This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2024.3505206

Smart Shield: Prevent Aerial Eavesdropping via Cooperative Intelligent Jamming Based on Multi-Agent Reinforcement Learning

Qubeijian Wang, *Member, IEEE*, Shiyue Tang, Wen Sun, *Senior Member, IEEE*, Yin Zhang, *Senior Member, IEEE*, Geng Sun, *Senior Member, IEEE*, Hong-Ning Dai, *Senior Member, IEEE*, Mohsen Guizani, *Fellow, IEEE*

***Abstract*—The spotlight on unmanned aerial vehicles (UAVs) is to enhance wireless communications while ignoring the potential risk of UAVs acting as adversaries. Due to their mobility and flexibility, UAV eavesdroppers pose an immeasurable threat to legitimate wireless transmissions. However, the existing fixed jamming scheme without cooperation cannot counter the flexible and dynamic UAV eavesdropping. In this article, a cooperative intelligent jamming scheme is proposed, authorizing ground jammers (GJs) to interfere with UAV eavesdroppers, generating specific jamming shields between UAV eavesdroppers and legitimate users. Toward this end, we formulate a secrecy capacity maximization problem and model the problem as a decentralized partially observable Markov decision process (Dec-POMDP). To address the challenge of the huge state space and action space with network dynamics, we leverage a deep reinforcement learning (DRL) algorithm with a dueling network and double-Q learning (i.e., dueling double deep Q-network) to train policy networks. Then, we propose a multi-agent mixing network framework (QMIX)-based collaborative jamming algorithm to enable GJs to independently make decisions without sharing local information. Additionally, we perform extensive simulations to validate the superiority of our proposed scheme and present useful insights into practical implementation by elucidating the relationship between the deployment settings of GJs and the instantaneous secrecy capacity.**

***Index Terms*—Anti-eavesdropping, MARL, collaborative jamming, power allocation, trajectory design, UAV.**

I. Introduction

With advancements in intelligent control, precision guidance, and energy supply technologies [1], Unmanned Aerial Vehicles (UAVs) have become invaluable in various fields, including environmental monitoring, military operations, and civilian applications. Their ability to perform rapid deployment and flexible networking makes them powerful assets in wireless communications [2], [3]. However, the increasing number of deployed UAVs, if controlled or disguised by adversaries, poses significant and unpredictable threats to wireless communication security [4]. Due to their high mobility and flexibility, UAVs can hover at strategic positions to intercept confidential data on wireless channels, acting as eavesdroppers. This threat is particularly critical in sensitive environments such as military operations, where UAV eavesdropping can compromise mission-critical information, and in civilian sectors, where it can lead to breaches of personal and corporate data. Moreover, UAV eavesdroppers benefit from high eavesdropping channel quality due to dominant line-of-sight (LoS) gain, enhancing their interception capabilities. Recent research has demonstrated the potential risks and technical challenges associated with UAV eavesdropping, emphasizing the need for robust security measures [5] [6]. Therefore, developing effective countermeasures against UAV eavesdroppers is both imperative and vital to protect the integrity of systems for wireless communication across various domains.

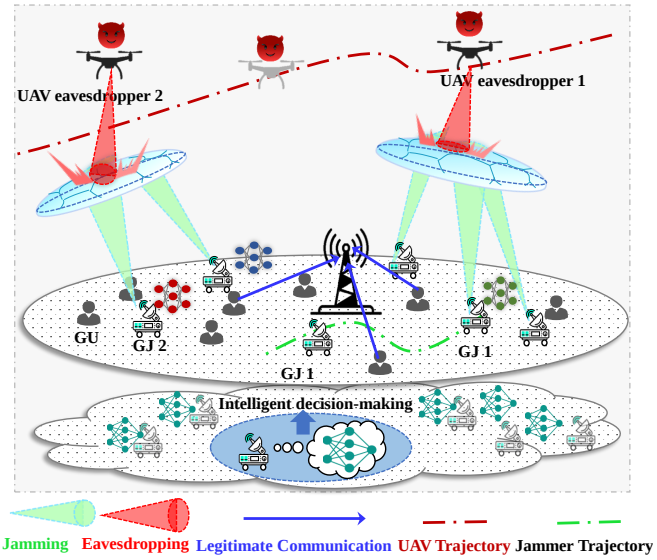


Fig. 1. Cooperative intelligent jamming scheme.

The physical layer security (PLS) technology has been demonstrated its superiority in mitigating the threat of eavesdropping by destroying eavesdropping channels [7], [8], which can be composed as complementary to traditional encryption. In particular, friendly jamming, as a classical approach to PLS techniques, reduces the signals to interference plus noise ratio

Q. Wang, S. Tang and W. Sun are with the School of Cybersecurity, Northwestern Polytechnical University, Xi'an, 710071, China (e-mail: qubeijian.wang@nwpu.edu.cn, tangshiyue@mail.nwpu.edu.cn, sunwen@nwpu.edu.cn). (Corresponding author: Wen Sun.)

Y. Zhang is with the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu. 611731, China (e-mail: yin.zhang.cn@ieee.org).

G. Sun is with the College of Computer Science and Technology, Jilin University, Changchun 130012, China (e-mail: sungeng@jlu.edu.cn).

Hong-Ning Dai is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong (e-mail: hndai@ieee.org).

M. Guizani is with the Machine Learning Department, Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI), UAE (e-mail:mguizani@ieee.org).

(SINR) of the eavesdropping channel by emitting artificial noise [9]. Compared to other PLS techniques such as beamforming,friendly jamming can be more effective in countering UAV eavesdropping, since it is flexibly deployed according to the location of the UAV eavesdropper, the network size and channel quality. Additionally, since the air-to-ground (A2G) link always has better channel quality than the ground-to-ground (G2G) link (A2G links are dominated by LoS gain), the jamming effect of ground jammers interfering with UAVs outweighs the effect on ground users at certain jamming locations and jamming powers. Therefore, we utilize mobile ground jammers (GJs) to find the optimal moving trajectory (i.e., jamming trajectory) and power to counter UAV eavesdropping while ensuring the quality of legitimate communications [10].

Nevertheless, existing studies on mobile ground jamming encounter several significant challenges in countering UAV eavesdropping. First, real-time dynamics in wireless channels contribute to increased complexity in task allocation for GJs [11]. Particularly, conventional optimization methods, e.g., successive convex approximation (SCA) and block coordinate descent (BCD) algorithms, struggle to accommodate the dynamics of networks and fulfill the intelligence of GJs. Furthermore, the trajectory design of the GJ is contingent upon having access to the complete eavesdropping trajectory, but owing to the unpredictability of the trajectory of the UAV eavesdropper, the complete and accurate eavesdropping trajectory is barely obtained in advance. It is inaccurate to speculate the optimal jamming trajectory through the UAV eavesdropper's positional information at the moment. Therefore, calculating the optimal jamming trajectory based on the real-time acquired UAV eavesdropper's position needs to be solved urgently. Additionally, the mobility speed of the GJ is obviously lower than that of the UAV eavesdropper, which leads to diminished effectiveness in a single GJ and fails to guarantee the secrecy capacity of the system.

Deep reinforcement learning (DRL), exploring the optimal sequential decisions through interactions between the agent and the environment [12], empowers GJs with intelligence against UAV eavesdropping in dynamic networks. Then, to improve the accuracy of predicting the optimal jamming trajectory, we utilize the real-time trajectory sequence of the UAV eavesdropper as the basis and dynamically compute the jamming trajectory through DRL. To overcome the unequal speed between the UAV eavesdropper and the jammer, collaborative jamming by multiple GJs becomes a key issue. Multi-agent reinforcement learning (MARL) provides a promising solution for GJs to execute collaborative jamming tasks, because MARL can facilitate collaboration between agents by sharing observations and decisions [13]. However, sharing information during collaborative jamming among GJs is challenging. Real-time UAV eavesdropper trajectory information will cause lag after being shared among GJs. Moreover, information-sharing also hampers the decision-making efficiency of GJs, as it is operated as a basis for decision-making in dynamic environments.

To overcome the aforementioned challenges, a collaborative intelligent jamming scheme is proposed as shown in Fig. 1. Through the collaboration of GJs emitting jamming signals, an invisible shield is generated between the UAV eavesdropper and the user. Each GJ can independently make decisions with its own observation, while jointly forming the best jamming strategy. This paper has the following major innovations.

- We first propose a cooperative intelligent jamming scheme to prevent UAV eavesdropping. Specifically, multiple GJs dynamically optimize their trajectories and jamming power to generate specific jamming shields between UAV eavesdroppers and legitimate users. Then, we formulate a joint optimization problem of trajectory and jamming power for GJs to maximize the secrecy capacity.
- We then transform the optimization problem into a Dec-POMDP, owing to the non-convexity of the problem. GJs can adapt their jamming trajectories and jamming power, without complete information from the dynamic networks. Moreover, to overcome the huge state space and action space, we employ a DRL algorithm featuring a dueling network and double-Q learning (i.e., dueling double deep Q-network) to train policy networks.
- We also propose a multi-agent mixing network framework (QMIX)-based collaborative jamming algorithm to enhance the effectiveness of collaboration among multiple GJs. The algorithm enables GJs to independently design their own trajectories and jamming power without sharing local information. However, the ultimate objective for all GJs is to maximize the overall security capacity.
- We further validate the performance of the proposed cooperative intelligent jamming scheme by simulations. Results show that the proposed scheme can maximumly prevent UAV eavesdropping in the premise of ensuring legitimate transmissions, outperforming benchmark schemes. Our results also reveal the relationship between the deployment settings of GJs and the instantaneous secrecy capacity with different parameters.

The rest of this paper is organized as follows. Section II briefly summarizes the related work. In section III, the system model and problem formulation are introduced in detail. In section IV, we formulate the optimization problem as Dec-POMDP and introduce our proposed QMIX-based collaborative jamming algorithm with the dueling double deep Q-network (dueling DDQN) to realize collaborative jamming. In section V, we also present the numerical results with analysis. In Section VI, we discuss the implementation of our scheme and future work with promising techniques. Finally, section VII gives a conclusion of this paper.

II. Related Work

Recently, many studies focused on terrestrial eavesdropping, while ignoring the threat from aerial eavesdropping [14], [15]. However, some studies [16], [17] have noticed that the UAV eavesdropper can bring unpredictable threats to future networks. As a physical layer security technology, friendly jamming has been proven to be an effective method to improve the security of wireless communications [18], [6]. The traditional jamming method for countering eavesdropping is to deploy fixed jammers or relays [14]. Meanwhile, some

TABLE I
THE COMPARISON OF EXISTING WORKING ANTI-EAVESDROPPING SCHEMES

Anti-Eavesdropping scheme	Eavesdropper category	Number of eavesdroppers	Eavesdropping location	Mobility of eavesdroppers	Extra relay	Scalability of protected area
Friendly jamming and aerial relay [14]	Ground	Single	Known	Fixed	Yes	Flexible
Beamforming and aerial relay [19]	Ground	Single	Unknown	Fixed	Yes	Flexible
Intelligent reflecting surface and beamforming [20]	Ground	Multiple	Known	Fixed	Yes	Inflexible
Intelligent reflecting surface and friendly jamming [21]	Ground	Multiple	Known	Fixed	Yes	Flexible
Intelligent reflecting surface and aerial relay [22]	UAV	Single	Unknown	Mobile	Yes	Inflexible
Friendly jamming and trajectory optimization [16]	UAV	Single	Known	Mobile	No	Inflexible
Mobile friendly jamming [10]	UAV	Single	Known	Mobile	No	Flexible
Our scheme	UAV	Multiple	Unknown	Mobile	No	Flexible

works also focus on beamforming [19] as well as intelligent reflecting surfaces (IRS) [20], [21] to dynamically adjust the beam of jamming signals. Then, to effectively prevent flexible UAV eavesdroppers, a secure aerial relay method is proposed by the orchestration of UAV and IRS [22]. However, a large number of active antennas and extra devices bring heavy hardware costs and system complexity. In [16], the authors propose a secure transmission scheme supported by a fixed-ground jammer to disturb UAV eavesdroppers. Nevertheless, the effectiveness of a single fixed-ground jammer is limited due to the flexibility of UAV eavesdroppers and their extensive eavesdropping range. Meanwhile, an increased number of jammers increases extra costs. Therefore, a mobile ground jammer scheme is proposed [10], to dynamically follow the trajectory of the UAV eavesdropper and interfere with it. Then, they explored optimal jamming strategies by the alternative-optimization method. However, the existing work has ignored the mobility limitations of mobile ground jammers, i.e., the speed of the mobile ground jammer is much lower than that of the UAV eavesdropper. Furthermore, dynamic networks with UAVs also increase the complexity of task assignments for jamming strategies. We summarize the differences between our work and relevant literature on anti-UAV eavesdropping in Table I.

Model-free DRL is regarded as a pivotal technology to handle dynamic environments, promoting intelligent networks [23], [24]. To fully utilize the flexibility of mobile nodes, some studies focus on DRL-based trajectory design and resource allocation. In [25], deep deterministic policy gradient (DDPG) is utilized to achieve real-time scheduling of the trajectory of a UAV base station (BS) by solving the problem of an infinite number of state-action pairs. Moreover, a DRL-based joint trajectory design and power allocation scheme is investigated to further enhance the performance of an aerial BS [26]. Additionally, DRL is also adopted in friendly jamming to secure wireless communications [27]. The authors in [28] designed an algorithm based on DDPG to address the joint optimization problem of jamming trajectory and jamming power for a single UAV jammer. However, the huge state and action space challenge and overestimation of action values by the policy network, arising from joint trajectory design and power allocation, makes convergence more prone to falling into a local optimum [29], [30]. Dueling DDQN is considered to be effective in addressing these challenges. In [11], the authors optimized the trajectory of the UAV using dueling DDQN, obtaining an improvement in convergence and performance compared to standard deep Q-network (DQN). Nevertheless, since eavesdroppers often appear in groups, a single jammer cannot withstand the attacks of many eavesdroppers from different. Besides, the mobility speed of GJs is obviously lower than that of UAV eavesdroppers, leading to diminished effectiveness in a single GJ.

The application of MARL in wireless networks for optimizing trajectory design and power allocation of flexible network nodes has been studied. In [31], the authors utilized the DQN-employed MARL framework to enable UAV base stations to move between multiple target service areas according to service requirements. Then, to enhance the mobility of UAV base stations (allowing them to move freely without being bounded in a specific region), a multi-agent deep deterministic policy gradient (MADDPG)-based algorithm is proposed in [32]. The collaboration of UAV base stations in [31], [32] is implemented with information sharing among each agent. However, sharing information challenges the effectiveness of collaborative jamming when using GJs against UAV eavesdroppers. Trajectory information of UAV eavesdroppers is hard to share between GJs in real-time, leading to a trajectory information lag. Besides, the efficiency of decision-making for GJs relies on the quality of information sharing. Therefore, MARL with information sharing is not suitable for collaborative problems of GJs, such as MADDPG, multi-agent proximal policy optimization.

Considering the training cost and scalability, value function factorization (VFF) is demanded as a promising method for MARL [33], [34]. For large-scale multi-agent systems, traditional MARL algorithms face huge computational complexity. VFF can decompose the learning problem of the joint policy network into multiple small-scale learning problems of the local policy network, thereby reducing computational complexity and improving learning efficiency [35]. Furthermore, when the new agent joins the system, it only needs to learn its own local policy network, and then combining with the other local policy networks [36]. By this approach, the scalability of MARL can be further enhanced. Therefore, MARL with

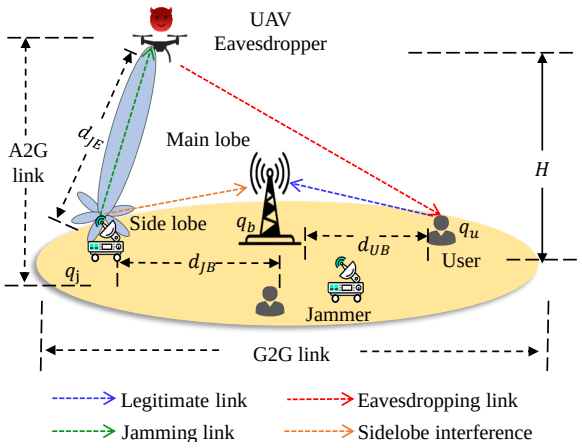


Fig. 2. Cooperative intelligent jamming system model.

VFF framework is wildly investigated in non-information sharing multi-agent learning algorithms, e.g., independent Q-learning (IQL), value decomposition networks (VDNs) [33], and QMIX. Particularly, the framework of the Actor-Critic network [37] and the idea of centralized training and decentralized execution have been integrated into QMIX, which has shown significant superiority [34].

From the aforementioned studies, we can observe that the implementation of cooperative intelligent jamming still faces obstacles. Especially, UAV eavesdropping with speed advantages and unpredictable eavesdropping trajectories challenge the effectiveness of collaboration jamming. Motivated by the superiority of QMIX, we proposed a QMIX-based collaborative jamming algorithm with the dueling network and double-Q learning for GJs to independently make decisions without sharing observations.

III. System Model And Problem Formulation

In this section, we first present a system model of our cooperative intelligent jamming scheme to protect the UAV eavesdropper wiretapped wireless terrestrial networks, as depicted in Fig. 2. Then, we optimally deploy GJs with the appropriate location and jamming power by formulating a problem to maximize overall secrecy capacity.

A. Network Model

In this paper, we mainly focus on a wiretapped terrestrial network, in which K legitimate users transmit confidential information to the BS while M malicious UAV eavesdroppers are present. UAV eavesdroppers hover in the air at the height H with flight speed V_e, wiretapping on legal transmissions. To continuously prevent such wiretapping from UAV eavesdroppers, we propose a ground mobile jamming scheme. Specifically, N GJs are responsible for degrading the eavesdropping channel to protect legitimate transmissions. GJs dynamically and intelligently adjust their moving trail with speed V_j and emit directional disturbing signals to UAV eavesdroppers according to their real-time locations and channel state information (CSI). Then, the location of UAV eavesdropper m projected on the ground is denoted as $q_{e,m} = \{x_{e,m}, y_{e,m}\}$. The location of user k can be represented by $q_{u,k} = \{x_{u,k}, y_{u,k}\}$. The location of GJ n is denoted as $q_{j,n} = \{x_{j,n}, y_{j,n}\}$. The locations of the BS can be represented by $q_b = \{x_b, y_b\}$.

B. Channel Model

In this paper, there are two types of communication links, including the G2G communication link and the A2G communication link.

G2G link: It is the transmission on the ground, which predominantly experiences Rayleigh fading and path loss effects, including transmissions from legitimate users to the BS and interference from jammers to the BS. Then, The received power of the BS is described, which can be described as $P_t h_{UB} d_{UB}^{-\alpha}$, where P_t denotes the user's transmit power, d_{UB} is the Euclidean distance between the user and the BS calculated by $d_{UB} = \sqrt{||q_{u,k} - q_b||^2}$, the channel coefficient is denoted by h_{UB} between the BS and the user, and the path loss factory represented by α. Correspondingly, the interference at the BS is represented as $P_j G_s h_{JB} d_{JB}^{-\alpha}$, where P_j denotes the jamming power, the Euclidean distance is expressed by d_{JB} between the GJ and the BS calculated by $d_{JB} = \sqrt{||q_{j,n} - q_b||^2}$, and the channel coefficient is denoted by h_{JB} between the BS and the GJ, and G_s represents the side/back-lobe gain of antennas at GJs. Note that this interference is caused by leaked jamming signals from directional antennas deployed at GJs.

A2G link: It is modeled by the transmission from the ground nodes (i.e., the user or the jammer) to the UAV eavesdropper. A2G communication mainly includes a Line-of-sight (LoS) link and a None LoS (NLoS) link. In particular, we denote the probability of LoS link as

$$\mathbb{P}_{\text{LoS}} = \frac{1}{1 + \tau \exp\left(-\psi\left(\varphi - \tau\right)\right)}, \tag{1}$$

where $\varphi = \frac{180}{\pi} \arctan\left(\frac{H}{d_{UE}}\right)$ represents the elevation angle of the UAV, ψ and τ are constant values depending on the environment. Moreover, we further calculate the probability of NLoS link $\mathbb{P}_{\text{NLoS}} = 1 - \mathbb{P}_{\text{LoS}}$. Referring to [38], NLoS links are affected by small-scale fading as well as path loss, while LoS links only experience path loss effect. Accordingly, the received power of the UAV can be given by

$$P_e = \mathbb{P}_{\text{LoS}} P_t G_e d_{UE}^{-\alpha_e} + \mathbb{P}_{\text{NLoS}} h_{UE} P_t G_e d_{UE}^{-\alpha_e}, \tag{2}$$

where G_e is the received antenna gain at the UAV eavesdropper, and the Euclidean distance between is represented as d_{UE} the UAV eavesdropper and the user, which is calculated by $d_{UE} = \sqrt{||q_u - q_e||^2 + H^2}$, α_e is the path loss factor, and h_{UE} is the small-scale fading factor. Similarly, we denote the interference received by the UAV eavesdropper as

$$\mathcal{I}_j = \mathbb{P}_{\text{NLoS}} P_j G_e G_j h_{JE} d_{JE}^{-\alpha_e} + \mathbb{P}_{\text{LoS}} P_j G_e G_j d_{JE}^{-\alpha_e}, \tag{3}$$

where G_j is the antenna gain at the GJ, and the Euclidean distance is denoted by d_{JE} from the user to the UAV eavesdropper, calculated by $d_{JE} = \sqrt{||q_j - q_e||^2 + H^2}$, the channel coefficient is represented by h_{JE} between the jammer and the eavesdropper.

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2024.3505206

Therefore, the SINR of the BS is represented by ζ_b, being denoted as follows

$$\zeta_b = \frac{P_t h_{UB} d_{UB}^{-\alpha}}{\sigma^2 + P_j G_s h_{JB} d_{JB}^{-\alpha}}, \tag{4}$$

where σ^2 is the Gaussian noise. In addition, the SINR of the UAV eavesdropper, represented by ζ_e, is denoted by

$$\zeta_e = \frac{P_e}{\sigma^2 + \mathcal{I}_j}. \tag{5}$$

Then, we define the instantaneous secrecy capacity, which is represented as

$$C(q_{j,n}, P_j) = \left[\log\left(1+\zeta_b\right) - \log\left(1+\zeta_e\right)\right]^+, \tag{6}$$

where $C(q_{j,n}, P_j)$ is non-negative, and $[x]^+ \triangleq \max(x, 0)$.

C. Problem Formulation

To assure the quality of the secure transmission, GJs are required to degrade the effect on the legitimate transmission (i.e., the transmission between the user and the BS) while enhancing the interference on the wiretapped transmission (i.e., the transmission between the UAV eavesdropper and the BS). In this case, GJs need to move toward the UAV eavesdropper as close as possible, meanwhile, away from the legitimate user. Additionally, to minimize the effect on legitimate transmissions, GJs need to efficiently and reasonably allocate the jamming power. Therefore, to maximize the secrecy capacity, an optimization problem is formulated through joint trajectory design and jamming power allocation. Then, our overall secrecy capacity maximization problem is expressed as

$$\begin{aligned}
(\mathrm{P0}): \max_{\{q_{j,n}, P_j\}} & \sum_{k=1}^{K} \overline{C}_k(q_{j,n}, P_j), && (7)\\
\text{s.t.} & V_j \le V_{\max}, && (7a)\\
& r_{\min} \le \|q_{j,n}\|_2 \le r_{\max}, && (7b)\\
& 0 \le P_j \le P_{\max}, && (7c)\\
& \zeta_b > \zeta_{th}, && (7d)
\end{aligned}$$

where $\overline{C}_k(q_{j,n}, P_j)$ represents the overall secrecy capacity of user k. Herein, constraint (7a) ensures that the movement speed of the GJ is within the maximum speed $V_{\max}$. Constraint (7b) ensures that the GJ moves within the target region. Note that $r_{\min}$ represents the minimum radius of the movement range (centered on the BS) to reduce the degradation of the legitimate transmission. Meanwhile, $r_{\max}$ represents the maximum radius of the movement range (centered on the BS) to guarantee the effectiveness of GJs since they can quickly respond to the next eavesdropping. In the constraint (7c), the jamming power is limited by the maximum jamming power $P_{\max}$. Finally, in constraint (7d), to assure essential transmission quality, the ζ_b need exceed a threshold value ζ_{th}. Obviously, the problem (P0) is highly non-convex and challenging to solve directly, owing to the dynamic network topology resulting from the mobility of both jammers and UAV eavesdroppers. However, DRL enables agents to make near-optimal decisions in dynamic networks with high-dimensional state spaces by modeling the optimization problem as a Markov decision process. Consequently, DRL effectively addresses the overall secrecy capacity maximization problem by leveraging its powerful prediction and decision-making in jamming trajectory design and jamming power allocation.

IV. Reinforcement Learning Solution

In this section, we propose a DRL-based solution to solve our overall secrecy capacity maximization problem. As shown in Fig. 3, the policy network is organized into agent networks and a mixing network. Specifically, each GJ deploys an agent network, and agent networks enable collaborative decision making through mixing networks. To solve (P0) using the DRL-based solution, the optimization problem (P0) is reformulated into a Dec-POMDP. Then, we present a dueling DDQN-based algorithm of each independent agent network. Finally, we introduced a mixing network based on the QMIX framework to enable collaborative jamming for all GJs.

A. Cooperative Markov Game Formulation

To model the Markov decision process, we first decompose the eavesdropping period into T time steps with a certain interval Δt. Note that Δt is small enough to ensure that the location $q_j[t]$ of GJs, the location $q_e[t]$ of UAV eavesdroppers and the jamming power $P_j[t]$ of GJs are approximately unchanged within a time step $t \in [1, 2, 3, ..., T]$. Then, $\|q_j[t] - q_j[t-1]\| \le \Delta s,\ \forall t$, where $\Delta s = V_{\max}\Delta t$ is the maximum movement distance per step. Theoretically, the optimal solution of (P0) requires that the GJs quickly move close to the UAV eavesdropper with the maximum speed. Therefore, the constraint (7a) can be rewritten as

$$q_j[t+1] = q_j[t] + \overrightarrow{V}_j[t] \triangle s,\ \forall t \tag{8}$$

where $\overrightarrow{V}_j[t]$ is the direction vector at the time step t, satisfying $\left\|\overrightarrow{V}_j\right\| = 1$. In addition, constraint (7c) can be rewritten as

$$0 \le P_j[t] \le P_{\max},\ \forall t \tag{9}$$

where $P_j[t]$ represents the jamming power of the GJ n at time step t. Since the location of the GJs and the eavesdroppers are quasi-static in each time step t, the overall secrecy capacity $\overline{C}_k(q_{j,n}[t], P_j[t])$ for user k is approximately given as follows

$$\overline{C}_k(q_{j,n}[t], P_j[t]) = \sum_{t}^{T} C_k(q_{j,n}[t], P_j[t]),\ \forall t \tag{10}$$

where $C_k(q_{j,n}[t], P_j[t])$ represents the instantaneous secrecy capacity of user k. Consequently, (P0) is approximately expressed as

$$\begin{aligned}
(\mathrm{P1}): \max_{\{q_{j,n}[t], P_j[t]\}} & \sum_{t}^{T}\sum_{k}^{K} C_k(q_{j,n}[t], P_j[t]), && (11)\\
\text{s.t.} & q_{j,n}[t+1] = q_{j,n}[t] + \overrightarrow{V}_j[t] \triangle s,\ \forall t && (11a)\\
& r_{\min} \le \|q_{j,n}[t]\|_2 \le r_{\max},\ \forall t && (11b)\\
& 0 \le P_j[t] \le P_{\max},\ \forall t && (11c)\\
& \zeta_b[t] > \zeta_{th},\ \forall t && (11d)
\end{aligned}$$

In our scheme, all GJs are independently coordinated in a decentralized way for cooperative jamming against eavesdropping, hence, the observations and decisions cannot be shared among GJs. This cooperative multi-agent task is determined as Dec-POMDP, which is defined by the five-tuple $\{\mathcal{S}, \mathbb{A}, \mathcal{O}, \mathcal{P}, r\}$, where $\mathcal{S}$ represents the global state space, and $\mathcal{O}$ is the observation space. At each time step, the state of the environment is given as $s \in \mathcal{S}$ and the observations of the agents in the environment are denoted as $o \in \mathcal{O}$. The joint action space of all GJs is denoted as $\mathbb{A} = \mathcal{A}_1 \times \ldots \mathcal{A}_n \ldots \times \mathcal{A}_N$, where $\mathcal{A}_n$ denotes the set of independent action space of GJ n. Furthermore, $\mathcal{P}$ is the state transition probability. Then, $\mathcal{P}(s[t+1]|s[t], a[t])$ specifies the probability of transitioning from the state $s[t]$ to the next state $s[t+1] \in \mathcal{S}$ by the action $a[t] \in \mathbb{A}$. Then, the single-step reward at time t is given as $r[t] = r(s[t], a[t])$, which is determined by the state $s[t]$ and action $a[t]$. Additionally, the action of the GJ is governed by its policy π, where the probability of taking action a in state s is given by $\pi(a|s)$.

To model the joint trajectory design and the jamming power selection problem strength into the DRL framework, we define the necessary elements in Dec-POMDP.

1) State: To find the optimal trajectory design and jamming power allocation, the GJ needs to observe necessary environmental information, including locations and transmit power. Since the speed of the GJ is much smaller than that of the UAV eavesdropper, the GJ is difficult to keep up with the UAV eavesdropper. To efficiently deploy GJs for enhancement of jamming performance, predicting the future location of the UAV eavesdropper is a reasonable method. However, relying solely on the current location $q_e[t]$ is difficult to predict the future location $q_e[t+1]$. Consequently, we take the part of the trajectory sequence of the UAV eavesdropper as a part of the observation $o[t]$. The trajectory sequence $w_e[t]$ is defined by

$$w_e[t] = \{q_e[t-Z+1], q_e[t-Z+2], ..., q_e[t]\}, \ \forall t. \quad (12)$$

where Z denotes the maximum length of the trajectory sequence. Note that the length of the trajectory sequence is equal to t when the time step $t < Z$. Then, the trajectory sequence of all UAV eavesdroppers can be represented as

$$\mathcal{W}_n[t] = \{w_{e,1}[t], w_{e,2}[t], \ldots, w_{e,M}[t]\}, \ \forall t \quad (13)$$

Likewise, the locations of all users can be presented by

$$\mathcal{X}_n[t] = \{q_{u,1}[t], q_{u,2}[t], \ldots, q_{u,K}[t]\}, \ \forall t \quad (14)$$

The GJ n can easily obtain the jamming power $P_{j,n}[t]$ emitted by itself. Then, the observation of GJ n at time step t can be expressed

$$o_n[t] = \{\mathcal{X}_n[t], q_{j,n}[t], q_b, \mathcal{W}_n[t], P_{j,n}[t]\}, \ \forall t \quad (15)$$

Note that the observation of a GJ is different from observations of others since the private information, i.e., location $q_{j,n}$ and the jamming power $P_{j,n}[t]$ are not shared. Then, the global state is constituted by the observations of all GJs, which can be expressed as

$$\mathbf{s}[t] = \{o_1[t], o_2[t], \ldots, o_N[t]\}, \ \forall t \quad (16)$$

2) Action: To effectively resist eavesdropping, the GJ needs to take action by dynamically adjusting its flight trajectory and jamming power. Therefore, the movement direction $\overrightarrow{v_{j,n}}[t]$ and jamming power allocation $P_{j,n}[t]$ are defined as the action of the GJ n at time step t. The action a of the GJ is denoted as

$$a_n[t] = \{\overrightarrow{v_{j,n}}[t], P_{j,n}[t]\}, \ \forall t \quad (17)$$

3) Reward: In addition, to perform the contribution of each GJ from the overall secrecy capacity maximization problem (P1), in our solution, the reward function r includes a penalty item and a reward item. Specifically, the penalty item enhances the guidance of the reward function to the GJ actions while facilitating the convergence of the algorithm. Since the increase in instantaneous secrecy capacity can only indicate that $\zeta_b - \zeta_e$ is increasing, it does not guarantee the minimum transmission requirement for legitimate transmissions, i.e., $\zeta_b > \zeta_{th}$. Therefore, we set a penalty item W_b according to constraint (11d) to guarantee the legitimate transmission, which is represented as

$$W_b[t] = \begin{cases} \log\{1 + \zeta_b[t]\}, & \text{if} \quad \zeta_b[t] \le \zeta_{th}, \\ 0, & \text{otherwise}, \end{cases} \quad (18)$$

Additionally, we set a penalty item W_r according to the constraint (11b) to rapidly respond to the next episode, which is represented as

$$W_r[t] = \begin{cases} 0 & , \text{ if} \quad r_0 \le \|q_{j,n}[t]\|_2 \le r_{th}, \\ \varrho & , \text{ otherwise}, \end{cases} \quad (19)$$

where ϱ is the penalty constant coefficient.

Furthermore, the reward item specifics the bonus of the action selection to maximize the secrecy capacity. As the intuitive feedback for the contribution of overall secrecy capacity maximization, we leverage the instantaneous secrecy capacity C as a reward item. However, when the eavesdropping channel exhibits superior quality compared to the legitimate channel, the feedback of the instantaneous secrecy capacity becomes invalid (the value is always zero), i.e., if $\zeta_b \le \zeta_e$, $C = 0$. Consequently, the jammer is unable to obtain effective feedback on the contribution of its own policy π for (P1) from the instantaneous secrecy capability. Therefore, we reformulate (6) as

$$C[t] = \log(1 + \zeta_b[t]) - \log(1 + \zeta_e[t]), \quad (20)$$

Then, the reward function $r[t]$ can be intuitively represented by

$$r[t] = \sum_{k}^{K} C_k[t] - \xi W_b[t] - \delta W_r[t], \forall t \quad (21)$$

where $\xi \in [0, 1]$ and $\delta \in [0, 1]$ are the penalty coefficients of W_b and W_r respectively. To prevent the algorithm converging to a local optimal solution, the initialization setting of the penalty coefficients cannot exceed 1. Excessive punishment may cause GJs to consistently choose prudent actions to avoid punishment, causing GJs to ignore the optimal policy $\pi(a[t]|o[t])$ based on the current observation $o[t]$. In conclusion, with such a reward function, the GJ can make a decision such that $\zeta_b > \zeta_e$ can get a reward to encourage the

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2024.3505206

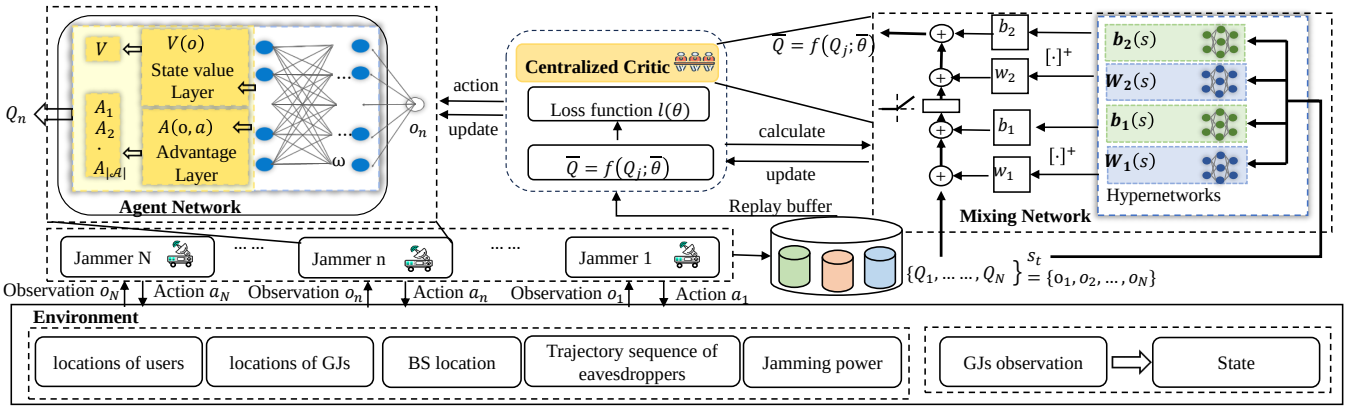


Fig. 3. Cooperative intelligent jamming framework for GJs.

GJ to increase the instantaneous secrecy capacity. Besides, if the action causes $\zeta_b < \zeta_{th}$ or exceeds the target region, an additional penalty will be incurred to reduce the probability of this policy occurring.

For the DRL solution, the goal of GJs is to maximize accumulative reward by improving its policy π, which is given as $\sum_{t=1}^{T} \gamma^{t-1} r[t]$. Thus, after being devised as a Dec-POMDP, (P1) can be further reformulated as

$$(\mathrm{P2}): \max_{\{q_{j,n}[t], P_j\}} \sum_{t}^{T} r[t]. \quad \text{s.t. } (11a), (11c), (18), (19)$$

Then, (P2) can be solved by applying the DRL algorithm. In section IV-B, we first introduce a DRL-based algorithm in agent networks to solve the challenge of learning the optimal policy for each GJ, and then in section IV-C propose a mixing network-based collaborative jamming solution to solve (P2).

B. Dueling DDQN Algorithm

In this work, multiple UAV eavesdroppers and GJs cause problems of huge global state space $\mathcal{S}$ and action space $\mathcal{A}$, which degrade the convergence of standard DQN. Fortunately, DQN with the dueling network (dueling DQN) can effectively solve such problems [29]. Dueling DQN incentivizes GJs to enhance their comprehension of the potential interrelations between jamming strategies, specific states, and secrecy capabilities. However, since both dueling DQN and standard DQN tend to choose the next action that maximizes the reward value, they suffer from overestimation bias. In intricate environments characterized by the presence of multiple collaborative jammers, overestimation bias becomes unavoidable and demands thorough attention. Currently, double Q-learning, which has been extensively employed, can alleviate the issue of overestimation bias. Consequently, the DQN algorithm with dueling network and double-Q learning (i.e., dueling DDQN) is applied in the agent network.

1) Dueling network: The key feature of dueling DDQN is to split the action-value function in standard DQN into the value function and the advantage function [29]. In particular, the value function is to map the contribution of a specific observation $o_n[t]$ to the optimization problem (P2). The advantage function is to map the contribution of each action a to the optimization problem (P2) in a specific observation $o_n[t]$. Then, combining the outputs of the value function and the advantage function, an estimate of the action-value function can be given. With the dueling network, the GJ can more efficiently learn the state value function. Especially, when the action-value gap between different actions of the same state is small, the action-value function derived from the dueling network demonstrates robustness to approximation errors.

In Fig. 3, we depict the dueling DDQN framework for the GJ. At each time step t, the observation o_n of the GJ n is set as an input to the network, and then the network outputs the action-value of all actions for the observation o_n. Particularly, the hidden layer consists of the state value layer and the advantage layer. We utilize the state value layer and the advantage layer to estimate the value function $V^{\pi}(o)$ and the advantage function $A^{\pi}(o,a)$, respectively. Then, the action-value function $Q^{\pi}(o_n, a_n)$ is obtained by a linear combination of the outputs of the state value function and the advantage function, which can be expressed as

$$Q(o_n[t], a_n[t]; \omega, \mu, \nu) = V\left(o_n[t]; \omega, \mu\right) + A\left(o_n[t], a_n[t+1]; \omega, \nu\right), \quad (22)$$

where ω denotes the parameter of the hidden layers, μ denotes the parameter of the state value layer, and ν is the parameter of the advantage layer. Note that the expected value of the advantage function $\mathbb{E}_{a\sim\pi(o)}\left\{A^{\pi}\left(o_n, a_n\right)\right\} = 0$. Since $Q(o_n, a_n; \omega, \mu, \nu)$ is a parametric approximation value of the action-value function, it is difficult for us to uniquely deduce the values of $V(o_n; \omega, \mu)$ and $A(o_n, a_n; \omega, \nu)$ from $Q(o_n, a_n; \omega, \mu, \nu)$. It means that the roles of $V(o_n; \omega, \mu)$ and $A(o_n, a_n; \omega, \nu)$ cannot be distinguished during the training process. To address the above challenge, we can centralize the advantage function. Therefore, the action-value function in dueling DDQN can be further represented by

$$\begin{aligned} Q(o_n[t], a_n[t]; \omega, \mu, \nu) &= V\left(o_n[t]; \omega, \mu\right) + \Big(A\left(o_n[t], a_n[t]; \omega, \nu\right) \\ &\quad - \frac{1}{|A|} \sum_{a[t+1]} A(o_n[t], a_n[t+1]; \omega, \nu)\Big). \end{aligned} \quad (23)$$

Algorithm 1 Dueling DDQN for Ground Jamming Anti-UAV Eavesdropping

Initialize: the maximum number of episodes N_e, the capacity of the replay buffer $\mathcal{R}$, the number of replay starts N_0, the step size of the synchronization network parameters N_{tar}, set exploration $\epsilon = \epsilon_0$, decaying rate α.
Initialize: the reward function r, the penalty coefficient ξ and δ, the threshold ζ_{th}, r_0 and r_{th}.
Initialize: the Dueling network parameters ω and , the target network parameters $\omega^- = \omega$.
for each episode $i = 1, 2, ..., N_e$ **do**
  Initialize the environmental information.
  for each time step $t = 1, 2, ..., T$ **do**
    GJ n locally stores the location $q_e[t]$ of the UAV eavesdropper observed in one-step t;
    if $t \geq Z$ **then**
      Obtain the $\mathcal{W}_n$, and then obtain observation $o_n[t]$;
    end if
    Choose action $\overrightarrow{V_j}[t]$ and $P_{j,n}[t]$. Specifically, for all GJs, actions are taken randomly based on the $\epsilon-$greedy policy starting from $\mathcal{A}$, or following (25) with probability $1-\epsilon$;
    GJ n obtain reward $r[t]$ by executing $a_n[t]$, where $r[t] = \sum_{k=1}^{K} C_k[t] - \xi W_b[t] - \delta W_r[t], t \in T$;
    Update the environmental information, GJ n obtain $o_n[t+1]$;
    Store $\{o_n[t], a_n[t], r_n[t], o_n[t+1]\}$ to $\mathcal{R}$;
    if $n > N_0$ **then**
      Sample a batch of experience from $\mathcal{R}$;
      Update target Q-value by $Y_t^{\text{DoubleQ}} = r + \gamma \hat{Q}(o_n[t+1], a_{\max}(o_n[t+1]; \omega); \omega^-)$
      Update ω, $\epsilon = (1-\alpha)\epsilon$;
    end if
    if $n \mod N_{tar} == 0$ **then**
      Update $\omega^- = \omega$;
    end if
  end for
end for

2) Double-Q learning: The overestimation is caused by that the policy always chooses the action with the maximum target action-value, which can be solved by the double Q-learning. Specifically, the double-Q learning decouples the selection of actions for the target action-value and the estimation of the target action-value into two steps. Herein, we denote the target action-value as

$$Y^{\text{DoubleQ}} = r + \gamma \hat{Q}(o_n[t+1], a_{\max}(o_n[t+1]; \omega); \omega^-), \quad (24)$$

where

$$a_{\max}(o_n[t+1]; \omega) = \arg \max_{a_n[t+1]} Q\left(o_n[t+1], a_n[t]; \omega\right), \quad (25)$$

where $\hat{Q}$ is target action-value function, and γ is the discount coefficient. In addition, $a_{\max}\left(o_n[t+1]; \omega\right)$ represents obtaining the action $a_n[t]$ with the largest action-value in the current action-value function $Q(o_n[t+1], a_n[t]; \omega)$, i.e., the most valuable trajectory and jamming power strength of the GJ n in the next observation $o_n[t+1]$. Note that we use the dueling network Q with coefficient ω in (25) to select actions, while we employ the target dueling network $\hat{Q}$ with coefficient ω^- in (24) to evaluate the actions by the greedy method.

The proposed algorithm for ground Jamming anti-UAV eavesdropping with dueling DDQN is summarized in Algorithm 1. It is worth mentioning that we utilize the experience replay technique to enhance the train performance of the dueling DDQN. By using an experience replay buffer, the past experience and the current experience are mixed to reduce data correlation. Furthermore, experience replay enhances learning efficiency by making samples reusable. Due to the huge state space $\mathcal{S}$ in the environment, the initial value of the exploration coefficient ϵ needs to be large enough for the GJ to abundantly and effectively explore the environment. Then, the GJ is possible to try more directions of movement and the strengths of the jamming power for the same observations o_n. With the accumulation of exploration experience, the GJ will relieve random exploration while focusing on learning. Therefore, it is necessary to initialize a suitable exploration decay coefficient α. Furthermore, as depicted in step 8 of Algorithm 1, the GJ can obtain the complete UAV trajectory $\mathcal{W}_n$ when the time step $t \geq Z$, so that predicting the future UAV trajectory. Thus, to ensure the effective prediction of UAV trajectory, the GJ will continue to observe the eavesdropping trajectory until it obtains the complete $\mathcal{W}_n$.

C. Cooperative MARL: QMIX-based Framework

As previously mentioned, each GJ moves and emits jamming signals only depending on its own local observations. Hence, the GJ disturbs eavesdroppers without information (e.g., without the observations and actions) exchange with others, resulting in low-efficiency jamming. Towards this end, we proposed a collaborative jamming scheme based on the QMIX-based collaborative MARL to further improve the collaborative jamming performance of GJs without information. We deploy a global action-value estimator to evaluate the contribution of the joint actions, preventing the GJ from greedily choosing an action based on its own independent action-value Q. The QMIX framework is used to efficiently deal with collaborative MARL in our collaborative jamming algorithm.

As depicted in Fig. 3, the cooperative intelligent jamming framework includes the mixing network and the agent network. Specifically, the mixing network acts as an estimator to evaluate the joint actions of GJs. The agent network consists of all GJs (regarded as agents) which was introduced in Section IV-B. We leverage the mixing network to aggregate all action-values from the agent network, obtaining the global action-value $\overline{Q}$, which is given as

$$\overline{Q} = f(Q; \theta), \quad (26)$$

where θ represents the parameter of the mixing network, and $f(Q; \theta)$ is used to approximate the mapping relationship between global action-value $\overline{Q}$ and the local action-value Q of each GJ.

With the assistance of the mixing network, we can realize that the optimal joint action u is composed of the optimal local

actions which are greedily selected by GJs according to their local observations. In particular, we need to ensure that the set composed of actions a obtained by $\arg\max Q$ is equivalent to the joint action u obtained by $\arg\max \overline{Q}$, i.e.,

$$\arg\max_{a} \overline{Q}(s,u) = \begin{pmatrix} \arg\max Q_1(o_1,a_1) \\ \vdots \\ \arg\max Q_N(o_N,a_N) \end{pmatrix}, \tag{27}$$

where $s = \{o_1, o_2, ..., o_N\}$ is the global state, and $u = \{a_1, a_2, ..., a_N\}$ is the joint action. To satisfy (27), the global action-value $\overline{Q}$ need to monotonically increases with the local action-value Q of each GJ, i.e.,

$$\frac{\partial \overline{Q}}{\partial Q_n} \geq 0, \forall n \tag{28}$$

As shown in Fig. 3, the parameters of the mixing network $\theta = \{w_1, w_2, b_1, b_2\}$ are determined by hypernetworks $\mathbf{W_1}(s)$, $\mathbf{W_2}(s)$, $\mathbf{b_1}(s)$, and $\mathbf{b_2}(s)$. In addition, the hypernetwork $\mathbf{W}$ mainly includes a linear and an absolute activation function to generate non-negative weights w, $w \geq 0$, to ensure the monotonicity (28). The hypernetwork $\mathbf{b}$ is similar to the hypernetwork $\mathbf{W}$ but necessitates no absolute activation function. Then, to improve the nonlinearity of the mixing network, we introduce the activation function ReLU between $\mathbf{b_1}(s)$ and $\mathbf{W_2}(s)$.

Finally, we denote the loss function as follows

$$l(\theta) = \sum_{i=1}^{B} (\hat{Y}_i - \overline{Q}(s[t], u[t]; \theta))^2, \tag{29}$$

where B denotes the number of samples randomly in a batch training, and $\hat{Y}_i$ is the target action-value for the mixing network, which can be represented as

$$\hat{Y}_i = r[t] + \gamma \max_{a[t+1]} \overline{Q}(s[t+1], u[t+1]; \theta^-), \tag{30}$$

The QMIX-based collaborative jamming algorithm is given by Algorithm 2. In particular, for updating the parameter θ of the mixing network, $s[t]$ and $u[t]$ are also stored in the replay buffer $\mathcal{R}$, in step 4. The update dueling network parameter ω is also performed by the loss function in step 8. Note that in the environment of multi-GJs, we set the initial location of each GJ to be at a certain distance from each other, and decentralized initial locations allow GJs to learn cooperative interference faster and improve secrecy capacity.

Algorithm 2 QMIX-based Collaborative Jamming Algorithm

1: **Initialize**: Step 1-5 of Algorithm 1;
2: **Initialize** the UAV eavesdroppers' flight path $\{q_{e,1}, q_{e,2}, ..., q_{e,M}\}$;
3: Steps 7-14 of Algorithm 1;
4: Then the global state $s[t] = \{o_1[t], o_2[t], ..., o_N[t]\}$, the joint action $u[t] = \{a_1[t], a_2[t], ..., a_N[t]\}$
5: Store $\{s[t], u[t], \{r_1[t], r_2[t], ..., r_N[t]\}, s[t+1]\}$ to $\mathcal{R}$;
6: Steps 16-19 of Algorithm 1;
7: Update the target action-value of the critic network mixing network by $\hat{Y}_i = r[t] + \gamma \max_{a[t+1]} \overline{Q}(s[t+1], u[t+1]; \theta^-)$.
8: Update the parameter ω of each agent network, and update the parameter θ of the mixing network. Then, the loss function is given by
$$l(\theta) = \sum_{i=1}^{B} (\hat{Y}_i - \overline{Q}(s[t], u[t]; \theta))^2,$$
9: Step 20 of Algorithm 1;
10: **if** $t \mod N_{tar} == 0$ **then**
11: Synchronize the parameters $\theta^- = \theta$ of the critic target network
12: For each GJ, update $\omega^- = \omega$;
13: **end if**
14: Steps 24-26 of Algorithm 1.

D. Analysis of Algorithm Complexity

In this subsection, we analyze the computational complexity of the QMIX-based collaborative jamming algorithm in terms of the training and execution phases.

Complexity analysis in training phase. For the agent network, each GJ is equipped with a dueling network. The computational complexity of the dueling network mainly includes the complexity of forward-propagation and back-propagation. The complexity of forward-propagation depends on the structure of the neural network. Assuming the network has L layers and the i layer has ι_i neurons, the complexity of forward-propagation is $O(\sum_{i=1}^{L+1} \iota_i \cdot \iota_{i-1})$. Back-propagation involves computing gradients and updating network weights. Its complexity is similar to forward-propagation, which is also $O(\sum_{i=1}^{L+1} \iota_i \cdot \iota_{i-1})$. Then, the complexity of experience replay mainly depends on the process of sampling from the replay buffer, which is usually $O(1)$ since the sampling is typically random. For each agent, it is necessary to update the parameters of two neural networks: the primary network and the target network. The complexity of each update is $O(\sum_{i=1}^{L+1} \iota_i \cdot \iota_{i-1})$.

Additionally, since actions of all GJs are aggregated into a global action-value, the collaborative jamming algorithm uses a mixing network to achieve this aggregation. For mixing network, assuming the mixing network has D layers and the j layer has ϖ_i neurons, the complexity of forward and back-propagation is $O(\sum_{i=1}^{D+1} \varpi_i \cdot \varpi_{i-1})$. Therefore, the overall complexity of the training phase is $O(N \cdot (\sum_{i=1}^{L+1} \iota_i \cdot \iota_{i-1} + \sum_{j=1}^{D+1} \varpi_i \cdot \varpi_{j-1}))$, where N is number of GJs.

Complexity analysis in the execution phase. The complexity of the execution phase mainly considers the computational cost of forward propagation, as no parameter updates occur during execution. The overall complexity of the execution phase is $O(N \cdot \sum_{i=1}^{L+1} \iota_i \cdot \iota_{i-1} + \sum_{j=1}^{D+1} \varpi_i \cdot \varpi_{j-1})$.

V. Simulation Results

In this section, we evaluate the superiority of our cooperative intelligent jamming scheme through comparison with existing benchmarks. Furthermore, we investigate the effects

TABLE II
MARL-RELATED PARAMETERS

Simulation parameter	Value
Maximum episodes N_e	5000
Initialize exploration ϵ	0.9
Exploration decay rate α	0.9998
Batch size	32
Batch learning frequency	200
penalty coefficient ξ	0.7
penalty coefficient δ	0.6
Learning rate (agent network)	10^{-4}
Learning rate (mixing network)	10^{-4}
Replay buffer capacity $\mathcal{R}$	2000
Trajectory sequence size Z	3
Jamming power set	[0,1,2,...,$P_{\max}$]
Update frequency for target network N_{tar}	200
Hypernetwork (Number of hidden layer neurons)	[128,128]

of parameters on security performance in accordance with the number of GJs, mobility speed, and penalty coefficient.

A. Simulation Settings

In the simulation, we concentrate on an area of $100\text{m} \times 100\text{m}$, where a single BS is fixed at $q_b = \{50\text{m}, 50\text{m}\}$. Users randomly appear in the network area. Note that the number of users is randomly chosen from 6 to 9. UAV eavesdroppers wiretap legitimate transmissions at the flight altitude $H = 50\text{m}$ with speed $V_e = 10\text{m/s}$. For GJs, the maximum jamming power is $P_{\max} = 25\text{mW}$, the maximum movement range is $r_{\max} = 45\text{m}$ and the minimum movement range is $r_{\min} = 5\text{m}$, respectively. It is worth mentioning that the specific parameters of UAV networks, such as flight speed and flight altitude of the UAV, are set referring to the 3rd Generation Partnership Project (3GPP) [39]. For the sake of illustration, we set the movement direction of GJ n as four types: up, down, left, and right.

For the proposed QMIX-based collaborative jamming algorithm, the agent network consists of 3 hidden layers, each of which uses a ReLU activation function. The first 2 hidden layers have 128 and 128 neurons, respectively. The last hidden layer is a dueling layer with $|\mathcal{A}|+1$ neurons ($|\mathcal{A}|$ denotes the number of actions), where one neuron corresponds to the estimation of the state value and the other $|\mathcal{A}|$ neurons correspond to the action advantage of the $|\mathcal{A}|$ actions. Additionally, the mixing network consists of 4 hypernetworks, as introduced in Section IV. Each hypernetwork consists of 2 fully connected layers and a ReLU activation function. Furthermore, the simulations are performed with Python 3.8 and PyTorch 1.11 under the PyCharm platform. All experiments are implemented on a computer equipped with Intel Core i7-1165G7 2.80GHz. Note that the simulation results are processed with the policy network trained and converged after 5000 episodes. Unless specified otherwise, the values for all other key simulation parameters [40] are provided in Table II.

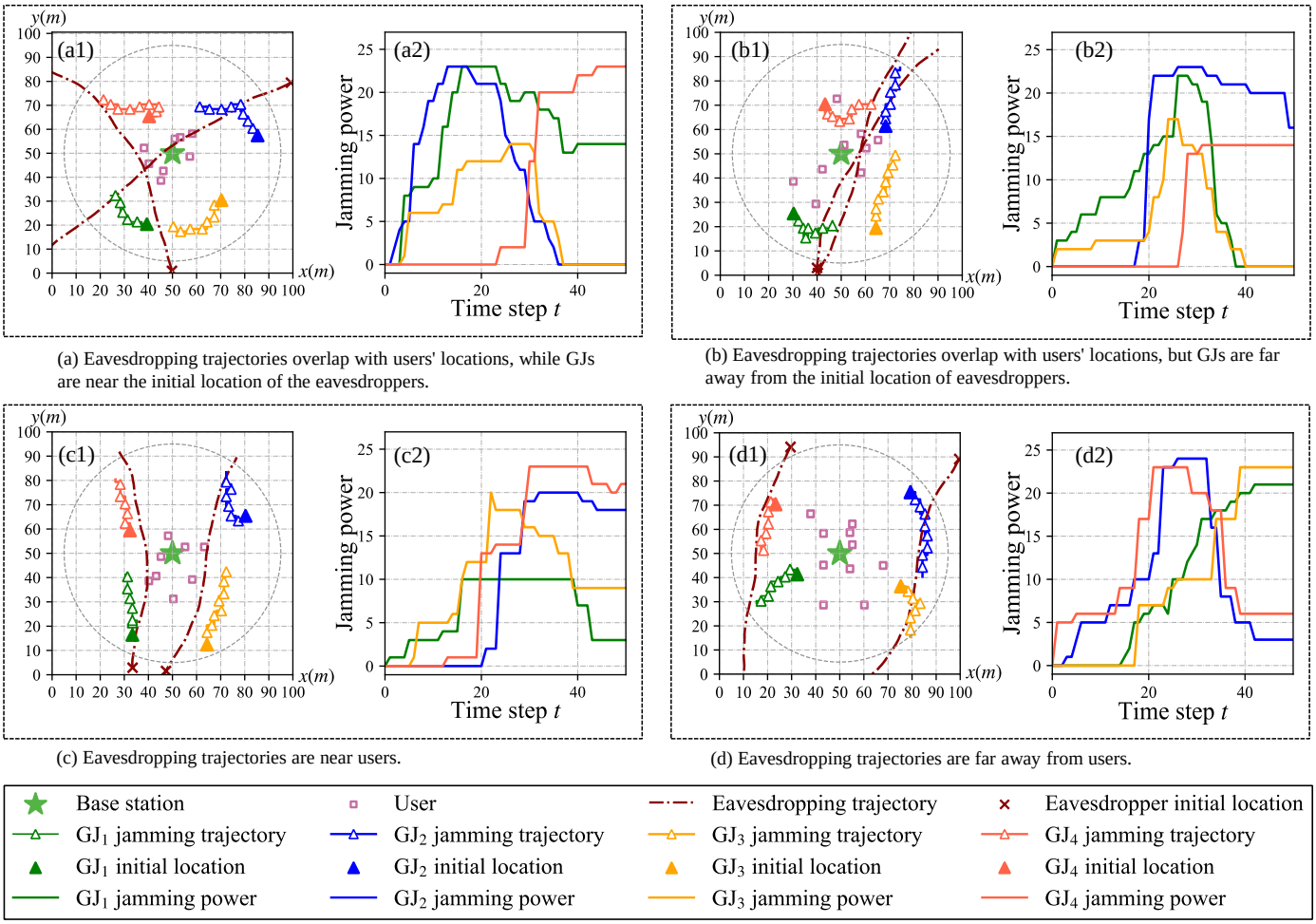
B. Analysis of Algorithm Effectiveness and Convergence

Herein, we verify the effectiveness of the proposed collaborative jamming algorithm and cooperative intelligent jamming scheme. First, we exhibit the effectiveness of the proposed algorithm for joint trajectory and jamming power optimization. Then, the effectiveness and convergence of the proposed algorithm are evaluated. Finally, the performance between different schemes as well as interference trajectories are compared.

Fig. 4 shows QMIX-based collaborative jamming against different eavesdropping trajectories. Figs 4 (a) to (d) illustrate different eavesdropping scenarios. Specifically, as shown in Figs. 4 (a1) and 4 (b1), when eavesdropping trajectories overlap with users' location, to degrade interference to legitimate transmissions, GJs are usually preferentially moved away from users. Then, as UAV eavesdroppers approach users, GJs move as close as possible to the eavesdropper while maintaining a distance from users. In Fig. 4 (c1), even if the eavesdropping trajectory is close to users, the jamming trajectory still maintains a certain distance from users to avoid undesirable interference. It can be seen from Fig. 4 (d1) that when eavesdropping trajectories are far away from users, the jammer is positioned between eavesdroppers and users to efficiently interfere with eavesdropping. Hence, the optimal jamming trajectory is always close to the eavesdropper and far from the BS, subject to safeguarding the quality of legitimate transmissions. When multiple GJs collaborate to design jamming trajectories, the GJ decides its jamming trajectories based on the observation. Taking Fig. 4 (b1) as an example, the initial locations of GJ_1 and GJ_3 are near the eavesdroppers. However, when GJ_1 approaches the eavesdropper, GJ_3 moves toward the future possible eavesdropping location of the eavesdropper. Furthermore, we can observe that, from Fig. 4 (a1) to (d1), the jamming trajectories do not overlap with each other, verifying the effectiveness of our collaborative jamming algorithm.

Furthermore, we can see from Fig. 4 (a2), the GJ_1 and GJ_3 gradually increase the jamming power as its location moves away from the user and close to the UAV eavesdropper (depicted in Fig. 4 (a1)). Then, as the UAV eavesdropper moves away from the GJs, the jamming power of jammer GJ_2 and jammer GJ_3 decreases by 0 to avoid the effect on the users. However, due to the proximity of the GJ_1 and GJ_4 to the UAV eavesdroppers after time step 30, the jamming is maintained at a high power. Therefore, when the GJ is in proximity to the user, it consistently reduces jamming power to maintain the quality of legitimate transmissions, otherwise, the jamming power is increased. The same variation can also be observed in Figs. 4 (b2) to (d2). Additionally, the GJs collaboratively adjust their own jamming power according to the location of other jammers. Particularly, when GJs are in proximity to the same UAV eavesdropper (in Figs. 4 (b1) and (b2)), the main jamming power is emitted by the GJ which is close to the UAV eavesdropper by the collaborative jamming algorithm. Then, the jamming power is decreased, when the UAV eavesdroppers are positioned on both sides of the users and close to the users (as shown in Figs.4 (c1) and (c2)). Meanwhile, the jamming power is increased, when the UAV

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2024.3505206



(a) Eavesdropping trajectories overlap with users' locations, while GJs are near the initial location of the eavesdroppers.

(b) Eavesdropping trajectories overlap with users' locations, but GJs are far away from the initial location of eavesdroppers.

(c) Eavesdropping trajectories are near users.

(d) Eavesdropping trajectories are far away from users.

Fig. 4. QMIX-based collaborative jamming trajectories and jamming power.

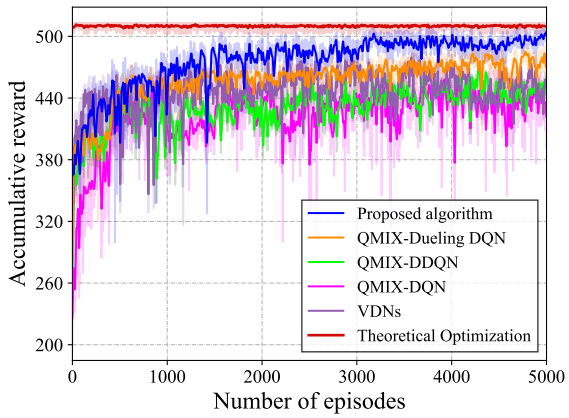


Fig. 5. Comparison of cumulative rewards among different algorithms over 5000 training episodes.

eavesdroppers are positioned on both sides of the users and far away from the users (as shown in Figs. 4 (d1) and (d2)).

To verify the effectiveness of our proposed QMIX-based collaborative jamming algorithm (using QMIX framework with Dueling DDQN), we compared it with the other five benchmarks.

1) QMIX-dueling DQN: QMIX framework with the dueling network.
2) QMIX-DDQN: QMIX framework with double Q learning.
3) QMIX-DQN: QMIX framework with standard DQN.
4) VDNs: MARL algorithm based on VDNs.
5) Theoretical Optimization: Alternating optimization algorithm with known system information (eavesdropping trajectories, channel parameters, etc.).

Fig. 5 compares accumulative rewards for different algorithms in 5000 training episodes. It is observed that all algorithms gradually converge after 2000 episodes. Compared to other MARL algorithms, our proposed algorithm has stable convergence. It means that the agent network with the dueling network has dominated superiority in the huge state space and action space for preventing multiple UAV eavesdropping. Then, the proposed algorithm obtains a higher accumulative reward compared to other algorithms, including non-information sharing MARL (i.e., VDNs). This shows that our proposed algorithm based on the QMIX framework is reasonable and effective. Moreover, it is evident that the proposed algorithm closely approximates the theoretical optimization.

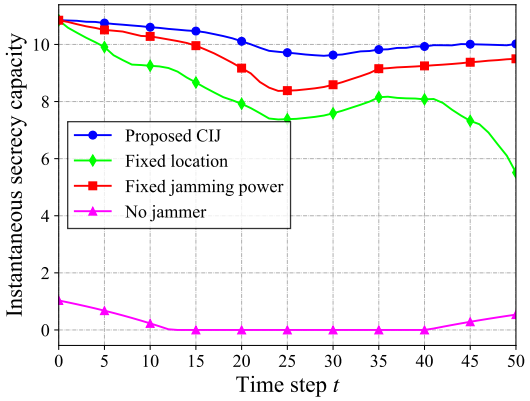


Fig. 6. Comparison of instantaneous secrecy capacity among different schemes.

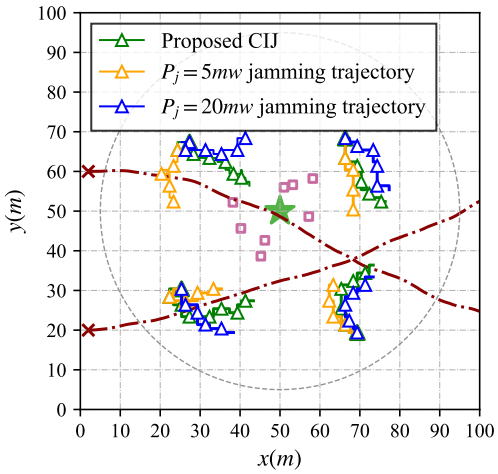


Fig. 7. Comparison of jamming trajectories among different schemes.

To summarize, the proposed QMIX-Dueling DDQN algorithm enables GJs to collaboratively interfere with eavesdropping by making the optimal decision without sharing their local observations.

Fig. 6 illustrates the variation of instantaneous secrecy capacity for different schemes in an episode. We compare our cooperative intelligent jamming scheme (denoted as proposed CIJ in Fig. 6) with the fixed location scheme, the fixed jamming power scheme and the no jammer scheme. Specifically, in the fixed location scheme, GJs can only change the strength of jamming power to disturb eavesdropping. Meanwhile, in the fixed jamming power scheme, GJs can change their positions to interfere with eavesdropping with a fixed jamming power. The no jammer scheme means that there are no jamming protection measures. It is noteworthy that GJs in the aforementioned schemes are driven by our proposed MARL algorithm. Since GJs are initiated at the same location, the instantaneous secrecy capacities of the three jamming schemes using GJ are the same when $t = 0$. Compared to the proposed scheme, the instantaneous secrecy capacity at the fixed location scheme drops significantly after time step 40. This decrease is caused by the fixed GJ's lack of mobility, and the UAV moves away from the GJ resulting in diminishing the jamming effect. Moreover, compared to the no jammer scheme, the other schemes can be effective against UAV eavesdropping. Furthermore, results also reveal that the proposed cooperative intelligent jamming scheme outperforms other jamming schemes, achieving higher instantaneous secrecy capacity. This superiority is attributed to the joint optimization of trajectory and jamming power, providing GJs with increased flexibility to adapt to complex locations and dynamic networks.

Fig. 7 demonstrates the variation of trajectories for different schemes. In the CIJ scheme, GJs can compensate for the lack of speed problem by dynamically adjusting the jamming power. In the fixed jamming power scheme, the speed resources of the GJs with higher jamming power are wasted away from the users, and GJs with lower jamming power consume the speed resources close to the eavesdropper. However, due to the limited speed resource ($V_j < V_e$), the CIJ scheme outperforms the other schemes.

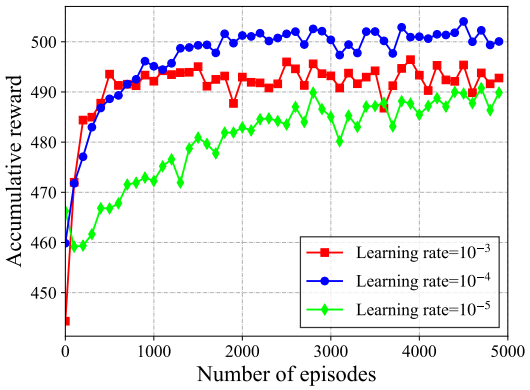
C. Performance Analysis with Various Settings

Then, we evaluate the performance of the proposed collaborative jamming algorithm with different learning rates. The learning rate of the agent network is set to three values, i.e., 10^{-3} (denoted as the red line), 10^{-4} (denoted as the blue line), and 10^{-5} (denoted as the green line) in Fig. 8(a). We can see that the red line initially exhibits a higher reward than the others. However, with the increasing number of episodes, the blue line rapidly surpasses the red line and converges after approximately 3000 episodes. The reason is that a large learning rate (e.g., the learning rate $= 10^{-3}$) can induce significant fluctuations in the training model, rendering it difficult to find the optimal policy. Additionally, a small learning rate (e.g., the learning rate $= 10^{-5}$) can result in excessively long training times for the model. Therefore, a reasonable learning rate is necessary for the practical implementation of our proposed algorithm, i.e., the learning rate is set to 10^{-4}.

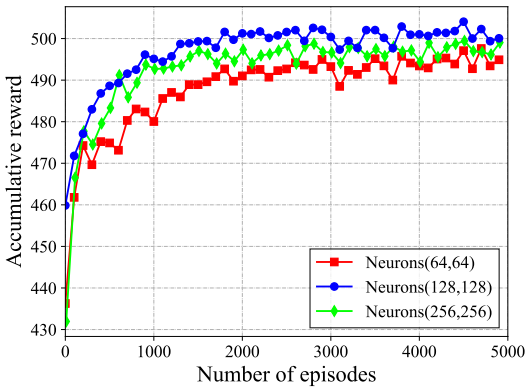
To assess the impact of varying numbers of neurons in the agent network on the algorithm's performance. We set the number of neurons in the hidden layer with $(64, 64)$, $(128, 128)$ and $(256, 256)$, respectively. In Fig. 8(b), a large number of neurons (ranging from $(64, 64)$ to $(128, 128)$) results in higher rewards and faster convergence. However, configuring an excessive number of neurons (e.g., $(256, 256)$) results in diminished rewards. This is because the potential overfitting of the neural network and the accompanying increase in the computational cost of training can result in a deterioration of the reward. Consequently, selecting a reasonable number of neurons in the agent network is essential for the proposed algorithm, i.e., the number of neurons is set as $(128, 128)$.

Furthermore, we analyze the performance of the proposed algorithm with different batch sizes. As shown in Fig. 8(c), a small batch size (i.e., batch size = 16) fails to learn all of the training data, resulting in the smallest accumulated reward for convergence. A large batch size (i.e., batch size = 64) results in repeatedly learning the training data. Additionally, a large batch size can consume more computing capability and

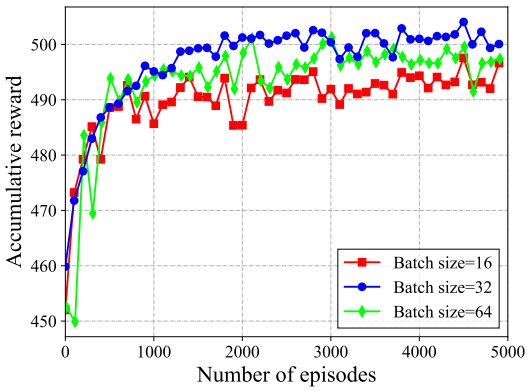
This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2024.3505206



(a) Algorithm performance under different learning rates.



(b) Algorithm performance under different number of neurons.



(c) Algorithm performance under different batch sizes.

Fig. 8. Algorithm performance analysis with various settings.

increase the training time. Then, a trade-off between convergence speed and training time is significant. As a consequence, the training batch size is set to 32 in the simulation.

Fig. 9 plots the instantaneous secrecy capacity of our proposed scheme in different GJ speeds. The blue, red, and green lines represent GJ speeds of $V_j = V_e$, $V_j = V_e/5$ and $V_j = V_e/10$, respectively. It is observed that an increased GJ speed can result in a high instantaneous secrecy capacity. The reason is that the increased speed enables GJs to rapidly move to a reasonable location to disturb eavesdroppers. Moreover, in Fig. 9, despite the secrecy capacities of the red and green lines

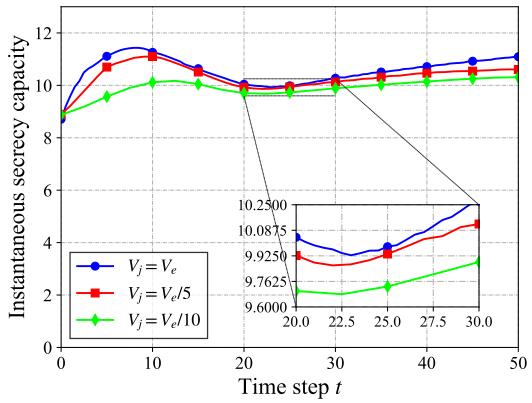


Fig. 9. Comparison of different GJ speeds on the instantaneous secrecy capacity.

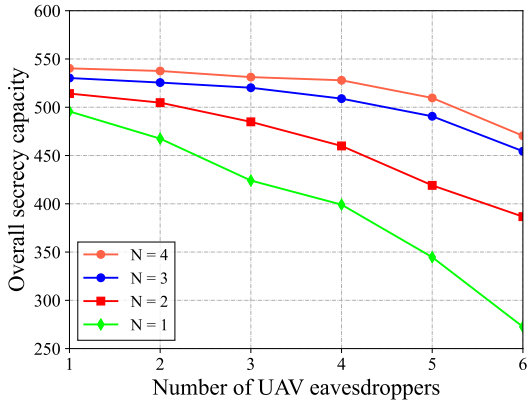


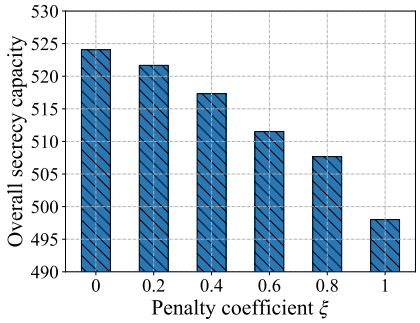
Fig. 10. Overall secrecy capacity versus the number of UAV eavesdroppers for different numbers of GJs.

being lower than that of the blue line at each time step, there is little difference among these secrecy capacities. This result demonstrates the effectiveness of our proposed algorithm in preventing UAV eavesdropping, even when the speed of the GJ is lower than that of the UAV eavesdropper.

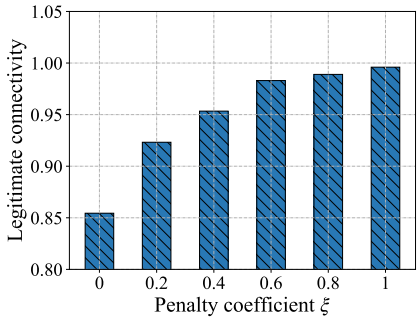
In Fig. 10, the overall secrecy capacity of our collaborative jamming algorithm is depicted, concerning the number of UAV eavesdroppers versus the number of GJs. Note that the dueling DDQN algorithm is employed when the number of GJs is $N = 1$. We observe that the secrecy capacity diminishes progressively with the rise in the number of UAV eavesdroppers. In Fig. 10, increasing the number of GJs effectively enhances jamming performance against eavesdropping. Moreover, when the number of GJ is less than the number of eavesdroppers, GJs can still provide a certain level of overall secrecy capacity. However, it is noticeable that the line for $N = 4$ exhibits the slowest decline trend as the number of eavesdroppers increases. This is because more GJs can rapidly respond to jamming deployment, enabling effective interference with UAV eavesdroppers.

Then, the impact of the penalty coefficient ξ for the penalty item W_b on the overall secrecy capability and the quality of legitimate transmissions is analyzed in Fig. 11. We set the number of GJs to $N = 2$, the number of UAV eavesdropping

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2024.3505206



(a) Penalty coefficient versus overall secrecy capacity.



(b) Penalty coefficient versus legitimate connectivity.

Fig. 11. Comparison of the penalty coefficient ξ.

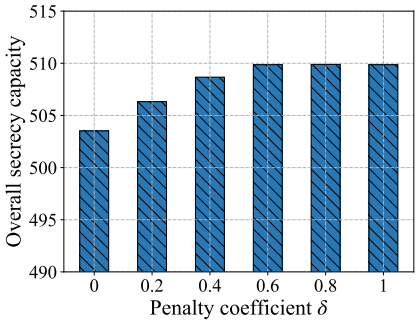


Fig. 12. Comparison of the penalty coefficient δ.

to $M = 2$, and $\zeta_{th} = 0\text{dBm}$. As shown in Fig. 11(a), we illustrate the overall secrecy capability with different penalty coefficients ξ. Then, we can find that as the penalty coefficient gradually increases, the overall secrecy capacity continuously decreases. Additionally, to investigate the effect on the quality of legitimate transmissions, we introduce legitimate connectivity which is defined as the probability that $\zeta_b[t] > \zeta_{th}$. As illustrated in Fig. 11(b), the legitimate connectivity increases, with the increase in the penalty coefficient. When $\xi = 1$, the penalty item W_b can elevate the legitimate connectivity of the cooperative intelligent jamming scheme to a maximum value of 99.6%. Meanwhile, $\xi = 0$ indicates that the reward function is not required to guarantee the legitimate connectivity, thus, GJs can obtain a large secrecy capacity as a reward. Hence, the setting of the penalty coefficient should trade-off the legitimate connectivity and the secrecy capacity.

Finally, in Fig. 12, we analyze the overall secrecy capability with different penalty coefficients δ for the penalty item W_r. We set the number of GJs as $N = 2$, the number of UAV eavesdroppers as $M = 2$, $r_{\min} = 5\text{m}$, and $r_{\max} = 45\text{m}$. From Fig. 12, we can observe that as the penalty coefficient δ gradually increases, the overall secrecy capability is first improved and then stabilized. In particular, in the absence of the penalty item W_r (i.e., the penalty coefficient $\delta = 0$), the overall secrecy capacity is significantly lower than that in the presence of the penalty item W_r. This is because, over multiple training episodes, the penalty item W_r ensures that the GJ never moves away from the protection region (i.e., the service region of the base station). Therefore, the GJ can timely respond to the next eavesdropping attack. This result shows that the penalty item W_r is essential in enhancing the overall secrecy capacity in the reward function.

VI. Discussion and Future Work

In this section, we discuss the potential challenges of the implementation of our cooperative jamming scheme. Moreover, we deliberate future work with promising techniques to further enhance the performance of cooperative intelligent jamming.

A. Discussion on the implementation of GJs

In the current study, we have analyzed that our collaborative jamming scheme can be used to prevent wiretapping from UAV eavesdroppers. Besides, with the QMIX-based collaborative jamming algorithm, the effectiveness of collaboration among multiple GJs can be further enhanced. Despite the advantages, we are aware of some challenges of multi-ground jamming collaboration in real-world deployments. In this subsection, we discuss these potential challenges as follows.

- *Hardware constraints:* The ground jammer, as an extra device, faces limitations in both energy and computing capability. Particularly, insufficient energy makes the GJ unable to achieve the optimal jamming power and movement speed, degrading the collaborative effectiveness of jamming. Inaccurate execution impacts the controllability of the policy network, as the actions output by the network are not aligned with the actions that the GJ performs. Moreover, limited computing capability prolongs both the training time and decision-making process, reducing algorithm stability and compromising the real-time effectiveness of jamming.
- *Environmental factors:* The deployment terrains can influent the performance of our cooperative jamming scheme. For example, some complex terrains, including steep slopes and obstructions, can impede the movement speed of GJs and alter the angle of their jamming emissions. Additionally, obstacles such as buildings and trees can significantly attenuate the jamming signal and disrupt the jamming trajectory, causing a lack of stability in the jamming performance.
- *Scalability:* To adapt the the diverse deployment scenarios, the deployed number of GJs is varied with the task requirement (e.g., the number of eavesdroppers). The training time for newly integrated GJs can hinder

the effectiveness of collaboration among existing GJs, thereby diminishing overall secrecy capacity. Additionally, the interdependence of GJ policies often results in convergence to a local optimum, particularly as the number of GJs increases. Furthermore, the complexity of finding a globally optimal policy also rises, thereby reducing algorithmic convergence.

B. Discussion on future work

As a promising technology, the cooperative intelligent jamming scheme provides secure data transmission in the presence of UAV eavesdroppers. To promote the implementation of this scheme, several future research directions are outlined as follows.

- *Tradeoff between energy consumption and jamming efficiency:* The jamming efficiency of GJs is extremely determined by energy resources. Investigating the tradeoff between energy consumption and jamming efficiency is an essential problem for the implementation of our scheme. The goal is to make GJs maintain high jamming effectiveness while degrading energy consumption (i.e., ensuring performance under constrained power resources). Through various energy allocation strategies generated by techniques like generative artificial intelligence algorithms, the optimal balance between jamming efficiency and energy consumption can be identified.
- *Multimodal data-driven collaborative jamming:* For practical deployment, complex environments hinder the performance of cooperative jamming. Therefore, multimodal data (including position and terrestrial data) is significant for collaborative jamming strategy. By collecting and analyzing data from multiple sensors, GJs make more informed and coordinated decisions in response to complex environments. For example, we can model complex environments, including building complexes and irregular terrain, by incorporating additional sensors to capture and process diverse types of environmental data. The modeled environment data refines the collaborative jamming strategy, enhancing its adaptability and effectiveness in challenging real-world implementation.
- *Game between jamming and eavesdropping:* With advancements in artificial intelligence, future eavesdroppers are foreseen to be intelligent, resulting in unpredictable eavesdropping trajectories. The UAV eavesdropper dynamically adjusts its trajectory by observing GJs, including the distribution of GJs, jamming trajectory, and jamming power. To counter intelligent eavesdroppers, the study of games between eavesdroppers and GJs is critical. By leveraging game theory, the current collaborative jamming algorithm can be further optimized, empowering GJs to effectively counter the sophisticated strategies employed by UAV eavesdroppers.

VII. Conclusion

In this paper, we proposed a cooperative intelligent jamming scheme to prevent UAV eavesdropping. We formulate an overall secrecy capability maximization problem with the constrained mobility of GJs. Then, we proposed a QMIX-based collaborative jamming algorithm with dueling DDQN for GJs to independently make decisions without sharing observations. The proposed scheme efficiently realizes the jamming trajectory design and jamming power allocation among multiple GJs, as indicated by the simulation results. Furthermore, the convergence of our proposed collaborative jamming algorithm for multiple ground jammers outperforms other benchmarks. Then, the designed penalty item effectively mitigates the interference of ground jammers on legitimate transmissions, while ensuring the secrecy capability. Moreover, the overall secrecy capability can be effectively guaranteed even if the movement speeds of GJs and UAV eavesdroppers are extremely uneven.

References

[1] B. Alzahrani, O. S. Oubbati, A. Barnawi, M. Atiquzzaman, and D. Alghazzawi, "UAV assistance paradigm: State-of-the-art in applications and challenges," *Journal of Network and Computer Applications*, vol. 166, p. 102706, 2020.

[2] S. Chai and V. K. N. Lau, "Multi-UAV Trajectory and Power Optimization for Cached UAV Wireless Networks With Energy and Content Recharging-Demand Driven Deep Learning Approach," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 10, pp. 3208–3224, 2021.

[3] L. Wang, H. Zhang, S. Guo, and D. Yuan, "Deployment and Association of Multiple UAVs in UAV-Assisted Cellular Networks With the Knowledge of Statistical User Position," *IEEE Transactions on Wireless Communications*, vol. 21, no. 8, pp. 6553–6567, 2022.

[4] V.-L. Nguyen, P.-C. Lin, B.-C. Cheng, R.-H. Hwang, and Y.-D. Lin, "Security and Privacy for 6G: A Survey on Prospective Technologies and Challenges," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 4, pp. 2384–2428, 2021.

[5] H. Wu, M. Li, Q. Gao, Z. Wei, N. Zhang, and X. Tao, "Eavesdropping and Anti-Eavesdropping Game in UAV Wiretap System: A Differential Game Approach," *IEEE Transactions on Wireless Communications*, vol. 21, no. 11, pp. 9906–9920, 2022.

[6] H. V. Poor and R. F. Schaefer, "Wireless physical layer security," *Proceedings of the National Academy of Sciences*, vol. 114, no. 1, pp. 19–26, 2017.

[7] Y. Zhou, C. Pan, P. L. Yeoh, K. Wang, M. Elkashlan, B. Vucetic, and Y. Li, "Secure Communications for UAV-Enabled Mobile Edge Computing Systems," *IEEE Transactions on Communications*, vol. 68, no. 1, pp. 376–388, 2020.

[8] W. Lu, Y. Ding, Y. Gao, Y. Chen, N. Zhao, Z. Ding, and A. Nallanathan, "Secure NOMA-Based UAV-MEC Network Towards a Flying Eavesdropper," *IEEE Transactions on Communications*, vol. 70, no. 5, pp. 3364–3376, 2022.

[9] B. Li, Y. Zou, J. Zhou, F. Wang, W. Cao, and Y.-D. Yao, "Secrecy Outage Probability Analysis of Friendly Jammer Selection Aided Multiuser Scheduling for Wireless Networks," *IEEE Transactions on Communications*, vol. 67, no. 5, pp. 3482–3495, 2019.

[10] Q. Wang, Y. Liu, H. Dai, M. Imran, and N. Nasser, "Ear in the Sky: Terrestrial Mobile Jamming to Prevent Aerial Eavesdropping," in *2021 IEEE Global Communications Conference (GLOBECOM)*, 2021, pp. 01–06.

[11] Y. Zeng, X. Xu, S. Jin, and R. Zhang, "Simultaneous Navigation and Radio Mapping for Cellular-Connected UAV With Deep Reinforcement Learning," *IEEE Transactions on Wireless Communications*, vol. 20, no. 7, pp. 4205–4220, 2021.

[12] J. Chen, H. Xing, Z. Xiao, L. Xu, and T. Tao, "A DRL Agent for Jointly Optimizing Computation Offloading and Resource Allocation in MEC," *IEEE Internet of Things Journal*, vol. 8, no. 24, pp. 17 508–17 524, 2021.

[13] K. Zhang, Z. Yang, and T. Başar, "Multi-agent reinforcement learning: A selective overview of theories and algorithms," *Handbook of reinforcement learning and control*, pp. 321–384, 2021.

[14] H. Dang-Ngoc, D. N. Nguyen, K. Ho-Van, D. T. Hoang, E. Dutkiewicz, Q.-V. Pham, and W.-J. Hwang, "Secure Swarm UAV-Assisted Communications With Cooperative Friendly Jamming," *IEEE Internet of Things Journal*, vol. 9, no. 24, pp. 25 596–25 611, 2022.

[15] Y. Zhou, P. L. Yeoh, C. Pan, K. Wang, Z. Ma, B. Vucetic, and Y. Li, "Caching and UAV Friendly Jamming for Secure Communications With Active Eavesdropping Attacks," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 10, pp. 11 251–11 256, 2022.

[16] W. Lu, Y. Ding, Y. Gao, S. Hu, Y. Wu, N. Zhao, and Y. Gong, "Resource and Trajectory Optimization for Secure Communications in Dual Unmanned Aerial Vehicle Mobile Edge Computing Systems," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 4, pp. 2704–2713, 2022.

[17] W. Lu, Y. Ding, Y. Gao, Y. Chen, N. Zhao, Z. Ding, and A. Nallanathan, "Secure NOMA-Based UAV-MEC Network Towards a Flying Eavesdropper," *IEEE Transactions on Communications*, vol. 70, no. 5, pp. 3364–3376, 2022.

[18] R. Jin, K. Zeng, and K. Zhang, "A Reassessment on Friendly Jamming Efficiency," *IEEE Transactions on Mobile Computing*, vol. 20, no. 1, pp. 32–47, 2021.

[19] A. S. Abdalla, A. Behfarnia, and V. Marojevic, "UAV Trajectory and Multi-User Beamforming Optimization for Clustered Users Against Passive Eavesdropping Attacks With Unknown CSI," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 11, pp. 14 426–14 442, 2023.

[20] H. Yang, Z. Xiong, J. Zhao, D. Niyato, L. Xiao, and Q. Wu, "Deep Reinforcement Learning-Based Intelligent Reflecting Surface for Secure Wireless Communications," *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 375–388, 2021.

[21] X. Tang, H. He, L. Dong, L. Li, Q. Du, and Z. Han, "Robust Secrecy via Aerial Reflection and Jamming: Joint Optimization of Deployment and Transmission," *IEEE Internet of Things Journal*, vol. 10, no. 14, pp. 12 562–12 576, 2023.

[22] E. T. Michailidis, M.-G. Volakaki, N. I. Miridakis, and D. Vouyioukas, "Optimization of Secure Computation Efficiency in UAV-Enabled RIS-Assisted MEC-IoT Networks With Aerial and Ground Eavesdroppers," *IEEE Transactions on Communications*, vol. 72, no. 7, pp. 3994–4009, 2024.

[23] N. Gao, Z. Qin, X. Jing, Q. Ni, and S. Jin, "Anti-Intelligent UAV Jamming Strategy via Deep Q-Networks," *IEEE Transactions on Communications*, vol. 68, no. 1, pp. 569–581, 2020.

[24] L. Xiao, H. Li, S. Yu, Y. Zhang, L.-C. Wang, and S. Ma, "Reinforcement Learning Based Network Coding for Drone-Aided Secure Wireless Communications," *IEEE Transactions on Communications*, vol. 70, no. 9, pp. 5975–5988, 2022.

[25] T. Ren, J. Niu, B. Dai, X. Liu, Z. Hu, M. Xu, and M. Guizani, "Enabling Efficient Scheduling in Large-Scale UAV-Assisted Mobile-Edge Computing via Hierarchical Reinforcement Learning," *IEEE Internet of Things Journal*, vol. 9, no. 10, pp. 7095–7109, 2022.

[26] N. Zhao, Y. Cheng, Y. Pei, Y.-C. Liang, and D. Niyato, "Deep Reinforcement Learning for Trajectory Design and Power Allocation in UAV Networks," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6.

[27] W. Chen, X. Qiu, T. Cai, H.-N. Dai, Z. Zheng, and Y. Zhang, "Deep Reinforcement Learning for Internet of Things: A Comprehensive Survey," *IEEE Communications Surveys Tutorials*, vol. 23, no. 3, pp. 1659–1692, 2021.

[28] H. Kang, X. Chang, J. Mišić, V. B. Mišić, J. Fan, and J. Bai, "Improving Dual-UAV Aided Ground-UAV Bi-Directional Communication Security: Joint UAV Trajectory and Transmit Power Optimization," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 10, pp. 10 570–10 583, 2022.

[29] Z. Wang, T. Schaul, M. Hessel, H. Hasselt, M. Lanctot, and N. Freitas, "Dueling network architectures for deep reinforcement learning," in *International conference on machine learning*. PMLR, 2016, pp. 1995–2003.

[30] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, no. 1, 2016.

[31] W. Shi, J. Li, H. Wu, C. Zhou, N. Cheng, and X. Shen, "Drone-Cell Trajectory Planning and Resource Allocation for Highly Mobile Networks: A Hierarchical DRL Approach," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9800–9813, 2021.

[32] R. Ding, F. Gao, and X. S. Shen, "3D UAV Trajectory Design and Frequency Band Allocation for Energy-Efficient and Fair Communication: A Deep Reinforcement Learning Approach," *IEEE Transactions on Wireless Communications*, vol. 19, no. 12, pp. 7796–7809, 2020.

[33] Q. Wei, Y. Li, J. Zhang, and F.-Y. Wang, "VGN: Value Decomposition With Graph Attention Networks for Multiagent Reinforcement Learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 1, pp. 182–195, 2024.

[34] T. Rashid, M. Samvelyan, C. S. De Witt, G. Farquhar, J. Foerster, and S. Whiteson, "Monotonic value function factorisation for deep multi-agent reinforcement learning," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 7234–7284, 2020.

[35] R. Pina, V. D. Silva, J. Hook, and A. Kondoz, "Residual Q-Networks for Value Function Factorizing in Multiagent Reinforcement Learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 2, pp. 1534–1544, 2024.

[36] Q. Wei, Y. Li, J. Zhang, and F.-Y. Wang, "VGN: Value Decomposition With Graph Attention Networks for Multiagent Reinforcement Learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 1, pp. 182–195, 2024.

[37] Z. Wang, V. Bapst, N. Heess, V. Mnih, R. Munos, K. Kavukcuoglu, and N. de Freitas, "Sample efficient actor-critic with experience replay," *arXiv preprint arXiv:1611.01224*, 2016.

[38] Y. Zhou, P. L. Yeoh, H. Chen, Y. Li, R. Schober, L. Zhuo, and B. Vucetic, "Improving Physical Layer Security via a UAV Friendly Jammer for Unknown Eavesdropper Location," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 11 280–11 284, 2018.

[39] 3rd Generation Partnership Project, "Unmanned Aerial System (UAS) support in 3GP," Tech. Rep. TS 22.125 v19.2.0 (2024-06028), 2024.

[40] S. Yin and F. R. Yu, "Resource Allocation and Trajectory Design in UAV-Aided Cellular Networks Based on Multiagent Reinforcement Learning," *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 2933–2943, 2022.

Qubeijian Wang (Member, IEEE) received the B.E. degree in electrical engineering from the University of Liverpool, U.K., in 2015, the M.E. degree in telecommunications from The University of Melbourne, Australia, in 2017, and the Ph.D. degree in electronic information technology from the Macau University of Science and Technology, Macau, in 2020. He is currently an Assistant Professor with the School of Cybersecurity, Northwestern Polytechnical University, China. His research interests include UAV-aided communications, physical-layer security, and large-scale network performance analysis. He serves as a TPC Member for conferences, including GLOBECOM2021-2023 and ICC 2024; and a reviewer for various prestigious IEEE journals.

Shiyue Tang received the B.Eng. degree in network engineering from the Xi'an University of Posts & Telecommunications, Xi'an, China, in 2022, and he is currently pursuing the M.Eng. degree in network and information security from Northwestern Polytechnical University, Xi'an, China, in 2025. His research interests include UAV aided communications, and wireless communication security.

Wen Sun (Senior Member, IEEE) received the BE degree from the Harbin Institute of Technology, in 2009, and the PhD degree in electrical and computer engineering from National University of Singapore, in 2014. She is currently a full professor with Northwestern Polytechnical University, China. Her research interests cover a wide range of areas including wireless mobile communications, IoT, 5G, and blockchain.She has published more than 50peer reviewed papers in various prestigious IEEE journals and conferences, including IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE NETWORK, IEEE WIRELESS COMMUNICATIONS. She was the recipient of the best paper award of GlobeCom2019.

Yin Zhang (Senior Member, IEEE) is currently a Full Professor with the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, China. He is the Co-Chair of the IEEE Computer Society Big Data Special Technical Community (STC). He serves as an Editor or an Associate Editor for IEEE Network, IEEE SYSTEMS JOURNAL, Information Fusion, and Journal of Circuits Systems and Computers.

Geng Sun (S'17-M'19) received the B.S. degree in communication engineering from Dalian Polytechnic University, and the Ph.D. degree in computer science and technology from Jilin University, in 2011 and 2018, respectively. He was a Visiting Researcher with the School of Electrical and Computer Engineering, Georgia Institute of Technology, USA. He is an Associate Professor in College of Computer Science and Technology at Jilin University, and His research interests include wireless networks, UAV communications, collaborative beamforming and optimizations.

Hong-Ning Dai (Senior Member, IEEE) received the Ph.D. degree in computer science and engineering from the Department of Computer Science and Engineering, The Chinese University of Hong Kong. Currently, he is an Associate Professor with the Department of Computer Science, at Hong Kong Baptist University, Hong Kong. His current research interests include the Internet of Things, big data, and blockchain technology. He has served as an Editor for Computer Communications (Elsevier), Connection Science (Taylor Francis), and IEEE Access, and a Guest Editor for IEEE Transactions on Industrial Informatics, IEEE Transaction Emerging Topics in Computing, and IEEE Open Journal of The Computer Society.

Mohsen Guizani (Fellow, IEEE) received the BS (with distinction), MS and PhD degrees in Electrical and Computer engineering from Syracuse University, Syracuse, NY, USA in 1985, 1987 and 1990, respectively. He is currently a Professor of Machine Learning at the Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, UAE. Previously, he worked in different institutions in the USA. His research interests include applied machine learning and artificial intelligence, smart city, Internet of Things (IoT), intelligent autonomous systems, and cybersecurity. He became an IEEE Fellow in 2009 and was listed as a Clarivate Analytics Highly Cited Researcher in Computer Science in 2019, 2020, 2021 and 2022. Dr. Guizani has won several research awards including the "2015 IEEE Communications Society Best Survey Paper Award", the Best ComSoc Journal Paper Award in 2021 as well 5 Best Paper Awards from ICC and Globecom Conferences. He is the author of 11 books, more than 1000 publications and several US patents. He is also the recipient of the 2017 IEEE Communications Society Wireless Technical Committee (WTC) Recognition Award, the 2018 AdHoc Technical Committee Recognition Award, and the 2019 IEEE Communications and Information Security Technical Recognition (CISTC) Award. He served as the Editor-in-Chief of IEEE Network and is currently serving on the Editorial Boards of many IEEE Transactions and Magazines. He was the Chair of the IEEE Communications Society Wireless Technical Committee and the Chair of the TAOS Technical Committee. He served as the IEEE Computer Society Distinguished Speaker and is currently the IEEE ComSoc Distinguished Lecturer.