Federated Deep Reinforcement Learning for Joint Coverage and Connectivity in Cooperative UAV Networks for Emergency Communications

Shafkat Khan Siam, Muhammad Yeasir Arafat, Krishnendu Guha, *Member, IEEE*, Zilong Liu, *Senior Member, IEEE*, Enric Pardo, Hyundong Shin, *Fellow, IEEE,* and Md. Noor-A-Rahim*, *Senior Member, IEEE*

***Abstract*—In disaster-affected areas where conventional communication infrastructure is severely damaged, unmanned aerial vehicles (UAVs) can serve as critical aerial platforms for providing emergency coverage to ground points of interest (PoIs). However, cooperative UAV deployment for area coverage poses several challenges, including energy-constrained trajectory planning, maintenance of reliable communication connectivity for data relaying, prevention of inter-UAV collisions, and maximization of coverage efficiency. In this paper, we propose FedJCC, a federated deep reinforcement learning framework for joint coverage and connectivity optimization in cooperative UAV networks for emergency communications. The problem is formulated as a constrained Markov decision process (MDP), and a hierarchical UAV architecture is adopted, in which cluster heads maintain communication relaying while cluster members provide ground coverage. To enable collision-free navigation, a virtual force-based collision avoidance mechanism derived from the improved adaptive artificial potential field (IAAPF) method is incorporated into the framework. FedJCC is built on a dueling double deep Q-network (D3QN) with prioritized experience replay, where each UAV is trained locally and model parameters are periodically aggregated through federated averaging. In addition, a dynamic deployment adjustment mechanism is introduced to reduce the number of active UAVs while preserving coverage performance. Simulation results show that FedJCC achieves a coverage ratio of 98.8%, substantially outperforming state-of-the-art combinatorial optimization baselines. The results further demonstrate that FedJCC surpasses existing baseline algorithms in terms of packet delivery ratio and average end-to-end delay across different scenarios. Finally, the coverage-connectivity tradeoff analysis confirms that FedJCC operates closest to the ideal region in which both objectives are jointly maximized, thereby validating the effectiveness of the proposed framework for disaster response applications.**

Shafkat Khan Siam, Muhammad Yeasir Arafat, Krishnendu Guha, and Md. Noor-A-Rahim are with the School of Computer Science and IT, University College Cork, Cork, T12 K8AF, Ireland (e-mail: 125115970@umail.ucc.ie,{ marafat, KGuha, md.noorarahim}@ucc.ie).

Zilong Liu is with the School of Computer Science and Electronic Engineering, University of Essex, Colchester Campus, UK (e-mail: zilong.liu@essex.ac.uk).

Enric Pardo is with the ITIS, Luxembourg Institute of Science and Technology (LIST), Luxembourg (e-mail: enric.pardo-grino@list.lu).

Hyundong Shin is with the Department of Electronic Engineering, Kyung Hee University, Korea (e-mail: hshin@khu.ac.kr).

This work was supported in part by the Science Foundation Ireland under Grant 13/RC/2077 P2. The work of Z. Liu was supported in part by the UK Engineering and Physical Sciences Research Council under Grants EP/Y037243/1 ('TITAN/REVOL6G' and 'TITAN/LEGEND6G'), EP/X040569/1 ('HASC/RETHIN6G'), EP/X035352/1 ('DRIVE'), EP/Y000986/1 ('SORT'), and by the Royal Society under Grants IEC\NSFC\233292 and IES\R1\241212.

*Corresponding author: Md. Noor-A-Rahim (e-mail: md.noorarahim@ucc.ie).

***Index Terms*—Cooperative coverage and connectivity, deep reinforcement learning, emergency communication, federated learning, unmanned aerial vehicle (UAV).**

I. Introduction

NATURAL disasters such as earthquakes, floods, and wildfires have become increasingly frequent worldwide, causing severe disruptions to communication infrastructure and posing significant challenges to rapid post-disaster response and reconstruction efforts [1], [2]. When disasters strike, affected areas that rely primarily on terrestrial base stations for communication services often experience complete network outages due to failures in power systems or damage to base station infrastructure [3]. This communication blackout creates critical obstacles for emergency response coordination, hazard monitoring, and public safety operations [4]. In such scenarios, the rapid deployment of reliable communication and computation systems becomes crucial for effective disaster management and rescue operations.

Unmanned Aerial Vehicles (UAVs) have emerged as a key enabling technology for post-disaster emergency communication networks due to their high flexibility, low cost, and ease of deployment [5]. UAVs can be rapidly deployed as aerial base stations to provide communication coverage for points of interest (PoI) in disaster-affected regions, thereby overcoming the limitations of damaged terrestrial infrastructure. However, the communication capacity and coverage area of a single UAV are inherently limited, making it insufficient to support long-range remote communications to PoIs distributed across large disaster zones [6]. To overcome these limitations, researchers have investigated UAV swarm networks to enhance system capacity and expand communication coverage, inspired by the principles of swarm intelligence [7]. Fig. 1 illustrates a generalized disaster response scenario where multiple UAVs cooperatively provide coverage over affected areas while maintaining multi-hop connectivity to the base station.

When disasters such as earthquakes or floods occur, the timely search and rescue (SAR) of survivors becomes critically important, particularly within the first 72 golden hours [8]. In such scenarios, UAVs can capture images and videos of PoIs in disaster zones, such as collapsed buildings where survivors may be trapped, and transmit the collected information to rescue stations to support decision-making. However, efficiently scheduling UAVs to monitor these PoIs while satisfying energy constraints remains a significant optimization

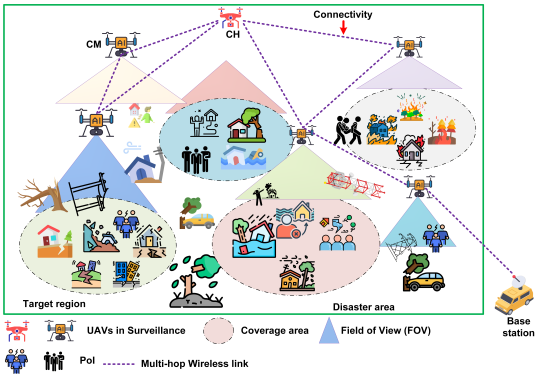


Fig. 1: Hierarchical UAV-assisted emergency communication network, where CM UAVs provide ground coverage for PoIs and CH UAVs maintain multi-hop wireless links to the base station.

challenge [9]. To address the complexity and uncertainty of disaster environments, deep reinforcement learning (DRL) has been increasingly adopted for UAV deployment, trajectory optimization, and task management [4], [10]. DRL combines the powerful feature representation capability of deep learning with the strong sequential decision-making ability of reinforcement learning, making it well suited for UAV path planning problems [3], [11]. Nevertheless, the effectiveness of DRL is often limited by its dependence on large amounts of training data, long convergence times, and the difficulty of maintaining a proper balance between exploration and exploitation, particularly in dynamic disaster scenarios [4], [12].

Multi-agent deep reinforcement learning (MADRL) has been developed to improve coordination among multiple UAVs [5], [13]. However, its performance often degrades as the number of agents increases, due to slow convergence and the growing computational burden associated with multi-agent training [4], [14]. In UAV-assisted SAR operations, UAVs collect disaster-area information and may execute computationally intensive tasks, such as real-time artificial intelligence (AI) inference and feature extraction [15]. While operating over disaster zones, UAVs continuously gather large volumes of image and video data, and real-time AI applications can assist in identifying survivors, hazardous areas, and other critical targets. However, because of their limited onboard computing resources and battery capacity, executing these tasks entirely on the UAVs can significantly degrade performance. Task offloading to edge computing servers has been proposed as a potential solution. Nevertheless, in severely affected disaster areas, fixed ground base-station (BS) infrastructure may be unavailable, making timely computational support difficult to obtain. In such dynamic and resource-constrained environments, machine learning (ML) approaches, particularly those capable of adaptive decision-making, offer a promising alternative to conventional optimization methods [15], [16].

Recent studies have increasingly adopted DRL algorithms for the adaptive management of task offloading and resource allocation in UAV networks [15], [17]. However, ensuring security and privacy in emergency communication networks remains a significant challenge. To protect sensitive information at distributed nodes, federated learning (FL) has been introduced, where only locally updated model parameters are shared instead of raw data [18], [19]. Nevertheless, the heterogeneous and highly dynamic nature of disaster environments makes it difficult for conventional FL global models to generalize effectively to previously unseen scenarios [15].

FL is a distributed learning framework in which multiple agents collaboratively train a shared model without sharing raw local data [20]. This paradigm has been applied in UAV networks to support tasks such as resource allocation, cooperative learning, and swarm coordination [21], [22]. Despite these advantages, FL faces several challenges in disaster recovery scenarios. In particular, the periodic aggregation of local models required by FL can become impractical in UAV swarms due to intermittent wireless connectivity and rapidly changing network topologies [23]. As a result, model synchronization may become delayed, limiting the suitability of conventional FL frameworks for time-critical decision-making without further enhancements.

Moreover, post-disaster emergency operations are inherently time-sensitive. Emerging technologies such as telemedicine surgery and autonomous rescue robots rely on reliable and low-latency communication networks to function effectively. Ensuring that sensor data are delivered to destination nodes completely and within strict time constraints reflects the deterministic communication requirements of time-sensitive flows [24]. At the same time, energy consumption directly affects UAV flight endurance. In SAR missions where every minute is critical for saving lives, minimizing energy consumption while maintaining effective coverage and connectivity becomes a fundamental design objective.

A. *Contributions of this Work*

To address these challenges, this paper proposes FedJCC, a novel federated deep reinforcement learning (FDRL) framework for joint coverage and connectivity optimization in cooperative UAV networks for emergency communication scenarios. Although conventional FL frameworks face practical challenges in disaster recovery scenarios, the FL paradigm remains well suited to this setting when appropriately tailored. By sharing only model parameters instead of raw sensory data, UAVs can collaboratively learn effective coverage and connectivity strategies while preserving the privacy of sensitive disaster-zone information and reducing communication overhead over bandwidth-constrained aerial links. At the same time, DRL enables UAVs to learn adaptive sequential decision-making policies that respond to the dynamic and unpredictable conditions of disaster environments, where conventional optimization approaches relying on complete prior knowledge and static assumptions are often inadequate. By integrating FL with DRL, the proposed framework supports privacy-preserving distributed training while enabling adaptive decision-making in complex and time-varying environments. The proposed framework aims to optimize UAV deployment and active swarm size while maximizing coverage performance, maintaining network connectivity, and improving energy efficiency. The proposed framework differs from conventional supervised federated learning, in which clients train on pre-existing static

datasets stored locally. In emergency UAV deployment, no such dataset exists prior to the mission. Instead, each UAV constructs its local dataset online in the form of a replay buffer that accumulates transition tuples obtained from interaction with the disaster environment. Federated aggregation is then applied to the parameters of the value network learned from these local experiences, which allows UAVs to share policy knowledge without exchanging raw sensory data, local PoI observations, or mission-specific information. FedJCC is therefore best interpreted as online federated reinforcement learning with periodic and asynchronous parameter aggregation rather than as conventional supervised federated learning.

The main contributions of this work are summarized as follows:

- First, we formulate cooperative UAV coverage and connectivity in emergency communication networks as a partially observable online federated reinforcement learning problem, in which each UAV relies on local sensing, neighbor information, and a CH-assisted belief map rather than global PoI knowledge.
- Second, we propose FedJCC, a communication-aware online FDRL framework based on D3QN with prioritized experience replay, in which UAVs learn from local replay buffers and periodically share model parameters without exchanging raw sensing data.
- Third, we introduce an asynchronous staleness-aware aggregation mechanism that supports intermittent UAV-to-BS connectivity, delayed model uploads, and network fragmentation by exponentially down-weighting stale updates.
- Fourth, we develop a failure-resilient hierarchical CH/CM architecture with dynamic CH reassignment, relay recovery, and deployment adjustment to maintain coverage and connectivity under UAV failure, low battery, and link outage.
- Fifth, we integrate a virtual-force safety shield that projects unsafe movement actions into a collision-free feasible region while preserving the learned policy's coverage and connectivity objectives.
- Sixth, we conduct extensive experiments, including ablation studies, learning-based baselines, multi-seed evaluations with variance bands, reward-weight trade-off discussion, and scalability analysis, to validate the effectiveness of FedJCC.

B. Outline of This Article

The remainder of this paper is organized as follows. Section II reviews related works on UAV deployment and reinforcement learning approaches for emergency communications. Section III presents the system model and problem formulation. Section IV describes the proposed FedJCC framework. Section V presents the simulation results and performance analysis. Finally, Section VI concludes the paper.

II. Related Works

The deployment of UAVs for communication coverage and connectivity optimization has received significant attention in recent years due to their rapid deployment, high mobility, and ability to provide line-of-sight (LoS) links to ground users. To address challenges such as multi-UAV coordination, trajectory design, and resource allocation, prior studies have proposed both heuristic and ML and DRL-based solutions. This section reviews the related literature in these two categories.

A. Heuristic-based Methods

Early studies on multi-UAV deployment and trajectory planning primarily relied on heuristic and optimization-based approaches. Prasad and Ramkumar [25] proposed a three-dimensional (3D) deployment and trajectory planning method for relay-based UAV-assisted cooperative communication in emergency scenarios using Dijkstra's algorithm. Their approach focused on establishing communication links between isolated ground users and rescue coordination centers through UAV relays. Gao et al. [26] proposed a heuristic approximation algorithm for UAV task allocation in multi-UAV-enabled Internet of Things (IoT) networks, considering task urgency and network cost constraints to minimize task completion time. Xu et al. [8] developed an approximation algorithm for scheduling energy-constrained homogeneous UAVs to monitor the maximum number of points of interest in disaster areas. In subsequent work, they further investigated heterogeneous UAV scheduling by considering different energy capacities and monitoring capabilities to maximize the total monitoring reward.

The vehicle routing problem formulation has also been applied to UAV scheduling. Ning et al. [27] studied the problem of designing UAV flight trajectories to serve community users through task offloading and proposed two heuristic algorithms to maximize system throughput. Although these heuristic approaches provide computationally efficient solutions with theoretical performance guarantees, they often lack the ability to adapt to highly dynamic environments and may fail to capture the complex spatiotemporal dependencies inherent in cooperative UAV systems.

B. ML and DRL based Methods

Learning-based and optimization-based techniques have been employed to address the limitations of purely heuristic approaches in UAV network optimization. Guan et al. [28] proposed a collaborative UAV trajectory design method for disaster-area emergency communications by combining an enhanced K-means algorithm with multi-agent proximal policy optimization (MAPPO). Sun et al. [29] developed an approximate convex optimization method that jointly optimizes UAV position, resource allocation, and task partitioning. Liu and Zheng [30] proposed a trajectory optimization algorithm based on block coordinate descent while considering message delay constraints. Although these approaches provide tractable and effective solutions, they often rely on simplified assumptions such as static or slowly varying environments, and may not fully capture the complex spatiotemporal dynamics of cooperative multi-UAV systems. Recent studies have further explored learning-based optimization for multi-UAV and IRS-assisted networks. In Khan et al. [31], an AI-empowered multi-UAV and IRS collaboration framework was developed for spectrum and energy optimization in B5G networks. In Ahmad

et al. [32], a learning-based power minimization framework was proposed for secure multiple UAV-aided MIMO networks. These works demonstrate the growing role of learning-assisted UAV resource optimization. However, they primarily target spectrum, energy, or secure-transmission objectives, whereas the present work focuses on emergency communication scenarios in which coverage, multi-hop connectivity, partial observability, collision avoidance, and dynamic UAV deployment must be jointly considered under unreliable connectivity.

DRL has emerged as a powerful paradigm for sequential decision-making in multi-UAV networks. Chang et al. [33] investigated UAV trajectory design and resource allocation using DRL, while Qin et al. [11] employed a centralized soft actor-critic (SAC) algorithm for joint communication and sensing optimization. Cao et al. [34] proposed an iterative two-stage multi-agent SAC framework for UAV association and dynamic deployment in emergency communication scenarios. Liu et al. [35] developed a DRL-based approach for energy-efficient UAV control while ensuring fair coverage among users. MADRL has further been explored to enable decentralized coordination among multiple UAVs. For example, Fu et al. [36] proposed a dense MADRL algorithm for UAV-assisted vehicular networks, Cheng et al. [37] integrated pheromone tracking with multi-agent deep deterministic policy gradient (MADDPG) for disaster-area coverage, and Huang et al. [38] applied the asynchronous advantage actor-critic (A3C) algorithm to UAV path planning in IoT systems.

Recent advances have explored more sophisticated architectures and learning paradigms for multi-UAV coordination. Chen et al. [3] proposed a spatiotemporal-aware DRL method that employs a Transformer as the Q-value function approximator, capturing spatial dependencies among UAVs and tasks as well as temporal dependencies through multi-head attention mechanisms. Hazarika et al. [4] proposed a framework that integrates generative artificial intelligence (GenAI) with graph neural networks for dynamic hover-point generation, combined with a multi-agent graph reinforcement learning (MAGRL) framework enhanced by graph attention to improve UAV coordination in disaster scenarios. Graph reinforcement learning (GRL) has also shown strong potential for multi-agent coordination by exploiting the communication network structure to support real-time state sharing and task adaptation [39], [40].

FL has gained increasing attention as a privacy-preserving distributed learning paradigm for UAV networks. Li et al. [18] employed FL to optimize UAV positioning and resource allocation while maximizing traffic offloading. Tang et al. [41] proposed a blockchain-based trusted traffic offloading framework for space-air-ground integrated networks using FDRL. Singh et al. [42] developed a DRL-enabled FL approach for vehicular caching management, while Qin et al. [15] proposed a blockchain-assisted meta-FL framework that combines FL with meta-learning to improve model adaptability across diverse disaster scenarios. Despite these advances, most existing studies address coverage and connectivity separately, and only limited attention has been given to their joint optimization through FDRL under the practical constraints of cooperative multi-UAV networks. To fill this gap, we propose a unified framework that jointly optimizes coverage and connectivity through federated model aggregation in cooperative multi-UAV networks.

III. Preliminaries and System Model

A. Network Architecture

We consider an emergency communication network deployed in a disaster-affected area where conventional communication infrastructure has been severely damaged as shown in Fig. 1. The network consists of three main components: a BS, a UAV swarm network, and a set of ground PoIs. Cluster head (CH) UAVs operate at higher altitude and provide relay connectivity toward the BS, while cluster member (CM) UAVs operate at lower altitude and provide ground coverage for PoIs. The BS serves as the central coordination and model aggregation node for FL. The UAV swarm is composed of N UAVs, denoted by $\mathcal{U} = \{u_1, u_2, \ldots, u_N\}$, which provide communication coverage over the affected area. The PoIs are represented by a set of M ground locations, denoted by $\mathcal{P} = \{p_1, p_2, \ldots, p_M\}$, corresponding to survivors, emergency facilities, or other critical communication nodes. The system operates in a time-slotted manner, where the total mission duration T is divided into discrete time slots of duration τ. Accordingly, the set of time slots is defined as $\{1, 2, \ldots, t, \ldots, T\}$.

B. UAV Mobility Model

At time slot t, the 3D position of UAV u_i is denoted by $\mathbf{l}_i(t) = [x_i(t), y_i(t), z_i(t)]$, where $x_i(t)$ and $y_i(t)$ represent the horizontal coordinates, and $z_i(t)$ denotes the altitude. The trajectory of each UAV is controlled by its horizontal flight direction, elevation angle, and travel distance in each time slot [15]. The position update is given by

$$x_i(t+1) = x_i(t) + d_i(t)\cos(\phi_i(t))\cos(\theta_i(t)) \quad (1)$$

$$y_i(t+1) = y_i(t) + d_i(t)\cos(\phi_i(t))\sin(\theta_i(t)) \quad (2)$$

$$z_i(t+1) = z_i(t) + d_i(t)\sin(\phi_i(t)), \quad (3)$$

where $\theta_i(t)$ denotes the horizontal flight direction, $\phi_i(t)$ denotes the elevation angle, and $d_i(t)$ represents the travel distance during time slot t. The movement distance, horizontal direction, and elevation angle are computed as, $d_i(t) = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$, $\theta_i(t) = \mathrm{atan2}(\Delta y, \Delta x)$, and $\phi_i(t) = \mathrm{atan2}\Big(\Delta z, \sqrt{\Delta x^2 + \Delta y^2}\Big)$. Due to the maximum flight-speed constraint of the UAVs, the trajectory variation is limited by $\|\mathbf{l}_i(t+1) - \mathbf{l}_i(t)\| \le v_{\max}\tau$, where $v_{\max}$ denotes the maximum flight speed, and τ is the duration of each time slot [15]. To avoid inter-UAV collisions, the following separation constraint is imposed: $d_{i,i'}(t) = \|\mathbf{l}_i(t) - \mathbf{l}_{i'}(t)\| \ge S_{\min}, \forall u_{i'} \in \mathcal{U}, i' \ne i$, where $S_{\min}$ represents the minimum safe distance between any two UAVs. Throughout the paper, τ denotes the environment and action time slot at which UAVs observe their local state and select control actions. It does not denote the interval at which federated aggregation is performed. Federated aggregation is performed every $T_{\mathrm{agg}} = K\tau$ slots, with $K \gg 1$ in the simulation study, so that the multi-hop relay latency associated with model uploads is comfortably absorbed within the aggregation interval.

To ensure collision-free navigation during cooperative coverage operations, we adopt a virtual force-based mechanism derived from the IAAPF method [5], [43]. Unlike conventional artificial potential field (APF) approaches, which include both attractive and repulsive components, the proposed framework considers only inter-UAV repulsive forces, since trajectory planning is handled separately by the RL agent. The repulsive potential field between UAV u_i and UAV u_j is defined as

$$U^*_{\text{rep}}(q_{ij}) = \begin{cases} \frac{1}{2}k_{\text{rep}}\left(\frac{1}{d_{i,j}(t)} - \frac{1}{d_0}\right)^2, & \text{if } d_{i,j}(t) \le d_0 \\ 0, & \text{if } d_{i,j}(t) > d_0 \end{cases} \quad (4)$$

where $d_{i,j}(t) = \|\mathbf{l}_i(t) - \mathbf{l}_j(t)\|$ denotes the Euclidean distance between UAVs, k_{rep} is the repulsion gain coefficient, and d_0 represents the influence range of the repulsive field. The repulsive force exerted on UAV u_i by UAV u_j is given by $\mathbf{F}_{\text{rep}}(i,j,t) = -\nabla U^*_{\text{rep}}(q_{ij})$, which yields

$$\mathbf{F}_{\text{rep}}(i,j,t) = \begin{cases} k_{\text{rep}}\left(\frac{1}{d_{i,j}(t)} - \frac{1}{d_0}\right)\frac{\hat{\mathbf{d}}_{ij}(t)}{d^2_{i,j}(t)}, & d_{i,j}(t) \le d_0 \\ 0, & d_{i,j}(t) > d_0 \end{cases} \quad (5)$$

where $\hat{\mathbf{d}}_{ij}(t) = \frac{\mathbf{l}_i(t) - \mathbf{l}_j(t)}{d_{i,j}(t)}$ is the unit vector pointing from UAV u_j to UAV u_i. Accordingly, the total virtual repulsive force acting on UAV u_i from all other UAVs is expressed as

$$\mathbf{F}_{\text{vf},i}(t) = \sum_{j \in \mathcal{U},\, j \neq i} \mathbf{F}_{\text{rep}}(i,j,t). \quad (6)$$

This virtual force is then used to adjust the intended UAV motion, such that the collision-aware next position is given by

$$\mathbf{l}'_i(t+1) = \mathbf{l}_i(t+1) + \alpha_{\text{vf}}\mathbf{F}_{\text{vf},i}(t), \quad (7)$$

where α_{vf} is a scaling factor controlling the influence of the virtual force, and $\mathbf{l}_i(t+1)$ denotes the intended next position determined by the RL agent.

C. Communication Model

Communication between UAVs and ground PoIs follows an air-to-ground (ATG) channel model that considers both LoS and non-line-of-sight (NLoS) propagation conditions [4]. The probability of establishing a LoS connection depends on the elevation angle $\theta_{m,n}$ between UAV m and PoI n, and is given by

$$P^{\text{LoS}}_{m,n} = \left(1 + a\exp\left(-b\left(\frac{180}{\pi}\theta_{m,n} - a\right)\right)\right)^{-1}, \quad (8)$$

where a and b are environment-dependent parameters determined by the propagation environment (e.g., urban, suburban, or rural) [4]. The probability of NLoS transmission is therefore

$$P^{\text{NLoS}}_{m,n} = 1 - P^{\text{LoS}}_{m,n}. \quad (9)$$

Accordingly, the composite path loss is expressed as

$$PL_{m,n} = L^{\text{LoS}}_{m,n}P^{\text{LoS}}_{m,n} + L^{\text{NLoS}}_{m,n}P^{\text{NLoS}}_{m,n}, \quad (10)$$

where $L^{\text{LoS}}_{m,n}$ and $L^{\text{NLoS}}_{m,n}$ denote the path losses under LoS and NLoS conditions, respectively. The channel gain between UAV u_i and PoI k is given by

$$G_{i,k}(t) = \frac{P^{\text{LoS}}_{i,k}}{L^{\text{LoS}}_{i,k}} + \frac{1 - P^{\text{LoS}}_{i,k}}{L^{\text{NLoS}}_{i,k}}. \quad (11)$$

For UAV-to-UAV or UAV-to-ground communication with clear LoS conditions, the channel gain can be approximated as

$$h_{i,j}(t) = \frac{g_0}{\|\mathbf{l}_i(t) - \mathbf{l}_j(t)\|^2}, \quad (12)$$

where g_0 denotes the reference channel gain at a unit distance [15]. The SINR for PoI k served by UAV u_i is computed as

$$\gamma_{i,k}(t) = \frac{P_t G_{i,k}(t)}{I_{i,k}(t) + \sigma^2}, \quad (13)$$

where P_t denotes the transmission power, $I_{i,k}(t)$ represents the interference from other UAVs, and σ^2 is the noise power. The achievable data rate follows the Shannon capacity formula

$$R_{i,k}(t) = B\log_2\left(1 + \gamma_{i,k}(t)\right), \quad (14)$$

where B denotes the allocated bandwidth. Accordingly, the amount of data transmitted by UAV u_i during time slot t is

$$W^o_i(t) = \min\left\{Q^o_i(t), \tau R_{i,k}(t)\right\}. \quad (15)$$

Unless otherwise stated, small-scale fading is not explicitly modeled in the main simulation. Instead, the dominant air-to-ground propagation effect is captured through the probabilistic LoS/NLoS path-loss model in Eqs. (8)–(11). Interference is computed from simultaneously active UAV transmissions sharing the same channel, while thermal noise is represented by the noise power σ^2 in Eq. (13).This abstraction allows the analysis to focus on mission-level coverage, relay connectivity, and energy-aware deployment dynamics. Robustness to fast fading and wind-induced trajectory perturbations will be considered in future hardware-in-the-loop validation.

D. Energy Consumption Model

The total energy consumption of UAV u_i at time slot t consists of propulsion, communication, and computation components [4], [15]. The propulsion energy consumption is given by

$$E^{\text{prop}}_i(t) = \frac{M_i^2}{\sqrt{2\rho A_r}} \cdot \frac{\tau}{\sqrt{v_i(t)^2 + \sqrt{v_i(t)^4 + \left(\frac{M_i}{2\rho A_r}\right)^2}}}, \quad (16)$$

where M_i denotes the UAV mass, ρ is the air density, A_r represents the rotor disc area, and $v_i(t) = \frac{\|\mathbf{l}_i(t) - \mathbf{l}_i(t-1)\|}{\tau}$ denotes the average UAV speed. The air-resistance energy consumption is expressed as, $E^{\text{air}}_i(t) = \frac{1}{8}C_{\text{air}}\rho A_r[v_i(t)]^3\tau$ where C_{air} denotes the aerodynamic drag coefficient [15].

The total flight energy consumption is therefore, $E^{\text{fly}}_i(t) = E^{\text{prop}}_i(t) + E^{\text{air}}_i(t)$ In addition, hovering and movement energy are modeled as, $E_{mh,i}(t) = P_{\text{hov}}T_{\text{hov},i}(t) + P_{\text{mov}}\|\mathbf{l}_i(t + \Delta t) - \mathbf{l}_i(t)\|$, where P_{hov} and P_{mov} denote the hovering power and movement power per unit distance, respectively. The communication energy required for data transmission is, $E^{\text{tra}}_i(t) = x_{i,j}(t)P^{\text{up}}_i \min\left\{\frac{Q^o_i(t)}{R_{i,j}(t)}, \tau\right\}$ where P^{up}_i denotes the transmission power and $x_{i,j}(t) \in \{0,1\}$ represents the binary offloading decision. The local computation energy on UAV u_i is given by

$$E^l_i(t) = \kappa[f_i(t)]^3 \min\left\{\frac{\phi_i Q^l_i(t)}{f_i(t)}, \tau\right\}, \quad (17)$$

TABLE I: Summary of Notations

Symbol	Description
$\mathcal{U}, \mathcal{P}$	Sets of UAVs and PoIs
N, M	Number of UAVs and PoIs
T, τ	Total mission duration and time-slot length
$\mathbf{l}_i(t)$	3D position of UAV u_i at time t
$\theta_i(t), d_i(t)$	UAV flight direction and movement distance
$v_{\max}$	Maximum UAV flight speed
$S_{\min}$	Minimum safe separation distance between UAVs
a, b	Environment-dependent LoS channel parameters
A_r	Rotor disc area of UAV
$P_{i,k}^{\mathrm{LoS}}$	LoS probability between UAV i and PoI k
$G_{i,k}(t)$	Channel gain between UAV u_i and PoI k
g_0	Reference channel gain at unit distance
$\gamma_{i,k}(t)$	Signal-to-interference-plus-noise ratio (SINR)
$R_{i,k}(t)$	Achievable transmission data rate
$E_i^{\mathrm{fly}}(t)$	Flight energy consumption
$E_i^{\mathrm{tra}}(t)$	Communication energy consumption
$E_i^{l}(t)$	Local computation energy consumption
$E_i^{\mathrm{total}}(t)$	Total UAV energy consumption
κ	Effective switched capacitance coefficient
$f_i(t)$	CPU frequency allocation
R, R_s	Coverage radius and sensing range
$P_d(k,i,t)$	Detection probability of PoI k by UAV i
$O_{k,i}(t)$	Obstacle impact severity factor
$o_{i,i'}(t)$	Coverage overlap between UAVs i and i'
$U_{\mathrm{rep}}^*(q_{ij})$	Repulsive potential field between UAVs i and j
$\mathbf{F}_{\mathrm{rep}}(i,j,t)$	Repulsive force exerted by UAV j on UAV i
$\mathbf{F}_{\mathrm{vf},i}(t)$	Total virtual force acting on UAV i
k_{rep}	Repulsion gain coefficient
d_0	Repulsive field influence distance
α_{vf}	Virtual-force scaling factor
y_i	UAV deployment indicator variable
$a_{k,i}$	Coverage assignment between PoI k and UAV i
$f_{i,j}$	Connectivity link indicator between UAVs i and j
$\beta_1, \beta_2, \beta_3$	Weighting coefficients in the objective function
$Q_i^l(t), Q_i^o(t)$	Local and offloading task queue backlogs
$A_i^l(t), A_i^o(t)$	Task allocation ratios
$x_{i,j}(t)$	Binary task offloading decision variable
$r_{\mathrm{cov}}(t)$	Coverage reward component
$r_{\mathrm{con}}(t)$	Connectivity reward component
$r_{\mathrm{bat}}(t)$	Battery management reward component
η_R	QoS violation penalty coefficient

where $\kappa \geq 0$ is the effective switched capacitance coefficient, $f_i(t)$ denotes the CPU frequency, and ϕ_i represents the number of CPU cycles required per bit. Accordingly, the total energy consumption of UAV u_i at time slot t is

$$E_i^{\mathrm{total}}(t) = E_i^{\mathrm{fly}}(t) + E_i^{\mathrm{tra}}(t) + E_i^{l}(t). \tag{18}$$

Environmental disturbances such as wind gusts may increase propulsion energy and perturb the nominal UAV position updates in Eqs. (1)–(3). In this work, wind-induced effects are not explicitly simulated; the model focuses on nominal propulsion, communication, and computation energy, while the battery-aware reward and virtual-force safety layer limit excessive motion and unsafe proximity.

E. Coverage Model

The ground coverage region of each UAV is assumed to be circular with radius R. The detection probability of PoI k by UAV u_i at time t is modeled as

$$P_d(k,i,t) = P_{i,k}^{\mathrm{LoS}} \exp\left(-\frac{\|\mathbf{l}_k - \mathbf{l}_i(t)\|}{R_s}\right), \tag{19}$$

where R_s denotes the probabilistic sensing range. Considering environmental obstacles, the detection probability becomes

$$P_d^{\mathrm{obs}}(k,i,t) = P_d(k,i,t)\left(1 - O_{k,i}(t)\right), \tag{20}$$

where $O_{k,i}(t) \in [0,1]$ represents the obstacle impact severity. To maintain efficient coverage, the overlap between UAV coverage areas should be minimized. The overlap area between UAV u_i and $u_{i'}$ is given by

$$o_{i,i'}(t) = \left[2R^2 \arccos\left(\frac{d_{i,i'}(t)}{2R}\right) - \frac{d_{i,i'}(t)}{2}\sqrt{4R^2 - d_{i,i'}^2(t)}\right] \times \mathbb{I}[d_{i,i'}(t) < 2R], \tag{21}$$

where $\mathbb{I}[\cdot]$ denotes the indicator function.

The obstacle severity factor $O_{k,i}(t) \in [0,1]$ represents the impact of blockage between UAV u_i and PoI p_k, where 0 indicates no blockage and 1 indicates complete obstruction. In practical UAV deployment, $O_{k,i}(t)$ can be estimated from onboard sensing or from prior map information. Possible sources include onboard camera or LiDAR-based obstacle detection, semantic segmentation of disaster imagery, pre-disaster or post-disaster topographic and structural maps, and signal-quality degradation used as a proxy for blockage severity. In the present simulation study, $O_{k,i}(t)$ is modelled as a normalized severity value drawn from a parameterized distribution that reflects the obstacle density of the environment.

F. Problem Formulation

The objective of the cooperative multi-UAV coverage problem is to maximize coverage efficiency while minimizing the number of deployed UAVs and maintaining reliable network connectivity. This problem is formulated as a constrained optimization problem that minimizes the total operational cost while satisfying coverage, connectivity, energy, mobility, and collision avoidance constraints. The optimization problem is formulated as

$$\min_{\mathbf{y},\mathbf{q},\mathbf{a},\mathbf{f}} \quad K = \beta_1 \sum_{i=1}^{N} y_i + \beta_2 \sum_{k=1}^{M}\sum_{i=1}^{N} a_{k,i}\|\mathbf{p}_k - \mathbf{q}_i\| + \beta_3 \sum_{i=1}^{N}\sum_{j=1}^{N} f_{i,j}\|\mathbf{q}_i - \mathbf{q}_j\| \tag{22a}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} a_{k,i} \geq 1, \quad \forall k \in \mathcal{P} \tag{22b}$$

$$\sum_{j=1}^{N} f_{i,j} \geq 1, \quad \forall i \in \mathcal{U}_{\text{active}} \tag{22c}$$

$$\sum_{t=1}^{T} E_i^{\mathrm{total}}(t) \leq E_{\max}, \quad \forall u_i \in \mathcal{U} \tag{22d}$$

$$x_i(t) \in [0, x_{\max}],\ y_i(t) \in [0, y_{\max}], \quad z_i(t) \in [z_{\min}, z_{\max}], \quad \forall u_i \in \mathcal{U} \tag{22e}$$

$$\|\mathbf{l}_i(t+1) - \mathbf{l}_i(t)\| \leq v_{\max}\tau, \quad \forall u_i \in \mathcal{U} \tag{22f}$$

$$d_{i,j}(t) \geq S_{\min}, \quad \forall i,j \in \mathcal{U},\ i \neq j \tag{22g}$$

$$y_i \in \{0,1\},\ a_{k,i} \in \{0,1\},\ f_{i,j} \in \{0,1\} \tag{22h}$$

The decision variables include y_i (binary UAV deployment indicator), $\mathbf{q}_i = [x_i, y_i, z_i]$ (3D UAV position), $a_{k,i}$ (coverage assignment between PoI k and UAV i), and $f_{i,j}$ (connectivity link indicator between UAVs i and j). The weighting coefficients β_1, β_2, and β_3 control the trade-offs between minimizing the number of deployed UAVs, reducing

the distance between UAVs and PoIs to improve coverage quality, and maintaining reliable inter-UAV connectivity.

Constraint (22b) ensures that each PoI is covered by at least one UAV. Constraint (22c) guarantees that each active UAV maintains at least one communication link, either directly with the base station or through multi-hop relaying via other UAVs, thereby preserving network connectivity. Constraint (22d) ensures that the cumulative energy consumption of each UAV does not exceed its maximum battery capacity $E_{\max}$. Constraint (22e) restricts UAV movements within the operational area. Constraint (22f) limits UAV displacement between consecutive time slots according to the maximum flight speed $v_{\max}$, ensuring physically feasible trajectories. Constraint (22g) enforces a minimum separation distance $S_{\min}$ between UAVs to avoid collisions, which is implemented through the virtual force mechanism described in Section III-B. Finally, constraint (22h) defines the binary nature of the deployment, coverage assignment, and connectivity variables.

G. Partially Observable MDP Formulation

The cooperative multi-UAV coverage problem is formulated as a partially observable online RL problem. The global PoI set $\mathcal{P}$ defines the disaster environment, but it is not assumed to be fully known by every UAV during execution. Instead, each UAV makes decisions based on its local observation, which is obtained from onboard sensing, neighboring UAV information, and limited information exchanged through the CH-assisted relay structure. Therefore, the local observation of UAV u_i at time slot t is defined as

$$s_i(t) = \left[\mathbf{l}_i(t), \mathcal{P}_i^{obs}(t), Q_i^l(t), Q_i^o(t), E_i^{rem}(t), \mathbf{N}_i(t)\right], \quad (23)$$

where the locally observable PoI set of u_i at time t, denoted by $\mathcal{P}_i^{obs}(t) = \{\mathbf{p}_k \in \mathcal{P} : \|\mathbf{l}_i(t) - \mathbf{p}_k\| \leq R_s\}$, $\mathbf{l}_i(t)$ denotes the 3D position of UAV u_i, $Q_i^l(t)$ and $Q_i^o(t)$ denote the local and offloading task queue backlogs, $E_i^{rem}(t)$ is the remaining battery energy, and $\mathbf{N}_i(t)$ denotes the set of neighboring UAVs within communication range. The action space of UAV u_i is defined as

$$a_i(t) = \left[\theta_i(t), d_i(t), A_i^l(t), A_i^o(t), f_i(t), x_{i,1}(t), \ldots, x_{i,M}(t)\right], \quad (24)$$

where $\theta_i(t)$ and $d_i(t)$ denote the UAV flight direction and movement distance, respectively. The variables $A_i^l(t)$ and $A_i^o(t)$ represent the task allocation ratios for local processing and task offloading, $f_i(t)$ denotes the CPU frequency allocation, and $x_{i,j}(t) \in \{0, 1\}$ indicates whether tasks are offloaded to server j. The reward function is designed to encourage efficient coverage and connectivity while penalizing undesirable behaviors. It is defined as

$$r_i(t) = r_{\text{cov}}(t) + r_{\text{con}}(t) + r_{\text{bat}}(t) - \eta_R \mathbb{I}[\text{QoS violation}], \quad (25)$$

where $r_{\text{cov}}(t)$ rewards successful PoI coverage, $r_{\text{con}}(t)$ incentivizes maintaining network connectivity, and $r_{\text{bat}}(t)$ reflects battery management efficiency. The parameter η_R denotes the penalty coefficient associated with QoS violations, and $\mathbb{I}[\cdot]$ represents the indicator function, i.e., $\mathbb{I}[\text{QoS violation}] \in \{0, 1\}$. The definitions of the state, action, and reward components are summarized in Table I.

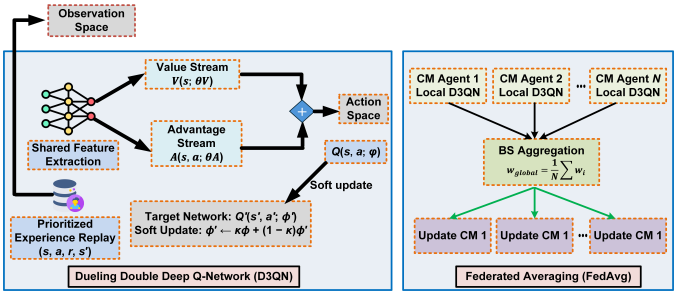


Fig. 2: Architecture of the FedJCC framework showing the D3QN agent structure (left) and the federated averaging (FedAvg) aggregation process (right).

IV. FedJCC for Cooperative UAV Coverage and Connectivity

The novelty of FedJCC does not reside in the D3QN architecture itself, which is adopted as a well-understood baseline for value-based deep RL with discrete action spaces. Rather, the contribution lies in adapting online value-based RL to partially observable, intermittently connected, multi-UAV emergency networks through four design elements: partially observable local observations with a CH-assisted belief map, communication-aware asynchronous federated aggregation, a virtual-force safety shield that projects nominal actions into a collision-free feasible set, and a failure-resilient hierarchical CH/CM architecture with dynamic role reassignment. Thus, D3QN is employed as the per-agent value-function approximator, while the methodological contribution lies in the online federated training, partial-observation state design, relay-connectivity-aware execution, and safety-constrained deployment framework built around it. The virtual-force mechanism is treated as a safety shield applied during action execution. The D3QN policy outputs a nominal action, while the safety layer modifies the resulting motion only when the minimum inter-UAV separation constraint may be violated. Thus, collision avoidance is enforced as a hard safety constraint, whereas the learned policy remains responsible for optimizing coverage, relay connectivity, and energy-aware operation.

These elements are described in the remainder of this section. In what follows, we use the term *online federated reinforcement learning* to emphasize that local datasets are not pre-existing but are generated online through environmental interaction, and that aggregation acts on learned model parameters rather than on raw observations or rewards. Algorithm 1 summarizes the overall framework, which iteratively executes four core phases until convergence.

A. FDRL Framework

The FDRL framework begins with an initialization phase that establishes the key components required for distributed learning and coordination. The BS first initializes a global neural network model $\mathbf{w}_{\text{global}}$, which is shared across all participating UAVs. This global model serves as the starting point for local training and ensures that all agents begin with a common policy representation. The framework parameters include the number of FL rounds R, the number of local training epochs E per round, and the mini-batch size B used

for gradient updates.

The system maintains a set of active UAVs $\mathcal{U}_{\text{active}}$ that are initially deployed based on the estimated coverage requirements of the target region. Each UAV is equipped with a deep Q-network $Q(\mathbf{s}, \mathbf{a}; \phi^Q)$ that approximates the state–action value function, together with a corresponding target network $Q'(\mathbf{s}, \mathbf{a}; \phi^{Q'})$ that provides stable learning targets. The target network parameters $\phi^{Q'}$ are periodically synchronized with the primary network parameters ϕ^Q using soft updates to improve training stability. In addition, each UAV maintains a local experience replay buffer Ω_i that stores transition tuples for sampling during gradient updates.

Communication parameters are configured to support the multi-hop relay architecture of the UAV network. Specifically, the maximum number of relay hops and the signal quality threshold are defined to determine whether a UAV can communicate directly with the base station or must rely on intermediate UAVs as relays. Once initialization is complete, the global model is broadcast to all UAVs in $\mathcal{U}_{\text{active}}$, enabling them to begin local training with identical initial policies. Fig. 2 illustrates the overall architecture of the proposed framework, showing the D3QN structure implemented at each UAV agent and the federated averaging process performed at the base station.

After local training, the participating UAVs upload their updated model parameters to the base station for aggregation. The global model is then updated using the FedAvg rule as

$$\theta_{global} = \theta_{global} - \psi \frac{1}{Y} \sum_{i=1}^{Y} \nabla_{\theta_i} L(\theta_i), \tag{26}$$

where ψ is the learning rate and Y is the number of participating UAVs. After aggregation, the updated global model is broadcast to the participating UAVs for the next training round. The overall FedJCC procedure is summarized in **Algorithm 1**.

Algorithm 1 Online FedJCC with Asynchronous Aggregation

Require: UAV set $\mathcal{U}$, FL rounds R, aggregation interval T_{agg}, local epochs E
Ensure: Optimal policy π^*, optimal deployment $\mathcal{U}^*$

```
Initialize global model w_global and local models w_i^local ← w_global
for each mission episode do
  for each time slot t do
    for each active UAV u_i in parallel do
      observe local observation s_i(t)
      select nominal action a_i^RL(t) using ε-greedy policy
      apply virtual-force safety shield to obtain a_i^safe(t)
      execute a_i^safe(t) and observe r_i(t) and next observation
      store transition in local replay buffer Ω_i
      perform local D3QN update on Q(s, a; φ_i^Q)
    end for
    if t mod T_agg = 0 then
      available UAVs upload local model updates to the BS
      BS performs asynchronous staleness-aware aggregation (Section IV-C)
      successfully connected UAVs receive the updated global model
      disconnected UAVs continue using their latest local model
    end if
    perform deployment adjustment and CH reassignment if needed
  end for
end for
return π*, U*
```

B. Local Training Phase

The local training phase, summarized in **Algorithm 2**, allows each UAV to update its policy using locally collected experiences. At the beginning of each federated learning round, every active UAV u_i receives the current global model and initializes its local parameters as $\mathbf{w}_i^{local} \leftarrow \mathbf{w}_{\text{global}}$.

During local training, UAV u_i interacts with the environment for E epochs. At time slot t, the UAV observes its state $s_i(t)$, which includes its current position, nearby PoI locations, task queue lengths, remaining battery energy, and information about neighboring UAVs. Based on this state, the UAV selects an action $a_i(t)$ using an ϵ-greedy strategy applied to the local Q-network $Q(s, a; \phi^Q)$. After executing the selected action, the UAV receives a reward

$$r_i(t) = w_1 r_{\text{cov}}(t) + w_2 r_{\text{con}}(t) + w_3 r_{\text{bat}}(t), \tag{27}$$

where w_1, w_2, and w_3 denote the non-negative weighting coefficients assigned to coverage, connectivity, and energy efficiency, respectively, with $w_1 + w_2 + w_3 = 1$. In the default setting, we use a coverage-prioritized configuration with (w_1>w_2>w_3), reflecting the emergency-response objective of maximizing PoI coverage while still preserving relay connectivity and energy efficiency. The transition tuple $(s_i(t), a_i(t), r_i(t), s_i(t+1))$ is stored in the replay buffer Ω_i. A mini-batch $\mathcal{B}$ is sampled from the buffer to update the Q-network. Following the Double DQN paradigm, the best next-state action is first selected by the online Q-network and then evaluated by the target network. The temporal-difference target for each sample $k \in \mathcal{B}$ is computed as

$$y_k = r_k + \gamma Q'\big(s'_k,\, \arg\max_{a'} Q(s'_k, a'; \phi^Q);\, \phi^{Q'}\big), \tag{28}$$

where γ is the discount factor, Q denotes the online network used for action selection, and Q' denotes the target network used for value evaluation. The loss function, weighted by the importance sampling coefficients w_k from the prioritized experience replay buffer to correct for non-uniform sampling bias, is given by

$$L(\phi^Q) = \frac{1}{|\mathcal{B}|} \sum_{k \in \mathcal{B}} w_k \left(y_k - Q(s_k, a_k; \phi^Q)\right)^2. \tag{29}$$

The network parameters are updated using gradient descent as

$$\phi^Q \leftarrow \phi^Q - \alpha_Q \nabla_{\phi^Q} L(\phi^Q), \tag{30}$$

Algorithm 2 Local Training Phase

Require: Global model weight $\mathbf{w}_{\text{global}}$, active UAVs $\mathcal{U}_{\text{active}}$, epochs E
Ensure: Updated local model weights $\{(\mathbf{w}_i^{\text{local}}, n_i)\}$
for each UAV $u_i \in \mathcal{U}_{\text{active}}$ **in parallel do**
  $\mathbf{w}_i^{\text{local}} \leftarrow \mathbf{w}_{\text{global}}$
  Collect PoI, signal strength, and battery level
  for epoch $e = 1, 2, \ldots, E$ **do**
    $s_i(t) \leftarrow$ observe local UAV observation
    $a_i(t) \leftarrow \epsilon$-greedy policy using $Q(s_i(t); \mathbf{w}_i^{\text{local}})$
    Execute action $a_i(t)$
    Compute reward: $r_i(t) = r_{\text{cov}}(t) + r_{\text{con}}(t) + r_{\text{bat}}(t)$
    Observe next state $s_i(t+1)$
    Store $(s_i(t), a_i(t), r_i(t), s_i(t+1))$ in replay buffer Ω_i
    Sample mini-batch $\mathcal{B}$ from Ω_i
    Compute Double DQN target and weighted loss
    $a_k^* = \arg\max_{a'} Q(s'_k, a'; \phi^Q)$ for each $k \in \mathcal{B}$
    $y_k = r_k + \gamma Q'(s'_k, a_k^*; \phi^{Q'})$ for each $k \in \mathcal{B}$
    $\delta_k = y_k - Q(s_k, a_k; \phi^Q)$ for each $k \in \mathcal{B}$
    Update priorities in Ω_i using $|\delta_k|$
    $L(\phi^Q) = \frac{1}{|\mathcal{B}|} \sum_{k \in \mathcal{B}} w_k \delta_k^2$
    Update Q-network
    $\phi^Q \leftarrow \phi^Q - \alpha_Q \nabla_{\phi^Q} L(\phi^Q)$
    Soft update target network:
    $\phi^{Q'} \leftarrow \kappa \phi^Q + (1-\kappa)\phi^{Q'}$
  end for
  $n_i \leftarrow$ number of local training samples
end for
return $\{(\mathbf{w}_i^{\text{local}}, n_i)\}$ for all $u_i \in \mathcal{U}_{\text{active}}$

where α_Q denotes the learning rate. The TD errors $\delta_k = y_k - Q(s_k, a_k; \phi^Q)$ are used to update the sampling priorities in the replay buffer Ω_i, ensuring that transitions with larger prediction errors are sampled more frequently in subsequent training iterations. To stabilize training, the target network parameters are updated using soft updates

$$\phi^{Q'} \leftarrow \kappa \phi^Q + (1-\kappa)\phi^{Q'}, \tag{31}$$

where κ is a small update coefficient. After completing the local training epochs, UAV u_i records the number of processed samples n_i, which will be used during the federated aggregation phase to weight its contribution to the global model update.

C. Model Upload and Global Aggregation

The model aggregation phase, presented in **Algorithm 3**, coordinates the collection of locally trained models and their integration into an improved global model. This phase addresses the communication challenges inherent in distributed UAV networks, where not all UAVs may have direct connectivity to the BS.

The first component of this phase handles the multi-hop upload of local models to the BS. Each UAV u_i first checks whether it can communicate directly with the BS by comparing its distance $\|\mathbf{l}_i - \mathbf{l}_{BS}\|$ with the communication range

Algorithm 3 Model Upload and Global Aggregation

Require: Local models weights $\{(\mathbf{w}_i^{\text{local}}, n_i)\}$, UAV positions, BS location
Ensure: Updated global model weight $\mathbf{w}_{\text{global}}$
Phase A: Multi-Hop Model Upload
for each UAV $u_i \in \mathcal{U}_{\text{active}}$ **do**
  if $\|\mathbf{l}_i - \mathbf{l}_{BS}\| \le R_{\text{comm}}$ **then**
    Send $(\mathbf{w}_i^{\text{local}}, n_i)$ directly to the BS
  else
    $\mathcal{N}_i \leftarrow \{u_j : \|\mathbf{l}_i - \mathbf{l}_j\| \le R_{\text{comm}}\}$
    if $\mathcal{N}_i \neq \emptyset$ **then**
      Compute relay scores using Eq. (32)
      $u_{\text{relay}} \leftarrow \arg\max_{u_j \in \mathcal{N}_i} \text{score}_j$
      Forward $(\mathbf{w}_i^{\text{local}}, n_i)$ via u_{relay}
    else
      Store the local model and retry in the next round
    end if
  end if
end for
Phase B: Global Aggregation at the BS
$\mathcal{S} \leftarrow$ set of UAVs with successful uploads
$n_{\text{total}} \leftarrow \sum_{i \in \mathcal{S}} n_i$
if $|\mathcal{S}| \ge N_{\min}$ **then**
  $\mathbf{w}_{\text{global}} \leftarrow \sum_{i \in \mathcal{S}} \frac{n_i}{n_{\text{total}}} \mathbf{w}_i^{\text{local}}$
else
  Retain the previous $\mathbf{w}_{\text{global}}$
end if
Phase C: Global Model Broadcast
for each UAV $u_i \in \mathcal{U}_{\text{active}}$ **do**
  Transmit $\mathbf{w}_{\text{global}}$ to u_i via direct or multi-hop communication
end for
return $\mathbf{w}_{\text{global}}$

R_{comm}. UAVs within direct range transmit their local model parameters $\mathbf{w}_i^{\text{local}}$ and sample count n_i directly to the BS.

For UAVs outside the direct communication range, the algorithm identifies the set of neighboring UAVs $\mathcal{N}_i$ that lie within communication range. If such neighbors exist, a relay selection process is used to determine the most suitable intermediate UAV for forwarding the model. The relay score of candidate UAV u_j is computed as

$$\text{score}_j = \alpha_1 \cdot \text{SNR}_{i,j} + \frac{\alpha_2}{\text{hop}_j} + \alpha_3 \cdot E_j^{\text{rem}}, \tag{32}$$

where $\text{SNR}_{i,j}$ denotes the signal-to-noise ratio of the link between UAVs i and j, hop_j is the number of hops from UAV u_j to the BS, and E_j^{rem} is the remaining energy of the candidate relay. The weighting coefficients α_1, α_2, and α_3 control the relative importance of link quality, path length, and relay energy availability, respectively. The neighbor with the highest score is selected as the relay, and the model is forwarded accordingly. If no eligible neighbor is available, the UAV stores its local model and retries transmission in the next round.

Once the local models have been collected at the BS, the global aggregation phase begins. Let $\mathcal{S}$ denote the set of UAVs

Algorithm 4 Action Execution Phase

Require: Global model $\mathbf{w}_{\text{global}}$, UAV states, locally observed PoIs
Ensure: Updated UAV positions and coverage status

for each UAV $u_i \in \mathcal{U}_{\text{active}}$ **in parallel do**
  Observe state
  $s_i(t) \leftarrow [\mathbf{l}_i(t), \mathcal{P}_i^{obs}(t), Q_i^l(t), Q_i^o(t), E_i^{\text{rem}}(t), \mathbf{N}_i(t)]$
  Select action
  $a_i(t) \leftarrow \text{ModelPredict}(\mathbf{w}_{\text{global}}, s_i(t))$
  Parse $a_i(t) = [\theta_i(t), \phi_i(t), d_i(t), A_i^l(t), A_i^o(t), f_i(t), \mathbf{x}_i(t)]$
  Execute movement
  Update the UAV position using Eqs. (1)–(3)
  $\mathbf{l}_i(t+1) \leftarrow [x_i(t+1), y_i(t+1), z_i(t+1)]$
  Update energy
  Update the remaining UAV energy using Eq. (35)
  Battery check
  if $E_i^{\text{rem}}(t+1) < E_{\text{threshold}}$ **then**
    Plan return-to-base path for u_i
  end if
  Idle check
  if no PoIs are within coverage range **and** u_i is not critical for connectivity **then**
    Mark u_i as `IDLE`
  end if
end for
Update coverage status
for each PoI $p_k \in \mathcal{P}$ **do**
  if $\exists\, u_i$ such that $\|\mathbf{l}_i(t+1) - \mathbf{p}_k\| \le R$ **then**
    Mark p_k as `COVERED`
  else
    Mark p_k as `UNCOVERED`
  end if
end for
return Updated positions $\{\mathbf{l}_i(t+1)\}$ and coverage status

whose local models are successfully received. If a sufficient number of models have been collected, i.e., $|\mathcal{S}| \ge N_{\min}$, the global model is updated using the federated averaging (FedAvg) rule

$$\mathbf{w}_{\text{global}} \leftarrow \sum_{i \in \mathcal{S}} \frac{n_i}{n_{\text{total}}} \mathbf{w}_i^{\text{local}}, \tag{33}$$

where

$$n_{\text{total}} = \sum_{i \in \mathcal{S}} n_i \tag{34}$$

is the total number of training samples contributed by all participating UAVs. This weighted aggregation ensures that UAVs with more local training samples exert a proportionally greater influence on the updated global model. If fewer than $N_{\min}$ models are received, the previous global model is retained to avoid degradation caused by insufficient aggregation. Following successful aggregation, the updated global model is disseminated to all active UAVs. The broadcast process transmits $\mathbf{w}_{\text{global}}$ to each UAV, using multi-hop relay paths when necessary to reach UAVs outside the direct communication range of the BS. Upon receiving the updated global model, each UAV refreshes its local policy parameters, ensuring that the swarm operates with a consistent and improved decision-making model in the next training round.

Algorithm 5 Dynamic Deployment Adjustment and Convergence Check

Require: Active deployment $\mathcal{U}_{\text{active}}$, coverage status, connectivity status, global model weight $\mathbf{w}_{\text{global}}$
Ensure: Updated deployment $\mathcal{U}_{\text{active}}$, convergence flag

Evaluate current status
$\mathcal{P}_{\text{uncovered}} \leftarrow \{p_k \in \mathcal{P} : p_k \text{ is not covered}\}$
$\mathcal{U}_{\text{idle}} \leftarrow \{u_i \in \mathcal{U}_{\text{active}} : u_i \text{ is marked } \texttt{IDLE}\}$
$connected \leftarrow \text{CheckNetworkConnectivity}(\mathcal{U}_{\text{active}}, \text{BS})$
Case 1: Insufficient coverage
if $|\mathcal{P}_{\text{uncovered}}| > 0$ **then**
  $n_{\text{needed}} \leftarrow \text{EstimateRequiredUAVs}(\mathcal{P}_{\text{uncovered}})$
  for $j = 1$ **to** $\min(n_{\text{needed}}, N_{\text{available}})$ **do**
    $u_{\text{new}} \leftarrow \text{DeployNewUAV}()$
    Initialize u_{new} with $\mathbf{w}_{\text{global}}$
    $\mathcal{U}_{\text{active}} \leftarrow \mathcal{U}_{\text{active}} \cup \{u_{\text{new}}\}$
  end for
end if
Case 2: Excess UAVs
if $|\mathcal{P}_{\text{uncovered}}| = 0$ **and** $|\mathcal{U}_{\text{idle}}| > 0$ **then**
  for each $u_i \in \mathcal{U}_{\text{idle}}$ **do**
    if not ConnectivityCritical(u_i) **then**
      Recall u_i to the base station
      $\mathcal{U}_{\text{active}} \leftarrow \mathcal{U}_{\text{active}} \setminus \{u_i\}$
    end if
  end for
end if
Case 3: Connectivity issues
if not $connected$ **then**
  Reposition UAVs or deploy relay UAVs
end if
Convergence check
$cov_{\text{ok}} \leftarrow (|\mathcal{P}_{\text{uncovered}}| = 0)$
$conn_{\text{ok}} \leftarrow connected$
$reward_{\text{stable}} \leftarrow (|\bar{r}_{\text{recent}} - \bar{r}_{\text{prev}}| < \epsilon_r)$
if cov_{ok} **and** $conn_{\text{ok}}$ **and** $reward_{\text{stable}}$ **then**
  $converged \leftarrow \texttt{TRUE}$
  $\mathcal{U}^* \leftarrow \text{MinimalCoveringSet}(\mathcal{U}_{\text{active}})$
else
  $converged \leftarrow \texttt{FALSE}$
end if
return $\mathcal{U}_{\text{active}}$, $converged$

D. Action Execution Phase

The action execution phase, described in **Algorithm 4**, translates the learned policies into physical actions that update UAV positions and the coverage status of the monitored region. This phase is executed in parallel across all active UAVs, with each agent making decisions based on the shared global model and its local observations. Each UAV first constructs its state observation $s_i(t)$, which contains the information required for decision-making. The state vector includes the current position $\mathbf{l}_i(t)$, the locations of known PoIs $\mathbf{l}_{\text{PoI}}$, the current queue

lengths for local computation $Q_i^l(t)$ and offloading $Q_i^o(t)$, the remaining energy $E_i^{\mathrm{rem}}(t)$, and information about neighboring UAVs $\mathbf{N}_i(t)$. This comprehensive state representation enables the policy to simultaneously account for spatial, computational, and communication factors.

The action selection process queries the global model using the current state to obtain the action $a_i(t)$. The action vector includes multiple control dimensions, namely the flight direction $\theta_i(t)$, the flight distance $d_i(t)$, the local computation allocation $A_i^l(t)$, the offloading allocation $A_i^o(t)$, the CPU frequency $f_i(t)$, and the offloading target selection $\mathbf{x}_i(t)$. Energy management is another critical aspect of action execution. After each action, the remaining energy is updated as

$$E_i^{\mathrm{rem}}(t+1) \leftarrow E_i^{\mathrm{rem}}(t) - E_i^{\mathrm{fly}}(t) - E_i^{\mathrm{tra}}(t) - E_i^{l}(t), \quad (35)$$

where the consumed energy includes flight, communication, and local computation components. The algorithm also includes a safety mechanism for low-battery operation. When the remaining energy drops below a threshold E_{th}, the UAV initiates a return-to-base procedure and plans a path back to the charging station to avoid complete battery depletion during the mission.

In addition, the algorithm identifies idle UAVs that neither cover any PoIs nor serve as critical relay nodes for maintaining network connectivity. Such UAVs are marked as `IDLE`, and this status is subsequently used in the deployment adjustment phase to improve resource utilization. At the end of the action execution phase, the coverage status of all PoIs is updated. For each PoI $p_k \in \mathcal{P}$, the algorithm checks whether there exists any UAV within the coverage radius R. If there exists a UAV u_i such that $\|\mathbf{l}_i(t) - \mathbf{p}_k\| \leq R$, the PoI is marked as `COVERED`; otherwise, it is marked as `UNCOVERED`. This coverage assessment is then used to guide the subsequent deployment adjustment decisions.

E. Dynamic Deployment Adjustment and Convergence

The deployment adjustment phase, presented in **Algorithm 5**, adapts the number and configuration of active UAVs according to the current coverage and connectivity status. This dynamic adjustment capability enables the system to respond to changing conditions and improve resource utilization throughout the mission. The phase begins by evaluating three key elements: the set of uncovered PoIs $\mathcal{P}_{\mathrm{uncovered}}$, the set of idle UAVs $\mathcal{U}_{\mathrm{idle}}$, and the overall network connectivity status. The connectivity check verifies whether all active UAVs can communicate with the base station, either directly or through multi-hop relay paths.

When uncovered PoIs remain, the algorithm estimates the number of additional UAVs required to achieve full coverage. For each required UAV, subject to the number of reserve UAVs available, a new UAV is deployed from the base station. The newly deployed UAV is initialized with the current global model $\mathbf{w}_{\mathrm{global}}$, allowing it to immediately benefit from the collective learning of the existing swarm. The active set is then updated as $\mathcal{U}_{\mathrm{active}} \leftarrow \mathcal{U}_{\mathrm{active}} \cup \{u_{\mathrm{new}}\}$. Conversely, when all PoIs are covered and idle UAVs exist, the algorithm considers recalling excess UAVs. For each idle UAV, a connectivity criticality check is performed to determine whether it serves as an essential relay node in the communication network. If the UAV is not critical for connectivity, it is recalled to the base station and removed from the active set, i.e., $\mathcal{U}_{\mathrm{active}} \leftarrow \mathcal{U}_{\mathrm{active}} \setminus \{u_i\}$. This recall mechanism avoids unnecessary energy expenditure by UAVs that no longer contribute to either coverage or connectivity.

If the connectivity check indicates that some UAVs have lost communication paths to the base station, the algorithm triggers a repositioning procedure. The same connectivity-checking mechanism also provides resilience to CH failure. If a CH becomes unavailable due to battery depletion, damage, or loss of communication, the CMs previously associated with that CH reassess the reachable CH candidates and reconnect to the nearest available CH or to a CH reachable through a feasible multi-hop relay path. If no suitable CH or relay path is available, the deployment-adjustment module repositions existing UAVs or deploys an additional relay UAV to restore connectivity. Moreover, UAVs with low residual energy initiate return-to-base behavior before complete battery depletion, and UAVs are recalled only when they are not connectivity-critical. Therefore, the swarm can reorganize around a failed or depleted CH instead of causing a global network failure.

This may involve adjusting the positions of existing UAVs to restore relay paths or deploying dedicated relay UAVs to bridge communication gaps. The convergence check evaluates three conditions. First, the coverage condition is satisfied when all PoIs are covered, i.e., $|\mathcal{P}_{\mathrm{uncovered}}| = 0$. Second, the connectivity condition requires that all active UAVs maintain communication paths to the base station. Third, the reward stability condition checks whether the average reward has stabilized, namely whether $|\bar{r}_{\mathrm{recent}} - \bar{r}_{\mathrm{prev}}| < \epsilon_r$, where ϵ_r is a small threshold. When all three conditions are satisfied, the algorithm is considered to have converged to an optimal or near-optimal deployment. The convergence flag is set to `TRUE`, and the minimal covering set $\mathcal{U}^*$ is extracted from the current active set, representing the most efficient UAV configuration that achieves full coverage while preserving connectivity. Otherwise, the convergence flag is set to `FALSE`, and the main training loop proceeds to the next round.

F. Complexity Analysis

The computational complexity of the proposed FedJCC algorithm is determined by its main operational phases. The local training phase dominates the overall cost, with complexity $\mathcal{O}(E \cdot |\mathcal{B}| \cdot d)$ per UAV, where E is the number of local epochs, $|\mathcal{B}|$ is the mini-batch size, and d denotes the neural network training complexity. For $|\mathcal{U}_{\mathrm{active}}|$ active UAVs, this becomes $\mathcal{O}(|\mathcal{U}_{\mathrm{active}}| \cdot E \cdot |\mathcal{B}| \cdot d)$ per round. The model aggregation phase requires $\mathcal{O}(|\mathcal{S}| \cdot |\mathbf{w}|)$ operations, where $|\mathcal{S}|$ is the number of participating UAVs and $|\mathbf{w}|$ is the model size. The action execution phase has complexity $\mathcal{O}(|\mathcal{U}_{\mathrm{active}}| + |\mathcal{P}|)$, while the deployment adjustment phase requires $\mathcal{O}(|\mathcal{U}_{\mathrm{active}}|^2)$ in the worst case due to connectivity verification. Hence, the overall per-round complexity is dominated by local training when deep neural networks are used.

The dynamic deployment adjustment module is intentionally implemented as a mission-level resource management and safety layer rather than as a learned RL component. This

design choice separates hard mission constraints (uncovered PoIs, low battery, connectivity- critical relays) from the soft, learned objectives optimized by the D3QN policy. In emergency deployment, mission-level safety decisions benefit from being interpretable and verifiable, which is more naturally achieved by a deterministic controller than by a learned policy. To preserve the learning-deployment coupling, an active-UAV efficiency signal is additionally encoded in the reward function, so that the learned policy is encouraged to maintain coverage and connectivity using fewer active UAVs whenever possible. The rule-based component is retained only to enforce hard constraints that are not safely left to a learned policy in safety-critical disaster operations.

The communication overhead is mainly incurred during model upload and broadcast. In each round, every participating UAV uploads a model of size $|\mathbf{w}|$ and receives the updated global model of the same size, resulting in a total communication cost of $\mathcal{O}(|\mathcal{U}_{\text{active}}| \cdot |\mathbf{w}|)$. Although the multi-hop relay mechanism may increase transmission latency, it does not significantly increase the overall communication volume, since each model update is forwarded only once along the relay path. Overall, the proposed framework supports privacy-preserving collaborative learning while maintaining scalability for cooperative UAV deployment in emergency communication scenarios. Closed-form convergence guarantees for federated value-based deep RL under partial observability remain open in the literature, since both deep Q-learning and federated aggregation of local updates introduce non-convex stochastic dynamics. We therefore do not claim global optimality for FedJCC. The framework instead relies on three well-established mechanisms that stabilize training in practice: target networks for bootstrapped value estimation, prioritized experience replay for sample-efficient updates, and staleness-aware aggregation that exponentially down-weights stale local contributions. Empirical convergence curves with confidence intervals over multiple random seeds, are used as the operational notion of convergence in the present work.

V. Performance Analysis

In this section, we evaluate the proposed FedJCC framework through extensive simulations. We first describe the simulation setup and baseline schemes, and then present the convergence behavior, comparative performance, and computational efficiency of the proposed method.

A. Simulation Setup

We consider a disaster-affected area modeled as a 2000×2000 m operational region, where $M = 200$ PoIs are randomly distributed in clusters with a standard deviation of 300 m to emulate realistic disaster scenarios with concentrated survivor locations. The UAV swarm follows a hierarchical architecture consisting of $N_{CH} = 4$ CHs and $N_{CM} = 16$ CMs, resulting in an initial deployment of $N = 20$ UAVs. The CHs operate at altitudes ranging from 150 to 200 m as communication relay nodes, whereas the CMs fly at altitudes ranging from 60 to 100 m to provide ground coverage. The coverage radius is set to $R = 30$ m, and the communication range is $R_{\text{comm}} = 50$ m, with multi-hop relaying supported for up to five hops. The main simulation parameters are summarized in Table II.

TABLE II: Simulation Parameters

Parameter	Value
Operational area	2000×2000 m
Number of PoIs (M)	200
Number of CHs (N_{CH}) / CMs (N_{CM})	4 / 16
CH altitude / CM altitude	200 m / 100 m
Coverage radius (R)	30 m
Communication range (R_{comm})	50 m
Max relay hops	5
Max flight speed (v_{max})	50 m/s
CM energy budget	8 kJ
Propulsion power / Hovering power	20 W / 15 W
Min UAV separation (S_{min})	30 m
Episode length / Time slot duration (τ)	150 steps / 1 s
Training episodes (Federated rounds R)	1000
Local epochs (E) / Mini-batch size (B)	5 / 32
Learning rate (α) / Discount factor (γ)	0.001 / 0.9
ϵ start / min / decay	1.0 / 0.01 / 0.98
Target update rate (κ)	0.005
Replay buffer size	50,000
PoI visit reward / Collision penalty / NFZ penalty	+20 / -20 / -5
Connectivity penalty	1.0
Overlap penalty factor	0.2
Virtual force k_{rep} / d_0 / α_{vf}	100 / 100 m / 0.5
Reward weights (w_1, w_2, w_3)	$(0.5, 0.3, 0.2)$

The FedJCC algorithm is implemented using a D3QN architecture with prioritized experience replay. The action space is discrete, comprising eight directional movement actions and one hover action. Training starts with the full deployment of 20 UAVs. Subsequently, the dynamic deployment adjustment module identifies and recalls spatially redundant or idle UAVs within the first few steps of each episode, typically reducing the active swarm to about 8–10 UAVs while maintaining both coverage effectiveness and critical relay connectivity.

B. Baseline Algorithms

To evaluate the coverage performance of the proposed FedJCC framework, we compare it with the following three representative baseline algorithms, which are state-of-the-art combinatorial optimization methods for heterogeneous UAV scheduling in disaster-area monitoring.

1) **approAlg** [8]: A $\frac{1}{3}$-approximation algorithm that constructs flight tours for K heterogeneous UAVs using a reward decomposition strategy combined with a primal–dual method for solving the underlying orienteering subproblem.
2) **clusterAlg** [44]: A partition-based algorithm that first divides the PoIs into K subsets according to the energy capacities of the UAVs, and then independently solves the orienteering problem for each subset using a $\frac{1}{2}$-approximation algorithm.
3) **greedyAlg** [45]: An iterative greedy algorithm that constructs UAV flight tours by repeatedly inserting the PoI with the highest reward-to-energy ratio into the current tour of each UAV.

These baselines are evaluated on the same 2000×2000 m disaster-area layout and PoI distribution as FedJCC. For comparison, $K = 12$ heterogeneous UAVs are deployed using real commercial UAV specifications from [8], with UAV types randomly assigned to preserve heterogeneity in energy capacities and monitoring capabilities. The baseline algorithms in [8] solve a static one-shot tour optimization problem under complete prior knowledge of all PoI locations. In contrast,

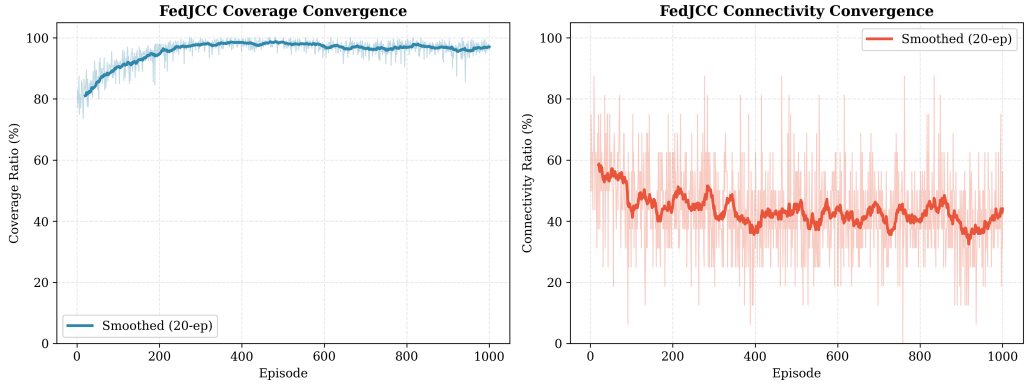


Fig. 3: FedJCC training convergence over 1000 episodes: (a) coverage ratio and (b) training-stage instantaneous connectivity ratio. Smoothed curves (20-episode moving average) are shown alongside the raw per-episode values.

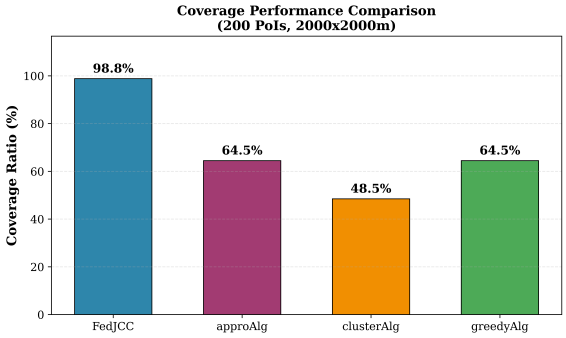


Fig. 4: Coverage performance comparison under the default scenario ($M = 200$ PoIs in a 2000×2000 m area).

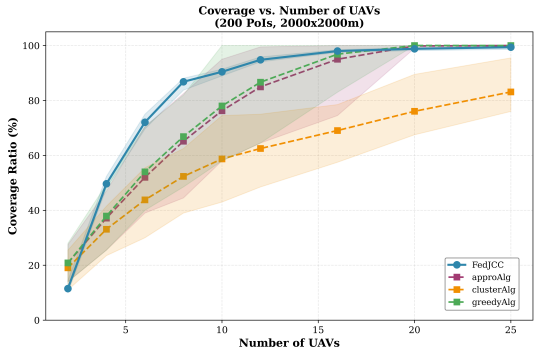


Fig. 5: Coverage ratio versus the number of UAVs ($M = 200$ PoIs in a 2000×2000 m area), with variance bands across multiple seeds.

FedJCC learns adaptive policies through sequential interaction with the environment. Moreover, while the baselines focus solely on reward maximization, FedJCC jointly accounts for coverage, network connectivity, energy efficiency, and collision avoidance. Thus, the comparison reflects both coverage performance and broader operational capability.

In addition to these combinatorial baselines, we include an Independent Deep Q-Network (IDQN) reference [46] to isolate the contribution of federated aggregation. IDQN uses the identical D3QN architecture, prioritized experience replay, reward formulation, and hierarchical deployment as FedJCC,

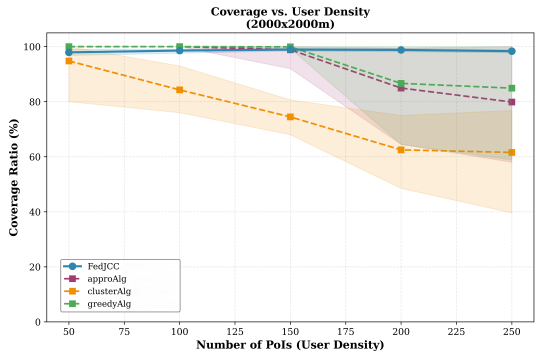


Fig. 6: Coverage ratio versus the number of PoIs (user density) in a 2000×2000 m operational area, with variance bands across multiple seeds.

but each UAV learns from its own experience without any model exchange. This learning-based reference is used in the ablation study of Section V-E and provides a direct deep reinforcement learning comparison alongside the heuristic baselines.

C. Training Convergence and Coverage Performance

Fig. 3 shows the training convergence behavior of FedJCC over 1000 federated learning episodes. The coverage ratio (left panel) increases rapidly from about 80% to 90% during the first 100 episodes, indicating that the UAVs quickly learn basic PoI-seeking behavior. From episodes 100 to 400, the coverage ratio continues to improve steadily and reaches about 95%, benefiting from effective knowledge sharing through federated averaging. After episode 400, the smoothed curve (20-episode moving average) stabilizes above 96%, with a final converged coverage ratio of about 98.8%.

In contrast, the connectivity ratio (right panel) fluctuates throughout training, while its smoothed curve stabilizes around 40–45%. This behavior reflects the inherent tradeoff between coverage and connectivity: as UAVs spread toward geographically distributed PoI clusters, the distances between them increase, which may weaken relay links. The learning process eventually reaches a steady operating point that balances these competing objectives. It is important to note that the connectivity ratio in Fig. 3(b) represents a training-stage,

episode-averaged instantaneous connectivity metric and should not be interpreted as the final operational reliability of the emergency network. During exploration and transient repositioning, UAVs may temporarily spread toward geographically distributed PoI clusters, which reduces instantaneous relay connectivity. Therefore, inference-stage PDR and end-to-end delay are reported as the primary operational communication metrics.

Table III and Fig. 4 compare FedJCC with the three baseline algorithms under the default scenario of $M = 200$ PoIs. FedJCC achieves a coverage ratio of 98.8%, successfully covering 199 out of 200 PoIs. This corresponds to improvements of 52.4% over approAlg (64.5%), 102.7% over clusterAlg (48.5%), and 52.4% over greedyAlg (64.5%). This superior performance stems from the sequential decision-making capability of FedJCC, which allows UAVs to adaptively move toward uncovered PoIs over 150 time steps, whereas the baselines compute one-shot flight tours under fixed energy constraints.

Scalability with UAV Fleet Size: Fig. 5 shows that coverage improves with the number of deployed UAVs for all methods, while FedJCC consistently outperforms the baselines across the full range. FedJCC achieves about 50% coverage with 4 UAVs, exceeds 87% with 8 UAVs, and reaches nearly complete coverage (about 99%) with 20 UAVs. In contrast, the baseline methods achieve lower coverage and exhibit higher variance; for example, at 12 UAVs, greedyAlg and approAlg reach about 85–87%, whereas clusterAlg achieves only 63%. At 20 UAVs and above, all methods approach near-complete coverage, but FedJCC remains more stable, as indicated by its lower variance, reflecting the stability of the learned policy. These results highlight the strong scalability and deployment flexibility of the proposed framework.

Robustness to User Density: Fig. 6 shows that FedJCC remains highly robust as the number of PoIs increases from 50 to 250, consistently maintaining coverage above 97% with low variance. In contrast, the baselines are more sensitive to user density. At low density ($M = 50$), greedyAlg and approAlg achieve near-complete coverage, but their performance degrades as the number of PoIs increases. In particular, clusterAlg drops to about 62% at $M = 200$. Although the baseline gap narrows slightly at $M = 250$ due to shorter inter-PoI distances in denser layouts, FedJCC consistently maintains the best performance across all tested densities.

Per-UAV Efficiency and Normalized Coverage: Fig. 7 shows that FedJCC achieves the highest per-UAV efficiency, covering 12.4 PoIs per UAV, compared with 10.8 for approAlg and greedyAlg and 8.1 for clusterAlg. This corresponds to a 14.8% improvement over the best-performing baseline. Fig. 10 shows two normalized coverage metrics with variance bands to enable fair comparison across different fleet sizes. In both the area-normalized (left) and user-normalized (right) views, FedJCC consistently outperforms the baselines, with the largest gain observed at smaller fleet sizes due to its more effective coordinated coverage strategy. For all methods, per-UAV normalized coverage decreases as fleet size increases, mainly because of higher spatial overlap.

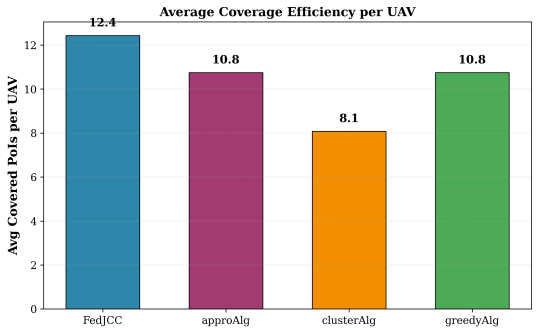


Fig. 7: Average coverage efficiency, measured by the number of PoIs covered per UAV under the default scenario.

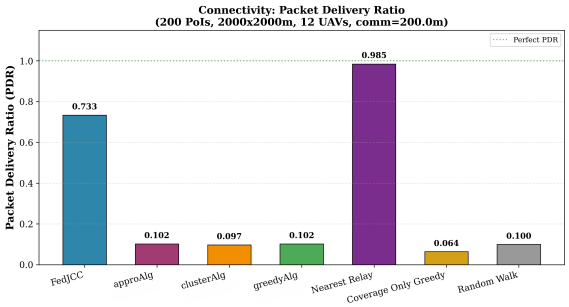


Fig. 8: Average PDR comparison across all algorithms ($M = 200$ PoIs, 2000×2000 m area).

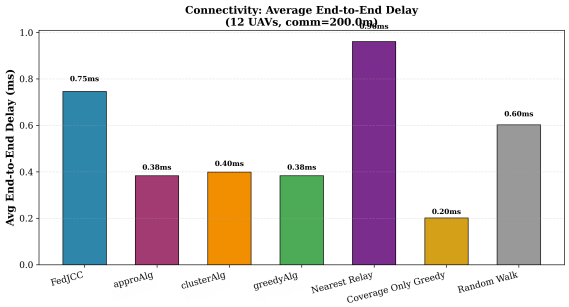


Fig. 9: Average end-to-end delay comparison across all algorithms ($M = 200$ PoIs, 2000×2000 m area).

D. Efficiency and Fairness Analysis

Per-UAV Coverage Efficiency: Fig. 7 compares the average number of PoIs covered per UAV under the default scenario. FedJCC achieves 12.4 PoIs per UAV, corresponding to improvements of 14.8% over approAlg and greedyAlg (10.8 each) and 53.1% over clusterAlg (8.1). This gain stems from the learned sequential policy, which adaptively targets uncovered PoIs and reduces redundant travel. The resulting Jain's fairness index of 0.8438 also indicates a well-balanced workload across UAVs.

PDR: We use PDR as the main connectivity metric, defined as the fraction of CMs that successfully deliver data to at least one CH through multi-hop relay paths at each time step. As shown in Fig. 11, the training PDR stabilizes around 70–75%, with persistent variance reflecting the tradeoff between coverage and connectivity. During inference, the baselines start with high PDR when UAVs remain clustered near deployment, but their PDR drops below 20% as the UAVs disperse. In contrast,

TABLE III: Performance comparison of FedJCC, the IDQN ablation (FedJCC without federated aggregation), and baseline algorithms ($M = 200$ PoIs in a 2000×2000 m area).

Algorithm	Coverage Ratio (%)	PoIs Covered	Avg PoIs per UAV	Area-Norm. Coverage	User-Norm. Coverage	Mission Execution Time (s)	Avg Energy / UAV(J)
FedJCC	**98.8**	**199**	**12.4**	**0.062**	3.30	3.3	2999.6
approAlg	64.5	129	10.8	0.076	0.054	1.90	4669.9
clusterAlg	48.5	97	8.1	0.0572	0.040	0.016	4547.8
greedyAlg	64.5	129	10.8	0.0760	0.054	1.90	4669.9
IDQN	74.6	136	8.5	0.060	0.043	4.46	3065.8

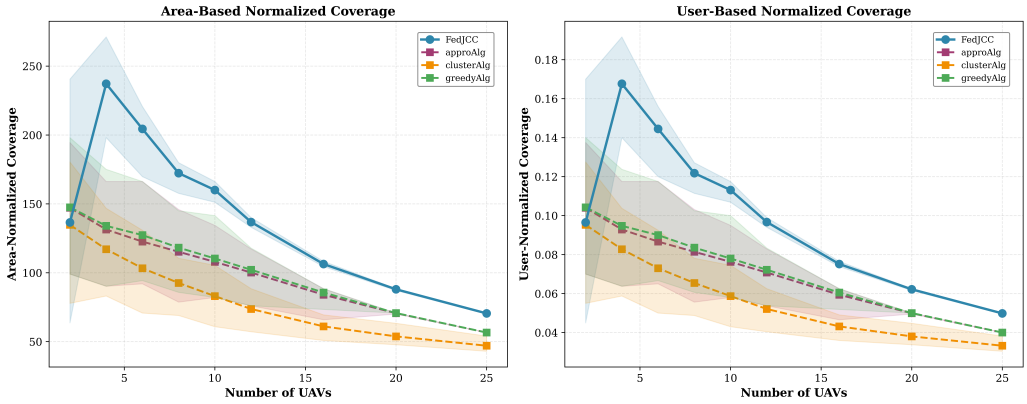


Fig. 10: Normalized coverage versus the number of UAVs, with variance bands: (a) area-normalized coverage and (b) user-normalized coverage.

FedJCC shows fluctuating PDR during active coverage (steps 0–80) and then stabilizes at 70–85% once coverage is largely achieved, indicating a learned temporally structured strategy.

Fig. 8 shows the episode-averaged PDR. FedJCC achieves 0.733, about 7× higher than the tour-based baselines (approAlg: 0.102, clusterAlg: 0.097, and greedyAlg: 0.102), which do not consider connectivity and perform similarly to Random Walk (0.100). Although the Nearest Relay heuristic reaches 0.985 PDR, its coverage drops to about 23%. By contrast, FedJCC maintains a more practical balance, achieving 73% PDR with over 93% coverage.

End-to-End Delay and Coverage-Connectivity Tradeoff: Fig. 9 shows that FedJCC achieves an average delay of 0.75 ms, higher than the tour-based baselines (0.38–0.40 ms) but lower than the Nearest Relay heuristic (0.96 ms). The lower delay of the baselines is mainly due to their very low PDR, since only a few packets are delivered over short paths. In contrast, FedJCC sustains connectivity for 73% of CMs through multi-hop relays, leading to slightly higher but still acceptable delay. Overall, FedJCC operates closest to the desired coverage-connectivity region, achieving 93% coverage and 73% PDR, whereas the other methods favor one objective at the expense of the other.

E. Ablation Study: Effect of Federated Aggregation

To quantify the contribution of federated aggregation, we compare FedJCC against the IDQN reference under an identical training budget, network configuration, and set of random seeds, with the only difference being whether model parameters are aggregated across UAVs. Table IV reports the results under the default scenario. Removing federated aggregation reduces coverage from 98.8% to 74.6%, a relative degradation of 21.6%, and lowers per-UAV efficiency from 12.4 to 8.5 PoIs, while PDR remains comparable. This confirms that federated averaging, rather than any incidental design choice, is a primary driver of FedJCC's performance.

TABLE IV: Ablation study: effect of federated aggregation under an identical training budget ($M = 200$ PoIs, 2000×2000 m area).

Variant	Coverage Ratio (%)	PoIs Covered	Avg PoIs per UAV	PDR
FedJCC	**98.8**	**199**	**12.4**	**0.745**
IDQN	74.6	136	8.5	0.728

The benefit is further illustrated by the training dynamics. Because both variants share identical initialization, their coverage trajectories are indistinguishable until the first aggregation round, after which only the federated variant continues to improve while the independent learners plateau. This isolates federated knowledge sharing as the mechanism that enables FedJCC to surpass independent multi-agent learning under the same architecture and hyperparameters.

F. Computational Efficiency

Fig. 12 compares the offline and online computational costs of all evaluated methods. FedJCC requires 1131.1 minutes (18.9 hours) for offline training, which is substantially higher than the baselines because it learns neural network policies through repeated environment interactions rather than solving a static optimization problem. However, the more relevant metric is online execution time. Once trained, FedJCC completes a full mission in 3.3 seconds, including initialization, dynamic UAV reduction, and 150 decision steps, which is of the same order as approAlg (2.1 s) and greedyAlg (2.0 s), although slower than clusterAlg (23 ms). This makes FedJCC practical for real-time disaster response. Moreover, the training cost is amortized across repeated deployments, whereas the baselines

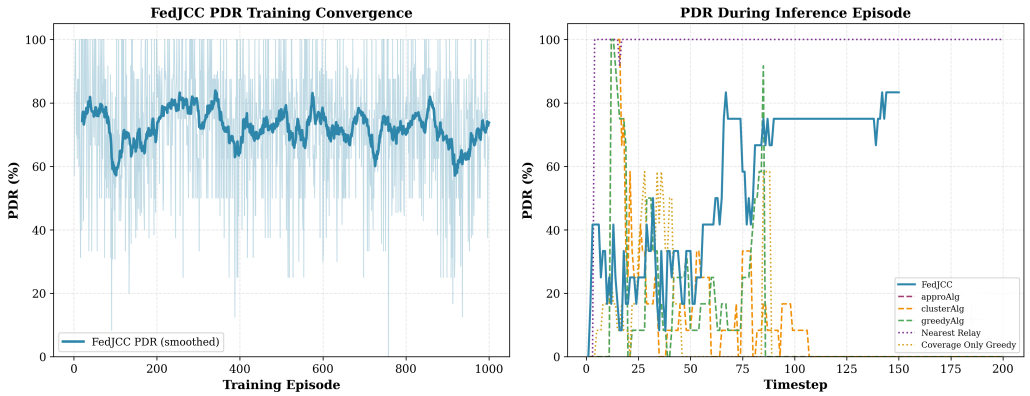


Fig. 11: PDR analysis: (a) FedJCC PDR convergence during training over 1000 episodes, and (b) PDR variation during a single inference episode across all algorithms ($M = 200$ PoIs, 2000×2000 m area).

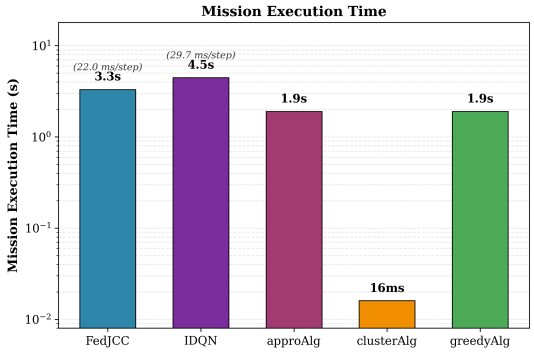


Fig. 12: Computational efficiency comparison with execution time.

must recompute their solutions for each new scenario. FedJCC also provides additional operational capabilities, including relay connectivity, collision avoidance, and dynamic swarm adjustment, which are not supported by the baseline methods.

VI. Conclusion

This paper presented FedJCC, a FDRL framework for joint coverage and connectivity optimization in cooperative UAV networks for emergency communications in disaster-affected areas. The proposed framework integrates a hierarchical CH–CM architecture for multi-hop connectivity, a D3QN with prioritized experience replay for local policy learning, federated averaging for privacy-preserving knowledge sharing, a dynamic deployment adjustment mechanism that reduces the active swarm from 20 to approximately 8 to 10 UAVs, and a virtual force-based collision avoidance mechanism derived from the IAAPF method. Extensive simulations in a 2000×2000 m disaster area with 200 PoIs show that FedJCC achieves a coverage ratio of 98.8%, outperforming combinatorial optimization baselines by 52.4% to 102.7%, while attaining a PDR of 0.733, corresponding to a $7\times$ improvement over connectivity-unaware baselines. The framework also demonstrates strong scalability, consistently outperforming the baselines across different fleet sizes and reaching nearly 99% coverage with 20 UAVs without retraining. In addition, the online inference time of 3.3 seconds confirms its practical suitability for time-critical disaster response scenarios. Future work will also extend the mobility and propulsion-energy models to account for environmental disturbances such as wind gusts, which are known to affect both UAV trajectory tracking and rotary-wing propulsion power.

References

[1] M. Y. Arafat and S. Moh, "Localization and clustering based on swarm intelligence in uav networks for emergency communications," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8958–8976, 2019.

[2] Z. Fan, M. Zhang, Y. Cao, Z. Liu, O. Kaiwartya, Y. Javed, and F. Bashir Hussain, "A novel uav-assisted vanet routing protocol for post-disaster emergency communications," *IEEE Transactions on Network Science and Engineering*, vol. 13, pp. 4863–4882, 2026.

[3] G. Chen, G. Zhao, C. Xu, Z. Han, and S. Yu, "Spatiotemporal-aware deep reinforcement learning for multi-uav cooperative coverage in emergency deterministic communications," *IEEE Transactions on Vehicular Technology*, 2025.

[4] B. Hazarika, P. Singh, K. Singh, S. L. Cotton, H. Shin, O. A. Dobre, and T. Q. Duong, "Generative ai-augmented graph reinforcement learning for adaptive uav swarm optimization," *IEEE Internet of Things Journal*, vol. 12, no. 8, pp. 9508–9524, 2025.

[5] M. Morshed Alam, M. A. Kabir, M. F. A. Omey, S. Ahmed, K. Bin Faruq, T. Ahmed, and M. Y. Arafat, "Integrated trajectory, association, and fair resource allocation for aoi and energy-aware data collection in air-ground collaborative networks," *IEEE Open Journal of the Communications Society*, vol. 7, pp. 685–700, 2026.

[6] J. Agrawal, A. Kumar, M. M. Alam, and M. Y. Arafat, "Hcpmr: A hierarchically coordinated proximal multi-hop routing scheme for fanets in mission-critical environments," *IEEE Access*, vol. 14, pp. 36 505–36 522, 2026.

[7] S. Khemiri, M. A. Kishk, and M.-S. Alouini, "Coverage analysis of tethered uav-assisted large-scale cellular networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, no. 6, pp. 7890–7907, 2023.

[8] W. Xu, C. Wang, H. Xie, W. Liang, H. Dai, Z. Xu, Z. Wang, B. Guo, and S. K. Das, "Reward maximization for disaster zone monitoring with heterogeneous uavs," *IEEE/ACM Transactions on Networking*, vol. 32, no. 1, pp. 890–903, 2023.

[9] S. Chaturvedi, Z. Liu, V. A. Bohara, A. Srivastava, and P. Xiao, "Resource allocation in scma-empowered multi-uav transmission system," *IEEE Transactions on Network Science and Engineering*, vol. 13, pp. 815–827, 2026.

[10] P. Hou, Y. Huang, H. Zhu, Z. Lu, S.-C. Huang, Y. Yang, and H. Chai, "Distributed drl-based intelligent over-the-air computation in unmanned aerial vehicle swarm-assisted intelligent transportation system," *IEEE Internet of Things Journal*, vol. 11, no. 21, pp. 34 382–34 397, 2024.

[11] Y. Qin, Z. Zhang, X. Li, W. Huangfu, and H. Zhang, "Deep reinforcement learning based resource allocation and trajectory planning in integrated sensing and communications uav network," *IEEE Transactions on Wireless Communications*, vol. 22, no. 11, pp. 8158–8169, 2023.

[12] B. Hazarika, K. Singh, A. Paul, and T. Q. Duong, "Hybrid machine learning approach for resource allocation of digital twin in uav-aided

internet-of-vehicles networks," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 2923–2939, 2023.

[13] Y. Hou, J. Zhao, R. Zhang, X. Cheng, and L. Yang, "Uav swarm cooperative target search: A multi-agent reinforcement learning approach," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 568–578, 2023.

[14] Y. Xu, L. Zheng, X. Wu, Y. Tang, W. Liu, and D. Sun, "Joint resource allocation for uav-assisted v2x communication with mean field multi-agent reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 74, no. 1, pp. 1209–1223, 2024.

[15] P. Qin, Y. Wang, J. Zhang, P. Li, and Y. Fu, "Multi-type disaster scenario task offloading in air-ground integrated search and rescue networks: A blockchain-assisted mfl approach," *IEEE Transactions on Vehicular Technology*, 2025.

[16] T. Zhao, F. Li, and L. He, "Secure video offloading in multi-uav-enabled mec networks: A deep reinforcement learning approach," *IEEE Internet of Things Journal*, vol. 11, no. 2, pp. 2950–2963, 2023.

[17] P. Qin, Y. Fu, Y. Xie, K. Wu, X. Zhang, and X. Zhao, "Multi-agent learning-based optimal task offloading and uav trajectory planning for agin-power iot," *IEEE Transactions on Communications*, vol. 71, no. 7, pp. 4005–4017, 2023.

[18] F. Li, K. Zhang, J. Wang, Y. Li, F. Xu, Y. Wang, and N. Tong, "Multi-uav hierarchical intelligent traffic offloading network optimization based on deep federated learning," *IEEE Internet of Things Journal*, vol. 11, no. 12, pp. 21 312–21 324, 2024.

[19] Z. Ji, Z. Qin, and X. Tao, "Meta federated reinforcement learning for distributed resource allocation," *IEEE Transactions on Wireless Communications*, vol. 23, no. 7, pp. 7865–7876, 2023.

[20] B. Hazarika, P. Saikia, K. Singh, and C.-P. Li, "Enhancing vehicular networks with hierarchical o-ran slicing and federated drl," *IEEE Transactions on Green Communications and Networking*, vol. 8, no. 3, pp. 1099–1117, 2024.

[21] B. Jiang, J. Du, C. Jiang, Z. Han, A. Alhammadi, and M. Debbah, "Over-the-air federated learning in digital twins empowered uav swarms," *IEEE Transactions on Wireless Communications*, vol. 23, no. 11, pp. 17 619–17 634, 2024.

[22] J. Xu, H. Yao, R. Zhang, T. Mai, S. Huang, and S. Guo, "Federated learning powered semantic communication for uav swarm cooperation," *IEEE Wireless Communications*, vol. 31, no. 4, pp. 140–146, 2024.

[23] K. T. Pauu, J. Wu, Y. Fan, Q. Pan *et al.*, "Differential privacy and blockchain-empowered decentralized graph federated learning-enabled uavs for disaster response," *IEEE Internet of Things Journal*, vol. 11, no. 12, pp. 20 930–20 947, 2023.

[24] J. Fan, D. Xu, T. Zhang, R. Mumtaz, and K. Yu, "Deep reinforcement learning based dynamic time slot allocation in unmanned aerial vehicle," in *ICC 2024-IEEE International Conference on Communications*. IEEE, 2024, pp. 1310–1315.

[25] N. L. Prasad and B. Ramkumar, "3-d deployment and trajectory planning for relay based uav assisted cooperative communication for emergency scenarios using dijkstra's algorithm," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 4, pp. 5049–5063, 2022.

[26] J. Wu, Y. Sun, D. Li, J. Shi, X. Li, L. Gao, L. Yu, G. Han, and J. Wu, "An adaptive conversion speed q-learning algorithm for search and rescue uav path planning in unknown environments," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 12, pp. 15 391–15 404, 2023.

[27] Z. Ning, Y. Yang, X. Wang, L. Guo, X. Gao, S. Guo, and G. Wang, "Dynamic computation offloading and server deployment for uav-enabled multi-access edge computing," *IEEE Transactions on Mobile Computing*, vol. 22, no. 5, pp. 2628–2644, 2021.

[28] Y. Guan, S. Zou, H. Peng, W. Ni, Y. Sun, and H. Gao, "Cooperative uav trajectory design for disaster area emergency communications: A multiagent ppo method," *IEEE Internet of Things Journal*, vol. 11, no. 5, pp. 8848–8859, 2023.

[29] G. Sun, Y. Wang, Z. Sun, Q. Wu, J. Kang, D. Niyato, and V. C. Leung, "Multi-objective optimization for multi-uav-assisted mobile edge computing," *IEEE Transactions on Mobile Computing*, vol. 23, no. 12, pp. 14 803–14 820, 2024.

[30] K. Liu and J. Zheng, "Uav trajectory planning with interference awareness in uav-enabled time-constrained data collection systems," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 2, pp. 2799–2815, 2023.

[31] A. Khan, B. Hayat, S. Ahmad, F. K. Karim, X. Liu, W. U. Khan, and S. M. Mostafa, "Ai-empowered multi-uav and irs collaboration for spectrum and energy optimization in b5g networks," *IEEE Internet of Things Journal*, 2025.

[32] S. Ahmad, J. Zhang, A. Khan, I. Ali, B. Hayat, and Y. Tian, "Learning-based power minimization in secure multiple uav-aided mimo networks," *IEEE Transactions on Consumer Electronics*, vol. 71, no. 2, pp. 4277–4291, 2025.

[33] Z. Chang, H. Deng, L. You, G. Min, S. Garg, and G. Kaddoum, "Trajectory design and resource allocation for multi-uav networks: Deep reinforcement learning approaches," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 5, pp. 2940–2951, 2022.

[34] Y. Cao, Y. Luo, H. Yang, and C. Luo, "Uav-based emergency communications: An iterative two-stage multiagent soft actor–critic approach for optimal association and dynamic deployment," *IEEE Internet of Things Journal*, vol. 11, no. 16, pp. 26 610–26 622, 2023.

[35] C. H. Liu, Z. Chen, J. Tang, J. Xu, and C. Piao, "Energy-efficient uav control for effective and fair communication coverage: A deep reinforcement learning approach," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 9, pp. 2059–2070, 2018.

[36] H. Fu, J. Wang, J. Chen, P. Ren, Z. Zhang, and G. Zhao, "Dense multiagent reinforcement learning aided multi-uav information coverage for vehicular networks," *IEEE Internet of Things Journal*, vol. 11, no. 12, pp. 21 274–21 286, 2024.

[37] X. Cheng, R. Jiang, H. Sang, G. Li, and B. He, "Trace pheromone-based energy-efficient uav dynamic coverage using deep reinforcement learning," *IEEE Transactions on Cognitive Communications and Networking*, vol. 10, no. 3, pp. 1063–1074, 2024.

[38] Z. Huang, H. Chen, B. Gu, S. Gong, Z. Su, and M. Guizani, "A learning-based iterative algorithm for aoi-optimal trajectory planning in uav-assisted iot networks," *IEEE Transactions on Wireless Communications*, 2025.

[39] Z. Liu, J. Zhang, E. Shi, Z. Liu, D. Niyato, B. Ai, and X. Shen, "Graph neural network meets multi-agent reinforcement learning: Fundamentals, applications, and future directions," *IEEE Wireless Communications*, vol. 31, no. 6, pp. 39–47, 2024.

[40] J. Agrawal, A. Kumar, M. M. Alam, and M. Y. Arafat, "Hcpmr: A hierarchically coordinated proximal multi-hop routing scheme for fanets in mission-critical environments," *IEEE Access*, vol. 14, pp. 36 505–36 522, 2026.

[41] F. Tang, C. Wen, L. Luo, M. Zhao, and N. Kato, "Blockchain-based trusted traffic offloading in space-air-ground integrated networks (sagin): A federated reinforcement learning approach," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 12, pp. 3501–3516, 2022.

[42] P. Singh, B. Hazarika, K. Singh, C. Pan, W.-J. Huang, and C.-P. Li, "Drl-based federated learning for efficient vehicular caching management," *IEEE Internet of Things Journal*, vol. 11, no. 21, pp. 34 156–34 171, 2024.

[43] P. Zhang, Z. Wang, Z. Zhu, Q. Liang, and J. Luo, "Enhanced multi-uav formation control and obstacle avoidance using iaapf-smc," *Drones*, vol. 8, no. 9, p. 514, 2024.

[44] C. Wang, J. Li, F. Ye, and Y. Yang, "A mobile data gathering framework for wireless rechargeable sensor networks with vehicle movement costs and capacity constraints," *IEEE Transactions on Computers*, vol. 65, no. 8, pp. 2411–2427, 2015.

[45] W. Xu, Z. Xu, J. Peng, W. Liang, T. Liu, X. Jia, and S. K. Das, "Approximation algorithms for the team orienteering problem," in *IEEE INFOCOM 2020-IEEE conference on computer communications*. IEEE, 2020, pp. 1389–1398.

[46] A. Tampuu, T. Matiisen, D. Kodelja, I. Kuzovkin, K. Korjus, J. Aru, J. Aru, and R. Vicente, "Multiagent cooperation and competition with deep reinforcement learning," *PloS one*, vol. 12, no. 4, p. e0172395, 2017.

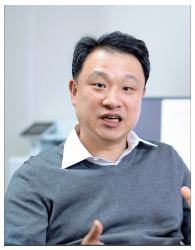
Shafkat Khan Siam received the B.Sc. degree in electronics and communication engineering from Khulna University of Engineering and Technology (KUET), Khulna, Bangladesh, in 2019, and the M.S. degree in computer engineering from Chosun University, Gwangju, South Korea, in 2023. He is currently pursuing the Ph.D. degree in computer science with the School of Computer Science and IT, University College Cork, Cork, Ireland. Prior to his doctoral studies, he worked as a Machine Learning Engineer and an Artificial Intelligence Engineer in the e-commerce and telecommunications industries, where he developed recommendation systems, data processing pipelines, and AI-powered applications. His research interests include machine learning, deep learning, explainable artificial intelligence, and their application to next-generation wireless networks, including UAV-assisted and federated learning systems. For more information, see https://khan022.github.io/.

Muhammad Yeasir Arafat received his received the Ph.D. degree in computer engineering from Chosun University, South Korea in 2020. He is currently a Senior Postdoctoral Fellow at the School of Computer Science and Information Technology at University College Cork, Ireland. Prior to this role, he was a Research Professor at Chosun University, South Korea. He also held the position of Assistant Professor in the Department of Computer Engineering, Chosun University, South Korea. He was a Korean Government Scholarship Program (KGSP) grantee and received Minister Award from the Ministry of Education, Korea for his excellent study. His current research interests include ad hoc networks, unmanned aerial vehicle networks, and wireless body area networks with a focus on network architectures and protocols.

Krishnendu Guha (Member, IEEE) received the M.Tech. (Hons.) and Ph.D. degrees from the University of Calcutta, India. In M.Tech., he secured the University Gold Medal from the University of Calcutta. He was an INSPIRE Fellow with the Department of Science and Technology, Government of India. He is currently an Assistant Professor/ Lecturer with the School of Computer Science and Information Technology, University College Cork (UCC), Ireland. Prior to this, he was a Postdoctoral Research Associate with the University of Florida, USA. He also held the position of an Assistant Professor (Temporary Faculty) with the National Institute of Technology, Jamshedpur, India. He was an Intel India Research Fellow and a Visiting Scientist with Indian Statistical Institute, Kolkata, India. Presently, he has received research grants from University College Cork, Research Ireland CONNECT Centre, EU EOSC, and Horizon Europe. His research arena essentially involves embedded systems and hardware security, quantum computing, real-time systems security with a focus on cloud, and IoT platforms. He also has a strong interest in artificial intelligence and nature-inspired approaches. He has served as a reviewer for various top tier journals, technical program committee member for several conferences, and also served as a chair for a few of them.

Zilong Liu (Senior Member, IEEE)is an Associate Professor, the 6G Lab Manager, and the 2024 Outstanding Mid-Career Researcher, in the Faculty of Science and Health, University of Essex. He is/was a Distinguished Lecturer of the IEEE Vehicular Technology Society and an awardee of the prestigious EPSRC New Investigator Award (NIA) in the UK. His research generally lies in the interplay of coding, signal processing, and communications, with a major objective of bridging theory and practice. He has widely published in quality journals such as IEEE Transactions on Information Theory and IEEE Transactions on Signal Processing. So far, he has made a series of original contributions to waveforms, sequences, and multiple access. Prior to his current post, he was with Huazhong University of Science and Technology (Bachelor, 2000-2004), Tsinghua University (MSc, 2004-2007), Nanyang Technological University Singapore (PhD and postdoc, 2008-2017), University of Melbourne (Visiting PhD, 2012), the Hong Kong University of Science and Technology (Visiting PhD, 2012), and the University of Surrey (Senior Research Fellow, 2018-2019). Details of his research can be found at: https://sites.google.com/site/zilongliu2357

Enric Pardo received the B.Sc. degree in telecommunications engineering from Universitat Politecnica de Catalunya, Spain, in 2016, and the Ph.D. degree from the Centre for Telecommunications Research, King's College London, U.K., in the area of multi-connectivity in 5G Networks, in 2021. He is currently a Research and Technology Associate with Luxembourg Institute of Science and Technology (LIST), where he has contributed to several national and European projects related to 5G and beyond communications with technical contributions, leading tasks and work packages, and budget acquisition. His research interests include mathematical analysis with the use of stochastic geometry for network performance evaluation, applied mathematics in 5G communications, the coexistence of terrestrial and aerial users, and vehicular communications.



Hyundong Shin (Fellow, IEEE) received the B.S. degree in Electronics Engineering from Kyung Hee University (KHU), Yongin-si, Korea, in 1999, and the M.S. and Ph.D. degrees in Electrical Engineering from Seoul National University, Seoul, Korea, in 2001 and 2004, respectively. During his postdoctoral research at the Massachusetts Institute of Technology (MIT) from 2004 to 2006, he was with the Laboratory for Information Decision Systems (LIDS). In 2006, he joined the KHU, where he is currently a Professor in the Department of Electronic Engineering. His research interests include quantum information science, wireless communication, and machine intelligence. Dr. Shin received the IEEE Communications Society's Guglielmo Marconi Prize Paper Award and William R. Bennett Prize Paper Award. He served as the Publicity Co-Chair for the IEEE PIMRC and the Technical Program Co-Chair for the IEEE WCNC and the IEEE GLOBECOM. He was an Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and IEEE COMMUNICATIONS LETTERS.

Md. Noor-A-Rahim (Senior Member, IEEE) received the Ph.D. degree from the School of Information Technology and Mathematical Sciences, University of South Australia, Australia, in 2015. He is currently serving as an Assistant Professor (Lecturer-Above the Bar) within the School of Computer Science and Information Technology, University College Cork, Ireland. Prior to this role, he was a Senior Researcher and a Marie Curie Fellow within the School of Computer Science and Information Technology. He also held the position of Postdoctoral Research Fellow with the Nanyang Technological University, Singapore. His research interests include control over wireless networks, intelligent transportation systems, machine learning, signal processing, and DNA-based data storage. In recognition of his academic excellence, he was honored with the Michael Miller Medal for presenting the most outstanding Ph.D. thesis in 2015. For more information, see https://narahim.github.io/.