From Static to Adaptive Defense: Federated Multi-Agent Deep Reinforcement Learning-Driven Moving Target Defense Against DoS Attacks in UAV Swarm Networks

Yuyang Zhou, *Member, IEEE*, Guang Cheng, *Member, IEEE*, Kang Du, *Student Member, IEEE*, Zihan Chen, *Member, IEEE*, Tian Qin, and Yuyu Zhao, *Member, IEEE*

***Abstract*—The proliferation of unmanned aerial vehicle (UAV) swarms has enabled a wide range of mission-critical applications, but also exposes UAV networks to severe Denial-of-Service (DoS) threats due to their open wireless environment, dynamic topology, and resource constraints. Traditional static or centralized defense mechanisms are often inadequate for such dynamic and distributed scenarios. To address these challenges, we propose a novel federated multi-agent deep reinforcement learning (FMADRL)-driven moving target defense (MTD) framework for proactive and adaptive DoS mitigation in UAV swarm networks. Specifically, we design three lightweight and coordinated MTD mechanisms, including leader switching, route mutation, and frequency hopping, that leverage the inherent flexibility of UAV swarms to disrupt attacker efforts and enhance network resilience. The defense problem is formulated as a multi-agent partially observable Markov decision process (POMDP), capturing the distributed, resource-constrained, and uncertain nature of UAV swarms under attack. Each UAV is equipped with a local policy agent that autonomously selects MTD actions based on partial observations and local experiences. By employing a policy gradient-based FMADRL algorithm, UAVs collaboratively optimize their defense policies via reward-weighted aggregation, enabling distributed learning without sharing raw data and thus reducing communication overhead. Extensive simulations demonstrate that our approach significantly outperforms state-of-the-art baselines, achieving up to a 34.6% improvement in attack mitigation rate, a reduction in average recovery time of up to 94.6%, and decreases in energy consumption and defense cost by as much as 29.3% and 98.3%, respectively, while maintaining robust mission continuity under various DoS attack strategies.**

***Index Terms*—Unmanned aerial vehicle swarm network, denial-of-service attacks, moving target defense, federated multi-agent deep reinforcement learning, policy gradient method.**

I. Introduction

THE rapid development of the unmanned aerial vehicle (UAV) technology [1] has enabled a wide range of applications, including environmental monitoring, disaster response, precision agriculture, logistics, aerial photography, and intelligent surveillance. By leveraging the collaborative capabilities of multiple UAVs, the UAV swarm [2] can achieve enhanced coverage, resilience, and real-time data processing, making them indispensable in both civilian and industrial domains. As the low-altitude economy continues to expand, UAV swarm networks [3] are expected to play an increasingly important role in smart cities, emergency management, and next-generation communication infrastructures.

Nevertheless, the widespread adoption of UAV swarms also brings new security challenges [4]. Due to their reliance on open wireless links, limited energy and processing capabilities, UAV networks are particularly vulnerable to *Denial-of-Service* (DoS) attacks [5], [6]. For example, UAVs often operate without robust authentication or traffic filtering mechanisms. Attackers can easily launch attacks by overwhelming communication channels or computational resources, leading to service disruptions or UAV disconnection from the swarm [7]. In mission-critical scenarios, even a brief loss of connectivity or control can have catastrophic consequences, including the failure of time-sensitive missions or the crash of drones.

In addition, UAV swarms operate in highly dynamic, resource-constrained, and often uncertain environments. The mobility of UAVs, the need for low-latency communication, and the distributed nature of control introduce unique challenges for both attack detection and defense [8]. Traditional security mechanisms, such as firewalls [9], intrusion detection systems (IDSs) [10], and traffic redirection, typically rely on prior knowledge of attack characteristics and are often designed for centralized infrastructures, making them ill-suited for UAV networks. Moreover, the static nature of these defenses results in several shortcomings: (i) limited adaptability to evolving and sophisticated attack patterns, (ii) increased false positive rates due to lack of contextual awareness, and (iii) reactive mitigation that only occurs after attacks have already caused damage. Therefore, ensuring the resilience of UAV networks against DoS attacks by more proactive and adaptive defense strategies is a critical research challenge.

Fortunately, *Moving Target Defense* (MTD) [11], [12] has emerged as a promising approach for proactively defending against various cyber threats. MTD aims to increase the uncertainty for attackers and disrupt attack reconnaissance-effort asymmetry by continuously or periodically changing the attack surface of a system [13]. However, current MTD implementations face three critical limitations in UAV contexts. (i) Existing MTD mechanisms are not primarily proposed for UAV networks, and the overhead of frequent adaptations may exceed the limited resources available on UAV platforms. (ii) Most MTD decision-making frameworks are designed to collect global information and coordinate defense actions in

Yuyang Zhou, Guang Cheng, Kang Du, Zihan Chen, Tian Qin, and Yuyu Zhao are with the School of Cyber Science and Engineering, Southeast University, Purple Mountain Laboratories, and Jiangsu Province Engineering Research Center of Security for Ubiquitous Network, Nanjing 211189, China. E-mail: {yyzhou, chengguang, dukang, zhchen_seu, 230208959, yyzhao}@seu.edu.cn.

Guang Cheng is the corresponding author.

a centralized manner, which cannot operate efficiently and reliably within the unique constraints of UAV swarm networks. (iii) Sophisticated attackers can reconstruct attack target states by tracking and learning the mutation patterns, enabling them to persistently target UAV networks despite the use of MTD. These limitations highlight the urgent need for distributed and intelligent MTD framework that can adaptively balance security and performance in UAV scenarios.

In this paper, we propose a novel *federated multi-agent deep reinforcement learning* (FMADRL)-driven MTD framework tailored for UAV swarm networks facing DoS attacks, where the bird's-eye view of the proposed method has been illustrated in Fig. 1. In our framework, we design three lightweight MTD actions, including (i) *Leader Switching*, (ii) *Route Mutation*, and (iii) *Frequency Hopping*, based on partial observations. The first mechanism allows the UAV swarm to promptly reassign the leader role among eligible UAVs when the current leader is persistently targeted, while the second mechanism dynamically reconfigures communication paths by selecting alternative relay UAVs, ensuring that critical messages can bypass compromised links and reach their destinations. Furthermore, frequency hopping periodically changes the communication frequency channels used by the swarm, significantly increasing the uncertainty for attackers and disrupting their ability to sustain effective attacks over time. To effectively mitigate adaptive DoS attacks while minimizing resource consumption and maintaining mission continuity, we first formulate the defense problem as a multi-agent partially observable Markov decision process (POMDP) and develop a policy gradient-based FMADRL (PG-FMADRL) method, where each UAV is treated as an independent agent that can learn and adapt its defense strategies based on local observations and experiences. Then, each agent periodically shares only its model parameters with a central aggregator using a federated learning scheme. This collaborative approach enables the UAV swarm to adaptively coordinate distributed defense strategies in real time without sharing raw data, thus respecting the stringent resource and latency constraints of UAV networks. The main contributions of this paper are summarized as follows:

- **Proactive and collaborative DoS defense framework.** We propose a novel framework for DoS mitigation in UAV swarm networks that enables self-adaptive defense among distributed UAV nodes. Our proposed approach eliminates the need for attack detection and advances security by establishing a proactive and collaborative paradigm.
- **Lightweight and adaptive MTD mechanisms.** Within the proposed framework, we consider the effects of heavy consumption and high delay issues that arise in existing MTD solutions, and thus design three lightweight, flexible, and adaptive MTD mechanisms specifically optimized for dynamic and resource-constrained UAV swarm environments.
- **Federated and intelligent defense decision-making.** We formulate the distributed defense problem as a multi-agent POMDP, capturing the uncertainty and limited observability in UAV swarm networks under attack. Based on this formulation, we develop a PG-FMADRL method that enables UAVs to collaboratively learn and optimize defense policies.
- **Comprehensive evaluation of system performance.** Through extensive simulations, we demonstrate that our method outperforms the state-of-the-art (SOTA) schemes in terms of attack mitigation rate, while imposing less recovery time and overhead. The source code is available at https://github.com/SEU-ProactiveSecurity-Group/PG-FMADRL.

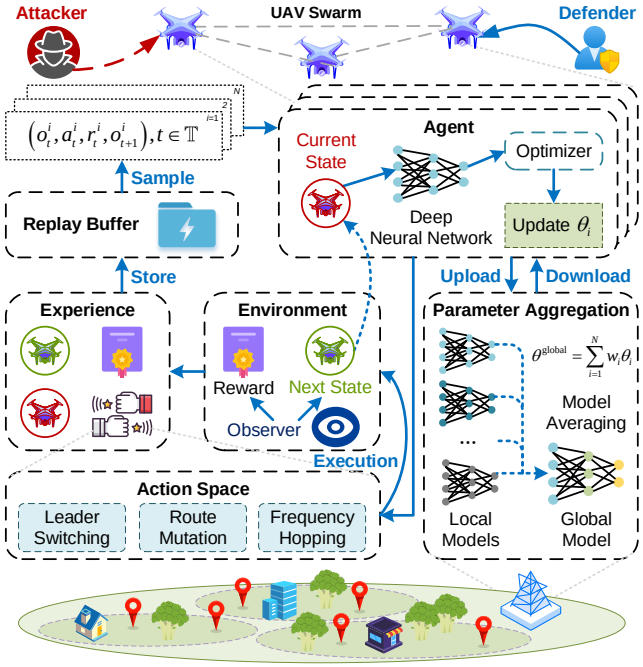


Fig. 1. Bird's-eye view of the proposed DoS mitigation approach.

The remainder of this paper is organized as follows. We first review related work on DoS mitigation and MTD techniques in Section II. Next, Section III introduces the system architecture and problem formulation. The details of the proposed FMADRL-driven MTD framework are presented in Section IV, followed by simulation results and performance analysis in Section V. Finally, Section VI concludes the paper and discusses potential future research directions.

II. Related Work

A. DoS Attacks Detection and Mitigation in UAV Networks

DoS attacks in UAV networks have been extensively studied, with various detection and mitigation techniques proposed. For instance, Fu et al. [14] proposed an IDS for UAV networks, integrating convolutional neural network (CNN) and long short-term memory (LSTM) to achieve high detection accuracy of 94.4% for DoS attacks. Similarly, Hassler et al. [15] employed a method that fuses UAV cyber and physical features, which achieves an accuracy of at most 98.5% for attacks. To overcome challenges such as small sample sizes and uneven data distribution among UAVs, He et al. [16] leveraged a conditional generative adversarial network (CGAN)-based intrusion detection algorithm, achieving an accuracy of 99% in detecting DoS attacks. However, these methods typically require large amounts of labeled data for training, which may not be available in real-world scenarios, especially when

timely response to novel or unseen attacks is required. Unlike existing methods, our approach operates without relying on prior knowledge or explicit attack signature identification, and instead dynamically orchestrates multiple MTD mechanisms to proactively respond to attacks in real time.

In addition to detection, several mitigation strategies have been proposed. For example, Gupta et al. [17] developed a machine learning (ML)-based Distributed DoS (DDoS) mitigation framework that leverages SDN's programmability and centralized control to enable intelligent traffic management in UAV environments. Recent research has also investigated adaptive control strategies to enhance the resilience of UAV swarms against DoS attacks. Wu et al. [18] proposed a zero-sum differential game approach to effectively handle connectivity disruptions caused by DoS attacks and ensure boundedness and consensus control performance among UAVs. Tang et al. [19] incorporated robustness constraints to enhance disturbance resilience and introduced a dynamic event-triggered mechanism to respond to varying attack durations. Nevertheless, these approaches usually enhance the robustness of UAV swarms through tolerance and isolation strategies, which may be insufficient against sophisticated adversaries. To cope with this challenge, our proposed method fully leverages MTD techniques to present dynamic external attributes for UAV swarms, and enables them to autonomously adjust their defense strategies according to real-time network conditions and attack patterns.

B. MTD-based Solutions for DoS/DDoS Attacks

Recently, various MTD-based approaches have been introduced to counter DoS/DDoS attacks [20]–[22]. These methods aim to increase the attackers' effort and cost by misleading them toward incorrect targets, thereby effectively diverting attacks away from the protected system. For example, Ribeiro et al. [23] proposed a novl architecture leveraging Software Defined Networking (SDN) for flow classification and MTD techniques for DDoS mitigation. Similarly, Zhang et al. [24] developed a collaborative mutation-based MTD (CM-MTD) framework to disrupt attacks. The authors formulated the MTD deployment as a semi-Markov decision process (SMDP), leveraging a hierarchical deep reinforcement learning (DRL) algorithm for scheduling of MTD actions. However, these MTD mechanisms are equipped with high computation resources, which are not suitable for resource-constrained environments such as UAV networks. In this study, we repurpose existing functionalities within UAVs as lightweight MTD mechanisms, enabling seamless integration into the dynamic defense of UAV swarms, while maintaining high effectiveness against DoS attacks.

To enable DoS/DDoS defense in resource-constrained environments, several recent studies have explored lightweight MTD mechanisms. For instance, Zhang et al. [25] designed configuration mutation mechanisms against DDoS attacks, where the communication ranges and capacities of roadside units (RSUs) are dynamically adjusted using a DRL algorithm. Similarly, Zhou et al. [26] introduced lightweight MTD mechanisms for the Internet of Things (IoT) environments, focusing on the dynamic control of device admission and service replica migration to defeat DDoS attacks. In the edge cloud context, the authors [13] designed several container-based MTD mechanisms and employed a deep Q-network (DQN) algorithm to optimize the trade-off between security and overhead when mitigating DDoS attacks. However, these mechanisms are not directly applicable to DoS defense in UAV networks. Furthermore, most existing studies that employ DRL for MTD optimization adopt centralized decision-making paradigms, which are ill-suited for the distributed and dynamic nature of UAV swarms. In contrast, our work proposes novel MTD mechanisms specifically designed for UAV environments and leverages a federated learning architecture to facilitate adaptive and collaborative defense across UAV swarm networks.

III. System Model and Problem Formulation

A. System Model

We consider a multi-UAV swarm system composed of a ground control station (GCS) and N UAVs, denoted as $\mathcal{V} = \{0, 1, ..., N\}$, where node 0 is the GCS and nodes 1 to N are UAVs. The system operates in a three-dimensional space and is tasked with persistent patrol and robust formation maintenance.

At any given time, one UAV is designated as the leader, responsible for receiving high-level commands from the GCS and relaying them to the rest of the swarm. The remaining UAVs act as followers, adjusting their positions based on local observations and received commands.

The communication network among the UAVs and the GCS is modeled as a dynamic undirected graph $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$. An edge $(i, j) \in \mathcal{E}(t)$ exists if and only if UAVs i and j are within a communication range R_c, operate on the same frequency channel f at time t. Formally,

$$(i,j) \in \mathcal{E}(t) \iff |\mathbf{p}_i(t) - \mathbf{p}_j(t)| \leq R_c, \quad f_i(t) = f_j(t) \quad (1)$$

where $\mathbf{p}_i(t)$ denotes the position of UAV i at time t.

The UAV swarm is required to maintain a circular formation of radius $\mathbf{r}$ centered at $\mathbf{c} = (c_x, c_y, h)$. The ideal position for UAV i at time t is given by

$$\mathbf{p}_i^*(t) = \mathbf{c} + \mathbf{r} \cdot \begin{bmatrix} \cos\theta_i(t) \\ \sin\theta_i(t) \\ 0 \end{bmatrix} \quad (2)$$

where $\theta_i(t)$ is the desired angular position of UAV i. The formation rotates at a constant angular speed ω, so that

$$\theta_i(t) = \theta_i(0) + \omega t \quad (3)$$

where $\theta_i(0)$ is the initial angular position of UAV i. The patrol (linear) speed of each UAV along the circle is given by $v_{\text{pat}} = r\omega$, where r is the formation radius. To ensure collision avoidance, the UAVs are required to maintain a minimum distance $d_{\min}$ from each other at all times such that

$$|\mathbf{p}_i(t) - \mathbf{p}_j(t)| \geq d_{\min}, \quad \forall i, j \in \mathcal{V} \setminus \{0\} \quad (4)$$

where $d_{\min}$ is the minimum separation distance between UAVs. The maximum speed of each UAV is denoted as $v_{\max}$, which is the upper limit on the flying speed of the UAVs.

At each time step, each UAV updates its position and heading to minimize the deviation from its ideal position $\mathbf{p}_i^*(t)$ with its velocity $v_i(t)$. If the deviation exceeds a threshold δ, i.e., $|\mathbf{p}_i(t) - \mathbf{p}_i^*(t)| > \delta$, the UAV will accelerate and move toward its ideal position at maximum speed $v_{\max}$ until the deviation falls below the threshold. Otherwise, it continues patrolling at the nominal speed v_{pat}.

B. Threat Model

In this work, we consider a UAV swarm network operating in an adversarial environment and the GCS is trusted, where external attackers aim to disrupt the swarm's communication and coordinated behavior. To comprehensively analyze the system's resilience and develop effective defense, we define the threat model in terms of the attack types and their strategies.

1) Attack Types: The attacker can launch two types of DoS attacks against the UAV swarm communication system according to attack targets as follows.

- **Node Attack**: The attacker targets a specific UAV i and occupies its computation resources by sending a large amount of malicious requests through the current frequency [27]. When the UAV i is under node attack at time t, it can be formally described as $\phi_i^{\mathcal{N}}(t) = 1$.
- **Link Attack**: The attacker targets a specific communication link (i, j) between two UAVs or the GCS and the leader UAV, jamming the wireless channel by transmitting interference signals [28]. When the link (i, j) is under DoS attack at time t, we can describe the communication between UAVs i and j as $\phi_{ij}^{\mathcal{L}}(t) = 1$.

2) Attacker Strategies: To capture a wide range of realistic adversarial behaviors, we consider three types of attacker strategies, such that

- **Fixed Attacker**: The attacker selects a specific UAV or communication link as the attack target at the beginning of the DoS attack and persistently attacks it throughout the whole attack-defense interaction [29]. For example, if the fixed attacker selects a UAV node i as the target with initialized attack frequency f_{atk}, then, this attack remains active for the entire interaction and can be described as $\phi_i^{\mathcal{N}}(t) = 1$ and $f_{\text{atk}}(t) = f_i(0)$ for all time steps $t \in \mathbb{T} = \{1, 2, \ldots, T\}$, where T denotes the time horizon.
- **Random Attacker**: At each attack opportunity, the attacker randomly selects a UAV node or communication link from the set of available targets [30]. The attacker launches an attack on the chosen target for a fixed attack duration τ_{atk}. After each attack, the attacker needs to wait for a reconnaissance period τ_{recon} before initiating the next attack. For example, if the random attacker selects a UAV i as the target at time step 0, then the attack can be described as $\phi_i^{\mathcal{N}}(t) = 1$ and $f_{\text{atk}}(t) = f_i(0)$ for $t \in [0, \tau_{\text{atk}} + \tau_{\text{recon}})$, and $\phi_j^{\mathcal{N}}(t) = 0$ for all other UAVs $j \neq i$. Thus, the next attack can only start after initial attack ends and the attack frequency is updated to match the current frequency of the another selected UAV j, i.e., $f_{\text{atk}}(\tau_{\text{atk}} + \tau_{\text{recon}}) = f_j(\tau_{\text{atk}} + \tau_{\text{recon}})$.
- **Greedy Attacker**: The adversary dynamically observes the network state, enabling adaptively selecting the target that is expected to cause the maximum disruption to the swarm [31]. For example, the greedy attacker can select the current leader UAV $\mathbb{L}^{\mathcal{N}}(t)$ or the core link $\mathbb{L}^{\mathcal{L}}(t)$ of the swam as the target at time step t. For the sake of not weakening the greedy attacker's ability, we assume that he/she may be aware of the existence of defense mechanisms (e.g., MTD), but not their specific deployment or timing. Therefore, the attacker can re-evaluate and update his/her target at time step t' to maximize attack effectiveness, but attack actions are subject to duration and reconnaissance constraints that are similar to those of the random attacker, i.e., $t' = t + \tau_{\text{atk}} + \tau_{\text{recon}}$.

The effectiveness of an attack depends on the frequency alignment between the attacker and the victim, as well as the lasting time of attack duration and reconnaissance. For the nth node attack, it can be formally described as

$$E_i^{\mathcal{N}}(t) = \begin{cases} 1, & f_i(t) = f_{\text{atk}}(t) \wedge t \in [(n-1)\tau_{\text{eff}}, n\tau_{\text{eff}}) \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where $\tau_{\text{eff}} = \tau_{\text{atk}} + \tau_{\text{recon}}$ is the lasting time of a round of attack. Similarly, the link attack can be described as

$$E_{ij}^{\mathcal{L}}(t) = \begin{cases} & (i,j) \in \mathcal{E}(t) \\ 1, & \wedge \ f_i(t) = f_j(t) = f_{\text{atk}}(t) \\ & \wedge \ t \in [(n-1)\tau_{\text{eff}}, n\tau_{\text{eff}}) \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

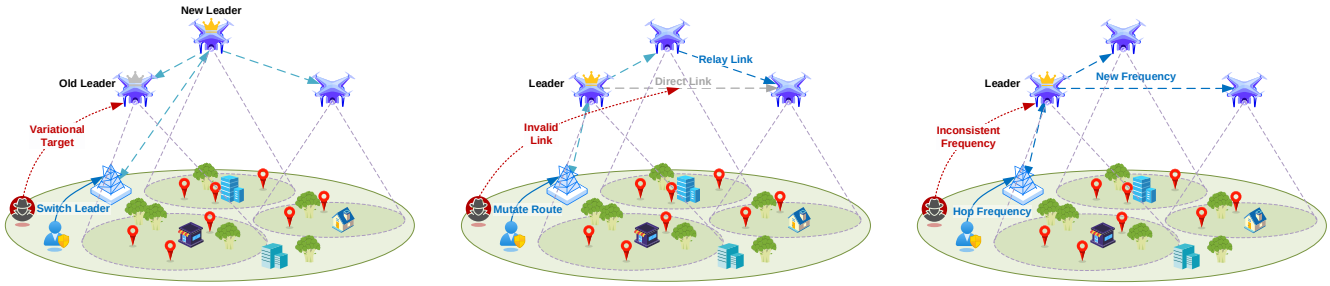
C. Moving Target Defense Mechanisms

Traditional defense mechanisms provide attackers with a stable attack surface, allowing them to gradually accumulate knowledge and optimize their strategies. To break this asymmetry and increase the attacker's uncertainty and cost, MTD has emerged as a proactive paradigm that dynamically shifts the system's attack surface, thereby disrupting the attacker's reconnaissance and exploitation process [32].

While many MTD techniques, such as IP address randomization and virtual machine migration, have been successfully deployed in traditional network environments with elastic resources, their high computational and communication overhead makes them unsuitable for resource-constrained UAV swarms. Moreover, the real-time requirements and dynamic topology of UAV networks further limit the applicability of existing MTD solutions.

To address these challenges, we propose a suite of lightweight and coordinated MTD mechanisms specifically tailored for UAV swarm networks. These mechanisms are designed to leverage the inherent dynamicity and flexibility of UAVs, enabling effective defense with minimal resource consumption and latency. Fig. 2 provides an overview of the proposed MTD-based mitigation mechanisms.

1) Leader Switching: As we discussed in Section III-B2, the leader UAV and the communication link between the GCS and the leader are often the primary targets for attackers, particularly those employing greedy strategies to maximize disruption. To counteract this vulnerability, we leverage the inherent dynamicity of the swarm to implement leader switching as a form of higher-level MTD mechanism.



(a) Leader Switching (b) Route Mutation (c) Frequency Hopping

Fig. 2. An overview of the proposed MTD-based mitigation mechanisms, including (a) leader switching, (b) route mutation, and (c) frequency hopping.

Leader switching designates a new UAV as the swarm leader when the current leader is persistently targeted or isolated by attacks. It is important to note that leader switching is a logical (virtual) role reassignment that can be executed rapidly and with minimal overhead and does not require any physical movement within the swarm. Let $\mathbb{L}^{\mathcal{N}}(t)$ denote the swarm leader at time t. The system can switch to a new leader chosen from the set of eligible UAVs after the leader switching occurs:

$$\mathbb{L}^{\mathcal{N}}(t+\tau^{\mathbb{L}}_{\text{exec}}) = i, \quad i \in \mathcal{V} \setminus \{0, \mathbb{L}^{\mathcal{N}}(t)\} \tag{7}$$

Here, i cannot be 0 (the GCS) or the current leader and $\tau^{\mathbb{L}}_{\text{exec}}$ denotes the execution time that the leader switching mechanism takes. After the switch, the new leader takes over the responsibility of receiving control commands from the GCS and disseminating them to the follower UAVs through the flying Ad-Hoc network (FANET).

2) Route Mutation: Route mutation is a mechanism that dynamically alters the communication paths within a network by changing the routing of data packets between nodes. In the context of UAV swarm networks, route mutation refers to the process of reconfiguring the communication topology so that messages can be delivered through different relay UAV nodes or links, rather than following a fixed, predictable path.

Usually, the command is transferred from the GCS to the leader, and is then forwarded to the follower UAVs. However, when a direct communication link is blocked by an attack, we exploit route mutation that selects an alternative relay UAV to forward messages. For example, when the communication link between the current leader $\mathbb{L}^{\mathcal{N}}(t) = i$ and a follower UAV j is under attack, then the relay UAV $r \in \mathcal{V} \setminus \{0, i, j\}$ should satisfy the following rules:

$$E^{\mathcal{L}}_{ir}(t) = 0 \wedge (r, j) \in \mathcal{E}(t) \tag{8}$$

This operation dynamically reconfigures the communication graph in real time, allowing the command to be forwarded through the relay UAV and successfully delivered to the follower, thus mitigating the impact of the targeted link attack, i.e., $E^{\mathcal{L}}_{ij}(t + \tau^{\mathbb{R}}_{\text{exec}}) = 0$, where $\tau^{\mathbb{R}}_{\text{exec}}$ is the time for updating the route table. When there is no ongoing attack or when the attack fails to produce a substantial impact, the system can further optimize network performance by restoring the relay-based communication path back to a direct link. This reduces communication latency and improves overall efficiency without compromising the defense capability.

3) Frequency Hopping: Despite the effectiveness of the aforementioned MTD mechanisms in mitigating DoS attacks, there remains a significant risk that a determined attacker could continuously target the leader UAV or the critical communication link between the GCS and the leader. Such persistent attacks may paralyze the entire UAV swarm's communication, rendering the swarm unable to coordinate or execute its patrol and mission tasks. Therefore, it is essential to introduce an additional defense mechanism that can further disrupt the attacker's ability to maintain effective defense and ensure the continuity of swarm operations.

Frequency hopping is a proactive MTD technique that addresses this challenge by dynamically and periodically changing the communication frequency channels used by all UAVs and the ground control station. At each defense opportunity, if the agent chooses the frequency hopping action, the swarm can switch from the last frequency to a new frequency selected from the available set $\mathcal{F}$, such that

$$f_i(t+\tau^{\mathbb{F}}_{\text{exec}}) \neq f_i(t), \quad \forall i \in \mathcal{V}, \quad f_i(t+\tau^{\mathbb{F}}_{\text{exec}}), f_i(t) \in \mathcal{F} \tag{9}$$

where $\tau^{\mathbb{F}}_{\text{exec}}$ denotes the time cost of frequency hopping between all UAVs and the GCS. This approach significantly increases the uncertainty for attackers, as they must continuously detect and adapt to the new frequency to sustain their attack.

D. Problem Formulation

For the UAV swarm network, its objective is to defend against the malicious DoS attacker. The UAVs need to choose the proper defensive actions to maximize the security performance, maintain the formation, and ensure communication connectivity while minimizing the defense consumption in a distributed setting. Therefore, we formulate the optimization problem for the UAV swarm network security as follows:

$$\begin{aligned}
\textbf{P1:} \Big(& \min_{(i,j)\in\mathcal{E}(t)} \sum_{t=1}^{T} E^{\mathcal{N}}_{i}(t) + E^{\mathcal{L}}_{ij}(t), \max_{t\in\mathbb{T}} |\mathcal{E}(t)|, \\
& \min_{i\in\mathcal{V}\setminus\{0\}} \sum_{t=1}^{T} |\mathbf{p}_i(t) - \mathbf{p}^*_i(t)|, \min_{i\in\mathcal{V}\setminus\{0\}} \sum_{t=1}^{T} C^i(t) \Big) \\
\textit{s.t.}\ & \textbf{C1:}\ Eqs.\ (1), (4), (5), (6), (8) \\
& \textbf{C2:}\ 0 \le \theta_i(t) < 2\pi, \quad \forall i \in \mathcal{V} \setminus \{0\}, \forall t \in \mathbb{T} \\
& \textbf{C3:}\ r\omega \le v_i(t) \le v_{\max}, \quad \forall i \in \mathcal{V} \setminus \{0\}, \forall t \in \mathbb{T} \\
& \textbf{C4:}\ f_i(t) = f_j(t) \in \mathcal{F}, \quad \forall i, j \in \mathcal{V}, \forall t \in \mathbb{T} \\
& \textbf{C5:}\ 0 < n\tau_{\text{eff}} \le T, \quad \forall n \in \mathbb{N}
\end{aligned} \tag{10}$$

where $|\mathcal{E}(t)|$ is the cardinality of this set and $C^i(t)$ is the defense cost of UAV i at each time slot. Constraints **C1** restrict the bound of the distance between UAVs, the attack effectiveness, and the communication links. Constraint **C2** ensures that the heading angle of each UAV is within the range $[0, 2\pi)$. The patrol speed of each UAV to be within the range of $[r\omega, v_{\max}]$ is enforced by Constraint **C3**, where r is the formation radius and ω is the angular speed. Constraint **C4** ensures that all UAVs operate on the same frequency channel at any time step, which is essential for maintaining communication connectivity. Finally, the attack duration and reconnaissance period are non-negative integers, allowing for a well-defined attack cycle, as delineated in constraint **C5**.

E. Optimization Analysis

It is worth noting that the multi-objective optimization problem above might not be solved by using traditional methods since obtaining accurate knowledge of the system dynamics and the attacker's behavior is impractical in real-world scenarios. Therefore, in this paper, we explore a DRL-based approach, leveraging the fitting capability of neural networks for approximating probability distributions and generating defense policies. We formulate the defense problem for the UAV swarm as a Partially Observable Markov Decision Process (POMDP) for each agent, reflecting the fact that each UAV can only access local and partial observations of the global environment. This formulation can be completely described through its state space, action space, transition probabilities, and reward function as follows.

1) State Space: Let S_t denotes the global environment state at time step t, which is related to the state and action of the previous time step. In this scenario, it can be represented as

$$S_t = [\mathbf{P}(t), \mathbf{V}(t), \mathbf{H}(t), \mathbf{L}(t), \mathbf{F}(t), \mathcal{E}(t)], \quad t \in \mathbb{T} \tag{11}$$

where $\mathbf{P}(t), \mathbf{V}(t), \mathbf{H}(t)$, and $\mathbf{F}(t)$ denote the positions, velocities, headings, and frequencies of all UAVs at time t, respectively. For example, $\mathbf{P}(t) = [\mathbf{p}_1(t), \mathbf{p}_2(t), \ldots, \mathbf{p}_N(t)]$ and $\mathbf{F}(t) = [f_1(t), f_2(t), \ldots, f_N(t)]$. Moreover, $\mathbf{L}(t)$ is the binary indicator of the current leader UAV, which is the UAV that receives commands from the GCS and coordinates the swarm, such that $\mathbf{l}_i(t) = 1$ when $i = \mathbb{L}^{\mathcal{N}}(t)$, otherwise 0. Finally, $\mathcal{E}(t)$ denotes the communication graph at time t, which captures the connectivity between UAVs based on their positions and frequencies (can be derived from Eq. (1)).

The state of UAV i at time step t can be represented as s_t^i. However, each agent i receives a local observation $o_t^i = O(S_t, i)$, which typically includes its own position, velocity, heading, the leadership indicator, its current frequency, and the local communication status. Formally, the observation space for agent i can be defined as

$$o_t^i = [\mathbf{p}_i(t), \mathbf{v}_i(t), \mathbf{h}_i(t), \mathbf{L}(t), f_i(t), e_i(t)] \tag{12}$$

where $\mathbf{p}_i(t), \mathbf{v}_i(t), \mathbf{h}_i(t)$, and $f_i(t)$ are the position, velocity, heading, and the current frequency of agent i at time t, respectively. For instance, when the UAV swarm patrols counterclockwise and each UAV always follows its ideal trajectory, the heading can be calculated as $h_i(t) = \theta_i(t) + \frac{\pi}{2}$.

It should be noted that each agent is aware of the global leadership status $\mathbf{L}(t)$, as every UAV, regardless of whether it is the leader, needs to receive control commands from the current leader. In addition, the local communication status $e_i(t)$ can include information such as whether it can receive commands from the GCS (as a leader) or the current leader UAV (as a follower). This information is crucial for the agent to make informed decisions about its defense actions, which can be represented as

$$e_i(t) = \mathbb{I}\left[\, (0, i) \in \mathcal{E}(t) \;\text{ or }\; (\mathbb{L}^{\mathcal{N}}(t), i) \in \mathcal{E}(t) \,\right] \tag{13}$$

2) Action Space: We assume that the agent makes defense decisions in an independent manner, implying that it may decide to take different actions on specific UAVs. At each time step, an agent i can execute one of three MTD mechanisms as described in Section III-C. Hence, we can define the action space for any state in S_t as

$$A_t = [\tilde{A}_t^{\mathbb{L}}, \tilde{A}_t^{\mathbb{R}}, \tilde{A}_t^{\mathbb{F}}], \quad t \in \mathbb{T} \tag{14}$$

Specifically, $\tilde{A}_t^{\mathbb{L}} = \{\mathbf{l}_1(t), \mathbf{l}_2(t), ..., \mathbf{l}_N(t)\}$ denotes the decisions of leader switching at time step t. Similarly, $\tilde{A}_t^{\mathbb{R}} = \{a_{t,1}^{\mathbb{R}}, a_{t,2}^{\mathbb{R}}, ..., a_{t,N}^{\mathbb{R}}\}$ denotes the actions of route mutation for all UAVs, where $a_{t,n}^{\mathbb{R}} = 1$ represents that the i-th UAV plays the role of relay node whereas $a_{t,n}^{\mathbb{R}} = -1$ means canceling its role. Furthermore, $\tilde{A}_t^{\mathbb{F}} = \{a_{t,1}^{\mathbb{F}}, a_{t,2}^{\mathbb{F}}, ..., a_{t,N}^{\mathbb{F}}\}$ shows a global action of frequency hopping on this UAV swarm, where $\tilde{A}_t^{\mathbb{F}} = [1, 1, \ldots, 1]$ triggers the change of the communication frequency for all UAVs and the GCS. It is worth noting that a zero value indicates no action, that is, this MTD mechanism will not be executed at this time. Therefore, the action selected by agent i at time step t can be represented as

$$a_t^i = \left[\mathbf{l}_i(t), a_{t,i}^{\mathbb{R}}, a_{t,i}^{\mathbb{F}}\right] \in A_t, \quad i \in \mathcal{V} \setminus \{0\} \tag{15}$$

3) Transition Probability: Let $S_t \times A_t \times S_{t+1} \to [0, 1]$ represent the state transition function. The environment transitions to a new state s' according to the transition probability $P(s'|s, a)$, where s is the current state at time t and $a \in A_t$ is the action selected by all agents. This probability is influenced by the system state, the attacker's strategy, and the defender's chosen actions. In the case of deterministic transitions, the probability becomes binary and can be formulated as follows:

$$P(s'|s, a) \in \{0, 1\}, \quad \forall s, s' \in S_{t+1}, a \in A_t \tag{16}$$

For instance, if the defender takes no action (i.e., all elements in A_t are zero) on all UAVs and the attacker continues with the same malicious requests, the system remains in its current state, resulting in $P(s'|s, a) = 0$. Conversely, if the defender initiates a global frequency hopping when there is no ongoing attack, the resulting system state can be determined directly by this action.

4) Reward Function: In a multi-agent setting, each agent i receives a reward based on its actions and the current state of the environment. In our scenario, we design to encourage the agents to maintain formation, ensure communication connectivity, and effectively respond to attacks while minimizing costs associated with defense actions. The reward function for agent i at time step t can be expressed as

$$r_t^i = \alpha R_{\text{com}}^i(t) + \beta R_{\text{form}}^i(t) - \zeta C^i(t) - \eta P_{\text{atk}}^i(t) - \xi P_{\text{vel}}^i(t) \tag{17}$$

where α, β, ζ, η, and ξ are coefficients for balancing the reward. First, the reward of communication connectivity is defined by $R^i_{\text{com}}(t) = e_i(t)$, which is a binary indicator that represents whether agent i is connected to the leader or the GCS at time step t. Second, the reward of formation is defined by $R^i_{\text{form}}(t) = 1 - |\mathbf{p}_i(t) - \mathbf{p}^*_i(t)|/\delta$, which captures the deviation of agent i from its ideal position $\mathbf{p}^*_i(t)$, normalized by a threshold δ.

Then, we can define the cost of defense actions as $C^i(t) = \sum_{i=1}^{N} \mathbf{l}_i(t) + a^{\mathbb{R}}_{t,i} + a^{\mathbb{F}}_{t,i}$, which represents the total number of relay actions taken by agent i at time step t. The penalty for being under effective attack is defined as $P^i_{\text{atk}}(t) = E^{\mathcal{N}}_i(t) + \frac{1}{2}E^{\mathcal{L}}_{i\mathbb{L}^{\mathcal{N}}(t)}(t)$, which captures the impact of node and link attacks on agent i. Finally, the penalty for excessive velocity is defined as $P^i_{\text{vel}}(t) = \frac{v_i(t) - v_{\text{pat}}}{v_{\max} - v_{\text{pat}}}$ which penalizes agent i if its speed exceeds the allowed patrol speed.

IV. Federated Multi-Agent Deep Reinforcement Learning Framework

To enable scalable, robust, and proactive defense in UAV swarm networks, we design a federated multi-agent deep reinforcement learning (FMADRL) framework. In this framework, each UAV is equipped with a local policy agent that learns to select MTD actions based on its own observations, while periodically participating in federated parameter aggregation to accelerate learning and improve generalization.

A. Framework Overview

The proposed FMADRL framework consists of N distributed agents, each deployed on a UAV in the swarm, and a central aggregator (e.g., the GCS) responsible for federated parameter aggregation.

During execution, each agent i interacts with the environment based on its local observation o^i_t (as defined in Eq. (12)), selects an action a^i_t from the action space A_t (see Eqs. (14) and (15)), receives a reward r^i_t (see Eq. (17)), collects local experience, and updates its policy network parameters. The proposed framework allows the UAV swarm to learn adaptive and coordinated strategies against DoS attacks in a distributed and resource-efficient manner while ensuring scalability.

After a fixed number of episodes, agents upload their local model parameters to the aggregator, which computes a global average and redistributes the updated parameters back to the agents. This process enables collaborative learning without sharing raw data, thus reducing communication overhead.

B. Local Policy Optimization

In the local scenario, deep neural network (DNN) models are utilized to build the learning agent. Each agent maintains a policy network $\pi_{\theta_i}(a|o)$ parameterized by θ_i. At each time step t, the agent receives a local observation o^i_t and samples an action a^i_t from the policy as

$$a^i_t \sim \pi_\theta(a^i_t|o^i_t) \tag{18}$$

The policy updating procedure adjusts the parameter θ_i to improve the expected long-term cumulative reward of the agent i for action a^i_t given state s^i_t. The values of actions for sequential observations are measured with the action-value function (or Q-function) to evaluate the expected total return per action. Here, the Q-function is formulated as

$$Q^\pi(s,a) = \mathbb{E}\left[\Omega^i_t | s = s^i_t, a = a^i_t\right] \tag{19}$$

where Ω^i_t is the expected return for agent i at time step t, which can be computed as the discounted sum of future rewards as:

$$\Omega^i_t = \sum_{t=1}^{T} \gamma^{t-1} r^i_t \tag{20}$$

where $\gamma \in [0,1]$ is the discount factor that balances immediate and future rewards. Thus, we can also have the Q-function as

$$Q^\pi(s,a) = \mathbb{E}\Big[r(s,a) + \gamma \mathbb{E}_{a' \sim \pi}\left[Q^\pi(s',a')\right]\Big] \tag{21}$$

where $r(s,a)$ is the immediate reward for taking action a in state s, and s' is the next state after taking action a.

The objective of policy learning is to develop an optimal policy $\pi^*_{\theta_i}$ that maps sequences to actions to maximize the objective function as

$$\pi^*_{\theta_i} = \arg\max_{\pi} J(\theta_i) \tag{22}$$

where $J(\theta_i)$ is the expected long-term cumulative reward of agent i, defined as

$$J(\theta_i) = \mathbb{E}_{s \sim d^{\pi_{\theta_i}},\, a \sim \pi_{\theta_i}}\left[Q^{\pi_{\theta_i}}(s,a)\right] \tag{23}$$

where $d^{\pi_{\theta_i}}$ is the state distribution under policy π_{θ_i}.

Each agent stores its local experience tuples $(o^i_t, a^i_t, r^i_t, o^i_{t+1})$ in a buffer $\mathcal{D}_i$. After collecting sufficient experience, the agent performs a policy gradient update. The gradient of the function should be calculated with respect to as follows:

$$\nabla_{\theta_i} J(\theta_i) = \mathbb{E}_{\pi_{\theta_i}}\left[\nabla_{\theta_i} \log \pi_{\theta_i}(a^i_t|o^i_t) Q^{\pi_{\theta_i}}(s^i_t, a^i_t)\right] \tag{24}$$

where $\nabla_{\theta_i} \log \pi_{\theta_i}(a^i_t|o^i_t)$ is the score function that measures how much the policy changes with respect to the action taken. This gradient can be estimated using Monte Carlo sampling or temporal difference methods.

For optimizing the objective function, the policy parameters are updated by minimizing the following loss function:

$$L(\theta_i) = -\mathbb{E}_{\pi_{\theta_i}}\left[\log \pi_{\theta_i}(a^i_t|o^i_t) Q^{\pi_{\theta_i}}(s^i_t, a^i_t)\right] \tag{25}$$

where the negative sign indicates that we want to maximize the expected return. In addition, we incorporate an entropy regularization term to encourage exploration and prevent premature convergence to suboptimal policies. The modified loss function becomes

$$L(\theta_i) = -\mathbb{E}_{\pi_{\theta_i}}\Big[\log \pi_{\theta_i}(a^i_t|o^i_t)\hat{R}^i_t - \mu H(\pi_{\theta_i}(\cdot|o^i_t))\Big] \tag{26}$$

where $\hat{R}^i_t$ is the normalized return, $H(\cdot)$ is the entropy regularization term to encourage exploration, and μ is the entropy coefficient. In practice, the loss function for each agent can be expressed as

$$L(\theta_i) = -\frac{1}{T}\sum_{t=1}^{T} \log \pi_{\theta_i}(a^i_t|o^i_t) \cdot \hat{R}^i_t - \mu H(\pi_{\theta_i}(\cdot|o^i_t)) \tag{27}$$

The agent can update its policy parameters θ_i using gradient descent or adaptive optimization algorithms such as Adam.

C. Federated Parameter Aggregation

In our FMADRL framework, the aggregation of policy parameters across UAV agents is designed to maximize both learning efficiency and model personalization.

After a number of local updates (e.g., K episodes), all agents upload their local parameters θ_i to the central aggregator. Instead of naive averaging, we perform a reward-weighted aggregation of policy parameters. For each participating agent i, we compute a weight w_i proportional to its recent average return over the most recent M episodes, denoted as:

$$\bar{R}_i^{(M)} = \frac{1}{M} \sum_{m=1}^{M} \left(\sum_{t=1}^{T} r_{t,m}^i \right) \tag{28}$$

where $r_{t,m}^i$ is the reward received by agent i at time step t in the m-th most recent episode. The weight for agent i is then calculated as:

$$w_i = \frac{\bar{R}_i^{(M)}}{\sum_{j=1}^{N} \bar{R}_j^{(M)}} \tag{29}$$

To balance generalization and personalization, we only aggregate the parameters of the shared layers (e.g., feature extraction layers) across agents. The output layers remain local to each agent, allowing for adaptation to individual UAV roles or local environments. Thus, the global parameter for each shared layer is computed as:

$$\theta^{\text{global}} = \sum_{i=1}^{N} w_i \theta_i \tag{30}$$

After receiving the updated shared parameters θ^{global}, each agent i synchronizes the shared layers of its local policy network. To further personalize the policy, each agent then performs T_{local} steps of local fine-tuning using its own recent experience buffer $\mathcal{D}_i$. Specifically, the agent updates its full parameter set θ_i by minimizing the local loss function (e.g., Eq. (27)). This process enables each agent to adapt the global model to its local environment and recent experiences, thus achieving a balance between collaborative learning and individual specialization.

D. Policy Gradient-Based FMADRL Algorithm

To provide a clear overview of the FMADRL framework for MTD deployment in defeating DoS mitigation, we summarize the overall training procedure in the proposed policy gradient-based FMADRL (PG-FMADRL) algorithm (see Algorithm 1). This algorithm integrates the local policy optimization and federated parameter aggregation steps described in previous sections, enabling distributed UAV agents to collaboratively learn robust MTD strategies for DoS attack mitigation.

1) Algorithm Description: Specifically, at the beginning of the training process, we first initialize the global shared parameters θ^{global} in Line 1. For each agent i, the local policy parameters θ_i are set to θ^{global}, and the local experience buffer $\mathcal{D}i$ is initialized as empty in Lines 2 and 3, respectively. Subsequently, Line 4 starts the main training loop, which iterates for a total of $K_{\max}$ episodes.

Algorithm 1: PG-FMADRL Algorithm

```
1  Initialize global shared parameters θ^global ;
2  Initialize local policy parameters θ_i ← θ^global ;
3  Initialize local experience buffer D_i ← ∅ ;
4  for k = 1 to K_max do
5      for i = 1 to N (in parallel) do
6          Reset environment and observe initial o_1^i;
7          for t = 1 to T do
8              Select action a_t^i to execute it (Eq. (18));
9              Receive r_t^i (Eq. (17)) and observe o_{t+1}^i;
10             Store (o_t^i, a_t^i, r_t^i, o_{t+1}^i) in D_i;
11         Compute Ω_t^i (Eq. (20)) and normalize it as R̂_t^i;
12         Update θ_i by minimizing L(θ_i) (Eq. (27))
             using policy gradient (Eq. (24));
13     if mod(k, K) == 0 then
14         for i = 1 to N (in parallel) do
15             Compute average return R̄_i^(M) over recent
                 M episodes (Eq. (28));
16             Upload θ_i and R̄_i^(M) to the aggregator;
17         Compute weights w_i (Eq. (29)) and aggregates
             parameters using Eq. (30);
18         for i = 1 to N (in parallel) do
19             Synchronize shared layers: θ_i ← θ^global;
20             for t = 1 to T_local do
21                 Sample mini-batch from D_i and update
                     θ_i (Eqs. (24) and (27));
```

Within each episode, as shown in Lines 5–12, each agent i resets its environment and observes the initial state. At each time step t, the agent selects an action a_t^i according to its current policy π_{θ_i}, receives the immediate reward r_t^i and observes the next state o_{t+1}^i. The transition tuple $(o_t^i, a_t^i, r_t^i, o_{t+1}^i)$ is then stored in the local experience buffer $\mathcal{D}_i$. After completing the episode, it computes the return Ω_t^i and normalized return $\hat{R}_t^i$, and subsequently updates θ_i by minimizing the loss function $L(\theta_i)$ using the policy gradient method.

Every K episodes, as indicated in Lines 13–21, all agents participate in the aggregation process. Each agent computes its average return $\bar{R}i^{(M)}$ over the recent window of M episodes and uploads both its shared parameters and average return to the central aggregator. The aggregator then calculates the reward-weighted aggregation coefficients w_i and aggregates the shared parameters to obtain the updated global parameters θ^{global}. After that, each agent synchronizes its shared layers and performs Tlocal steps of local fine-tuning, further updating its policy parameters to adapt to local environments.

2) Complexity Analysis: During execution, each UAV agent makes decisions at every time step by performing a forward pass through its local policy network π_{θ_i}, which is typically implemented as a DNN. For a policy network with L_{P} layers and N_n neurons in the n-th layer, the computational complexity of a single forward pass is $O(\sum_{n=1}^{L_{\text{P}}-1} N_n N_{n+1})$. The local policy update, based on the policy gradient method, involves

TABLE I
KEY PARAMETERS USED IN THE SIMULATION.

Parameter	Value or Range
Area size	1000×1000 m^2
GCS position $\mathbf{p}_0$	[500, 500, 0] m
Patrol radius r and height h	300 m, 100 m
Number of UAVs N	5
Number of frequency channels $\lvert\mathcal{F}\rvert$	5
Communication range R_c	500 m
Patrol speed v_{pat} and maximum speed $v_{\max}$	15 m/s, 20 m/s
Minimum separation $d_{\min}$	20 m
Deviation threshold δ	40 m
Attack duration τ_{atk}	15 s
Reconnaissance period τ_{recon}	5 s
MTD Execution time $\tau^{\mathbb{L}}_{\text{exec}}$, $\tau^{\mathbb{R}}_{\text{exec}}$, and $\tau^{\mathbb{F}}_{\text{exec}}$	1 s
Time steps T per episode	50
Training episodes $K_{\max}$	2×10^3
Discount factor γ	0.99
Hidden layers	[64, 64]
Learning rate (Adam)	1×10^{-3}
Batch size	128
Activation function	ReLU
Memory size of $\mathcal{D}_i$	2×10^4
Federated aggregation interval K	20
Reward averaging window M	20
Local fine-tune steps T_{local}	100
Reward coefficients $(\alpha, \beta, \xi, \eta, \zeta)$	(0.5, 0.5, 1, 2, 0.5)
Entropy coefficient μ	0.01

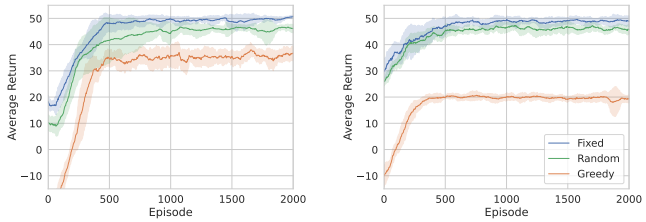
both forward and backward passes, resulting in a per-update complexity of the same order.

In each episode, the agent interacts with the environment for T time steps, leading to a per-episode complexity of $O(T\sum_{n=1}^{L_{\text{P}}-1} N_n N_{n+1})$. After every K episodes, the federated aggregation step involves computing the weighted average of the shared parameters across N agents, which has a complexity of $O(NP)$, where P is the number of shared parameters. The local fine-tuning phase after aggregation consists of T_{local} gradient updates per agent, each with complexity $O(\sum_{n=1}^{L_{\text{P}}-1} N_n N_{n+1})$. Therefore, the overall computational complexity of the proposed method is $O(NKT\sum_{n=1}^{L_{\text{P}}-1} N_n N_{n+1} + NP + NT_{\text{local}}\sum_{n=1}^{L_{\text{P}}-1} N_n N_{n+1})$.

V. PERFORMANCE EVALUATION

A. Simulation Setup

We investigate a coverage area of 1×1 Km serviced by a single GCS and multiple UAVs. The GCS is located at the center of the area, specifically at coordinates [500, 500, 0] m. Each UAV in our method is responsible for patrolling and monitoring the ground area, and all of them build a formation with a radius of 300 m and a height of 100 m. The minimum separation distance between UAVs is set to 20 m, and the deviation threshold for maintaining formation is set to 40 m. The attack duration is set to 15 s, and the reconnaissance period is set to 5 s. Each MTD mechanism has an execution time of 1 s. Other simulation parameters and their default values are summarized in Table I with reference to [33]–[35]. All the experiments were conducted on a workstation with 2.20 GHz Intel Xeon Gold 5220R, NVIDIA RTX A5000, 128 GB RAM, and Ubuntu 20.04 operating system. During the training phase, we trained the models for 2000 episodes and repeated



(a) Node attack (b) Link attack

Fig. 3. The average return of the proposed PG-FMADRL method under different DoS attack types and strategies.

the training process with random seeds ranging from 1 to 10 to ensure statistical reliability. In the testing phase, each method was evaluated over 10 episodes using random seeds from 1 to 100 to reduce the uncertainty of the environment.

To verify the effectiveness of the proposed PG-FMADRL algorithm, the following schemes are used as benchmarks and the performance is compared with that of the proposed algorithm, which are described as follows.

- **WF-MTD** [36]: It introduces rationality parameters to describe the learning capabilities of both the attacker and the defender. Then, it develops a Wright-Fisher process-based method for selecting the optimal MTD strategy.
- **RE-MTD** [13]: This method formulates the interaction between attacks and MTD deployment as a Markov decision process (MDP), and adopt a DQN algorithm to achieve a trade-off between effectiveness and overhead.
- **ID-HAM** [37]: This scheme models an MDP to describe the MTD mutation process, and designs an advantage actor-critic algorithm to learn from scanning behaviors and slow down network reconnaissance intelligently.
- **DESOLATER** [38]: It is a multi-agent deep reinforcement learning (MADRL)-based MTD technique that enables the agents to learn robustly in the presence of partial observations when defending against attacks.

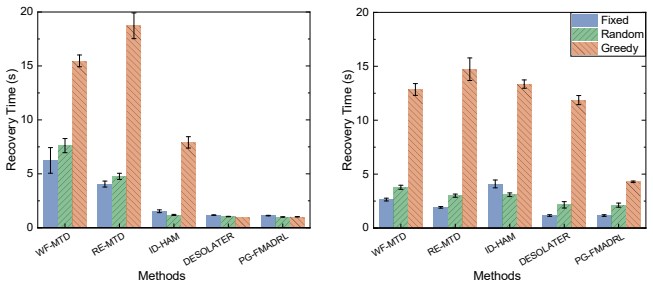
B. Convergence Analysis

To evaluate the convergence of our proposed method, we measured the average reward under fixed, random, and greedy attackers in both link and node scenarios. In Fig. 3, the x-axis indicates the number of training episodes, while the y-axis presents the average reward at each episode. In all cases, the average return increases steadily with the number of episodes and reaches convergence at approximately episode 500, which indicates that the proposed method achieves consistently good defense performance across different attack scenarios.

As shown in Fig. 3, fixed and random attackers lead to relatively high average rewards. In both link-level and node-level scenarios, the fixed attacker achieves a return close to 50, while the random attacker yields approximately 47. The reason is that fixed and random attackers select their targets once, and the applied defense mechanisms remain effective throughout the episode. In contrast, greedy attackers continuously adjust their targets in response to the defense, which results in the decrease of reward.

TABLE II
ATTACK MITIGATION RATE UNDER DIFFERENT ATTACK TYPES AND STRATEGIES

Defense Method	Node Attack						Link Attack					
	Fixed Attacker		Random Attacker		Greedy Attacker		Fixed Attacker		Random Attacker		Greedy Attacker	
	Avg.	StdDev.	Avg.	StdDev.	Avg.	StdDev.	Avg.	StdDev.	Avg.	StdDev.	Avg.	StdDev.
WF-MTD	0.8974	0.0236	0.8706	0.0129	0.7192	0.0104	0.9684	0.0025	0.9465	0.0036	0.7721	0.0105
RE-MTD	0.9413	0.0053	0.9276	0.0054	0.6535	0.0236	0.9823	0.0015	0.9615	0.0028	0.7360	0.0201
ID-HAM	0.9897	0.0024	0.9964	0.0010	0.8646	0.0106	0.9405	0.0070	0.9598	0.0033	0.7625	0.0074
DESOLATER	0.9966	0.0004	0.9992	0.0004	1.0000	0.0000	0.9969	0.0015	0.9778	0.0057	0.7910	0.0081
PG-FMADRL	0.9975	0.0004	0.9999	0.0000	0.9996	0.0002	0.9969	0.0015	0.9782	0.0039	0.9367	0.0014



(a) Node attack (b) Link attack

Fig. 4. The recovery time of the UAV swarm under different solutions for defeating DoS attacks with different types and strategies.

Besides, the figure also indicates that the proposed method performs well under both link and node attack scenarios. For fixed and random attackers, the average return remains similar across the two scenarios, suggesting that these relatively simple strategies of attackers have limited impact regardless of the attack scenarios. However, a notable performance drop is observed under greedy attackers, where the return decreases from 35 in the node scenario to 20 in the link scenario. This result suggests that link attacks are more difficult to defend, likely due to their broader influence on system communication and resource availability.

C. Effectiveness and Efficiency Analysis

To further analyze the defense effectiveness in this work, we measured the attack mitigation rate of proposed method PG-FMADRL under the previously defined attack scenarios and compare it with four representative MTD-based approaches including WF-MTD, RE-MTD, ID-HAM, DESOLATER.

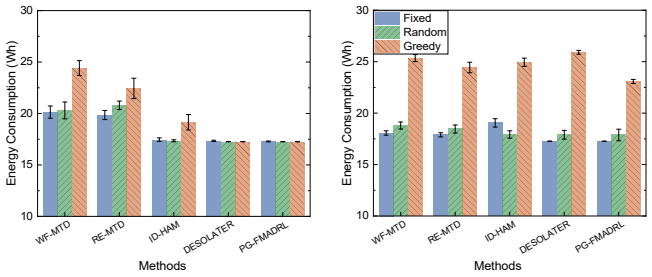
The defense attack mitigation rate is defined as the proportion of time steps within an episode during which attacks are successfully mitigated. Specifically, At each step, a heartbeat detection is performed for each UAV to compute its communication score, which is then used to assess connectivity. A UAV is considered disconnected if the score falls below a predefined threshold, indicating a defense failure at that step. The final attack mitigation rate is computed as the average proportion of UAVs that remain connected throughout the episode. The results are summarized in the Table II.

It can be seen that PG-FMADRL consistently achieves the highest or near-highest attack mitigation rates across all attack scenarios. For node attack scenario, PG-FMADRL achieves highest attack mitigation rates of 0.9975 and 0.9999 against fixed and random attackers, respectively, while also maintaining the lowest standard deviation among all compared methods. This indicates superior stability and robustness. In the link attack scenario, although the attack mitigation rates show a slight decline, the proposed method still achieves the highest values among all methods, reaching 0.9969 for fixed and 0.9782 for random attackers. These results demonstrate the method's strong and consistent defense effectiveness across different scenarios.

Notably, PG-FMADRL performs exceptionally well even against greedy attackers. In the node-level scenario, it achieves a attack mitigation rate of 0.9996, slightly below DESOLATER but 34.6% higher than RE-MTD. In the more challenging link-level scenario, PG-FMADRL reaches an attack mitigation rate of 0.9367, significantly outperforming the other methods, which achieve only 0.7910, 0.7625, 0.7360, and 0.7721, respectively. This result indicates that, compared to other methods, PG-FMADRL is more capable of continuously and adaptively responding to dynamic attackers, enabling more intelligent and effective defense decisions.

Beyond defense effectiveness, we also assess the efficiency of each method using recovery time as an indicator, which is defined as the duration required for the system to restore normal connectivity following an attack-induced disruption. Fig. 4 presents the recovery times of different defense methods.

As we can see, in the node attack scenario, WF-MTD shows the longest recovery times under fixed and random attackers, reaching 6.2 and 7.6 seconds, respectively. Under greedy attacks, RE-MTD performs the worst, with a recovery time of 18.7 seconds. In contrast, our PG-FMADRL method dramatically reduces the recovery time by approximately 94.6%, requiring only about 1 second to restore a secure state. In the link attack scenario, the worst recovery times are observed for different methods depending on the attacker type: ID-HAM for fixed attackers with 4.1 seconds, WF-MTD for random attackers with 3.7 seconds, and RE-MTD for greedy attackers with 14.7 seconds. Such prolonged recovery durations leave the UAV swarm vulnerable for extended periods, increasing both operational risk and energy consumption. Although DESOLATER performs relatively well under greedy attacks in the node scenario, its recovery time increases significantly to 11.8 seconds in the link scenario. On the other hand, PG-FMADRL achieves the shortest recovery times across all attack scenarios and attacker types, enabling rapid mitigation and minimizing disruption.



(a) Node attack (b) Link attack

Fig. 5. The energy consumption of the UAV swarm under different solutions for defeating DoS attacks with different types and strategies.

D. Energy Consumption and Defense Cost Analysis

To estimate the energy consumption of the UAV swarm when equipped with different defense methods, we adopt a simplified model introduced by Liu et al. [39] to express the power of a UAV according to its velocity as:

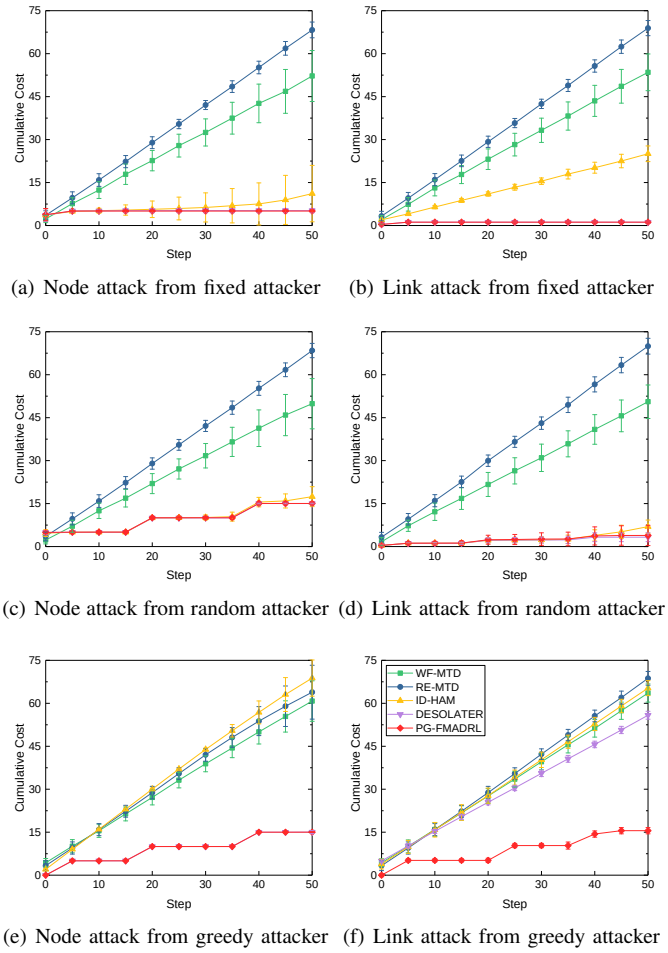
$$P(v_i) = (c_1 + c_2) \cdot (mg)^{3/2} + c_3 \cdot v_i^3 \tag{31}$$

where the coefficients are set to $c_1 = 2.8037\,(\mathrm{m/kg})^{1/2}$, $c_2 = 0.3177\,(\mathrm{m/kg})^{1/2}$, $c_3 = 0.0296\,\mathrm{kg/m}$, the mass of an UAV is set to $m = 1.283\,\mathrm{kg}$, and the gravitational acceleration is set to $g = 9.8\,\mathrm{m/s^2}$ with reference to [39].

Therefore, the total energy consumption in this section is computed as the sum of the energy consumed during the patrolling and return phases for all UAVs, as illustrated in Fig. 5. From an overall perspective, the greedy attack strategy leads to significantly higher energy consumption for the UAV swarm compared to the fixed and random strategies. This is primarily because the greedy attacker exhibits greater attack intensity and adaptability, dynamically adjusting its targets in response to defense actions. As a result, the swarm is often forced to deviate from its optimal formation, causing UAVs to travel longer distances and thus to consume more energy. Nevertheless, our proposed method consistently achieves the lowest energy consumption across all scenarios. Notably, under node attack conditions, PG-FMADRL reduces energy consumption by 29.3% compared to the WF-MTD method when facing greedy attackers, thereby effectively extending the patrol duration of the UAV swarm.

We can also observe that the energy consumption of DESOLATER is very close to our method under node attacks, which can be attributed to the fact that DESOLATER also employs a policy gradient-based approach. However, in the link attack scenario, PG-FMADRL enables more effective global agent coordination and learns superior action sequences by leveraging the federated learning framework. Consequently, under greedy attack strategies, PG-FMADRL achieves a 10.9% reduction in energy consumption compared to DESOLATER, further highlighting its effectiveness in collaborative defense and energy efficiency.

In addition to energy consumption, we also analyze the cumulative defense cost of the UAV swarm, which is defined as the total cost associated with MTD execution during the entire episode. As we can see in Fig. 6, the cumulative



(a) Node attack from fixed attacker (b) Link attack from fixed attacker

(c) Node attack from random attacker (d) Link attack from random attacker

(e) Node attack from greedy attacker (f) Link attack from greedy attacker

Fig. 6. The cumulative defense cost of the UAV swarm under different solutions for defeating DoS attacks with different types and strategies.

defense cost varies significantly across different methods and attack scenarios. It is evident that the proposed method consistently achieves the lowest cumulative cost, regardless of the attack type or attacker strategy. This demonstrates the superior efficiency of our approach in minimizing unnecessary defense actions while maintaining robust protection. Furthermore, baseline methods such as WF-MTD and RE-MTD incur significantly increasing costs, as they tend to trigger more frequent or redundant actions due to their lack of adaptive coordination. In contrast, for both node and link attacks from fixed and random attackers, PG-FMADRL maintains a nearly flat cost curve, indicating that the swarm can effectively mitigate attacks with minimal resource expenditure. Specifically, PG-FMADRL achieves a cumulative cost of 1.16 under fixed link attacks when the episode ends, having a significant reduction of 98.3% compared to RE-MTD.

When facing the more challenging greedy attacker, which dynamically adapts its strategy to maximize disruption, the advantage of PG-FMADRL becomes even more pronounced. While all baseline methods experience a sharp rise in cumulative cost, reflecting their struggle to efficiently counter adaptive threats, PG-FMADRL still maintains a much slower cost growth. This is attributed to its federated learning framework,

which enables global agent collaboration and more intelligent, context-aware defense decisions. Notably, DESOLATER and ID-HAM, which also employ multi-agent DRL, perform better than WF-MTD and RE-MTD but still lag behind PG-FMADRL, especially in the link attack scenario. This further highlights the benefit of parameter aggregation and reward-weighted policy updates in our approach, which allow for both effective adaptation and cost control.

VI. Conclusion and Future Work

In this paper, we proposed a novel FMADRL-driven MTD framework to proactively and adaptively mitigate DoS attacks in UAV swarm networks. By designing three lightweight and coordinated MTD mechanisms (e.g., leader switching, route mutation, and frequency hopping) and formulating the defense problem as a multi-agent POMDP, our approach enables each UAV to autonomously select defense actions based on local observations while benefiting from collaborative learning through reward-weighted federated aggregation. Extensive simulation results demonstrated that the proposed PG-FMADRL method significantly outperforms state-of-the-art baselines in terms of attack mitigation rate, recovery time, energy consumption, and defense cost, while maintaining robust mission continuity under various attack scenarios. Our source code can be available at https://github.com/SEU-ProactiveSecurity-Group/PG-FMADRL.

Despite these promising results, several avenues remain for future research. First, this work primarily considers attackers with fixed, random, or greedy strategies. In practical scenarios, adversaries may also leverage advanced techniques such as DRL to dynamically adapt their attack strategies in response to defense mechanisms. Therefore, a promising direction is to investigate adaptive defense strategies against intelligent attackers that can also employ DRL-based decision-making, leading to a more realistic and challenging adversarial environment. Additionally, future work will explore the scalability of the proposed framework to larger-scale UAV swarms and more complex mission scenarios, as well as the integration of additional lightweight MTD mechanisms to further enhance system robustness and security.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No. 62202097 and Grant No. 62072100, in part by China Postdoctoral Science Foundation under Grant No. 2024T170143 and Grant No. 2022M710677, and in part by Jiangsu Funding Program for Excellent Postdoctoral Talent under Grant No. 2022ZB137.

References

[1] Z. Zuo, C. Liu, Q.-L. Han, and J. Song, "Unmanned aerial vehicles: Control methods and future challenges," *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 4, pp. 601–614, 2022.

[2] S. Javed, A. Hassan, R. Ahmad, W. Ahmed, R. Ahmed, A. Saadat, and M. Guizani, "State-of-the-art and future research challenges in uav swarms," *IEEE Internet of Things Journal*, vol. 11, no. 11, pp. 19 023–19 045, 2024.

[3] P. Cao, L. Lei, S. Cai, G. Shen, X. Liu, X. Wang, L. Zhang, L. Zhou, and M. Guizani, "Computational intelligence algorithms for uav swarm networking and collaboration: A comprehensive survey and future directions," *IEEE Communications Surveys & Tutorials*, 2024.

[4] K.-Y. Tsao, T. Girdler, and V. G. Vassilakis, "A survey of cyber security threats and solutions for uav communications and flying ad-hoc networks," *Ad Hoc Networks*, vol. 133, p. 102894, 2022.

[5] H. Tang, Y. Chen, and I. Ali, "Secure distributed model predictive control for heterogeneous uav-ugv formation under dos attacks," *IEEE Transactions on Intelligent Vehicles*, 2024.

[6] H. Yang, Z. Yu, M. Fu, and Y. Zhang, "Resilient consensus control for multiple uavs with input saturation under dos attacks," *IEEE Transactions on Cybernetics*, vol. 55, no. 3, pp. 1159–1171, 2025.

[7] S. Shi, S. Wu, and B. Wei, "Neural-network-based event-triggered formation tracking for nonlinear multi-uav systems with switching topologies under dos attacks," *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 11 656–11 667, 2025.

[8] Z. Wang, Y. Li, S. Wu, Y. Zhou, L. Yang, Y. Xu, T. Zhang, and Q. Pan, "A survey on cybersecurity attacks and defenses for unmanned aerial systems," *Journal of Systems Architecture*, vol. 138, p. 102870, 2023.

[9] G. Uçtu, M. Alkan, İ. A. Doğru, and M. Dörterler, "A suggested testbed to evaluate multicast network and threat prevention performance of next generation firewalls," *Future Generation Computer Systems*, vol. 124, pp. 56–67, 2021.

[10] M. Zhong, M. Lin, C. Zhang, and Z. Xu, "A survey on graph neural networks for intrusion detection systems: methods, trends and challenges," *Computers & Security*, p. 103821, 2024.

[11] J. Tan, H. Jin, H. Zhang, Y. Zhang, D. Chang, X. Liu, and H. Zhang, "A survey: When moving target defense meets game theory," *Computer Science Review*, vol. 48, p. 100544, 2023.

[12] T. Zhang, F. Kong, D. Deng, X. Tang, X. Wu, C. Xu, L. Zhu, J. Liu, B. Ai, Z. Han *et al.*, "Moving target defense meets artificial intelligence-driven network: A comprehensive survey," *IEEE Internet of Things Journal*, vol. 12, no. 10, pp. 13 384–13 397, 2025.

[13] Y. Zhou, G. Cheng, Z. Ouyang, and Z. Chen, "Resource-efficient low-rate ddos mitigation with moving target defense in edge clouds," *IEEE Transactions on Network and Service Management*, vol. 22, no. 1, pp. 168–186, 2025.

[14] R. Fu, X. Ren, Y. Li, Y. Wu, H. Sun, and M. A. Al-Absi, "Machine-learning-based uav-assisted agricultural information security architecture and intrusion detection," *IEEE Internet of Things Journal*, vol. 10, no. 21, pp. 18 589–18 598, 2023.

[15] S. C. Hassler, U. A. Mughal, and M. Ismail, "Cyber-physical intrusion detection system for unmanned aerial vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 6, pp. 6106–6117, 2024.

[16] X. He, Q. Chen, L. Tang, W. Wang, and T. Liu, "Cgan-based collaborative intrusion detection for uav networks: A blockchain-empowered distributed federated learning approach," *IEEE Internet of Things Journal*, vol. 10, no. 1, pp. 120–132, 2023.

[17] B. B. Gupta, A. Gaurav, V. Arya, and K. T. Chui, "Machine learning-based ddos mitigation framework for unmanned aerial vehicles (uav) environment using software-defined networks (sdn)," in *GLOBECOM 2023-2023 IEEE Global Communications Conference*. IEEE, 2023, pp. 2178–2183.

[18] Y. Wu, M. Chen, and M. Chadli, "Zero-sum-game-based distributed fuzzy adaptive self-triggered control of swarm uavs under intermittent communication and dos attacks," *IEEE Transactions on Fuzzy Systems*, vol. 32, no. 9, pp. 5371–5384, 2024.

[19] H. Tang and Y. Chen, "Dynamic event-triggered distributed mpc for heterogeneous uavs-ugvs against dos attacks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 60, no. 6, pp. 7931–7944, 2024.

[20] B. Groza, L. Popa, P.-S. Murvay, Y. Elovici, and A. Shabtai, "{CANARY}-a reactive defense mechanism for controller area networks based on active {RelaYs}," in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 4259–4276.

[21] C. Wu, W. Yao, W. Pan, G. Sun, J. Liu, and L. Wu, "Secure control for cyber-physical systems under malicious attacks," *IEEE Transactions on Control of Network Systems*, vol. 9, no. 2, pp. 775–788, 2022.

[22] N. Sanoussi, K. Chetioui, G. Orhanou, and S. El Hajji, "Itc: Intrusion tolerant controller for multicontroller sdn architecture," *Computers & Security*, vol. 132, p. 103351, 2023.

[23] M. A. Ribeiro, M. S. P. Fonseca, and J. de Santi, "Detecting and mitigating ddos attacks with moving target defense approach based on automated flow classification in sdn networks," *Computers & Security*, vol. 134, p. 103462, 2023.

[24] T. Zhang, C. Xu, Y. Lian, H. Tian, J. Kang, X. Kuang, and D. Niyato, "When moving target defense meets attack prediction in digital twins: A

convolutional and hierarchical reinforcement learning approach," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 10, pp. 3293–3305, 2023.
[25] T. Zhang, C. Xu, P. Zou, H. Tian, X. Kuang, S. Yang, L. Zhong, and D. Niyato, "How to mitigate ddos intelligently in sd-iov: A moving target defense approach," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 1, pp. 1097–1106, 2023.
[26] Y. Zhou, G. Cheng, Y. Zhao, Z. Chen, and S. Jiang, "Toward proactive and efficient ddos mitigation in iiot systems: A moving target defense approach," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 4, pp. 2734–2744, 2022.
[27] L. Wang, Y. Chen, P. Wang, and Z. Yan, "Security threats and countermeasures of unmanned aerial vehicle communications," *IEEE Communications Standards Magazine*, vol. 5, no. 4, pp. 41–47, 2021.
[28] X. Gong, M. V. Basin, Z. Feng, T. Huang, and Y. Cui, "Resilient time-varying formation-tracking of multi-uav systems against composite attacks: A two-layered framework," *IEEE/CAA Journal of Automatica Sinica*, vol. 10, no. 4, pp. 969–984, 2023.
[29] Y. Yu, W. Yang, W. Ding, and J. Zhou, "Reinforcement learning solution for cyber-physical systems security against replay attacks," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 2583–2595, 2023.
[30] A. Boualouache and T. Engel, "Federated learning-based scheme for detecting passive mobile attackers in 5g vehicular edge computing," *Annals of Telecommunications*, vol. 77, no. 3, pp. 201–220, 2022.
[31] M. Huang, K. F. E. Tsang, Y. Li, L. Li, and L. Shi, "Strategic dos attack in continuous space for cyber-physical systems over wireless networks," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 8, pp. 421–432, 2022.
[32] A. H. Celdrán, P. M. S. Sánchez, J. Von Der Assen, T. Schenk, G. Bovet, G. M. Pérez, and B. Stiller, "Rl and fingerprinting to select moving target defense mechanisms for zero-day attacks in iot," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 5520–5529, 2024.
[33] H. Lei, D. Meng, H. Ran, K.-H. Park, G. Pan, and M.-S. Alouini, "Multi-uav trajectory design for fair and secure communication," *IEEE Transactions on Cognitive Communications and Networking*, 2024.
[34] R. Chai, B. Wang, R. Sun, X. Jing, and Q. Chen, "System cost function optimization-based data scheduling and flight trajectory for multi-antenna uav-assisted communication and sensing integration systems," *IEEE Transactions on Cognitive Communications and Networking*, pp. 1–1, 2024.
[35] J. Xu, H. Yao, R. Zhang, T. Mai, and M. Guizani, "Low latency and accuracy-guaranteed dnn inference for uav-assisted iot networks," *IEEE Transactions on Cognitive Communications and Networking*, 2025.
[36] J. Tan, H. Jin, H. Hu, R. Hu, H. Zhang, and H. Zhang, "Wf-mtd: Evolutionary decision method for moving target defense based on wright-fisher process," *IEEE transactions on dependable and secure computing*, vol. 20, no. 6, pp. 4719–4732, 2022.
[37] T. Zhang, C. Xu, J. Shen, X. Kuang, and L. A. Grieco, "How to disturb network reconnaissance: A moving target defense approach based on deep reinforcement learning," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 5735–5748, 2023.
[38] S. Yoon, J.-H. Cho, D. S. Kim, T. J. Moore, F. Free-Nelson, and H. Lim, "Desolater: Deep reinforcement learning-based resource allocation and moving target defense deployment framework," *IEEE Access*, vol. 9, pp. 70 700–70 714, 2021.
[39] Z. Liu, R. Sengupta, and A. Kurzhanskiy, "A power consumption model for multi-rotor small unmanned aircraft systems," in *2017 international conference on unmanned aircraft systems (ICUAS)*. IEEE, 2017, pp. 310–315.

Yuyang Zhou received the B.S. degree in Electronic Information Engineering from Nanjing University of Science and Technology in 2016 and the Ph.D. degree in Cyberspace Security from Southeast University in 2021. He is currently working as a postdoc with the School of Cyber Science and Engineering, Southeast University. His major research interests include moving target defense, DDoS mitigation, and intrusion detection. He has published in some of the topmost journals and conferences like IEEE TIFS, IEEE TII, IEEE TNSM, and ACM CCS, and is involved as reviewer and in technical program committees of several journals and conferences in the field. He is a Member of IEEE and CCF.

Guang Cheng received the B.S. degree in Traffic Engineering from Southeast University in 1994, the M.S. degree in Computer Application from Hefei University of Technology in 2000, and the Ph.D. degree in Computer Network from Southeast University in 2003. He is a Full Professor in the School of Cyber Science and Engineering, Southeast University, Nanjing, China. He has authored or coauthored seven monographs and more than 200 technical papers, including top journals and conferences like IEEE ToN, IEEE TIFS, IEEE TDSC, IEEE TII, and IEEE INFOCOM. His research interests include network security, network measurement, and traffic behavior analysis. He is a Member of IEEE and a Distinguished Member of CCF.

Kang Du received the B.S. degree in computer science and technology from Xidian University in 2023. He is currently pursuing the master's degree with the School of Cyber Science and Engineering, Southest University. His major research interests include moving target defense, DDoS mitigation, and microservices security.

Zihan Chen obtained his Ph.D. degree in Cyber Security from Southeast University in 2023 and B.S. degree in Software Engineering from Central South University in 2017. He is currently working as a postdoc with the School of Cyber Science and Engineering at Southeast University. His major research interests include cyber security, encrypted traffic classification, encrypted traffic feature engineering, and deep learning. He is a Member of IEEE, Member of CCF and works as a reviewer for multiple Journals.

Tian Qin received the B.S. degree in information and computing sciences from HoHai University (HHU) in 2020. He is currently a Doctor candidate at the School of cyber science and engineering, Southeast University, Nanjing, China. His current research interests include malicious traffic detection and federated learning.

Yuyu Zhao received the B.S. degree in software engineering from the Nanjing University of Science and Technology in 2016 and the M.S. and Ph.D. degrees in cyber security from Southeast University, Nanjing, China, in 2019 and 2023, respectively. He is a currently a Lecturer with the School of Cyber Science and Engineering, Southeast University. His research interests include in-band telemetry, blockchain, and network processors.