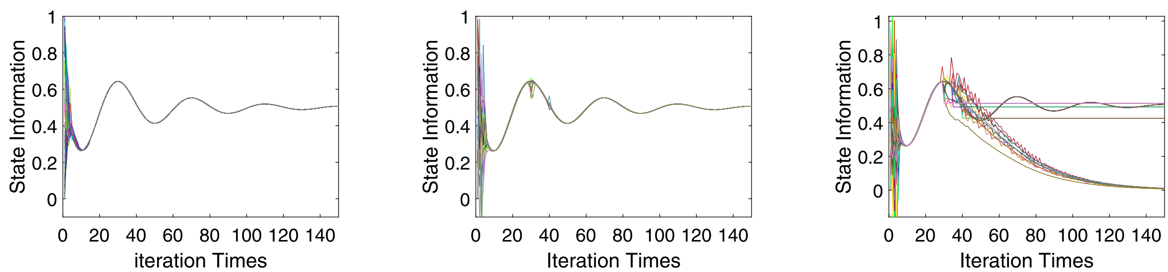
As shown in Figs. 7(a) and 7(b), an unprocessed coordination model leads to significant errors, and the system cannot converge. In contrast, using the Runge–Kutta method reduces the errors, and the state variations of the drones in the swarm tend to be consistent, although a notable delay can be observed compared to the leader.

The proposed prediction mechanism offers a reliable and efficient way to predict the state of the father node, thereby greatly enhancing the overall performance of the consensus algorithm and reducing tracking errors. These advantages are clearly demonstrated through simulation results. As shown in Fig. 7(b) and 7(c), the prediction mechanism brings a significant improvement in coordination speed.



(a) The coordination model without Runge-Kutta and prediction methods (b) The coordination model with Runge-Kutta method (c) The coordination model with Runge-Kutta method and prediction mechanism

Fig. 7. The coordination state change under different processing methods.



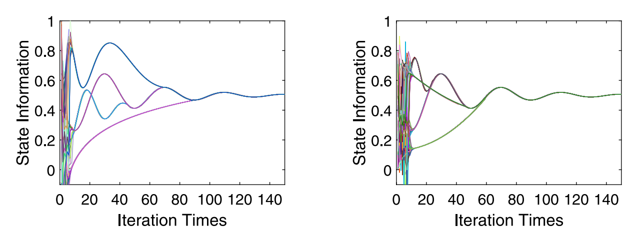
(a) The state information without drones losing (b) The state information of topology changing (c) The state information of topology splitting

Fig. 8. The changing of state information with key drones losing.

Algorithm 2 The multi-drone swarms merging algorithm

Input: The leader priority queue.

Initialization: The merging command is sent to each leader from the ground station.
for Each drone i and refresh cycle k in parallel **do**
  if Received state from neighbor drone j **then**
    if Drone i is LEADER **then**
      if The leader of drone j > drone i in priority queue **then**
        Set drone i as FOLLOWER.
        Set drone j as the father drone of i.
        Set $\langle x_T \rangle_i = \langle x_F \rangle_i$.
      end if
    end if
    if Drone i is FOLLOWER **then**
      if The leader of drone j > the leader of drone i in priority queue **then**
        Set drone i as LEADERLESS.
        Set $\langle x_T \rangle_i = \langle x_R \rangle_i$.
      end if
    else if Drone i is LEADERLESS **then**
      Set drone i as FOLLOWER.
      Set drone j as the father drone of i.
      Set $\langle x_T \rangle_i = \langle x_F \rangle_i$.
    end if
    Put $\langle x_T \rangle_i$ into the state information queue.
    Calculate $\langle x_P(t) \rangle_i$ by (12) ∼ (14).
    Update consensus state by using (15).
  end if
  Broadcast the state information to its one-hop neighbors.
end for



(a) A scenario with 4×25 drones (b) A scenario with 4×50 drones

Fig. 9. The changing of state information in the process of merging four drone swarms.

The drones in the swarm achieve convergence of coordination state within 20 iterations and maintain stable coordination, providing strong evidence that the coordination state prediction mechanism plays a crucial role in facilitating rapid coordination of large-scale drone swarms.

Moreover, we simulated the scenario where the key drones in the swarm were lost in a confrontational environment. The simulation result is shown in Fig. 8. As shown in Fig. 8(b), the state of the drone swarm fluctuates slightly when a few drones are lost, and the coordination can be quickly restored. Fig. 8(c) shows that some drones can directly or indirectly interact with the leader to maintain stable coordination even though the key drone in the drone swarm is lost. Some drones that are unable to interact with the leader from the new swarms can achieve local leaderless consensus. The other individual drones maintain the previous state information. Although the drone swarm is split, it still maintains local stability and is capable to complete coordinative work, which means that the proposed system architecture has good robustness to the loss of key drones in confrontational environments.

Finally, we conducted the merging simulation of multiple drone swarms. Two scenarios were deployed, with one containing 4×25 drones and the other containing 4×50 drones. The leader drones in the two scenarios have different dynamic state information. Fig. 3 shows the simulation results in which four drone swarms reach different consensus on their own firstly. After that, the swarms begin to merge when the drones of different swarms are able to communicate immediately with each other. Fig. 9 shows the changes in state information when the four drone swarms are merging. It turns out that the proposed fast coordination approach is able to achieve multi-swarm merged coordination.

6. Conclusion

This paper proposed a system architecture and semi-autonomous coordination algorithm based on local interaction for large-scale drone swarms. Drones can interact with adjacent drones and then achieve global coordination with regard to leaders by using the proposed approaches. The architecture and algorithm ensure that large-scale drone swarms can quickly reach coordination under the conditions of limited communication resources and node loss. A prediction mechanism is introduced to address the problem of time delay caused by local interaction during information delivery. Moreover, we adapt the Runge–Kutta method to discretize the algorithm, enabling the algorithm to be applied in large-scale drone swarm systems. A merging algorithm is also proposed to solve the multi-drone swarms merging problem. The simulation results show that the fast coordination approach can achieve global coordination rapidly for large-scale drone swarms and merge multi-swarm in confrontational environments, which has promising applications in practical scenarios.

In this paper, our main focus is on the leader–follower coordination approach of drone swarms. Given that drone swarms operate in a confrontational environment, it is possible for the leader drone to lose its connection with the control center. Therefore, future research should prioritize enhancing the collaborative decision-making capability of drone swarms. This would enable drones to possess autonomous decision-making abilities, continuous learning capabilities, and adaptability to dynamic environments. Consequently, it would facilitate the achievement of a fully decentralized and distributed cooperative working mode for the drone swarm system.

CRediT authorship contribution statement

Wu Chen: Conceptualization, Methodology, Software, Writing – original draft. **Jiayi Zhu:** Validation, Software, Data curation, Visualization, Investigation, Writing – original draft. **Jiajia Liu:** Supervision, Project administration, Funding acquisition. **Hongzhi Guo:** Formal analysis, Writing – reviewing & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

This work was supported by National Key R&D Program of China (2022YFB3104200), in part by National Natural Science Foundation of China (62202386), in part of Project funded by China Postdoctoral Science Foundation (2023M732861), and in part by Special Funds for Central Universities Construction of World-Class Universities (Disciplines) and Special Development Guidance (0639023GH0202091 and 0639023SH0201091).

References

Basiri, M., Schill, F., Floreano, D., Lima, P.U., 2014. Audio-based localization for swarms of micro air vehicles. In: 2014 IEEE International Conference on Robotics and Automation (ICRA). pp. 4729–4734.

Berlinger, F., Gauci, M., Nagpal, R., 2021. Implicit coordination for 3D underwater collective behaviors in a fish-inspired robot swarm. Science Robotics 6 (50), 1–15.

Brambilla, M., Ferrante, E., Birattari, M., Dorigo, M., 2013. Swarm robotics: a review from the swarm engineering perspective. Swarm Intell. 7 (1), 1–41.

Carli, R., Cavone, G., Epicoco, N., Ferdinando, M.D., Scarabaggio, P., Dotoli, M., 2020. Consensus-based algorithms for controlling swarms of unmanned aerial vehicles. In: International Conference on Ad-Hoc Networks and Wireless(ADHOC-now 2020). pp. 84–99.

Chen, W., Liu, J., Guo, H., 2020a. Achieving robust and efficient consensus for large-scale drone swarm. IEEE Trans. Veh. Technol. 69 (12), 15867–15879.

Chen, W., Liu, J., Guo, H., Kato, N., 2020b. Toward robust and intelligent drone swarm: Challenges and future directions. IEEE Netw. 34 (4), 278–283.

Chmaj, G., Selvaraj, H., 2015. Distributed processing applications for UAV/drones: a survey. In: Progress in Systems Engineering. pp. 449–454.

Coppola, M., McGuire, K.N., De Wagter, C., De Croon, G.C., 2020. A survey on swarming with micro air vehicles: Fundamental challenges and constraints. Front. Robot. AI 7 (20), 1–18.

Cui, Q., Liu, P., Wang, J., Yu, J., 2017. Brief analysis of drone swarms communication. In: 2017 IEEE International Conference on Unmanned Systems (ICUS). pp. 463–466.

Doostmohammadian, M., Taghieh, A., Zarrabi, H., 2021. Distributed estimation approach for tracking a mobile target via formation of UAVs. IEEE Trans. Autom. Sci. Eng. 19 (4), 3765–3776.

Dorigo, M., Floreano, D., Gambardella, L.M., Mondada, F., Nolfi, S., Baaboura, T., Birattari, M., Bonani, M., Brambilla, M., Brutschy, A., et al., 2013. Swarmanoid: a novel concept for the study of heterogeneous robotic swarms. IEEE Robot. Autom. Mag. 20 (4), 60–71.

Hindi, H., 2004. A tutorial on convex optimization. In: Proceedings of the 2004 American Control Conference(ACC). pp. 3252–3265.

Hönig, W., Preiss, J.A., Kumar, T.S., Sukhatme, G.S., Ayanian, N., 2018. Trajectory planning for quadrotor swarms. IEEE Trans. Robot. 34 (4), 856–869.

Hou, R., Zhou, S., Zheng, Y., Dong, M., Ota, K., Zeng, D., Luo, J., Ma, M., 2021. Cluster routing-based data packet backhaul prediction method in vehicular named data networking. IEEE Trans. Netw. Sci. Eng. 8 (3), 2639–2650.

Huang, S., Zeng, D., Qu, Z., 2022. Toward performance efficient UAV task scheduling in cloud native edge. In: GLOBECOM 2022 - 2022 IEEE Global Communications Conference. pp. 4517–4522.

Hussain, M.Z., Irshad, M., Sarfraz, M., Zafar, N., 2015. Interpolation of discrete time signals using cubic spline function. In: 2015 19th International Conference on Information Visualisation(IV). pp. 454–459.

Hussen, H.R., Choi, S.-C., Kim, J., Park, J.-H., 2017. Stateless and predictive geographic multicast scheme in flying ad-hoc networks. In: 2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN). pp. 685–690.

Kallenborn, Z., 2018. The era of the drone swarm is coming, and we need to be ready for it. Modern War Institute at West Point 25.

Li, S.Z., 2009. Markov Random Field Modeling in Image Analysis. Springer-Verlag, NY, USA.

Liu, X., Cao, J., Xie, C., 2019. Finite-time and fixed-time bipartite consensus of multi-agent systems under a unified discontinuous control protocol. J. Franklin Inst. B 356 (2), 734–751.

Liu, C., Wang, X., Ren, Y., Wang, X., Zhang, J., 2022. A novel distributed secondary control of heterogeneous virtual synchronous generators via event-triggered communication. IEEE Trans. Smart Grid 13 (6), 4174–4189.

McCune, R.R., Madey, G.R., 2013. Swarm control of UAVs for cooperative hunting with DDDAS. Procedia Comput. Sci. 18 (3), 2537–2544.

Moreau, L., 2005. Stability of multiagent systems with time-dependent communication links. IEEE Trans. Autom. Control 50 (2), 169–182.

Nägeli, T., Conte, C., Domahidi, A., Morari, M., Hilliges, O., 2014. Environment-independent formation flight for micro aerial vehicles. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS). pp. 1141–1146.

Onthoni, D.D., Sahoo, P.K., Atiquzzaman, M., 2023. ASAA: Multihop and multiuser channel hopping protocols for cognitive-radio-enabled internet of things. IEEE Internet Things J. 10 (9), 8305–8318.

Orfanus, D., De Freitas, E.P., Eliassen, F., 2016. Self-organization as a supporting paradigm for military UAV relay networks. IEEE Commun. Lett. 20 (4), 804–807.

Oubbati, O.S., Atiquzzaman, M., Lim, H., Rachedi, A., Lakas, A., 2022. Synchronizing UAV teams for timely data collection and energy transfer by deep reinforcement learning. IEEE Trans. Veh. Technol. 71 (6), 6682–6697.

Oubbati, O.S., Lakas, A., Lorenz, P., Atiquzzaman, M., Jamalipour, A., 2021. Leveraging communicating UAVs for emergency vehicle guidance in urban areas. IEEE Trans. Emerg. Top. Comput. 9 (2), 1070–1082.

Pal, M., 2022. A sliding mode based finite-time consensus protocol for heterogeneous multi agent UAS. In: 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS). pp. 1–7.

Pasek, P., Kaniewski, P., 2022. A review of consensus algorithms used in distributed state estimation for UAV swarms. In: 2022 IEEE 16th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET). pp. 472–477.

Ren, W., Beard, R.W., 2007. Distributed Consensus in Multi-Vehicle Cooperative Control: Theory and Applications. Springer Publishing Company, Incorporated.

Ren, W., Cao, Y., 2010. Distributed Coordination of Multi-Agent Networks: Emergent Problems, Models, and Issues. Springer-Verlag, NY, USA.

Rikos, A.I., Charalambous, T., Johansson, K.H., Hadjicostis, C.N., 2022. Distributed event-triggered algorithms for finite-time privacy-preserving quantized average consensus. IEEE Trans. Control Netw. Syst. 1–12.

Rubenstein, M., Cornejo, A., Nagpal, R., 2014. Programmable self-assembly in a thousand-robot swarm. Science 345 (6198), 795–799.

Saballa, J., 2022. US army to test largest-ever drone swarm. https://www.thedefensepost.com/2022/04/26/us-army-drone-swarm/.

Saffre, F., Hildmann, H., Karvonen, H., 2021. The design challenges of drone swarm control. In: International Conference on Human-Computer Interaction(HCII 2021). pp. 408–426.

Tahir, A., Böling, J., Haghbayan, M.-H., Toivonen, H.T., Plosila, J., 2019. Swarms of unmanned aerial vehicles—a survey. J. Ind. Inf. Integr. 16 (4), 1–16.

Tegicho, B.E., Geleta, T.N., Bogale, T.E., Eroglu, A., Edmonson, W., Bitsuamlak, G., 2021. Effect of wind on the connectivity and safety of large scale UAV swarms. In: 2021 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom). pp. 1–6.

Tian, W., Zhao, Y., Hou, R., Dong, M., Ota, K., Zeng, D., Zhang, J., 2023. A centralized control-based clustering scheme for energy efficiency in underwater acoustic sensor networks. IEEE Trans. Green Commun. Netw. 7 (2), 668–679.

Wang, Z., Guo, J., Chen, Z., Yu, L., Wang, Y., Rao, H., 2022a. Robust secure UAV relay-assisted cognitive communications with resource allocation and cooperative jamming. J. Commun. Netw. 24 (2), 139–153.

Wang, J., Zhang, Z., Tian, B., Zong, Q., 2022b. Event-based robust optimal consensus control for nonlinear multiagent system with local adaptive dynamic programming. IEEE Trans. Neural Netw. Learn. Syst. 1–14.

Yanmaz, E., Yahyanejad, S., Rinner, B., Hellwagner, H., Bettstetter, C., 2018. Drone networks: Communications, coordination, and sensing. Ad Hoc Netw. 68 (1), 1–15.

Zou, Y., Zhu, J., Wu, T., Guo, H., Wei, H., 2021. Cooperative drone communications for space-air-ground integrated networks. IEEE Netw. 35 (5), 100–106.

Wu Chen (chenwu@nwpu.edu.cn)

National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology, School of Cybersecurity, Northwestern Polytechnical University, Xi'an, China.

Wu Chen received the B.S. and M.S. degrees in navigation guidance and control from Northwestern Polytechnical University, Xi'an, China, in 1992 and 1997, respectively, and the Ph.D. degree in control theory and control engineering from Northwestern Polytechnical University, in 2000. He is currently an Associate Professor with the School of Cybersecurity, Northwestern Polytechnical University. His research interests include Ad Hoc network, intelligent control, embedded system technology, and information security.

Jiayi Zhu (15039565129@mail.nwpu.edu.cn)

National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology, School of Cybersecurity, Northwestern Polytechnical University, Xi'an, China.

Jiayi Zhu received the B.S. degree in computer science and technology from Hainan University, Haikou, China, in 2020. He is currently working toward the M.S. degree in cyberspace security with Northwestern Polytechnical University, Xi'an, China. His research interests include drone countermeasures, Ad Hoc network, and cybersecurity.

Jiajia Liu (Liujiajia@nwpu.edu.cn)

National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology, School of Cybersecurity, Northwestern Polytechnical University, Xi'an, China.

Jiajia Liu received the B.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2004, the M.S. degree in computer science from Xidian University, Xi'an, China, in 2009, and the Ph.D. degree in information sciences from Tohoku University, Sendai, Japan, in 2012.

He was a Full Professor (Vice Dean) with the School of Cybersecurity, Northwestern Polytechnical University, Xi'an, China. He has authored or coauthored more than 220 peer-reviewed papers in many high quality publications, including prestigious IEEE journals and conferences. His research interests include intelligent and connected vehicles, mobile/edge/cloud computing and storage, IoT security, wireless and mobile ad hoc networks, and SAGIN. He was the recipient of the IEEE ComSoc Best YP in Academia Award in 2020, the IEEE VTS Early Career Award in 2019, the IEEE ComSoc Asia–Pacific Outstanding Young Researcher Award in 2017, and the IEEE ComSoc Asia–Pacific Outstanding Paper Award in 2019. He has been actively joining the society activities, including an Associate Editor for IEEE Transactions on Wireless Communications since May 2018, IEEE Transactions on Computers from October 2015 to June 2017, and IEEE Transactions on Vehicular Technology from January 2016 to December 2020, and the Editor for IEEE Network since July 2015, and IEEE Transactions on Cognitive Communications and Networking since January 2019. He is the Chair of IEEE IoT, Ad Hoc and Sensor Networks Technical Committee and the Distinguished Lecturer of the IEEE Communications Society and Vehicular Technology Society.

Hongzhi Guo (hongzhi.guo@nwpu.edu.cn)

National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology, School of Cybersecurity, Northwestern Polytechnical University, Xi'an, China.

Hongzhi Guo (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in computer science and technology from the Harbin Institute of Technology, Harbin, China, in 2004, 2006, and 2011, respectively. He is currently an Associate Professor with the School of Cybersecurity, Northwestern Polytechnical University, Xi'an, China. He has authored or coauthored more than 40 peer-reviewed papers in many prestigious IEEE journals and conferences. His research interests include edge computing, SAGSIN, IoT security, AI security, and 5G security. He was the recipient of the WiMob Best Paper Award in 2019, and the IEEE TVT Top Reviewer Award in 2019. He has been actively joining the society activities, including an Associate Editor for IEEE Transactions on Vehicular Technology since January 2021, Frontiers in Communications and Network since January 2021, and the Editor for International Journal of Multimedia Intelligence and Security since March 2019.