DRL-based Energy-Efficient Trajectory Planning for Multiple UAVs under Centralized Control

Shahnila Rahim[1], Limei Peng[2], and Pin-Han Ho[3]
[1]Applied Data Science, Noroff University College, Kristiansand, Norway
[2]School of Computer Science and Engineering, Kyungpook National University, Daegu, South Korea
[3]Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada

Non-terrestrial networks (NTNs), comprising unmanned aerial vehicles (UAVs) with varying battery and computational capacities, are key technologies for implementing 5G and beyond (B5G) as well as 6G. However, the integration and orchestration of different NTN layers remain underexplored. This paper investigates a two-layer NTN architecture, featuring a high-altitude platform (HAP) with either satellites or high-performance UAVs, and low-altitude UAVs (LAUs) with limited capacity, tasked with collecting data from terrestrial Internet-of-Things (IoT) nodes. We delve into the dynamics of the NTN, focusing on a framework where multiple capacity-constrained LAUs are coordinated by a centralized HAP. The proposed research involves devising optimized trajectories for these cooperative LAUs, under HAP guidance, to boost energy efficiency in data collection. The proposed work tackles this challenge by developing two integer linear programming (ILP) optimization models and introducing a novel algorithm named collaborative multi-agent energy-efficient trajectory design and data collection (CoMETD). The proposed CoMETD, operating within the HAP, leverages a deep reinforcement learning (DRL)-based dueling double deep Q-learning network (D3QN) to dynamically plan multi-LAU trajectories, eliminating the need for prior knowledge of IoT node locations. The effectiveness of the proposed algorithm is validated through extensive simulations, where its performance is compared with contemporary state-of-the-art methods.

Index Terms—**Deep reinforcement learning, UAVs, energy efficiency, data collection, trajectory planning, HAP, aerial computing**

I. Introduction

THE integration of terrestrial networks (TNs) with non-terrestrial networks (NTNs), forming a hierarchical space-aerial-surface computing (SASC) architecture [1], has emerged as a key enabler for efficiently provisioning both legacy and on-demand mobile services, paving the way for sixth-generation (6G) communications and services. NTNs, consisting of high-altitude platforms (HAPs) and unmanned aerial vehicles (UAVs), present significant potential for scalable and energy-efficient data collection, particularly in emergency response scenarios where rapid deployment of on-demand wireless networks is critical [2]. However, UAV backhaul's heavy reliance on terrestrial infrastructure poses substantial challenges in scenarios lacking ground connectivity. Furthermore, the limited battery capacity of UAVs necessitates additional computational support. To address these challenges, attention has shifted toward deploying high-performance UAVs equipped with renewable energy sources and advanced computational capabilities, forming HAPs that assist capacity-constrained low-altitude UAVs (LAUs) in delivering cloud-like services. While the hierarchical SASC architecture offers an efficient communication framework by harnessing the computational capabilities of HAPs and the flexibility of UAVs, optimizing the energy efficiency of UAV trajectory planning in such dynamic and resource-constrained environments remains an ongoing and critical challenge.

The SASC HAP typically operates at an altitude of around 20 km in the stratosphere and is capable of maintaining a quasi-stationary position in the air, providing line-of-sight (LoS) communications with a wide coverage radius that spans 50 to 500 km [3]. Leveraging the advantages of HAP, such as a wide coverage range, long-term solar power utilization, and significant data storage and processing capabilities, and combining these with the deployment flexibility and cost-efficiency of LAUs [4], synthesizing HAP with LAUs to form the hierarchical two layers of the NTN is crucial for maximizing the potential of SASC. Enhancing the efficiency of this synthesis is a key research area, wherein the trajectory design for LAUs (hereafter LAU and UAV are used interchangeably) to facilitate energy efficiency through collaboration across layers presents one of the key challenges.

Despite notable progress in UAV trajectory optimization, current research has limitations [4] [5]. Conventional methods often assume static network conditions and fail to account for the dynamic and unpredictable nature of real-world NTNs. This leads to reduced adaptability in scenarios involving mobile IoT nodes, fluctuating wireless channels, and energy constraints. Additionally, existing approaches frequently treat HAPs and UAVs as separate entities, underutilizing their collaborative potential to optimize system performance. Multi-UAV collaboration introduces further complexity, which is not effectively addressed by traditional decentralized methods, resulting in suboptimal energy efficiency and coverage. These limitations underscore the need for robust solutions capable of addressing dynamic network conditions and coordinating multiple UAVs under centralized control. To address these challenges, we propose an innovative framework that leverages centralized HAP control and advanced deep reinforcement learning to enable efficient multi-UAV collaboration in real-world NTNs.

To address these challenges, we propose the collaborative multi-agent energy-efficient trajectory design and data collection (CoMETD) framework, which uniquely integrates

Manuscript received September 13, 2024; revised December 22, 2024.
Corresponding author: Limei Peng (email: auroraplm@knu.ac.kr).

HAPs and UAVs within a hierarchical architecture. CoMETD employs a dueling double deep Q-network (D3QN) algorithm, enabling centralized control via the HAP to dynamically optimize UAV trajectories. Unlike existing methods, CoMETD eliminates the need for prior knowledge of IoT node locations, adapting to dynamic environments through collaborative learning among UAVs. This approach not only enhances energy efficiency but also ensures optimal data collection by maximizing the coverage of IoT nodes while adhering to energy constraints.

By bridging these gaps, CoMETD demonstrates significant advancements in optimizing trajectory planning in NTNs. Its practical applications extend to emergency response scenarios, where rapid and energy-efficient deployment of UAVs is critical, as well as large-scale IoT networks requiring robust data collection solutions. Through extensive simulations, we validate CoMETD's superiority over counterpart methods in terms of energy efficiency, coverage, and system scalability, solidifying its relevance in the field of next-generation NTN architectures.

A. Literature review

Much of the existing pertinent work has overlooked the capacity for optimizing UAV paths through the collaboration across SASC layers. Most of these studies have focused on the trajectory planning of either a single UAV or multiple UAVs, without the collaboration of HAP, using either conventional methods or machine learning (ML) approaches. For instance, authors in [6] focused on maximizing minimum rates among terrestrial IoT devices through path planning and IoT device scheduling for a single UAV. [7] explored multiple UAVs for data collection, emphasizing optimized trajectory design through clustering and minimizing power consumption. In [8], authors optimized the trajectories of multi-UAVs to minimize energy consumption. [9] proposed a map compression technique and leveraged dynamic programming to efficiently design the UAV trajectory. [10] explored the 3D deployment of UAVs to maximize user coverage, addressing the optimization by separately optimizing the horizontal and vertical locations of the UAVs. ML techniques such as reinforcement learning (RL) and deep reinforcement learning (DRL) have been widely used for solving UAV trajectory planning. In [11], UAV data transmission and hovering power minimization were addressed through a DRL-solved energy-efficient path optimization problem. In [5], the authors proposed RL-based collaborative UAVs to perform energy-efficient trajectories within minimal flying time. However, these methods, whether conventional or ML-driven, often struggle to adapt to the dynamic and unpredictable nature of real-world network conditions, necessitating more flexible and responsive approaches.

Recent research has begun to investigate the integration of HAPs with LAUs to enhance service efficiency and reliability. [12] devised a heuristic greedy algorithm to address power and sub-carrier allocation issues within a fixed setup of HAPs and UAVs. [13] introduced a content caching prediction method for HAP-assisted multi-UAV networks, utilizing a deep regression model within a UAV-HAP hierarchical federated learning (FL) framework, incorporating the federated averaging algorithm. In [14], authors proposed a DRL-based method for allocating coverage areas in a heterogeneous UAV network with HAPs, aiming for maximum coverage while preserving energy efficiency. [15] focused on joint trajectory optimization and channel allocation, nonetheless, neglected the potential for collaborative data collection and computational load division between HAPs and LAUs.

In summary, most existing research has primarily focused on UAVs and HAPs as distinct problems, without fully exploring the potential advantages of their integration, particularly in terms of UAV trajectory optimization. The under-utilization of HAP capabilities limits the full potential of HAP-UAV integration in real-world networks. This highlights the need for further research exploring efficient UAV trajectory optimization in a broader context, considering practical complexities, and integrating multiple UAVs within SASC hierarchical network architectures.

B. Motivation and Contributions

To bridge the research gap and driven by the advantages of the collaborative network architecture of HAPs and UAVs, this paper explores the integration of these two SASC entities to enhance the network coverage and energy efficiency of the hierarchical SASC. This integration allows UAVs to optimize their trajectories for maximum data collection from IoT nodes, with the support of HAP positioned at higher altitudes to centrally manage UAV trajectories. By leveraging the strengths of both HAP and UAVs, this study proposes a dynamic environment solution, namely dueling double deep Q-network (D3QN)-based collaborative multi-UAV energy-efficient trajectory design and data collection (CoMETD), tailored for highly dynamic environments. The contribution of this paper is summarized as follows:

- Unlike previous studies that primarily developed performance metrics for NTNs, this research evaluates the overall SASC performance, including both NTN and TN, by accounting for the roles of both HAP and UAVs in dynamic scenarios and considering the mobility of IoT nodes across various environments.
- We formulate an optimization problem to address multi-UAV trajectory planning under HAP guidance, dividing it into two sub-problems: the first aims to maximize the coverage of IoT nodes, which is used as the input for the second subproblem that focuses on minimizing the overall energy consumption of UAVs for data collection. To bypass the computational complexity typical of traditional methods, we propose a D3QN-based CoMETD algorithm, which is implemented within a streamlined online framework, specifically designed to handle dynamic multi-UAV environments, all under the purview of the HAP.
- Extensive simulations are conducted to evaluate the performance of the proposed CoMETD algorithm, focusing on criteria such as IoT node service, energy consumption, and utilization rate. By comparing it with the state-of-the-art DQN [15] and scenarios without HAP, the study

highlights the significant impact of the centralized system and the advantages of collaborative learning in enhancing data collection efficiency. The results demonstrate the D3QN-based CoMETD algorithm's superior performance in dynamic environments and under various wireless channel conditions.

C. Organization

This paper proceeds as follows: Section II describes the system model, including network architecture, energy consumption, and communication models. Section III discusses the optimization problem formulations. Section IV details the D3QN-based CoMETD process for multi-UAV data collection in the HAP scenario and explains the associated algorithm. Section V presents the experimental results, and Section VI concludes the paper.

II. System Model

A. Network Model

In this study, we investigate uplink data transmission in an area of interest (AoI), denoted as ψ, where a HAP, designated as h, controls the trajectories of a set of UAVs, denoted as $\mathcal{L} = \{1, 2, \ldots, L\}$. We assume that the AoI ψ consists of $c \times c$ equal-sized cells, with c being a natural number, i.e., $c \in \mathbb{N}$. These UAVs travel at a uniform speed, denoted as v, and their mission is to provide communication coverage to a set of ground IoT nodes within the $c \times c$ cell region, all without relying on any terrestrial communication infrastructure. The IoT nodes, denoted as $\mathcal{M} \in \{1, 2, \ldots, M\}$, are randomly distributed and move with a random walk at a speed of v_m within the given AoI, as described in [16]. To maintain generality, we position the UAVs above the AoI at an altitude H_l, where $l \in \mathcal{L}$, and their objective is to maximize the number of served IoT nodes while adhering to a given energy constraint, $\mathcal{E}_l^{max}$. It is important to note that the UAVs operate without any prior knowledge of the IoT node locations.

As depicted in Fig. 1, UAVs initiate their trajectories within the centralized system from random starting positions. Time is divided into T equal-length discrete intervals, defined as $\mathcal{T} = \{0, 1, 2, \ldots, T\}$. The 3D coordinates of UAV l at time step t are represented as $\mathcal{Z}_l(t) = (x_l(t), y_l(t), H_l)$, where $t \in \mathcal{T}$, and H_l denotes the altitude of UAV l. Furthermore, the designated resting position of the UAVs is indicated as $\mathcal{P}_l^{\text{final}}$. The HAP flies at an altitude of H_h above the center of the AoI, which is denoted as $\mathcal{Z}_h(t) = (x_h(t), y_h(t), H_h)$.

In an unobserved environment, a HAP acts as the centralized system, controlling the trajectories of all UAVs to ensure efficient performance. Within this centralized system, the HAP computes relevant information that facilitates optimal decision-making for UAV coordination and trajectory planning. To prevent collisions and ensure mission safety, the HAP not only provides future navigation instructions to the UAVs but also maintains a record of their current locations. A collision is recognized when the distance between any pair of UAVs falls below a predetermined safety threshold, denoted as $D_{\min}$. Therefore, the system enforces a strict threshold on the UAVs, limiting their movement if their mutual distance breaches $D_{\min}$, as depicted in Fig. 2. This seamless coordination between the HAP and UAVs enables efficient data collection while ensuring safe and uninterrupted operations. Moreover, the UAVs utilize the frequency division multiple access (FDMA) technique to allocate distinct frequency bands, enabling simultaneous communication with IoT nodes without interference.

We assume that each UAV is equipped with a directional antenna that can be adjusted to control the UAV's beamwidth for data collection from IoT nodes. For simplicity, we consider that the UAV's antenna has equal half-power in both azimuth

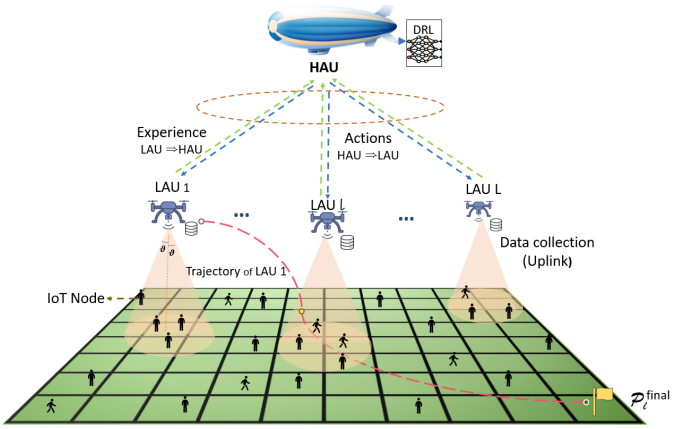


Figure 1: System architecture (HAU: high-altitude UAV).

Parameters	Values
$\mathcal{M}$	Set of IoT nodes
$\mathcal{L}$	Set of UAVs
h	HAP
ψ	Area of interest
$c \times c$	Number of unit cells in an area of interest
$\mathcal{T}$	Set of time steps
E_{max}	Maximum power of the UAV l (W)
H_l, H_h	Flying altitude of UAV l and HAP h (m)
$\mathcal{P}_l^{\text{final}}$	Final position of UAVs
$\mathcal{Z}_l(t)$, $\mathcal{Z}_h(t)$	Coordinates of UAV l and HAP h at time t
ξ_0	Blade profile
ξ_1	Induced power of UAV
v_0	Rotor induced velocity (m/s)
μ_{tip}	Tip of the rotor blade (m/s)
z_0	Fuselage drag ratio (m^2)
τ	Rotor solidity
κ	Air density (kg/m^3)
A	Rotor disc area (m^2)
B	Bandwidth
α	Path-loss exponent
σ	Noise variance
$\mathcal{Y}_{m,l} \in \{0, 1\}$	Binary variable, 1 if UAV l can successfully serve IoT node m, and 0 otherwise
D_{min}	Minimum distance between UAVs (m)

Table I: List of parameter notations.

and elevation angles, with both angles measured as 2ϑ in radians, where $\vartheta \in (0, \frac{\pi}{2})$ [17]. Additionally, the corresponding antenna gain in the direction (θ, ϕ) is approximately modeled as:

$$G(\theta,\phi) = \begin{cases} \frac{G_0}{\vartheta^2}, & -\vartheta \le \theta \le \vartheta, -\vartheta \le \phi \le \vartheta, \\ g \approx 0, & otherwise \end{cases} \quad (1)$$

where $G_0 = \frac{30000}{2^2} \times \left(\frac{\pi}{180}\right)^2 \approx 2.2846$, and θ and ϕ represent the azimuth and elevation angles, respectively, as defined in [17]. Note that in practice, g satisfies the condition $0 < g \ll \frac{G_0}{\vartheta^2}$, and for simplicity, we assume $g = 0$. The beamwidth angle is adjusted based on the number of detected IoT nodes. At time step t, UAV l detects a number of $N_l(t)$ IoT nodes. To strengthen the received signal, UAV l adjusts the antenna angle to narrow the lobe until at least $\frac{N_l(t)}{2}$ IoT nodes are within its range.

B. UAV Energy Consumption Model

We consider HAP to be equipped with solar power, and to operate at an altitude where cloud-free conditions prevail, ensuring uninterrupted charging capabilities [18]. The energy consumption of the UAVs can be divided into two primary components: (i) propulsion energy and (ii) energy related to communication. The latter, used when the UAVs transmit, analyze, and receive signals, is negligible compared to the propulsion energy. We assume the communication-related power to be a constant, denoted as $\mathcal{E}_l^{comm}(t)$, for UAV $l \in \mathcal{L}$ at time step $t \in \mathcal{T}$. The propulsion energy for UAV l at time step t, denoted as $\mathcal{E}_l^{prop}(t)$, is consumed during movement and hovering, and is formulated as follows [19].

$$\mathcal{E}_l^{prop}(t) = \mathcal{P}_0 \left(1 + \frac{3v_l^2}{\mu_{tip}^2}\right) + \mathcal{P}_1 \left(\sqrt{1 + \frac{v_l^4}{4v_0^2}} - \frac{v_l^2}{2v_0^2}\right)^{\frac{1}{2}} + \left(\frac{1}{2} z_0 \tau \kappa A v^3\right) \quad (2)$$

where $\mathcal{P}_0$ and $\mathcal{P}_1$ are constant parameters that denote blade profile and induced power, respectively. While the UAV is in its hovering state, μ_{tip} represents the tip of the rotor blade, and v_0 represents the induced rotor velocity during hovering. Furthermore, κ, τ, z_0, and A are parameters that represent air density, rotor disc area, rotor solidity, and fuselage drag ratio, respectively.

To calculate the energy consumed by UAV l at time step t during hovering, we set the speed of UAV l as $v_l = 0$ in Eq. (2). Then, the hovering energy, denoted as $\mathcal{E}_l^{hover}(t)$, can be expressed as:

$$\mathcal{E}_l^{hover}(t) = \mathcal{P}_0 + \mathcal{P}_1 \quad (3)$$

When a UAV collects data from an IoT node, it hovers above the IoT nodes and consumes power related to communication. Thus, the total power consumed during data collection by UAV l at time step t, denoted as $\mathcal{E}_l^{DC}(t)$, is as follows:

$$\mathcal{E}_l^{DC}(t) = \mathcal{E}_l^{hover}(t) + \mathcal{E}_l^{comm}(t) \quad (4)$$

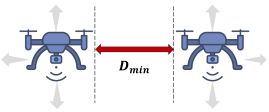


Figure 2: Safety distance to avoid collision between two UAVs.

Whereas, the total power consumed by UAV l at time step t, denoted as $\mathcal{E}_l^{tot}(t)$, can be calculated as:

$$\mathcal{E}_l^{tot}(t) = \mathcal{E}_l^{DC}(t) + \mathcal{E}_l^{prop}(t) \quad (5)$$

During the mission, UAV l follows a trajectory controlled by HAP represented by a sequence of cells $\hat{i}^l = [\hat{i}_1^l, \hat{i}_2^l, ..., \hat{i}_{j_l}^l]$, where each cell corresponds to a visited location, and the index j_l indicates the last visited cell by UAV l. This trajectory represents the path followed by the UAV to visit and serve IoT nodes. At each step of the trajectory, the UAV can choose one of four discrete directions: east, west, north, or south, to move from its current position, as detailed in Section IV-A. The total energy consumption of UAV l during the mission, defined by the trajectory $\hat{i}^l$, includes energy used for UAV flying, hovering, and communication, as described in Eq. (5). This total energy consumption is denoted as $\tilde{\mathfrak{J}}(\hat{i}^l)$ and calculated as follows:

$$\tilde{\mathfrak{J}}(\hat{i}^l) = \sum_{t \in \mathcal{T}} (\mathcal{E}_l^{DC}(t) + \mathcal{E}_l^{prop}(t)) \quad (6)$$

C. Ground-Air-Space Communication Model

1) *Ground-to-UAV channel model:* A UAV operated at a sufficiently high altitude tends to create line-of-sight (LoS) links with the ground IoT nodes. However, it also experiences small-scale fading caused by the presence of rich scattering in the environment such as buildings [20]. We utilize the probabilistic LoS channel model [21] to simulate the environment, which accounts for a combination of LoS and Non-LoS (NLoS) conditions. The large-scale coefficients $\beta_{l,m}$ of the IoT node $m \in \mathcal{M}$ and the UAV $l \in \mathcal{L}$ at time step $t \in \mathcal{T}$ channels can be calculated by:

$$\beta_{l,m}(t) = \begin{cases} \beta_0 d_{l,m}^{-\alpha}(t), & \text{LoS} \\ \kappa \beta_0 d_{l,m}^{-\alpha}(t), & \text{NLoS} \end{cases} \quad (7)$$

where α is the path loss exponent that usually has a value between 2 and 6 and β_0 is the average channel power gain at the reference distance of $d_0 = 1$ m [19]. The parameter κ is the attenuation loss for NLoS scenario [21]. $d_{m,l}$ denotes the horizontal distance between UAV $l \in \mathcal{L}$ and IoT node $m \in \mathcal{M}$ at height H_l at time step t, is given as:

$$d_{l,m}(t) = \sqrt{(x_l(t) - x_m(t))^2 + (y_l(t) - y_m(t))^2 + H_l^2} \quad (8)$$

where (x_m, y_m) is the 2D location of IoT node m. The probability of LoS link depends on the angle $\phi_{l,m}$ between the UAV l and IoT node m at time step t can be evaluated as [21]:

$$P_{LoS}(\phi_{l,m})(t) = \frac{1}{a + \exp\left(-b(\phi_{l,m} - a)\right)} \tag{9}$$

where a and b are the environment parameters. The probability of NLoS can be calculated as:

$$P_{NLoS}(\phi_{l,m})(t) = 1 - P_{LoS}(\phi_{l,m})(t) \tag{10}$$

The channel gain between UAV l and IoT node m at time step t is expressed as:

$$\begin{aligned} g_{l,m}(t) = P_{LoS}(\phi_{l,m})(t)\beta_0 d_{l,m}^{-\alpha}(t) \\ + P_{NLoS}(\phi_{l,m})(t)\kappa\beta_0 d_{l,m}^{-\alpha}(t) \end{aligned} \tag{11}$$

UAV l detects N_l IoT nodes for communication using a directional antenna with a variable beamwidth. A data connection link is formed after each of the N_l IoT nodes has been detected, and UAV l then starts collecting data from IoT nodes. The achievable data transmission rate between IoT node m and UAV l at time step t, denoted as $\mathcal{K}_{l,m}(t)$, is calculated as follows:

$$\mathcal{K}_{l,m}(t) = B_{l,m} \log\left(1 + \frac{\rho_{l,m}(t) G g_{l,m}(t)}{\sigma^2}\right) \tag{12}$$

where $B_{l,m}$ is the channel bandwidth of the link between UAV l and IoT node m, $\rho_{l,m}(t)$ is the transmitted power of the IoT node at time step t, G is the antenna power gain of the IoT node to the flying UAV link, and σ^2 is the noise variance. We define Γ as $\frac{\rho_{l,m}(t) G g_{l,m}(t)}{\sigma^2}$, which is signal-to-noise-ratio.

2) *UAV-to-HAP channel model:* UAVs fly at higher altitudes, ensuring LoS conditions between UAVs and the HAP for transferring their data. The distance between UAV l and HAP h at time step t, denoted as $d_{h,l}(t)$, is:

$$\begin{aligned} d_{h,l}(t) = \\ \sqrt{(x_h(t) - x_l(t))^2 + (y_h(t) - y_l(t))^2 + (H_h - H_l)^2} \end{aligned} \tag{13}$$

Channel gain from UAV l to HAP h at time step t, denoted as $g_{h,l}(t)$, uses the formula:

$$g_{h,l}(t) = \beta_0 \times d_{h,l}^{-\alpha}(t) \tag{14}$$

where β_0 is the average channel power gain. According to [22], the achievable data transmission rate from UAV $l \in \mathcal{L}$ to HAP h at time step t is:

$$\mathcal{K}_{h,l}(t) = B_{h,l} \log\left(1 + \frac{\rho_{h,l}(t) g_{h,l}(t) \tilde{c}}{4\pi f \mu_h \varsigma_B}\right) \tag{15}$$

where $B_{h,l}$ represents the bandwidth of the communication link between UAV l and HAP h, $\rho_{h,l}(t)$ is the transmission power of UAV l, $\tilde{c}$ is the speed of light, and f is the carrier frequency. μ_h denotes the system noise temperature, and ς_B represents Boltzmann's constant.

III. Problem Formulation

In the context of optimal trajectory planning in a centralized system, the given problem is formulated into two objective functions to be achieved by the HAP. The first sub-problem aims to maximize the total coverage of IoT nodes within the flight time T of all UAVs. For instance, for serving IoT node m in a given coverage, UAV l should collect its data D_m entirely within a specified time constraint. We introduce the binary variable $\mathcal{Y}_{m,l} \in [0,1]$, where $m \in \mathcal{M}$ and $l \in \mathcal{L}$, which equals 1 if UAV l can successfully serve IoT node m and 0 otherwise. The formulated optimization problem, with the goal of maximizing the overall served IoT nodes shown in Eq. (16a), is expressed as follows:

$$\max \quad \sum_{l \in \mathcal{L}} \sum_{m \in \mathcal{M}} \mathcal{Y}_{m,l} \tag{16}$$

$$\text{s.t.} \quad \tilde{\mathfrak{J}}(\hat{i}^l) \leq \mathcal{E}_l^{max}, \forall l \in \mathcal{L}, \forall m \in \mathcal{M} \tag{17}$$

$$\widehat{\mathcal{S}}_m \leq \widehat{\mathcal{S}}_{max}, \forall m \in \mathcal{M} \tag{18}$$

$$\mathcal{Z}_l = \mathcal{P}_l^{\text{final}}, \forall l \in \mathcal{L} \tag{19}$$

$$\mathcal{Y}_{m,l} \in \{0,1\}, \forall m \in \mathcal{M}, \forall l \in \mathcal{L} \tag{20}$$

$$\sum_{l,l' \in \mathcal{L}} |\mathcal{Z}_l(t) - \mathcal{Z}_{l'}(t)| > D_{min}, \forall l \in \mathcal{L}, l \neq l', \forall t \in \mathcal{T} \tag{21}$$

Constraint (17) ensures that the energy consumed by UAV l when following trajectory $\hat{i}^l$ remains below the maximum available energy of UAV l. Constraint (18) ensures that each served IoT node m uploads its data $\widehat{\mathcal{S}}_m$ within the specified serving time, $\widehat{\mathcal{S}}_{\text{max}}$. Constraint (19) specifies the desired final location of UAV l. Constraint (20) guarantees that $\mathcal{Y}_{m,l}$ can be either 0 or 1 at time t. Constraint (21) guarantees that the absolute difference in positions between UAV l and all other UAVs l' at time t should be greater than the minimum distance D_{min}.

The second sub-problem, using the output of the first sub-problem as its input, aims to minimize the total energy consumed by UAVs, which is given as follows:

$$\min \quad \sum_{l \in \mathcal{L}} \sum_{t \in \mathcal{T}} \mathcal{E}_l^{tot}(t) \tag{22}$$

$$\text{s.t.} \quad (17), (20), (21)$$

$$\sum_{l \in \mathcal{L}} \sum_{m \in \mathcal{M}} \mathcal{Y}_{m,l} \geq U_l \tag{23}$$

$$\mathcal{E}_l^{tot}(t) \leq \mathcal{E}_l^{max}, \forall l \in \mathcal{L}, \forall t \in \mathcal{T} \tag{24}$$

Constraint (23) guarantees that during its mission, the UAV l must cover a minimum number of U_l IoT nodes, and constraint (24) ensures that the energy consumption of the UAV l at time t remains below the maximum power limit.

Solving these optimization problems using traditional methods leads to serious scalability issues and requires intensive computation due to the dynamic nature of this large network and the complexity of its dimensions. Fortunately, DRL has the capability to explore a vast state space and achieve efficient trajectory and energy management through its formidable data processing capabilities.

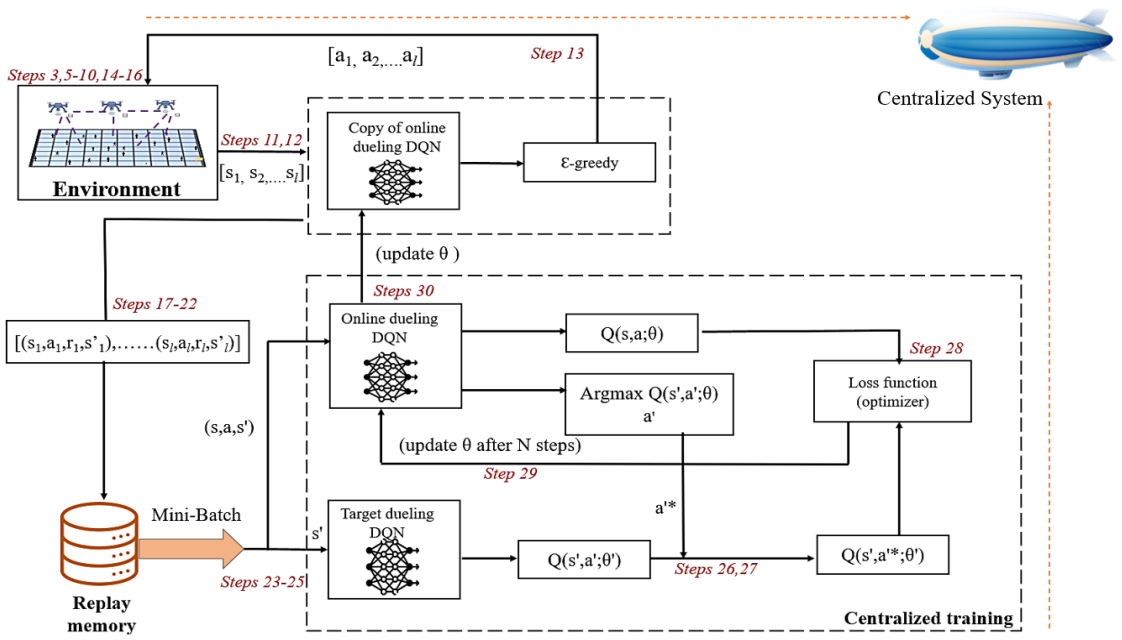


Figure 3: D3QN-based CoMETD Framework.

IV. Proposed D3QN-based CoMETD Scheme

This section introduces the centralized collaborative trajectory planning algorithm, CoMETD, as previously mentioned. The proposed CoMETD algorithm employs the powerful D3QN at HAP to optimize UAV trajectories, ensuring efficient data collection from terrestrial IoT nodes under energy constraints. All UAVs collaborate and share information at the HAP, storing state-action pair transitions to achieve globally optimal solutions and enhanced system efficiency.

The proposed CoMETD framework leverages the dueling double deep Q-network (D3QN) algorithm to address the challenges inherent in dynamic, multi-agent environments like NTNs. as shown in Fig. 3. D3QN combines the advantages of dueling architecture and double Q-learning, making it particularly effective for resource-constrained scenarios. The dueling architecture separates the estimation of state values and action advantages, allowing the algorithm to differentiate the intrinsic importance of states from the effect of specific actions. This results in more precise decision-making and enhanced learning efficiency. Additionally, D3QN employs two independent Q-networks: one for action selection and the other for evaluation. This separation mitigates the Q-value overestimation issue commonly associated with traditional DQN approaches, thereby improving stability and value estimation accuracy. These features make D3QN well-suited for applications requiring adaptive and robust learning mechanisms, such as the optimization of UAV trajectories in non-terrestrial networks [23], [24].

A. DRL-based Problem Formulation

In the proposed DRL-based formulation, the D3QN-based CoMETD algorithm is executed within the HAP to allocate trajectories that maximize user coverage while adhering to energy constraints. We consider UAVs as agents that interact with the system environment in a sequence of discrete time instances. The HAP takes on the role of executing the D3QN algorithm and storing the UAVs' experiences. It communicates with and controls the trajectories of all UAVs, ensuring effective coordination and optimization across the system.

The state and action of the proposed D3QN-based CoMETD problem are given as follows:

1) **State Space**: The coverage range of the UAVs can vary based on their beamwidth parameters, enabling them to encompass multiple cells. The state $s(t)$ at time t is represented as a four-element tuple, i.e., $s(t) = (\psi, \mathcal{Z}_l(t), \eta_l(t), \varphi_l(t))$, where ψ is the grid in which UAVs collect data and plan their trajectories, $\mathcal{Z}_l(t)$ is the cell that UAV l is located at time slot t, $\eta_l(t)$ is the set of cells within the range of UAV l at time t that is subject to the beamwidth parameters, and $\varphi_l(t)$ is the remaining energy of UAV l at time t.
2) **Action Space**: An agent may take one of the five moving actions at each state, denoted as $A_l = \{+x, +y, -x, -y, 0\}$ to represent the action, where $-y$, $+y$, $-x$, or $+x$ indicates that UAV l changes its state by moving downwards, upwards, to the right, or to the left, respectively. Meanwhile, '0' represents UAV l hovering in place for data collection. However, in a centralized system, the HAP provides actions in an array that includes all UAVs' actions at time step t, denoted as $A(t) = \{a_1, a_2, \ldots, a_l\}$, where $l \in \mathcal{L}$.
3) **Reward Function**: The main objective of the proposed D3QN-based CoMETD problem is to maximize the expected reward achieved by UAV l when completing a mission from its initial to the final position. The trajectory reward, i.e., $R1_l(t)$, is given as:

Algorithm 1: CoMETD algorithm

1 **Input:** Initialize environment; current network parameter; replay memory $\mathcal{D}$, epsilon probability $\epsilon \in [0,1]$, learning rate $\alpha \in [0,1]$, discount factor $\gamma \in [0,1]$, θ and the target network $\tilde{\theta}$;
2 Initialize main network $Q_1(s(t), a(t), \theta)$ with weights θ and the target network $Q_2(s(t), a(t), \tilde{\theta})$ with weights $\tilde{\theta}$;
3 **Output:** The optimal policy
4 $\mathcal{A} \leftarrow$ sample_Action_Space();
5 **while** *episode* $\leq$ *Episodes* **do**
6 **Foreach** UAV l
7 $\delta \leftarrow$ reset_Environment();
8 $t \leftarrow 0$;
9 **while** ($\mathcal{Z}_l \neq \mathcal{P}_l^{final}$) **do**
10 $\sigma \leftarrow$ random_Sample([0,1]);
11 $s(t) \leftarrow$ observe_State(δ);
12 Select action:
$\begin{cases} a(t) \leftarrow argmax Q(s(t), a(t), \theta), & \text{if } \sigma \geq \epsilon \\ a(t) \leftarrow random\_Action(\mathcal{A}), & \text{Otherwise.} \end{cases}$
13 **if** *UAV l collects data from IoT node m* **then**
14 mark m as collected
15 **end if**
16 $\mathcal{R}_l(t) \leftarrow$ obtain_Reward($a(t)$);
17 $s_l(t+1) \leftarrow$ observe_NewState($a(t)$);
18 Checks safety distance d_{min};
19 **if** $|\mathcal{Z}_l(t)$ - $\mathcal{Z}_{l'}(t)| < D_{min}$ **then**
20 $s_l(t+1) = s(t)$
21 **end if**
22 Observe $s(t)$ and adjust beamwidth;
23 Store the transition tuple $(s(t), a(t), r(t), s(t+1))$ in common $\mathcal{D}$;
24 Sample mini batch of B_m tuples;
25 Calculate target;
26 $Y(t) = R(t+1) + \gamma Q_{2,\tilde{\theta}}(s(t+1), \underset{a(t+1)}{\text{argmax}}\, Q_{1,\theta}(s(t+1), a(t+1)))$, Perform the gradient decent step to minimize loss function;
27 Calculate the loss $L(\theta) = \mathbb{E}[(Q_{1,\theta(}s(t), a(t))) - Y(t)r]$;
28 Soft update of target parameters,
29 $\tilde{\theta} = (1-\tilde{x})\tilde{\theta} + \tilde{x}\theta$ (update factor $\tilde{x}$ = [0,1]);
30 episode = episode + 1
31 **end while**
32 **end while**

$$R1_l(t) = \begin{cases} +\chi_1, & \text{if } \mathcal{Z}_l = \mathcal{P}_l^{\text{final}} \\ -\chi_2, & \text{if } \mathcal{Z}_l(t) - \mathcal{Z}_{l'}(t) < D_{min} \\ -1, & \text{otherwise} \end{cases} \quad (25)$$

where reward χ_1 is obtained when UAV l successfully reaches the final destination, while penalty χ_2 is incurred for violating the safety distance. Additionally, a negative penalty of -1 is assigned for each step taken without completing the mission. By utilizing the problems (16), (22), (12), and (25), we design a reward function incorporating parameters ξ and ζ, motivating the agents to maximize their rewards within the range $[0, 1)$. This reward function motivates UAVs to maximize their rewards by efficiently serving IoT nodes while minimizing energy consumption. The overall reward for UAV l at time t, i.e., $R_l(t)$, is expressed as follows:

$$R_l(t) = \xi \frac{\mathcal{Y}_{m,l}}{\mathcal{E}_l^{tot}(t)} + \mathcal{K}_{l,m}(t) + \zeta R1_l(t) \quad (26)$$

where $\mathcal{Y}_{m,l}$ and $\mathcal{E}_l^{tot}(t)$ are defined based on the problems (16) and (22), respectively. The total reward achieved by UAV l upon completion of the mission can be formulated as follows:

$$\mathcal{R}_l = \xi \frac{\mathcal{Y}}{\mathcal{E}_l} + \mathcal{K}_l + \zeta R1_l \quad (27)$$

where $\mathcal{Y} = \sum_{l=1}^{\mathcal{L}} \sum_{m=1}^{\mathcal{M}} \mathcal{Y}_{m,l}$, $\mathcal{E}_l = \sum_{t=0}^{T} \mathcal{E}_l^{tot}(t)$, $\mathcal{K}_l = \sum_{t=0}^{T} K(t)_{l,m}$, and $R1_l = \sum_{t=1}^{T} R1_l(t)$. The total reward of the episode including all UAVs can be calculated as:

$$\mathcal{R}_{total} = \sum_{l=1}^{\mathcal{L}} \mathcal{R}_l \quad (28)$$

B. Propose D3QN-based CoMETD Algorithm

We introduce the novel D3QN-based CoMETD algorithm designed to address the DRL-based formulation within the framework illustrated in Fig. 3. The algorithm is detailed in **Algorithm** 1 and introduced in the following.

During the training phase, we first initialize the replay memory $\mathcal{D}$ and set parameters such as learning rate α, discount factor γ, and epsilon probability ϵ. Then, we initialize the evaluation and target networks, as well as other necessary parameters (steps 1-2), and the output is the optimal policy π^* (step 3). In step 4, an action space is generated. During each training episode, every UAV navigates within the AoI to provide communication to the IoT nodes and reach the final destination. Notably, the environment is reset at the start of every episode (step 7). At each time step t, each UAV independently observes the environment and, following the ϵ-greedy policy, either randomly selects an action with a probability of ϵ or chooses the action with the maximum Q-value, continuing this process until they reach their final destination $\mathcal{P}_l^{\text{final}}$ (as described in steps 8–12).

After executing the chosen action and collecting data from the IoT node m, the UAV marks the IoT node m as collected (steps 13–14). The UAV receives a reward $R_l(t)$ based on Eq. (26) and observes the new state. If the UAV violates the safety distance D_{min}, it will remain in the same state and get a negative reward. It also adjusts the beamwidth according to the section (II-A) (steps 15–22). In step 23, the transition tuples $(s(t), a(t), r(t), s(t+1))$ are stored in a common replay memory $\mathcal{D}$ in HAP. For training the evaluation network θ, a mini-batch of tuples B_m is randomly sampled from the replay memory $\mathcal{D}$ in step 24. The evaluation network θ is fed into the optimizer, which calculates the loss function between the target $Y(t)$ and the estimated experiences $(s(t), a(t), r(t), s(t+1))$ as shown in Fig. 3. Further, the optimizer minimizes the loss function over the mini-batch B_m using an equation and updates the parameters θ of the online dueling DQN, thereby training the network (as described in steps 25–30). Finally, the episode concludes when all UAVs reach their destination. Steps 7 to 30 are repeated for all episodes. Once the training

Parameters	Values
Number of UAVs	10, 20, 30
Number of HAP	1
Number of IoT nodes	100, 200, 300, 400, 500
v	25 m\s
B	1 MHz [19]
H_l	150 m
H_h	20 KM [17]
v_0	7.2 [19]
$\mathcal{P}_0$,$\mathcal{P}_1$	79.9 W , 88.6 W
μ_{tip}	200 m/s [19]
z_0	0.3 m^2 [19]
τ	0.05 [19]
κ,	1.225 kg/m^3 [19]
A	0.79 m^2 [19]
ϑ	80°- 140°
γ	0.8
α	0.01
ϵ, ϵ_{min}, ϵ_{decay}	1.0, 0.01, 0.0005
Optimizer	Adam
Mini-batch size	128
Replay memory size	5000

Table II: Simulation parameters

is completed, a policy is obtained with a well-trained DNN that allows HAP to manage all UAVs to navigate in real-time environments.

V. Simulation Results

In the simulation environment, we consider 2000 $\times$ 2000 m^2 area size, and a set of IoT nodes are randomly distributed and moving throughout the area. The simulation parameters specified in Table II are employed to train the DRL models. The experimental parameters for DRL-based approaches, such as α, ϵ, and γ, are fine-tuned through an iterative process of trial and error. Various parameter values are tested and evaluated to determine the optimal settings. The proposed D3QN-based CoMETD includes dueling DQN by introducing two separate neural networks: the online network and the target network, which are presented in Fig. 3. We compare the proposed D3QN-based CoMETD algorithm with the benchmark algorithm as follows:

1) **D3QN without HAP:** To show the benefits of integrating HAP in the proposed system, the D3QN without HAP scheme is implemented and compared. This approach solely utilizes UAVs for data collection and trajectory design, excluding the involvement of HAP. The UAVs work individually and optimize their trajectories using the D3QN algorithms.
2) **DQN [15]:** In the DQN-based approach, both UAVs and HAP are utilized to serve the IoT nodes as the proposed framework. The trajectory of the UAVs is optimized using the DQN technique.
3) **DQN without HAP [5]:** This scheme operates without the use of HAP, relying solely on UAVs to serve the IoT nodes. Each UAV works independently and optimizes its trajectory using the DQN technique.

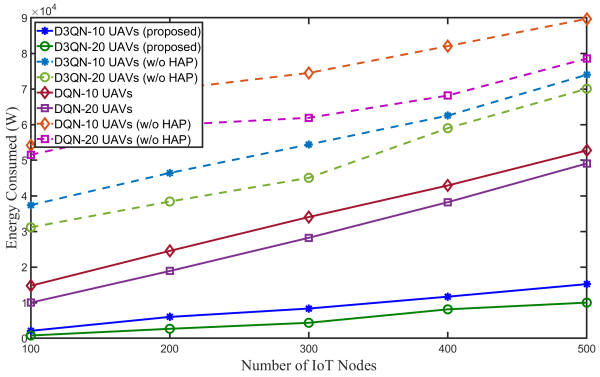


Figure 4: Comparison of energy consumption under different numbers of IoT nodes.

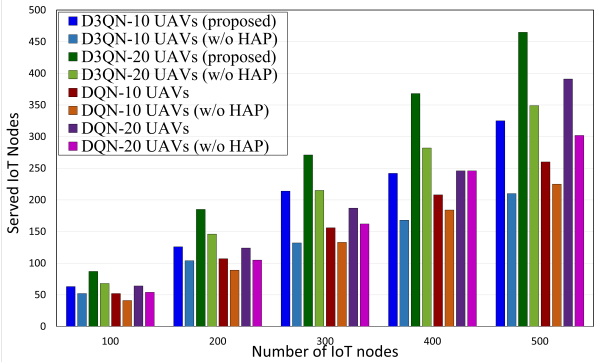


Figure 5: Successfully served IoT nodes with different numbers of IoT nodes.

Fig. 4 illustrates the impact of the number of IoT nodes and the number of UAVs on the total consumed energy by UAVs to complete the data collection mission. The graph reveals that as the number of IoT nodes increases, energy consumption also increases. This can be attributed to the fact that with more IoT nodes, the UAVs need to collect data from a larger number of nodes, resulting in higher energy consumption. The proposed D3QN-based CoMETD with 20 UAVs algorithm outperforms other benchmark algorithms. By increasing the number of UAVs, the collaborative centralized system exhibits improved performance due to the shared experiences among the UAVs. On the contrary, the proposed D3QN-based CoMETD approach without HAP (w/o HAP), specifically with 10 and 20 UAVs, demonstrates significantly higher energy consumption. In contrast, the proposed D3QN-based CoMETD approach without HAP, specifically with 10 and 20 UAVs, demonstrates significantly higher energy consumption. This is because they are learning independently without a centralized system. Similarly, both the DQN-based approach with HAP and the DQN-based approach without HAP consume at least 30% to 50% more energy when compared to our proposed D3QN-based CoMETD approach.

In Fig. 5, we examine the number of served IoT nodes in relation to different numbers of IoT nodes on the ground. The graph includes results from the D3QN-based CoMETD (with and without HAP), and DQN (with and without HAP) approaches with 10 UAVs and 20 UAVs. It is evident that

	Served Iot Nodes	Energy Saved
10 UAVs	65	34.8
20 UAVs	93	42.7
30 UAVs	94.4	45.4

Table III: System utilization rate.

the proposed D3QN-based CoMETD approach with 20 UAVs outperforms the other approaches, demonstrating significantly better results. However, D3QN-based CoMETD (both with 10 and 20 UAVs) covered approximately 20% more IoT nodes than the D3QN-based CoMETD without HAP.

The experimental results in Table. III demonstrate the overall utilization of the system. The graph illustrates the resource utilization when employing different numbers of UAVs. When using 10 UAVs, approximately 35% of energy is saved while serving 65% of the IoT nodes. Increasing the number of UAVs to 20 results in serving 93% of the IoT nodes and saving 43% more energy compared to 10 UAVs. However, when further increasing the number of UAVs to 30, there is no significant difference in serving IoT nodes, but approximately 45% of energy is saved. Therefore, these findings lead to the conclusion that using 20 UAVs in this scenario is an optimal choice, as it achieves a high IoT node serving rate while also saving a considerable amount of energy.

Fig. 6 presents the achieved data rate concerning the number of episodes having 500 IoT nodes and 20 UAVs with adaptive beamwidths in the environment setup. The data rate consistently increases as the learning iteration progresses, indicating the effectiveness of the learning algorithm as it learns to serve more IoT nodes. The proposed D3QN-based CoMETD and DQN exhibited superior performance compared to the versions of these algorithms that did not incorporate HAP in terms of achieving data rate. However, the proposed D3QN-based CoMETD method exhibits greater robustness compared to the other approaches considered. Moreover, it illustrates the impact of the achieved data rate based on the adaptive and fixed beamwidths utilized by UAVs. As the number of episodes increases, it can be observed that the achieved data rate also increases for both approaches. However, the adaptive beamwidth approach consistently outperforms the fixed beamwidth approach, demonstrating higher data rates across all episode values. This can be attributed to the adaptive beamwidths' ability to dynamically adjust and optimize the communication parameters based on the network conditions. These findings highlight the significance of incorporating adaptive beamwidth techniques for improving the data rate in wireless communication systems.

Fig. 7 shows the efficacy of each scheme DQN (with and without HAP) and proposed D3QN-based CoMETD (with and without HAP) with 500 IoT nodes and 20 UAVs in the environment setup. The graph illustrates the performance of each scheme in terms of efficacy, which is measured as the ratio of the number of IoT nodes served to the number of steps taken by the respective scheme. It is evident that the proposed D3QN-based CoMETD algorithm outperformed other schemes in terms of efficacy, serving the highest percentage

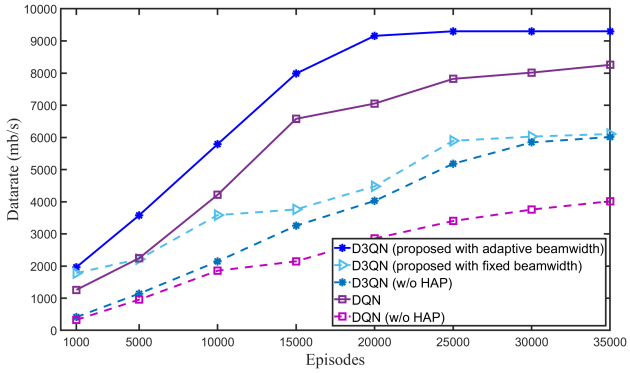


Figure 6: Achieved data rate versus number of episodes in a scenario with 500 IoT nodes and 20 UAVs.

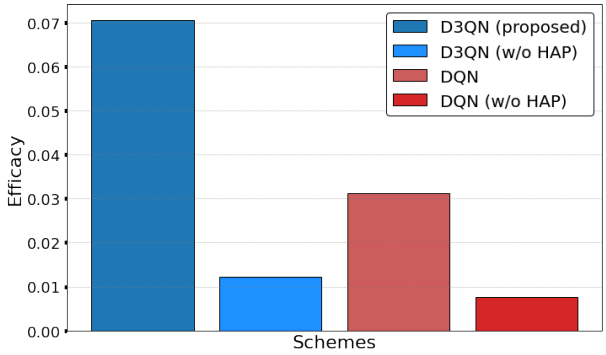


Figure 7: Comparison of efficacy measured as the ratio of the number of IoT nodes served to the number of steps taken by the respective scheme in a scenario with 500 IoT nodes and 20 UAVs.

of IoT nodes per step taken.

Fig. 8 illustrates the average reward plotted against the number of episodes, providing an evaluation of the performance of all the methods with 500 IoT nodes and 20 UAVs in the environment setup. As the number of episodes increases, the reward also increases, indicating improved performance, and the proposed scheme converges faster, within 10,000 episodes, outpacing other baseline methods. Notably, the D3QN-based CoMETD algorithm outperforms the benchmark algorithms as it combines the benefits of both the dueling and double DQN architectures. This combination contributes to higher rewards and enhanced performance when compared to the benchmark. The average reward shown in Fig. 8 is computed as the cumulative reward achieved by the UAVs during each episode. To reduce noise and better visualize the learning trend, a sliding average with a window size of 150 episodes was applied. The graph represents the smoothed reward progression, highlighting the algorithm's ability to learn and improve over time.

In a simulated 16x16 grid with 25 IoT nodes and 3 UAVs, Fig. 9 provides a visual representation. Fig. 9a shows trajectories generated using the DRL without the HAP approach. Here, the UAVs navigate through numerous steps to complete their trajectory, leaving roughly four IoT nodes isolated. This outcome demonstrates the consequences of UAVs independently

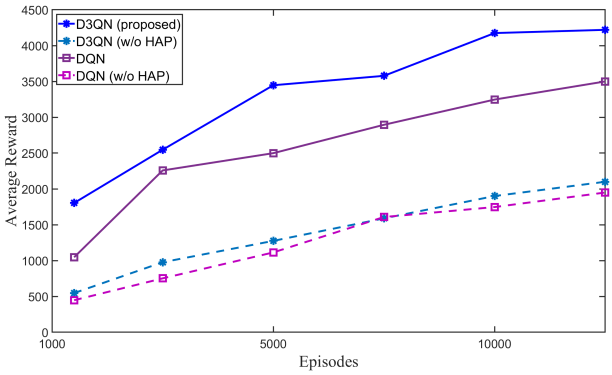
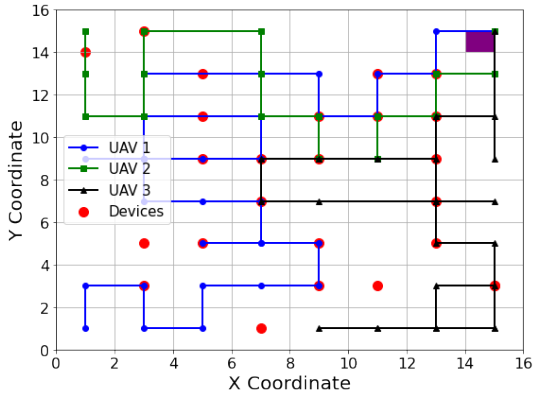
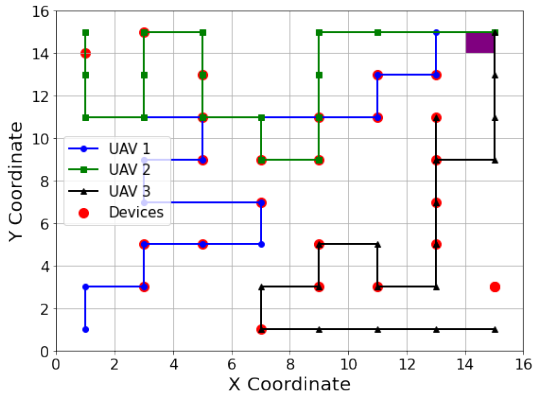


Figure 8: Average reward per scheme versus number of episodes.



(a) DRL (w/o HAP)



(b) DRL (Proposed)

Figure 9: UAVs' Trajectories in a 16x16 Grid with 25 IoT Nodes. red circles are IoT nodes and a purple square is the final location.

	10x10 Cells	20x20 Cells
ILP Model	3419.22	4345.47
D3QN-based CoMETD	4045.75	4936.15

Table IV: Comparative analysis of energy consumption (W): CoMETD versus ILP Scheme.

executing DRL, which require more steps to complete the mission. Conversely, Fig. 9b shows the proposed CoMETD-based DRL. This optimized trajectory requires fewer steps, resulting in a marginal exclusion of only two IoT nodes. These figures distinctly emphasize the superiority of the CoMETD-based DRL method with a centralized system over the independent operational approach.

In this study, an energy efficiency comparison was conducted between our proposed CoMETD scheme and the ILP model. Table. IV presents the outcomes derived from two distinct scenarios within an area of 200 m^2. The first scenario entails the subdivision of the area into a 10×10 cell configuration accommodating 60 IoT nodes, while the second scenario involves a 20×20 cell configuration accommodating 110 IoT nodes. In the 10×10 cell scenario, the ILP model exhibited an energy consumption of 3419.22 W, contrasting with the CoMETD scheme, which registered consumption of 4245.75 W. Similarly, in the 20×20 cell scenario, the ILP model demonstrated an energy consumption of 4545.47 W, whereas the CoMETD scheme exhibited a consumption of 4936.15 W. These results confirm that the CoMETD scheme exhibits a notable ability to approximate optimal performance, as indicated by its close proximity to the energy consumption levels achieved by the ILP model.

VI. Conclusions

In this paper, we introduced the collaborative multi-agent energy-efficient trajectory design and data collection (CoMETD) algorithm to solve the data collection problem for multiple UAVs under the centralized control of a HAP. The CoMETD algorithm utilized the D3QN architecture, enabling efficient real-time learning and decision-making without prior knowledge of the network environment. By leveraging a centralized system with shared memory at the HAP, the CoMETD algorithm maximized energy efficiency and coverage of IoT nodes by the multiple UAVs through collaborative learning. Extensive simulations proved the proposed D3QN-based CoMETD algorithm's superiority compared to its state-of-the-art counterparts in terms of achieved data rate, coverage, and energy efficiency. These findings highlighted the potential of the CoMETD algorithm for optimizing trajectory and data collection in centralized multi-UAV systems.

VII. Acknowledgement

This work was supported by the Brain Pool program funded by the Ministry of Science and ICT through the National Research Foundation of Korea (grant number: NRF-2022H1D3A2A01063679) and by the 2023 Kyungpook National University BK21 FOUR Graduate Innovation Project (International Joint Research Project for Graduate Students).

REFERENCES

[1] H. Mei and L. Peng, "On multi-robot data collection and offloading for space-aerial-surface computing," *IEEE Wireless Communications*, vol. 30, no. 2, pp. 90–96, 2023.

[2] M. Li, N. Cheng, J. Gao, Y. Wang, L. Zhao, and X. Shen, "Energy-efficient uav-assisted mobile edge computing: resource allocation and trajectory optimization," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 3424–3438, 2020.

[3] G. K. Kurt, M. G. Khoshkholgh, S. Alfattani, A. Ibrahim, T. S. Darwish, M. S. Alam, H. Yanikomeroglu, and A. Yongacoglu, "A vision and framework for the high altitude platform station (haps) networks of the future," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 729–779, 2021.

[4] S. Rahim and L. Peng, "Intelligent space-air-ground collaborative computing networks," *IEEE Internet of Things Magazine*, vol. 6, no. 2, pp. 76–80, 2023.

[5] S. Rahim, M. M. Razaq, S. Y. Chang, and L. Peng, "A reinforcement learning-based path planning for collaborative uavs," in *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, pp. 1938–1943, 2022.

[6] Q. Wu, Y. Zeng, and R. Zhang, "Joint trajectory and communication design for multi-uav enabled wireless networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 2109–2121, 2018.

[7] B. Zhu, E. Bedeer, H. H. Nguyen, R. Barton, and J. Henry, "Joint cluster head selection and trajectory planning in uav-aided iot networks by reinforcement learning with sequential model," *IEEE Internet of Things Journal*, 2021.

[8] S. Rahim, L. Peng, S. Chang, and P.-H. Ho, "On collaborative multi-uav trajectory planning for data collection," *Journal of Communications and Networks*, vol. 25, no. 6, pp. 722–733, 2023.

[9] O. Esrafilian, R. Gangula, and D. Gesbert, "Learning to communicate in uav-aided wireless networks: Map-based approaches," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1791–1802, 2019.

[10] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3-d placement of an unmanned aerial vehicle base station (uav-bs) for energy-efficient maximal coverage," *IEEE Wireless Communications Letters*, vol. 6, no. 4, pp. 434–437, 2017.

[11] Y. Yuan, L. Lei, T. X. Vu, S. Chatzinotas, and B. Ottersten, "Actor-critic deep reinforcement learning for energy minimization in uav-aided networks," in *2020 European Conference on Networks and Communications (EuCNC)*, pp. 348–352, 2020.

[12] H. Ahmadinejad and A. Falahati, "Forming a two-tier heterogeneous air-network via combination of high and low altitude platforms," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 2, pp. 1989–2001, 2021.

[13] A. Masood, T.-V. Nguyen, T. P. Truong, and S. Cho, "Content caching in hap-assisted multi-uav networks using hierarchical federated learning," in *2021 International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 1160–1162, 2021.

[14] D. T. Hua, D. S. Lakew, and S. Cho, "Drl-based energy efficient communication coverage control in hierarchical hap-lap network," in *2022 international conference on information networking (ICOIN)*, pp. 359–362, IEEE, 2022.

[15] A. H. Arani, P. Hu, and Y. Zhu, "Haps-uav-enabled heterogeneous networks: A deep reinforcement learning approach," *arXiv preprint arXiv:2303.12883*, 2023.

[16] B. Jabbari, Y. Zhou, and F. Hillier, "Simple random walk models for wireless terminal movements," in *1999 IEEE 49th Vehicular Technology Conference (Cat. No. 99CH36363)*, vol. 3, pp. 1784–1788, IEEE, 1999.

[17] H. He, S. Zhang, Y. Zeng, and R. Zhang, "Joint altitude and beamwidth optimization for uav-enabled multi-user communications," *IEEE Communications Letters*, vol. 22, no. 2, pp. 344–347, 2017.

[18] Q. Ren, O. Abbasi, G. K. Kurt, H. Yanikomeroglu, and J. Chen, "Caching and computation offloading in high altitude platform station (haps) assisted intelligent transportation systems," *IEEE Transactions on Wireless Communications*, vol. 21, no. 11, pp. 9010–9024, 2022.

[19] S. S. Khodaparast, X. Lu, P. Wang, and U. T. Nguyen, "Deep reinforcement learning based energy efficient multi-uav data collection for iot networks," *IEEE Open Journal of Vehicular Technology*, vol. 2, pp. 249–260, 2021.

[20] C. You and R. Zhang, "3d trajectory optimization in rician fading for uav-enabled data harvesting," *IEEE Transactions on Wireless Communications*, vol. 18, no. 6, pp. 3192–3207, 2019.

[21] H. Cao, G. Yu, and Z. Chen, "Cooperative task offloading and dispatching optimization for large-scale users via uavs and hap," in *2023 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, IEEE, 2023.

[22] Z. Jia, M. Sheng, J. Li, D. Niyato, and Z. Han, "Leo-satellite-assisted uav: Joint trajectory and data collection for internet of remote things in 6g aerial access networks," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9814–9826, 2020.

[23] J. Ren, Z. Chai, and Z. Chen, "Joint spectrum allocation and power control in vehicular communications based on dueling double dqn," *Vehicular Communications*, vol. 38, p. 100543, 2022.

[24] D. J. Birabwa, D. Ramotsoela, and N. Ventura, "Multi-agent deep reinforcement learning for user association and resource allocation in integrated terrestrial and non-terrestrial networks," *Computer Networks*, vol. 231, p. 109827, 2023.

[25] "The promise and challenges of airborne wind energy." https://physicsworld.com/a/thepromise-and-challenges-of-airborne-wind-energy/, 2022-04-26.