Apollo Journals
Engineering and Computing Insights
ECI
Vol. 01 Issue 01 2026

Article

A Communication-Aware Multi-Agent Deep Reinforcement Learning Framework for UAV-Swarm Relay Deployment via 3D Trajectory Control

Shanshan He

Department of Communication Engineering, Jilin University, Changchun 130012, China; Email: hess2025@mails.jlu.edu.cn

Abstract: Unmanned aerial vehicle (UAV) swarms can restore communication after infrastructure failure and extend low-altitude Internet of Things (IoT) services, but trajectory decisions interact with access links, backhaul bottlenecks, data queues, battery use, service fairness, and flight constraints. This paper formulates these couplings as a communication-aware constrained Markov game and implements a light-weight communication-aware multi-agent dueling double DQN (CA-MAD3QN). The implemented learner controls only discrete three-dimensional UAV motion; user association, bandwidth, access power, and the MEC term are fixed or rule based and are not claimed as learned optimization variables. Backhaul capacity is propagated to every user through an explicit service-scaling factor before queue evolution. The revised numerical study compares CA-MAD3QN with discrete MAPPO, vanilla DQN, one-step model-predictive optimization, greedy tracking, and static deployment over three independent training seeds and reports 95% confidence intervals. Under the stress setting, CA-MAD3QN achieves an episode utility of 85.36 ± 28.94 (95% CI); the evidence does not establish universal superiority over the competitive baselines, so no such claim is made. In the two-UAV testbed, an exact joint-action projection eliminates modeled realized violations; disabling that projection for the same trained policy yields 0.863 violations per slot on average. Preliminary small-scale tests with two to four UAVs show increasing aggregate throughput but also expose projection and training-variance limitations. The results are therefore presented as an auditable proof of concept rather than hardware validation.

Keywords: UAV Swarm; Communication Relay; Three-Dimensional Trajectory Optimization; Multi-Agent Deep Reinforcement Learning; Mobile Edge Computing; Safety Projection; Federated Learning

Received: 6 June 2026
Revised: 15 September 2026
Accepted: 10 November 2026
Published: 10 November 2026

Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

1.1. Motivation and Scope

UAVs are increasingly considered mobile network nodes rather than flying antennas. Their altitude and mobility can improve line-of-sight probability and coverage, while their limited batteries, backhaul dependence, and safety constraints make deployment fundamentally different from terrestrial base-station placement [1]–[5]. In emergency communication, several UAVs may need to restore service over damaged areas; in low-altitude IoT, they collect bursty traffic from moving or sparsely distributed devices; and in UAV-assisted mobile edge computing (MEC), they must coordinate communication and computation decisions. These applications align with the integration of communication, sensing, computing, artificial intelligence, and resilience emphasized in IMT-2030 and non-terrestrial-network development [6]–[9].

The central difficulty is coupling. A higher altitude may improve line-of-sight probability but increase propulsion energy and interference. Flying closer to a user cluster can strengthen the access link while weakening the backhaul. Serving users with favorable channels maximizes instantaneous throughput but can leave remote or deadline-sensitive users unserved. Static optimization can solve a snapshot accurately, yet repeated recomputation becomes expensive when users, traffic, weather, and obstacles change. Deep reinforcement learning (DRL) can transfer part of this cost to offline training and return an online policy, but a useful policy must be communication-aware, safe, and evaluated against transparent physical models.

1.2. Contributions and Organization

This study develops and evaluates a focused engineering proof of concept. Its main contributions are as follows:

- A formal multi-layer model links probabilistic air-to-ground access, backhaul capacity, effective per-user service, queue evolution, battery-aware discrete-action energy, and flight constraints; the same service quantity is used consistently in throughput and queue updates.
- The implemented decision scope is stated explicitly: CA-MAD3QN optimizes only discrete 3D UAV motion in the proof-of-concept, while association, bandwidth, transmit power, and MEC quantities follow fixed or rule-based mechanisms.
- A communication-aware multi-agent dueling double DQN (CA-MAD3QN) combines parameter sharing, double-Q targets, a dueling head, mobility-prediction features, and a joint action-projection layer; none of these standard components is claimed as individually novel.
- A reproducible multi-seed evaluation adds discrete MAPPO and a dynamic one-step model-predictive baseline, reports 95% confidence intervals, separates five module ablations, and includes 2/3/4-UAV scalability tests. Unsupported superiority claims are removed.
- Engineering implications are drawn from safety-intervention burden, model calibration, fairness, and IMT-2030/NTN practice; federated model exchange is identified solely as a prospective architecture requiring further study.

The remainder of the paper is organized as follows. Section 2 reviews related work and identifies the research gap. Section 3 formulates the system and optimization problem. Section 4 presents CA-MAD3QN and a prospective federated architecture. Section 5 describes the experiments and results. Section 6 discusses deployment implications and limitations, and Section 7 concludes the paper.

2. Related Work and Research Gap

2.1. Conventional UAV Communication Optimization

Foundational work established the distinctive geometry of UAV communication and the altitude-coverage trade-off [1]–[4]. Probabilistic line-of-sight models make path loss depend on elevation angle, while trajectory optimization couples communication rates with kinematic feasibility. Convex approximation, block coordinate descent, and mixed-integer methods remain important because they expose structure and provide reliable benchmarks. Rotary-wing propulsion models further show that energy is a nonlinear function of speed, hovering, and vertical movement [5]. The limitation is not accuracy in a fixed scenario; it is the repeated optimization burden and sensitivity to model mismatch when users and channels evolve online.

2.2. DRL and Multi-Agent Coordination

DRL has been used for three-dimensional trajectory control, fair bandwidth allocation, task offloading, and resource management [10]–[12], [13]. Multi-agent formulations are particularly relevant when UAVs interact through interference, shared backhaul, collision constraints, and overlapping service regions. Centralized training with decentralized execution is widely used because global information can stabilize learning while each UAV executes with local observations [14], [15]. Value-based methods such as DQN [16] are attractive for discrete motion and association decisions, whereas actor-critic methods are more natural for continuous controls. The choice of algorithm is therefore secondary to the definition of physically meaningful state variables, constraints, and rewards.

2.3. Robustness, Safety, and Privacy

Three gaps recur in the literature. First, access-link rate, backhaul capacity, queue service, and fairness are often modeled at different abstraction levels, which can make the optimized objective inconsistent with the simulated queue dynamics. Second, reward penalties alone do not guarantee boundary, altitude, or inter-UAV separation constraints. Third, centralized training may collect sensitive user-location and traffic traces. Constrained multi-UAV tracking [17] demonstrates that safety constraints can be embedded in multi-agent control, while the recent safety-network framework in [18] learns centralized collision-avoidance filtering for decentralized execution. Those studies are primarily tracking and collision-avoidance studies; they do not propagate a communication backhaul bottleneck into per-user relay service and queue evolution. The present work therefore uses safety filtering as an enabling mechanism rather than claiming it as a new primitive.

https://doi.org/10.67728/eci.2026.003

2.4. Comparative Synthesis and Research Gap

The remaining gap addressed here is a reproducible communication-relay decision loop in which access rate, backhaul-limited per-user service, queue evolution, energy accounting, and flight feasibility are defined at the same level of abstraction. The novelty claim is deliberately integration-level: double Q-learning, dueling networks, MAPPO, federated averaging, and safety filtering are established techniques. The contribution is their communication-aware coupling and auditable implementation in a small UAV-relay testbed, together with explicit reporting of negative ablations, intervention burden, and validity limits (Table 1).

Table 1. Comparison of representative studies and the remaining gap. Source: Author.

Study	Scenario	Method	Coupled Layers	Privacy/Robustness	Main Limitation
Al-Hourani et al. [3]	Single-UAV placement	Probabilistic LoS altitude optimization	Channel/coverage	No	Static geometry; no queues or learning
Zeng et al. [5]	Single rotary-wing UAV	SCA and path discretization	Rate/propulsion energy	No	Accurate offline optimization; repeated recomputation
Ding et al. [11]	Single UAV, 3D	DDPG trajectory and band allocation	Energy/fairness	Limited	No multi-UAV coordination or hard projection
Wang et al. [10]	Multi-UAV MEC	MADDPG trajectory planning	Offloading/energy/fairness	Limited	No explicit backhaul and safety projection
Zhang et al. [19]	Multi-UAV network	DRL 3D trajectory	Coverage/trajectory	Limited	No privacy mechanism; simplified constraints
Bai et al. [12]	Multi-UAV survey	RL taxonomy	Broad	Discussed	Identifies gaps but does not instantiate a formal model
McMahan et al. [20]	Distributed learning	Federated averaging	Privacy/communication	Yes	Not UAV-specific and does not enforce flight constraints
Chen et al. [17]	Multi-UAV tracking	Constrained multi-agent RL	Trajectory/safety	Safety-aware	Tracking objective; no relay backhaul-to-queue propagation
Wang et al. [18]	UAV swarm tracking	MADRL + safety network	Tracking/collision	Safety network	No access/backhaul queue model or relay-resource evaluation
This work	Multi-UAV relay	CA-MAD3QN + joint projection	Access/backhaul/queues/battery/fairness	Explicit intervention logs	Discrete proof of concept; no claim of universal superiority

3. System Model and Problem Formulation

3.1. Network and UAV Mobility Model

Consider N UAVs serving K ground users over T discrete slots of duration Δt. UAV n has a three-dimensional position vector $\mathbf{q}_n(t) = \begin{bmatrix} x_n(t) \\ y_n(t) \\ h_n(t) \end{bmatrix}$, residual battery $b_n(t)$, and a backhaul link to a terrestrial or non-terrestrial gateway. User k has horizontal position $u_k(t)$, queue backlog $Q_k(t)$, and an optional MEC task descriptor (D_k, C_k, τ_k), representing input bits, CPU cycles, and deadline, respectively. The binary variable $a_{nk}(t)$ indicates user association.

$$\mathbf{q}_n(t+1) = \mathbf{q}_n(t) + \mathbf{v}_n(t)\Delta t, \quad \| \mathbf{v}_n(t) \|_2 \leq V_{\max} \tag{1}$$

$$h_{\min} \leq h_n(t) \leq h_{\max}, \quad \| \mathbf{q}_n(t) - \mathbf{q}_m(t) \|_2 \geq d_{\min}, \; n \neq m \tag{2}$$

The first constraint limits speed; the second imposes altitude and inter-UAV separation. Boundary, acceleration, no-fly-zone, and return-energy constraints can be added to the feasible action set $A_{feas(s_t)}$

https://doi.org/10.67728/eci.2026.003

3.2. Air-to-Ground Access and Backhaul Model

The elevation angle from UAV n to user k is $\theta_{nk}(t) = \arctan[h_n(t)/\|q_{n,xy}(t) - u_k(t)\|_2]$. Following the standard urban probabilistic line-of-sight (LoS) model [3], the LoS probability and average path loss are

$$p_{nk}^{\mathrm{LoS}}(t) = \frac{1}{1 + a\exp\left[-b\left(\frac{180\theta_{nk}(t)}{\pi} - a\right)\right]} \tag{3}$$

$$L_{nk}(t) = 20\log_{10}(4\pi f_c d_{nk}(t)/c) + p_{nk}^{\mathrm{LoS}}(t)\eta_L + \left(1 - p_{nk}^{\mathrm{LoS}}(t)\right)\eta_N + L_w(t) \tag{4}$$

where $d_{nk}(t)$ is the three-dimensional UAV–user distance, ηL and ηN denote excess LoS and NLoS losses, $Lw(t)$ is weather/blockage attenuation, and F is the receiver noise figure. The access rate uses per-user bandwidth $B_{nk}(t)$ and transmit power $p_{nk}(t)$, while the gateway link provides backhaul capacity.

$$r_{nk}(t) = a_{nk}(t)B_{nk}(t)\log_2\left(1 + \frac{p_{nk}(t)10^{-L_{nk}(t)/10}}{N_0 F B_{nk}(t)}\right) \tag{5}$$

$$C_n^{\mathrm{bh}}(t) = B_n^{\mathrm{bh}}\log_2\left(1 + \frac{P_n^{\mathrm{bh}} g_n^{\mathrm{bh}}(t)}{N_0 F B_n^{\mathrm{bh}}}\right) \tag{6}$$

$$R_n(t) = \min\left\{\sum_{k=1}^{K} r_{nk}(t),\ C_n^{\mathrm{bh}}(t)\right\} \tag{7}$$

$$\alpha_n(t) = \min\left\{1, \frac{C_n^{\mathrm{bh}}(t)}{\sum_j r_{nj}(t) + \varepsilon}\right\}, \quad \mu_{nk}(t) = \alpha_n(t) r_{nk}(t) \tag{8}$$

Equation (7) limits aggregate relay capacity by the available backhaul. Equation (8) propagates that bottleneck to each associated user. The effective service rate $\mu_{nk}(t)$, rather than unconstrained access rate $r_{nk}(t)$, determines the service budget. Per-slot data actually delivered are further capped by the available queue backlog according to (9). The formal resource constraints are $\sum_k B_{nk}(t) \le B_n$ and $\sum_k p_{nk}(t) \le P_{n,\max}$. In the proof of concept, association is nearest-UAV and both access bandwidth and power are shared equally among users associated with each UAV.

3.3. Queue, MEC, and Energy Models

Let $A_k(t)$ denote newly arrived data, available after service in slot t. The backhaul-limited service budget is $\Delta t\sum_n \mu_{nk}(t)$. Since data served cannot exceed the backlog $Q_k(t)$ available at the start of that slot, define the actually delivered data $S_k^{\mathrm{srv}}(t)$ and update the queue consistently as

$$\begin{aligned} S_k^{\mathrm{srv}}(t) &= \min\left\{Q_k(t), \Delta t \sum_{n=1}^{N} \mu_{nk}(t)\right\} \\ Q_k(t+1) &= Q_k(t) - S_k^{\mathrm{srv}}(t) + A_k(t) \end{aligned} \tag{9}$$

The proof-of-concept uses the explicit per-slot energy surrogate

$$E_n(t) = E_{\mathrm{hov}} + E_{\mathrm{tx}} + E_{\mathrm{move}}\,\mathbb{1}[a_n(t) \neq \mathrm{hover}] + E_{\mathrm{vert}}\,\mathbb{1}[a_n(t) \in \{\mathrm{climb}, \mathrm{descend}\}] \tag{10}$$

The values used in the simulator are $E_{\mathrm{hov}} = 0.12$ kJ, $E_{\mathrm{tx}} = 0.04$ kJ, $E_{\mathrm{move}} = 0.20$ kJ, and $E_{\mathrm{vert}} = 0.04$ kJ. This discrete surrogate is an implementation model, not a claim of platform-calibrated rotary-wing aerodynamics.

Battery dynamics are

$$b_n(t+1) = \max\{0,\ b_n(t) - E_n(t)\} \tag{11}$$

Boundary, altitude, inter-UAV separation, and one-slot reserve-energy checks enter the feasible joint-action set. The formal framework can retain MEC task variables, but the numerical experiment fixes the MEC/CPU term and does not optimize offloading or CPU allocation; those quantities are therefore excluded from the implemented learning action.

https://doi.org/10.67728/eci.2026.003

3.4. Implemented Decision Scope, Metrics, and Markov Game

The learned action at slot t is only the vector of discrete UAV motion commands. Communication-resource variables are environmental rules in this experiment: nearest-UAV association, equal access-bandwidth and power sharing, a fixed gateway-backhaul model, and the backhaul scaling in (8). This distinction is important: the framework is compatible with future joint resource control, but the results below validate trajectory/relay control under coupled communication constraints rather than learned joint radio-resource optimization.

The reported throughput R_t is the total data actually served per slot, not the sum of unconstrained access rates. Equation (12) also defines the Jain fairness index using that same realized service. Coverage C_t is the fraction of users with $S_k^{srv}(t) \geq 0.35$ Mbit in the slot; delay proxy $D_t = (1/K)\sum_k Q_k(t)$; normalized energy $\bar{E}_t = \sum_n E_n(t)/(N \times 80\ \text{kJ})$; I_t counts UAV actions altered by the safety projection; and V_t counts realized modeled safety violations. This convention retains the original Mbit/slot reporting and makes the throughput, coverage, fairness, and queue recursion consistent.

$$R_t = \sum_{k=1}^{K} S_k^{\text{srv}}(t)$$
$$J_t = \frac{\left(\sum_{k=1}^{K} S_k^{\text{srv}}(t)\right)^2}{K\sum_{k=1}^{K}\left(S_k^{\text{srv}}(t)\right)^2 + \varepsilon} \tag{12}$$

Training maximizes the discounted expected return $\max_\pi E_\pi[\sum_{t=0}^{T-1} \gamma^t r_t]$, with

$$r_t = 0.30(R_t/20) + 1.8C_t + 0.9J_t - 8\bar{E}_t - 0.03D_t - 2V_t - 0.035I_t \tag{13}$$

The reported episode utility is the undiscounted sum of r_t over the 40 evaluation slots so that methods can be compared on the same scale.

All normalization constants and thresholds above are fixed before evaluation. The 20 Mbit/slot throughput reference is used only for reward scaling; coverage, fairness, delay, energy, interventions, and violations are also reported separately so that a high scalar utility cannot hide unsafe or unfair behavior.

The resulting cooperative Markov game is $G = \langle N, S, \{O_n\}, \{A_n\}, P, r, \gamma\rangle$. The global state contains UAV positions, batteries, queues, user locations, channel/backhaul quantities, and neighbor geometry; each UAV executes from its locally encoded observation. The seven-action set consists of hover, ±x, ±y, climb, and descend.

4. Communication-Aware Multi-Agent Dueling Double DQN

4.1. Communication-Aware State Encoding

A discrete seven-action space is used because the implemented control variable is waypoint-level UAV motion rather than continuous radio-resource control. This choice permits an exact finite joint-action feasibility check in the two-UAV testbed and keeps the proof-of-concept computationally transparent. It should not be interpreted as evidence that value-based control is universally preferable to continuous actor-critic methods; MAPPO is included as a competitive baseline in Section 5.

For feature construction, users are assigned to the nearest UAV. Each local observation contains normalized UAV position and battery, an urgency-weighted centroid of associated users, a one-step predicted centroid, mean local queue, the relative position of the nearest other UAV, and normalized backhaul capacity. The predictor can be removed without changing the action space, enabling a clean ablation. Bandwidth, transmit power, association, and CPU allocation are not action dimensions in the reported implementation.

4.2. Dueling Double-Q Architecture and Parameter Sharing

All UAVs share one Q-network, which reduces the number of trainable parameters and makes agent identity exchangeable in this small homogeneous swarm. The dueling decomposition separates state value from action-specific advantage [21]:

$$Q(o,a;\theta) = V(o;\theta_V) + A(o,a;\theta_A) - \frac{1}{|\mathcal{A}|}\sum_{a'\in\mathcal{A}} A(o,a';\theta_A) \tag{14}$$

Double Q-learning reduces optimistic target bias by selecting the next action with the online network and evaluating it with the target network [22]. The target used in training is given in (15).

$$y_t = r_t + \gamma Q\left(o_{t+1}, \operatorname*{argmax}_a Q(o_{t+1},a;\theta);\theta^-\right) \tag{15}$$

https://doi.org/10.67728/eci.2026.003

Experience from all UAVs is stored in a common replay buffer. Parameter sharing, dueling decomposition, and double-Q targets are implementation choices rather than novelty claims. The discrete MAPPO baseline in Section 5 uses the same observation variables, reward, simulator, and joint safety projection so that policy-class differences are not confounded with different physical constraints.

4.3. Safety Projection and Reward Construction

A reward penalty does not guarantee feasible exploration. We therefore project the entire proposed multi-UAV joint action, not each action independently, onto the modeled feasible joint-action set as in (16). The distance is the number of UAV commands that must be changed (Hamming distance).

$$\tilde{\mathbf{a}}(t) = \arg\min_{\mathbf{a}' \in \mathcal{A}_{\text{feas}}(s_t)} \sum_{n=1}^{N} \mathbb{1}\left[a_n' \neq a_n(t)\right] \tag{16}$$

For two UAVs with seven actions each, all 49 joint actions are enumerated. The feasible candidate with minimum Hamming distance from the learned proposal is executed; ties prefer fewer non-hover motions. Consequently, boundary, altitude, separation, and reserve-energy feasibility are jointly guaranteed in the simulator whenever the feasible set is nonempty. Projection interventions are logged as modified UAV actions. This guarantee is limited to the modeled constraints and does not constitute a real-world flight-safety certification. For swarms larger than two UAVs, the scalability experiment uses a deterministic coordinate-repair approximation rather than exhaustive joint enumeration.

The numerical reward is exactly the expression in (13). Reporting I_t separately is essential because a zero realized-violation count can be produced either by a genuinely safe proposal policy or by frequent rule-based correction. Section 5.4 therefore evaluates the same learned policy both with and without the projection at execution time.

4.4. Prospective Federated-Learning Extension

A possible future multi-domain implementation could avoid centrally collecting raw user traces by training local parameters θ_i and periodically sending clipped model updates to an aggregator. A potential aggregation rule is federated averaging, $\theta \leftarrow \sum_i (n_i/\sum_j n_j)\theta_i$ [20]; secure aggregation and differential privacy could be evaluated under explicit threat and overhead assumptions [23]. Federated training was not implemented in this study. Its convergence, communication cost, robustness, and privacy leakage remain unmeasured; the optional exchange in Figure 1 is therefore a proposed architecture, not an experimentally validated contribution.

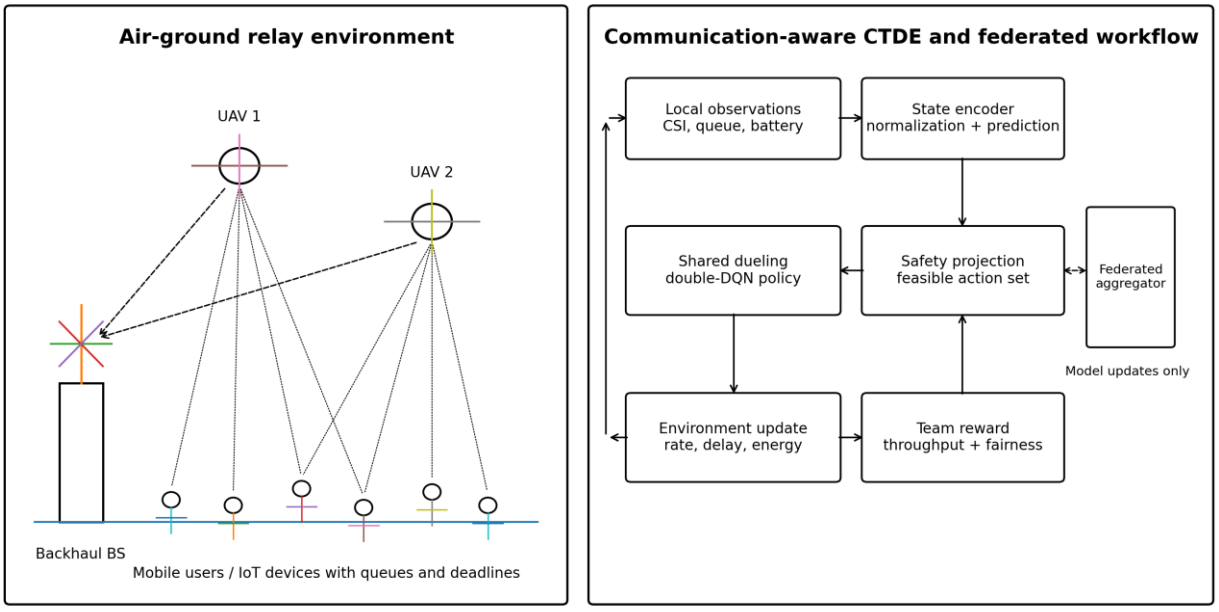


Figure 1. Air-ground relay scenario and communication-aware learning workflow. Solid arrows show observation, action, and reward flow; the dashed model exchange is a prospective federated extension that was not evaluated. Source: Author.

https://doi.org/10.67728/eci.2026.003

Algorithm 1. CA-MAD3QN training, joint safety projection, and evaluation procedure. Source: Author.

Step	Operation
1	Initialize the shared online/target Q-networks, replay buffer, random seed, and fixed simulator configuration.
2	At each slot, construct local observations from UAV state, queues, user centroids/prediction, neighbor geometry, and backhaul capacity.
3	Each UAV proposes one of seven motion actions using epsilon-greedy Q values.
4	Project the proposed joint action onto A_{feas}; for the two-UAV testbed enumerate all 49 joint actions and log every modified UAV command.
5	Execute motion; apply nearest-UAV association and equal bandwidth/power sharing; compute access rates, backhaul scaling $alpha_n$ and effective user rates mu_{nk}
6	Update queue backlogs and realized service according to (9), energy and battery with (10)–(11), and log throughput, coverage, fairness, interventions, and realized violations.
7	Store each agent transition with the shared team reward from (13).
8	Sample replay mini-batches; update the online network with the double-Q target in (15); copy parameters to the target network every 200 environment steps.
9	Evaluate without exploration over ten episodes per training seed and report 95% t-confidence intervals across the three seed-level means.

5. Evaluation Methodology and Results

5.1. Reproducible Proof-of-Concept Setup

The primary experiment retains the two-UAV/sixteen-user 1 km × 1 km testbed and 40-slot horizon, but the evaluation protocol is expanded. User positions follow correlated random walks around two moving hotspots and queues receive Poisson traffic. The stress scenario increases the Poisson event rate from 0.8 to 1.25 events/slot (0.45 Mbit per event), increases the mobility standard deviation from 7 to 22 m/slot, adds 10 dB access attenuation, and adds 10 dB backhaul attenuation. Learned methods are trained independently with three random seeds for 60 episodes. Each trained seed is evaluated for ten nominal and ten stress episodes with exploration disabled. Table and figure confidence intervals are 95% Student-t intervals across the three seed-level means: mean ± $t_{(0.975,2)}$s/sqrt(3). Non-learning baselines use three independent evaluation-seed blocks of the same size (Table 2).

Table 2. Main simulator, network, and training hyperparameters. Source: Author.

Parameter	Value
Area/horizon	1000 m × 1000 m/40 slots
UAVs/users	2/16 (main); 2/3/4 UAVs with 8 users/UAV in scaling
Altitude/minimum separation	80–140 m/120 m
Horizontal/vertical step	55 m/15 m
Carrier/access bandwidth	2 GHz/8 MHz per UAV
Access/backhaul transmit power	24 dBm/32 dBm
Backhaul bandwidth/extra attenuation	1 MHz/20 dB nominal, 30 dB stress
Urban LoS parameters/noise figure	a = 9.61, b = 0.16, eta_L=1 dB, eta_N = 20 dB/7 dB
Initial battery/slot energy	80 kJ; E_{hov}=0.12, E_{tx} = 0.04, E_{move} = 0.20, E_{vert} = 0.04 kJ
Traffic/mobility	Poisson 0.8 nominal, 1.25 stress; 0.45 Mbit/event; mobility sigma 7/22 m
DQN network/optimizer	64-64 MLP; Adam; learning rate 4 × 10^{-4}
Replay/batch/gamma	50,000/96/0.985
Exploration/target update	epsilon 1.0 to 0.05 over 72% of training/every 200 environment steps
Training/evaluation	60 episodes; 3 training seeds; 10 evaluation episodes/seed/scenario
MAPPO settings	64-64 actor, centralized critic; lr 3 × 10^{-4}; clip 0.2; GAE λ = 0.95; 3 PPO epochs

https://doi.org/10.67728/eci.2026.003

5.2. Baselines and Metrics

Six methods are compared under the same channel, queue, energy, reward, and safety definitions. CA-MAD3QN is the full method. Vanilla DQN removes the dueling head, double-Q target, and mobility-prediction feature. Discrete MAPPO uses a parameter-shared categorical actor and centralized critic with the same seven actions and joint projection. MPC-1 is a coordinate-wise one-step model-predictive dynamic optimizer that evaluates at most 7N candidate motion changes per slot using the same simulator model. Greedy tracking moves each UAV toward the urgency-weighted centroid of its associated users. Static deployment moves toward the initial user-cluster centroids and then holds those positions. The reported metrics are the quantities defined in Section 3.4. Learned-policy uncertainty is reported across independent training seeds, not merely across repeated episodes of one trained network (Figure 2).

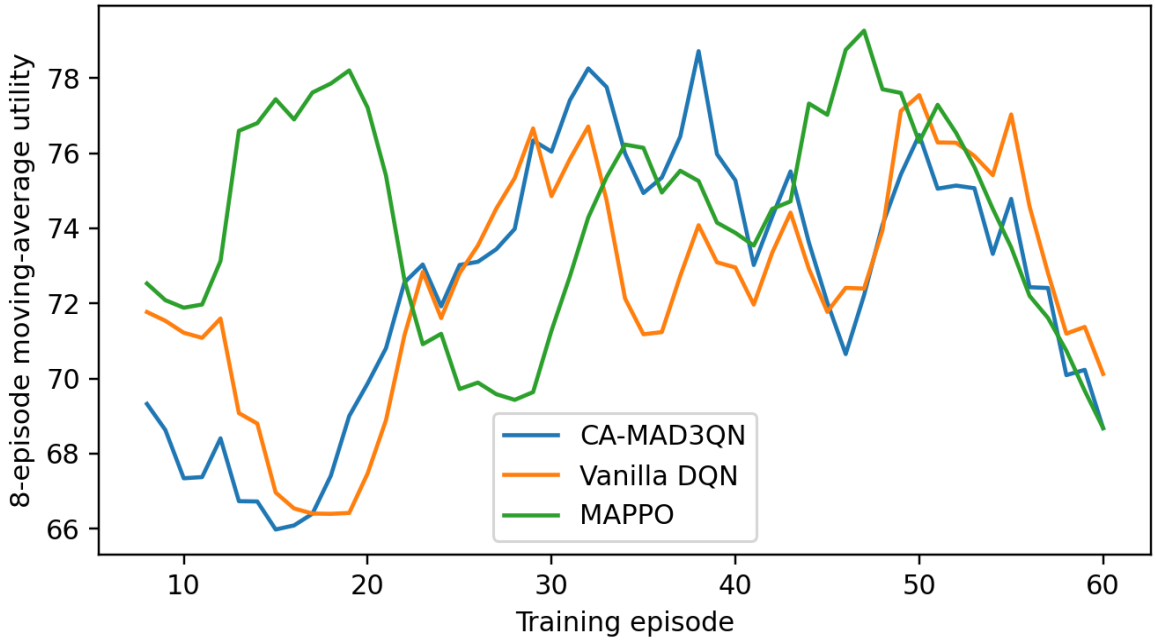


Figure 2. Eight-episode moving-average training utility for CA-MAD3QN, vanilla DQN, and discrete MAPPO, averaged across three independent training seeds. Curves document finite-budget learning behavior rather than asymptotic convergence. Source: Author.

5.3. Baseline Comparison

The expanded comparison does not support a claim that CA-MAD3QN universally outperforms the alternatives. In the stress scenario, CA-MAD3QN achieves throughput 8.048 ± 1.846 Mbit/slot and episode utility 85.36 ± 28.94. Discrete MAPPO obtains utility 69.32 ± 53.03, while MPC-1, greedy tracking, and static deployment obtain 95.85 ± 2.03, 94.69 ± 2.90, and 91.93 ± 1.22, respectively. The wide CA-MAD3QN/MAPPO intervals reflect training-seed sensitivity under the deliberately small 60-episode budget. Under nominal traffic, CA-MAD3QN utility is 59.02 ± 6.51 and greedy tracking is 61.41 ± 0.98, again indicating comparable behavior rather than dominance. The principal conclusion of this experiment is therefore reproducibility and transparent trade-off characterization, not state-of-the-art superiority (Table 3, Figures 3–5).

Table 3. Stress-scenario performance. Entries are mean ± 95% CI across three seed-level means; intervention and violation columns show the mean count per slot. Source: Author.

Method	Throughput	Coverage	Fairness	Delay	Energy	Utility	Interv./slot	Viol./slot
CA-MAD3QN	8.048 ± 1.846	0.778 ± 0.255	0.812 ± 0.162	1.960 ± 1.980	0.0022 ± 0.0005	85.36 ± 28.94	1.198	0.000
MAPPO	6.838 ± 3.351	0.619 ± 0.463	0.743 ± 0.335	3.340 ± 4.045	0.0030 ± 0.0029	69.32 ± 53.03	0.797	0.000
MPC-1	8.549 ± 0.245	0.866 ± 0.015	0.860 ± 0.021	1.549 ± 0.010	0.0024 ± 0.0001	95.85 ± 2.03	0.000	0.000
Vanilla DQN	7.799 ± 2.914	0.740 ± 0.419	0.830 ± 0.084	2.232 ± 3.151	0.0023 ± 0.0011	82.92 ± 39.47	1.081	0.000
Greedy	8.564 ± 0.255	0.854 ± 0.027	0.861 ± 0.025	1.515 ± 0.024	0.0036 ± 0.0001	94.69 ± 2.90	0.000	0.000
Static	8.429 ± 0.332	0.827 ± 0.009	0.840 ± 0.005	1.655 ± 0.072	0.0029 ± 0.0006	91.93 ± 1.22	0.000	0.000

https://doi.org/10.67728/eci.2026.003

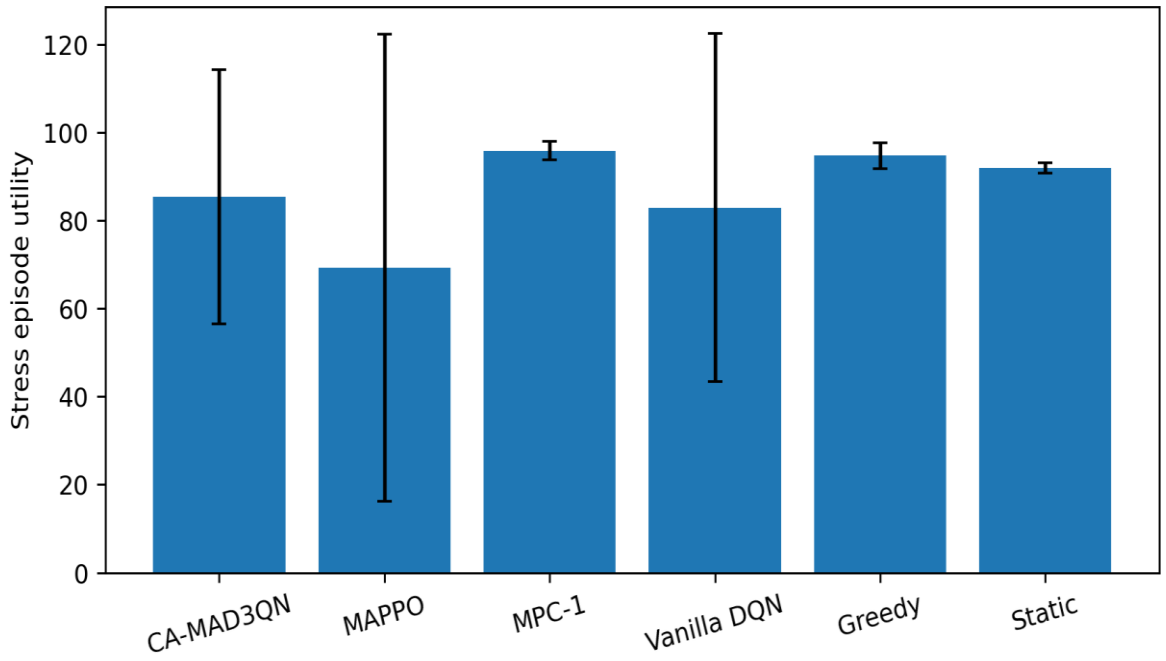


Figure 3. Stress-scenario episode utility. Error bars are 95% Student-t confidence intervals across three independent seed-level means. Source: Author.

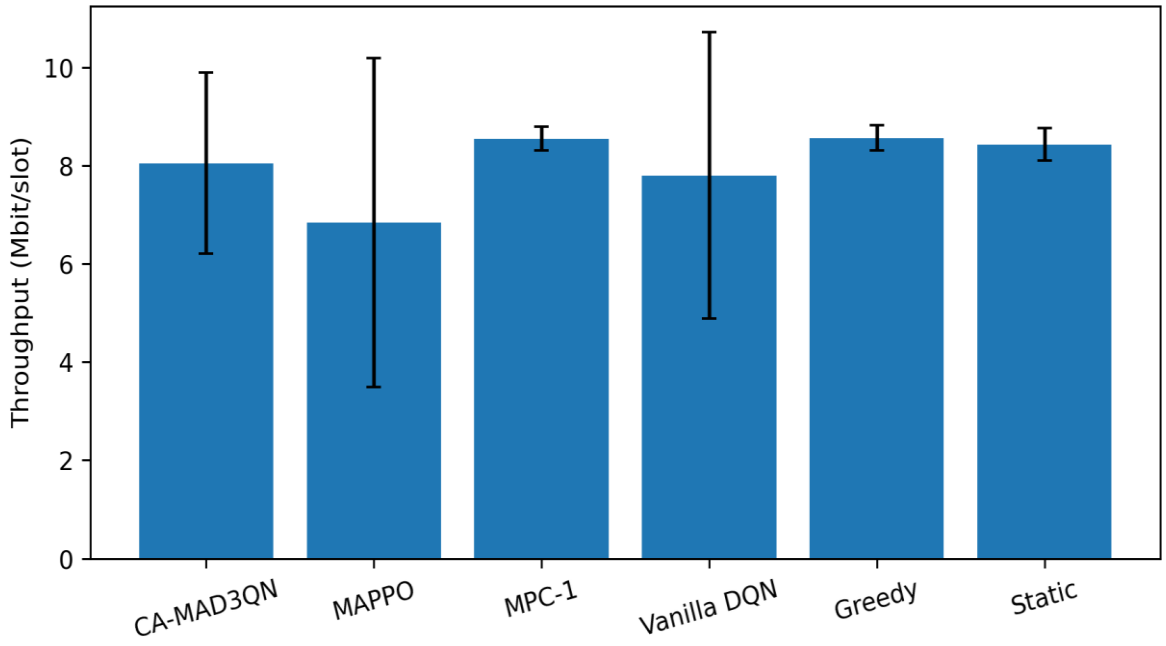


Figure 4. Stress-scenario throughput. Error bars are 95% Student-t confidence intervals across three independent seed-level means. Source: Author.

https://doi.org/10.67728/eci.2026.003

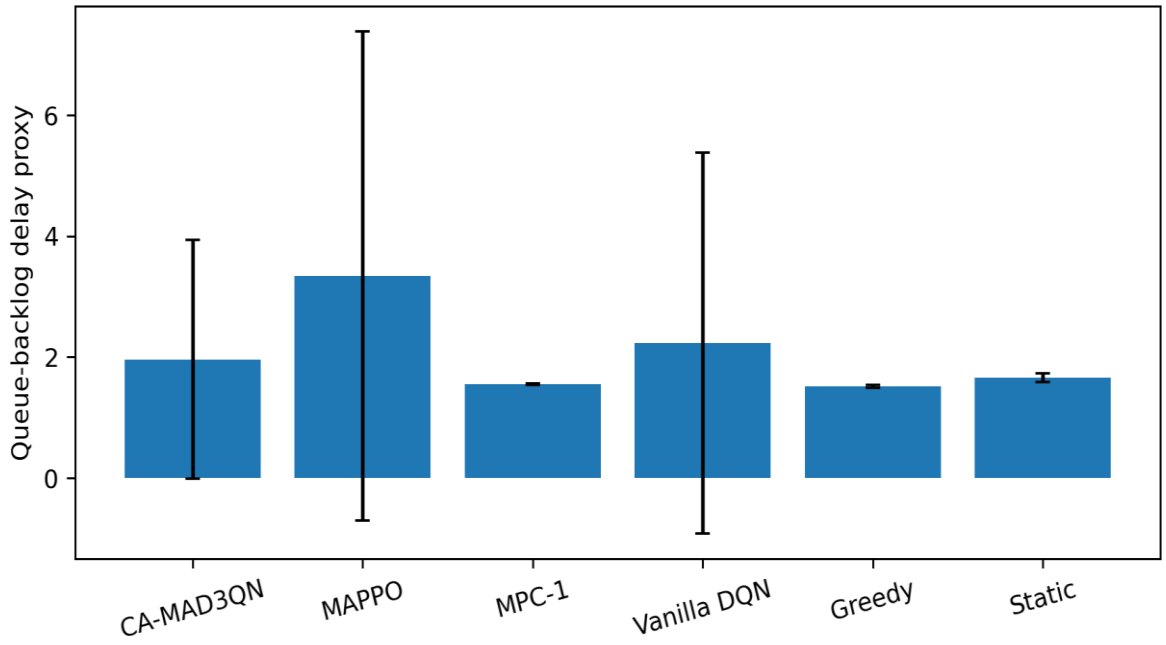


Figure 5. Stress-scenario queue-backlog delay proxy; lower is better. Error bars are 95% Student-t confidence intervals. Source: Author.

5.4. Ablation and Safety Analysis

The paired safety test separates the learned proposal policy from the rule-based projection. With the full trained policy and the joint projection enabled, the evaluation records zero realized modeled violations. When the projection is disabled for the same network weights and evaluation seeds, the controller produces 0.863 realized violations per slot on average and the mean utility falls to 17.44 with very large cross-seed variance. The projected controller modifies roughly 1.20 UAV actions per slot in the main stress evaluation (about 60% of the two proposed actions), so the zero-violation result cannot be attributed to the learned policy alone. The projection is therefore effective as an execution-time guard, but the high intervention burden is itself a limitation and motivates constrained-policy learning that proposes fewer inadmissible actions (Table 4).

Table 4. Stress-scenario module ablations and paired safety-layer test. Source: Author.

Variant	Throughput	Delay	Utility	Interv./slot	Viol./slot
Full CA-MAD3QN	8.048	1.960	85.36	1.198	0.000
No dueling	8.363	1.574	90.52	1.252	0.000
No double-Q	8.049	1.898	81.21	1.717	0.000
No prediction	8.506	1.395	88.35	1.783	0.000
No fairness term*	8.370	1.573	86.11	1.449	0.000
Projection off, same policy	7.941	1.985	17.44	0.000	0.863

* For the no-fairness-term variant, utility was re-evaluated using the common full reward metric for comparability.

The remaining ablations are deliberately reported without forcing a positive interpretation. Removing the dueling head gives a higher mean stress throughput (8.363 vs. 8.048 Mbit/slot) and utility (90.52 vs. 85.36) in this small testbed; removing double-Q yields similar mean throughput but high variance; removing the one-step mobility prediction also changes mean performance only modestly. Training without the fairness term changes the reward trade-off but does not reliably improve the standalone Jain index. These results show that the empirical contribution of individual modules is scenario dependent and that the strongest reproducible role in this experiment is the explicit communication/queue consistency and safety-accounting framework rather than any single neural-network embellishment (Figure 6).

https://doi.org/10.67728/eci.2026.003

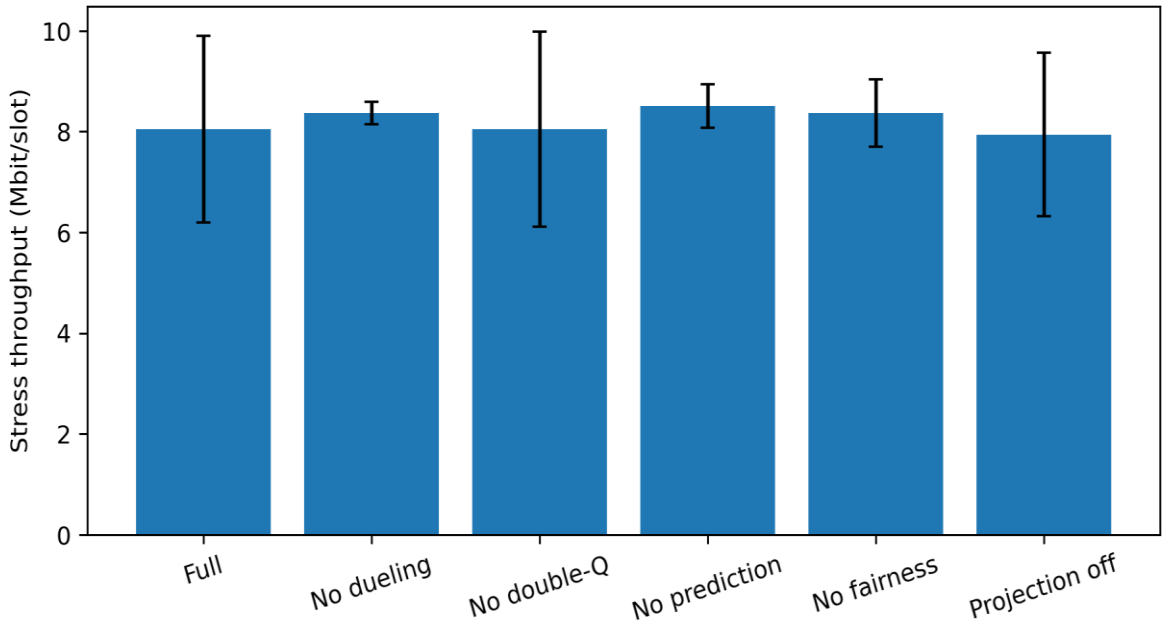


Figure 6. Stress-scenario throughput under separate module ablations; error bars are 95% confidence intervals. The projection-off bar evaluates the same learned full-policy weights without the safety layer. Source: Author.

5.5. Preliminary Small-Scale Scalability and Validity Limits

A preliminary small-scale test uses the same full architecture with three independent 60-episode training seeds and eight stress-evaluation episodes per seed at N = 2, 3, and 4 UAVs, with K = 8N users. Aggregate throughput is 8.44 ± 0.37, 12.71 ± 0.68, and 16.88 ± 0.78 Mbit/slot, respectively, corresponding to approximately stable per-UAV throughput in these settings. This shows only feasibility in a small simulator, not general large-swarm scalability. Exact enumeration of 7^N joint actions is used for N = 2; the N > 2 experiments instead use coordinate-wise feasibility repair, which does not guarantee the globally nearest feasible joint action. Its computational scaling, repair quality, and safety properties at larger swarm sizes require separate study (Table 5, Figure 7).

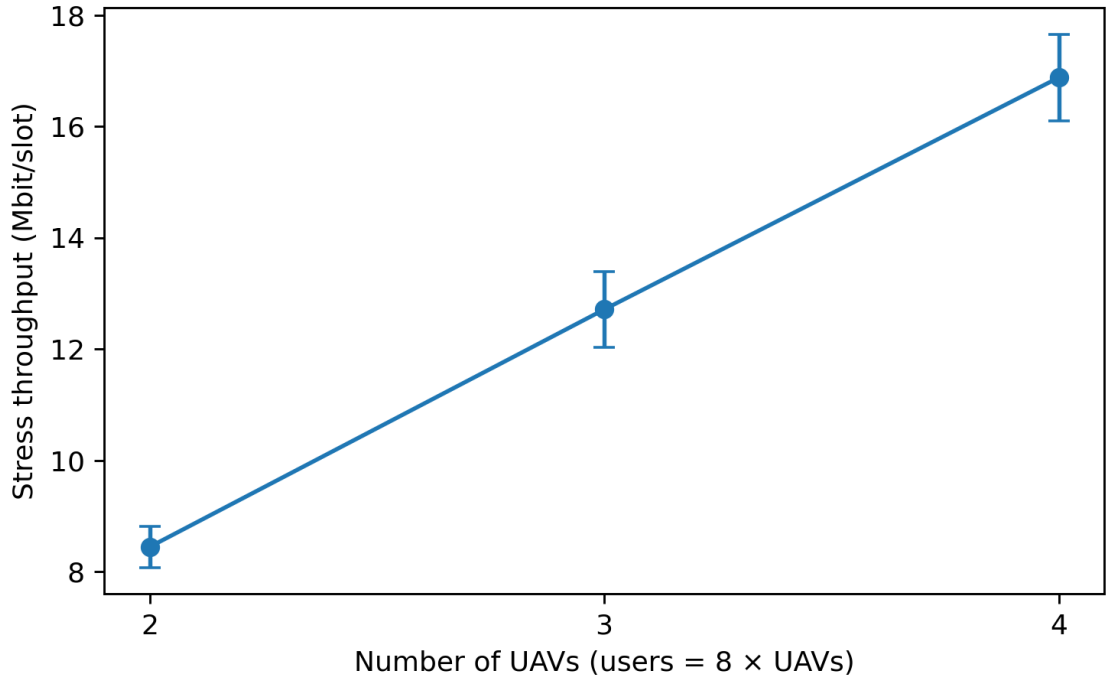


Figure 7. Stress throughput in the 2/3/4-UAV scalability test (8 users per UAV). Error bars are 95% confidence intervals across three independent training seeds. Source: Authors.

https://doi.org/10.67728/eci.2026.003

Table 5. Scalability results (mean ± 95% CI). Source: Author.

UAVs/Users	Throughput	Coverage	Delay	Utility
2/16	8.443 ± 0.369	0.832 ± 0.067	1.592 ± 0.241	91.14 ± 3.11
3/24	12.709 ± 0.683	0.809 ± 0.117	1.403 ± 0.779	89.51 ± 14.87
4/32	16.876 ± 0.781	0.839 ± 0.062	1.558 ± 0.085	95.38 ± 5.40

The study remains a simplified numerical proof of concept. It uses discrete waypoint actions, equal-share communication resources, a probabilistic urban channel, and a discrete per-slot energy surrogate rather than antenna orientation, wind dynamics, calibrated rotorcraft power, real battery discharge, packet-level protocol effects, hardware latency, or field measurements. Three training seeds are enough to expose variance but not to characterize all stochastic uncertainty; indeed, the wide CA-MAD3QN and MAPPO confidence intervals are a material result. The experiments therefore establish internal model/implementation consistency and comparative behavior only within the stated simulator. They do not establish real-world performance, universal superiority over continuous-control MARL, or certified flight safety.

The nominal case further reinforces the restrained interpretation: simple greedy and static controllers are close to the learned methods because the short horizon and low traffic allow myopic motion to work well. In the stress case, MPC-1 and greedy tracking also have higher point-estimate utility than CA-MAD3QN in this particular simulator. We therefore removed earlier statements that the proposed method necessarily dominates under distribution shift. The practical value claimed here is instead an auditable framework in which communication bottlenecks, queues, safety intervention, and learning variability can be inspected separately.

6. Engineering, Industrial, and Societal Implications

6.1. Representative Emergency Deployment Path

The testbed maps to a small emergency-restoration mission in which battery-limited UAVs provide temporary relay service while maintaining a constrained gateway link. The revised experiments do not translate the simulated utility into guaranteed field gains. Instead, they identify two operationally relevant diagnostics: backhaul-limited per-user service must be propagated into queue evolution, and a zero realized-violation count must be interpreted together with the projection-intervention rate. A deployment team would still need a calibrated digital map, platform-specific channel and energy measurements, hardware-in-the-loop timing tests, geofencing, regulatory constraints, and a verified fallback controller before any outdoor trial.

6.2. Alignment with IMT-2030 and NTN Engineering

ITU-R M.2160 identifies ubiquitous connectivity, artificial intelligence integration, sustainability, and enhanced security and resilience as central IMT-2030 directions [8]. 3GPP NTN studies similarly require explicit treatment of non-terrestrial propagation, mobility, timing, and service continuity [9]. The proposed framework contributes an orchestration layer rather than a replacement for those standards: standardized links provide measurements and control interfaces, the system model defines feasible operating envelopes, and the learned policy selects relay and trajectory actions within them. Regulatory altitude, remote identification, spectrum authorization, and local no-fly requirements remain external hard constraints and must never be left to reward learning.

6.3. Privacy, Fairness, and Teaching Value

User locations, traffic queues, and channel traces can reveal rescue-team movement or private-device activity. A federated implementation reduces raw-data concentration but does not automatically prevent model inversion, poisoned updates, or metadata leakage. A deployment should therefore minimize collected features, encrypt communication, authenticate participants, audit updates, and report privacy metrics alongside throughput. Likewise, fairness should be measured through Jain's index, worst-user service, and deadline violations rather than asserted qualitatively. For teaching and reproducibility, the compact simulator demonstrates how channel, queue, energy, and safety terms enter a Markov game and how negative ablations can inform revision rather than be suppressed.

Potential industrial use of multi-domain federated UAV learning remains speculative. In a multi-operator setting, locally trained model updates might reduce the need to exchange raw user traces, but they could still reveal sensitive information and would incur communication and coordination overhead. This study provides no measurements of federation overhead, privacy leakage, operating costs, cross-domain transfer, or market adoption. Commercial feasibility and any prospective UAV-as-a-Service benefits should therefore be investigated only after privacy, security, communication, and hardware-level evaluations.

https://doi.org/10.67728/eci.2026.003

6.4. Roadmap to Semi-Physical and Field Validation

- Stage 1—calibrated simulation: fit channel, traffic, and propulsion parameters to a selected platform and conduct multi-seed sensitivity analysis.
- Stage 2—hardware-in-the-loop: connect the policy server to autopilot and radio emulators; measure inference latency, packet loss, and safety-intervention frequency.
- Stage 3—controlled outdoor trial: use a small authorized area, geofencing, a fallback controller, and conservative speed and altitude limits.
- Stage 4—multi-domain federation: evaluate communication overhead, non-independent data, poisoning resilience, and privacy leakage before exchanging operational models.
- Stage 5—operational pilot: integrate incident-command procedures, spectrum permissions, remote identification, and human override.

7. Conclusions

This paper formulates UAV-swarm relay deployment through discrete three-dimensional motion as a communication-aware constrained Markov game and states the numerical decision scope precisely. CA-MAD3QN learns only UAV motion; association, bandwidth, access power, and MEC quantities are fixed or rule based. A central modeling clarification defines the actually served per-slot data as the minimum of available backlog and backhaul-limited service budget, and uses this quantity for throughput, fairness, coverage, and queue evolution. The experiment also uses explicit battery recursion and reports safety interventions separately from realized violations.

The expanded three-seed evaluation compares CA-MAD3QN with discrete MAPPO, vanilla DQN, MPC-1, greedy tracking, and static deployment and reports 95% confidence intervals. The data do not establish universal superiority: CA-MAD3QN stress utility is 85.36 ± 28.94, while MPC-1 and greedy tracking have higher point estimates and tighter intervals in this small simulator. The module ablations likewise show that dueling, double-Q, and mobility prediction are not uniformly beneficial. These negative or neutral findings are retained because they define the actual empirical contribution more accurately than a dominance claim.

In the two-UAV finite-action testbed, the exact joint safety projection guarantees modeled joint feasibility whenever a feasible action exists, but it intervenes frequently. Disabling the projection for the same trained policy yields 0.863 realized violations per slot on average, so the learned proposal policy should not be conflated with the rule-based safety guard. The preliminary 2/3/4-UAV test shows increasing aggregate throughput only in this small simulator; for more than two UAVs, projection is replaced by an approximate coordinate-repair scheme. General large-swarm scalability, continuous resource control, calibrated energy models, more independent training seeds, packet-level effects, and hardware-in-the-loop and authorized field validation remain future work. Federated learning is likewise an untested future extension. CA-MAD3QN is therefore an auditable trajectory-control proof of concept, not a universally superior or field-validated controller.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable. This study uses simulated UAVs and synthetic traffic and does not involve human or animal participants.

Informed Consent Statement: Not applicable.

Data Availability Statement: The numerical results supporting this study are reported in its tables and figures. The study uses synthetic traffic and mobility generated in the described simulation setup; no external dataset is analyzed.

Acknowledgments: The author thanks Jilin University for institutional support.

Conflicts of Interest: The author declares no conflicts of interest.

References

[1] Y. Zeng, R. Zhang, and T. J. Lim, “Wireless communications with unmanned aerial vehicles: Opportunities and challenges,” IEEE Communications Magazine, vol. 54, no. 5, pp. 36–42, 2016, doi: 10.1109/MCOM.2016.7470933.

[2] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah, “A tutorial on UAVs for wireless networks: Applications, challenges, and open problems,” IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2334–2360, 2019, doi: 10.1109/COMST.2019.2902862.

[3] A. Al-Hourani, S. Kandeepan, and S. Lardner, “Optimal LAP altitude for maximum coverage,” IEEE Wireless Communications Letters, vol. 3, no. 6, pp. 569–572, 2014, doi: 10.1109/LWC.2014.2342736.

[4] C. Yan, L. Fu, J. Zhang, and J. Wang, "A comprehensive survey on UAV communication channel modeling," IEEE Access, vol. 7, pp. 107769–107792, 2019, doi: 10.1109/ACCESS.2019.2933173.

[5] Y. Zeng, J. Xu, and R. Zhang, "Energy minimization for wireless communication with rotary-wing UAV," IEEE Transactions on Wireless Communications, vol. 18, no. 4, pp. 2329–2345, 2019, doi: 10.1109/TWC.2019.2902559.

[6] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, D. Niyato, O. A. Dobre, and H. V. Poor, "6G Internet of Things: A comprehensive survey," IEEE Internet of Things Journal, vol. 9, no. 1, pp. 359–383, 2022, doi: 10.1109/JIOT.2021.3103320.

[7] W. Jiang, B. Han, M. A. Habibi, and H. D. Schotten, "The road towards 6G: A comprehensive survey," IEEE Open Journal of the Communications Society, vol. 2, pp. 334–366, 2021, doi: 10.1109/OJCOMS.2021.3057679.

[8] ITU-R, Framework and Overall Objectives of the Future Development of IMT for 2030 and Beyond, Recommendation ITU-R M.2160-0, Nov. 2023.

[9] 3GPP, Study on New Radio (NR) to Support Non-Terrestrial Networks, 3GPP TR 38.811, Release 15.

[10] L. Wang, K. Wang, C. Pan, W. Xu, N. Aslam, and L. Hanzo, "Multi-agent deep reinforcement learning-based trajectory planning for multi-UAV assisted mobile edge computing," IEEE Transactions on Cognitive Communications and Networking, vol. 7, no. 1, pp. 73–84, 2021, doi: 10.1109/TCCN.2020.3027695.

[11] R. Ding, F. Gao, and X. S. Shen, "3D UAV trajectory design and frequency band allocation for energy-efficient and fair communication: A deep reinforcement learning approach," IEEE Transactions on Wireless Communications, vol. 19, no. 12, pp. 7796–7809, 2020, doi: 10.1109/TWC.2020.3016024.

[12] Y. Bai, H. Zhao, X. Zhang, Z. Chang, R. Jäntti, and K. Yang, "Toward autonomous multi-UAV wireless network: A survey of reinforcement learning-based approaches," IEEE Communications Surveys & Tutorials, vol. 25, no. 4, pp. 3038–3067, 2023, doi: 10.1109/COMST.2023.3323344.

[13] S. F. Abedin, M. S. Munir, N. H. Tran, Z. Han, and C. S. Hong, "Data freshness and energy-efficient UAV navigation optimization: A deep reinforcement learning approach," IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 9, pp. 5994–6006, 2021, doi: 10.1109/TITS.2020.3039617.

[14] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in Advances in Neural Information Processing Systems 30 (NIPS 2017), 2017.

[15] C. Yu, A. Velu, E. Vinitsky, J. Gao, Y. Wang, A. Bayen, and Y. Wu, "The surprising effectiveness of PPO in cooperative multi-agent games," in Advances in Neural Information Processing Systems 35 (NeurIPS 2022), pp. 24611–24624, 2022, doi: 10.52202/068431-1787.

[16] V. Mnih et al., "Human-level control through deep reinforcement learning," Nature, vol. 518, pp. 529–533, 2015, doi: 10.1038/nature14236.

[17] Y.-J. Chen, D.-K. Chang, and C. Zhang, "Autonomous tracking using a swarm of UAVs: A constrained multi-agent reinforcement learning approach," IEEE Transactions on Vehicular Technology, vol. 69, no. 11, pp. 13702–13717, 2020, doi: 10.1109/TVT.2020.3023733.

[18] Y. Wang, Y. Ling, Y. Wang, J. Ma, and Q. Jin, "Multiple targets tracking with unmanned aerial vehicle swarm based on multi-agent deep reinforcement learning framework with safety network," Engineering Applications of Artificial Intelligence, vol. 181, Art. no. 115370, 2026, doi: 10.1016/j.engappai.2026.115370.

[19] W. Zhang, Q. Wang, X. Liu, Y. Liu, and Y. Chen, "Three-dimension trajectory design for multi-UAV wireless network with deep reinforcement learning," IEEE Transactions on Vehicular Technology, vol. 70, no. 1, pp. 600–612, 2021, doi: 10.1109/TVT.2020.3047800.

[20] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS), PMLR, vol. 54, pp. 1273–1282, 2017.

[21] Z. Wang, T. Schaul, M. Hessel, H. van Hasselt, M. Lanctot, and N. de Freitas, "Dueling network architectures for deep reinforcement learning," in Proceedings of the 33rd International Conference on Machine Learning (ICML), PMLR, vol. 48, pp. 1995–2003, 2016.

[22] H. van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 30, no. 1, pp. 2094–2100, 2016, doi: 10.1609/aaai.v30i1.10295.

[23] P. Kairouz et al., "Advances and open problems in federated learning," Foundations and Trends in Machine Learning, vol. 14, nos. 1–2, pp. 1–210, 2021, doi: 10.1561/2200000083.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of Apollo and/or the editor(s). Apollo and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.