



ARTICLE

Predictive-Q Learning Based Interference-and-Mobility Aware Dual-Path Routing for UAV Swarm Networks with Mobile Edge Computing

Zhihao Dong^{1,2}, Huakui Sun^{1,2,*}, Yueyue Tao² and Daosen Zhai²

¹School of Airspace Science and Engineering, Shandong University, Weihai, China

²School of Electronics and Information, Northwestern Polytechnical University, Xi'an, China

*Corresponding Author: Huakui Sun. Email: sunhuakui@sdu.edu.cn

Received: 20 April 2026; Accepted: 03 June 2026; Published: 23 July 2026

ABSTRACT: High-mobility Unmanned Aerial Vehicle (UAV) swarm networks suffer from fast-varying connectivity and interference, and therefore routing decisions must jointly account for link instability and topology changes. By leveraging mobile edge computing (MEC) capabilities, each UAV can perform online routing decisions locally without relying on centralized controllers. This paper develops a Predictive-Q learning framework for dynamic routing under interference and mobility, where the Q-value is trained by a multi-factor reward that explicitly models retransmission costs, predicts link lifetime from relative motion, and anticipates forward connectivity and neighbor redundancy. To further enhance reliability under harsh interference, we extend Predictive-Q with a lightweight dual-path forwarding mechanism. Specifically, it conditionally splits a backup routing when the primary next hop becomes unreliable, and terminates the backup early when continued forwarding is unlikely to be beneficial, thereby controlling overhead. Simulation results demonstrate that, compared with existing methods, the proposed Predictive-Q-Dual improves packet delivery ratio by more than 15% over GPSR under strong interference while maintaining low delay and energy consumption across varying interference intensity, node density, and mobility speed.

KEYWORDS: UAV swarm networks; routing; reinforcement learning; dual-path forwarding; mobile edge computing

1 Introduction

Unmanned aerial vehicle (UAV) swarms are increasingly adopted for challenging missions such as emergency response, wide-area reconnaissance, and beyond-visual-line-of-sight operations. In these scenarios, UAVs must exchange coordination, control, and payload data over highly dynamic wireless links, forming flying ad hoc networks (FANETs). Compared with conventional mobile ad hoc networks and vehicular networks, FANETs feature rapid topology variation, intermittent connectivity, and structural fragility, which makes reliable multi-hop forwarding a severe challenge [1]. In addition, aerial links may experience intentional and unintentional interference, causing abrupt reliability degradation and significant retransmission-induced delay [2–5]. Therefore, robust forwarding that explicitly accounts for both mobility and interference is essential. With the integration of mobile edge computing (MEC) through onboard edge resources, each UAV can execute lightweight learning-based routing algorithms in real time, eliminating reliance on centralized controllers and enabling per-packet adaptive forwarding at the network edge.

Geographic routing offers a lightweight solution for position-based forwarding but struggles with 3D routing voids, short-lived links, and interference-induced reliability degradation [6–9]. Reinforcement learning (RL) provides an adaptive alternative by learning effective policies through environment interaction,

yet existing RL-based schemes often suffer from excessive exploration overhead or fail to explicitly model the reliability–overhead tradeoff under interference [10,11]. A detailed review of these approaches is provided in [Section 2](#).

Motivated by these gaps, this paper develops a dual-path predictive learning-based routing framework for UAV swarms under highly dynamic and interference-prone conditions. The core component is a Predictive-Q forwarding strategy that integrates multi-factor prediction (e.g., link quality, expected retransmission delay, and mobility-induced link stability) into Q-values for lightweight online routing selection on MEC-enabled UAV nodes [12–15]. To further enhance reliability under harsh interference, we propose a lightweight dual-path extension with controlled redundancy, achieving reliability improvement with bounded overhead under severe interference.

The main contributions are summarized as follows:

- We present a Predictive-Q forwarding strategy that embeds multi-factor link and mobility prediction into Q-values, enabling robust and lightweight online routing in highly dynamic UAV swarms. Unlike single-objective RL schemes, Predictive-Q employs distance-adaptive weighting to dynamically balance progress, reliability, and stability as packets approach the destination, making it suitable for distributed deployment on resource-constrained MEC-enabled UAV nodes.
- We propose a dual-path extension with controlled redundancy, where a backup branch is created only when the primary next hop is deemed unreliable. An early termination (backup-branch death) rule further prevents redundant transmissions, thereby improving packet delivery ratio (PDR) with bounded replication overhead under interference.
- Extensive simulations evaluate the reliability–delay–energy tradeoffs and demonstrate clear performance gains over representative geographic, metric-driven, and learning-based baselines. Specifically, the proposed scheme achieves 8%–16% higher PDR than GPSR and 4%–13% lower delay than QGeo, while bounding dual-path energy overhead within 0%–29%.

The remainder of this paper is organized as follows. [Section 2](#) reviews related work. [Section 3](#) introduces the system model. [Sections 4](#) and [5](#) present the proposed Predictive-Q forwarding and dual-path extension mechanism. [Section 6](#) reports simulation results, and [Section 7](#) concludes the paper.

2 Related Work

2.1 Geographic and Learning-Based Routing in FANETs

Recent surveys [1] highlight that FANETs face fast topology variation, intermittent connectivity, and stringent Quality of Service (QoS) requirements, motivating reliability-aware routing designs.

Geographic routing remains a foundational approach due to its minimal state requirements. Karp and Kung [6] pioneered Greedy Perimeter Stateless Routing (GPSR) with greedy forwarding and perimeter recovery, but its 2D assumptions limit applicability in aerial environments. De Couto et al. [7] introduced the expected transmission count (ETX) metric for link-quality awareness, yet it captures only instantaneous conditions without predictive capability. Cui et al. [8] addressed 3D void regions through effective transmission range modeling, while Singh et al. [9] incorporated connectivity and link-quality metrics via a multi-criteria decision framework. Yang et al. [16] proposed route discovery based on connected dominating sets for MANETs. A common limitation across these methods is their reactive nature: forwarding decisions respond to current conditions rather than anticipating future link states.

Learning-based routing addresses this limitation by enabling adaptive policy learning. Q-learning [17], a model-free reinforcement learning algorithm, provides the foundation for these approaches. Jung et al. [10]

pioneered QGeo, integrating Q-learning with geographic progress. Mai et al. [18] further introduced link-quality and rate-control penalties into Q-learning rewards, improving PDR awareness, yet their fixed reward weighting lacks adaptability to mobility-induced topology changes. Li et al. [11] applied RL to energy-efficient path selection, demonstrating the potential of reward shaping for multi-objective optimization. Recent works further explore RL for UAV communications in 6G and smart-city scenarios [19,20], while multi-objective RL has also been studied for data transmission optimization in edge computing [21]. Despite these advances, a critical gap remains: most learning-based schemes either require substantial online exploration impractical under rapid topology changes, or optimize single objectives without explicitly addressing the reliability-overhead tradeoff under interference. Moreover, these methods rarely exploit the onboard edge computing capability of UAVs to perform predictive inference locally, missing the opportunity for real-time link-aware decision-making. Our work addresses this gap by integrating multi-factor predictive rewards with controlled dual-path redundancy.

2.2 Interference-Aware and Mobility-Aware Routing

Interference and spectrum denial can significantly reduce link reliability and increase retransmission-induced delay. Mittal et al. [2] and Zhan et al. [22] proposed cross-layer interference-aware designs jointly optimizing power control and routing. Li and Chen [5] developed radio-map based prediction to anticipate future interference conditions. Huang et al. [4] and Zheng et al. [3] investigated anti-interference and spectrum-denial-aware routing for hostile environments. From a resource optimization perspective, Li et al. [23] addressed robust position and power optimization for full-duplex UAV relay networks under interference, and Zhai et al. [24] jointly optimized UAV positioning, user association, and resource allocation for load-balanced aerial networks. These studies highlight the importance of interference-aware resource management but do not address per-packet routing decisions with predictive link-quality awareness, nor have they been integrated with MEC-based online routing intelligence.

Because FANET links may break quickly, mobility-aware link prediction is important for stable forwarding. Long et al. [12] exploited link prediction and traffic load balancing in UAV networks. Zheng and Chen [15] analyzed how mobility and propagation constraints impact aerial connectivity. Pieska et al. [14] modeled retransmission-aware delay under uncertain links, and Khawaja et al. [13] predicted the lifespan of multipath components for high-frequency channels. Nevertheless, link prediction alone does not specify how to systematically combine progress, link quality, and predicted stability into a lightweight online next-hop policy. Furthermore, these efforts typically do not provide a controlled redundancy mechanism with explicit early termination to bound overhead during severe link degradation.

2.3 Multipath Routing and Controlled Redundancy

Multipath routing improves reliability via path diversity but may incur significant overhead if redundancy is uncontrolled. Zhan et al. [22] investigated interference-aware multipath routing for UAV networks, demonstrating the benefit of path diversity under dynamic interference but without explicit overhead control mechanisms. This work motivates our controlled dual-path design: redundancy is triggered only under low-quality primary links, and a backup-branch death rule limits unnecessary cost while preserving reliability gains.

Table 1 provides a structured summary of the key categories discussed above, together with the open issues that motivate our work.

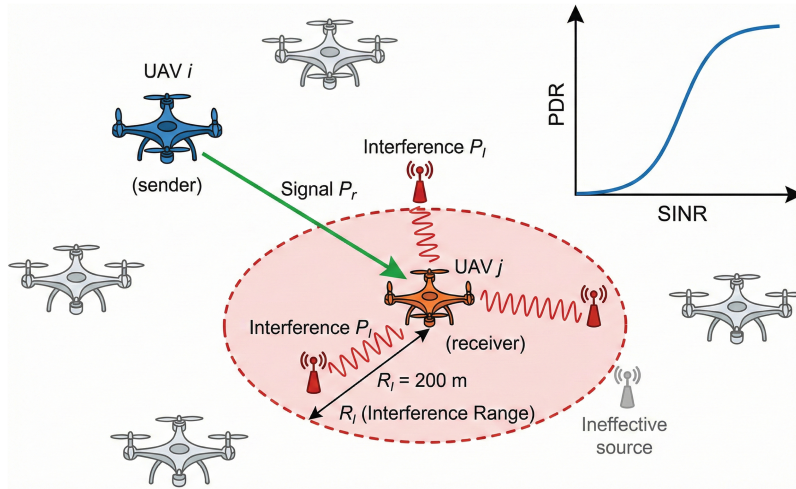
Table 1: Summary of related work.

Category	References	Characteristics
Survey	[1]	Comprehensive FANET survey highlighting fast topology variation as open challenge
Geographic/Metric-Driven	[6–9,16]	Position-based or hop-count-based forwarding lacking predictive link awareness
Learning-Based Routing	[10,11,17–21]	RL-driven routing with fixed or single-objective rewards, without interference–overhead tradeoff
Interference-/Mobility-Aware	[2–5,12–15,22–24]	Interference prediction or mobility modeling not integrated with per-packet online routing decisions

3 System Model

3.1 Time-Varying Network Topology

We consider a UAV swarm comprising N autonomous nodes operating in a three-dimensional airspace subject to external interference, as depicted in Fig. 1. The figure illustrates the UAV nodes distributed across the 3D airspace, a data sink node, and interference sources whose coverage radii affect nearby communication links. Directed edges between UAVs represent symmetric communication links established through mutual neighbor discovery. At time slot t , each node $i \in V = \{1, 2, \dots, N\}$ is characterized by its position $\mathbf{p}_i(t) \in \mathbb{R}^3$ and velocity $\mathbf{v}_i(t) \in \mathbb{R}^3$. Node mobility follows a 3D Gauss–Markov model. One designated node $d \in V$ serves as the stationary data sink.

**Figure 1:** UAV swarm networks under interference.

The instantaneous topology is modeled as a directed graph $G(t) = (V, E(t))$. A directed link $(i, j) \in E(t)$ exists if and only if $\|\mathbf{p}_i(t) - \mathbf{p}_j(t)\| \leq R_{\text{comm}}$ and $\text{PDR}_{i \rightarrow j}(t) > \tau$. Each node i maintains its neighbor set $\mathcal{N}_i(t)$ via periodic beacon exchanges, retaining only symmetric links (Section 3.3).

3.2 SINR-Based Link Reliability Model

Each UAV estimates the per-link packet success probability from the measured signal-to-interference-plus-noise ratio (SINR). Denote $P_r(i \rightarrow j, t)$ as the received signal power at node j (in dBm), P_N as the thermal noise floor, and $P_I(j, t)$ as the aggregate interference power observed at j from external sources such as interference emitters. The instantaneous SINR (in dB) is given by

$$\text{SINR}_{i \rightarrow j}(t) = P_r(i \rightarrow j, t) - 10 \log_{10}(P_N + P_I(j, t)). \quad (1)$$

The SINR value is then mapped to a packet delivery probability. When $\text{SINR}_{i \rightarrow j}(t) < 0$, we set $\text{PDR}_{i \rightarrow j}(t) = 0$; otherwise,

$$\text{PDR}_{i \rightarrow j}(t) = \left(1 - Q\left(\sqrt{2\kappa \cdot \text{SINR}_{i \rightarrow j}(t)}\right)\right)^L, \quad (2)$$

where $Q(\cdot)$ denotes the Gaussian Q-function, κ is a modulation-specific constant, and L represents the packet length in bits.

Because reliable routing requires both successful data delivery and acknowledgment reception, we define the bidirectional link reliability as

$$\text{PDR}_{ij}(t) = \text{PDR}_{i \rightarrow j}(t) \cdot \text{PDR}_{j \rightarrow i}(t). \quad (3)$$

3.3 Symmetric Neighbor Table Construction

The neighbor table at each node is formed by applying a PDR threshold in both directions and retaining only symmetric links

$$j \in \mathcal{N}_i(t) \iff \text{PDR}_{i \rightarrow j}(t) > \tau \wedge \text{PDR}_{j \rightarrow i}(t) > \tau, \quad (4)$$

This filtering eliminates asymmetric or highly unreliable links prior to the learning and forwarding stages, thereby enhancing routing stability under interference.

3.4 Retransmission-Induced Cost Amplification

A key observation from the link model is that both per-hop delay and energy consumption scale inversely with $\text{PDR}_{ij}(t)$. This scaling corresponds to the expected transmission count (ETX) under a geometric success model.

Let $\text{ETX}_{ij}(t) = 1/\text{PDR}_{ij}(t)$. Letting t_q and t_p denote the nominal queueing and processing delays per hop, the expected single-hop delay is

$$\mathbb{E}[D_{ij}(t)] = (t_q + t_p) \cdot \text{ETX}_{ij}(t) = \frac{t_q + t_p}{\text{PDR}_{ij}(t)}. \quad (5)$$

Similarly, the energy consumed per transmission attempt is

$$e_{ij}(t) = E_{\text{elec}}L + f_s L \cdot d_{ij}(t)^2, \quad d_{ij}(t) = \|p_i(t) - p_j(t)\|, \quad (6)$$

thus expected per-hop energy is

$$\mathbb{E}[E_{ij}(t)] = e_{ij}(t) \cdot \text{ETX}_{ij}(t) = \frac{e_{ij}(t)}{\text{PDR}_{ij}(t)}. \quad (7)$$

Key insight: As interference degrades PDR, both delay and energy grow approximately as $1/\text{PDR}$. This motivates Predictive-Q to explicitly incorporate link reliability into its reward while simultaneously controlling the hop count.

4 Predictive-Q Based Algorithm Design

To improve readability, this section is organized in a top-down manner following Fig. 2. Each subsection expands one stage of the pipeline: Step (1) neighbor discovery and MDP abstraction, Step (2) feature precomputation and warm-start initialization, Step (3) distance-adaptive weight selection, Step (4) reward synthesis, and Step (5) TD update with weighted exploration.

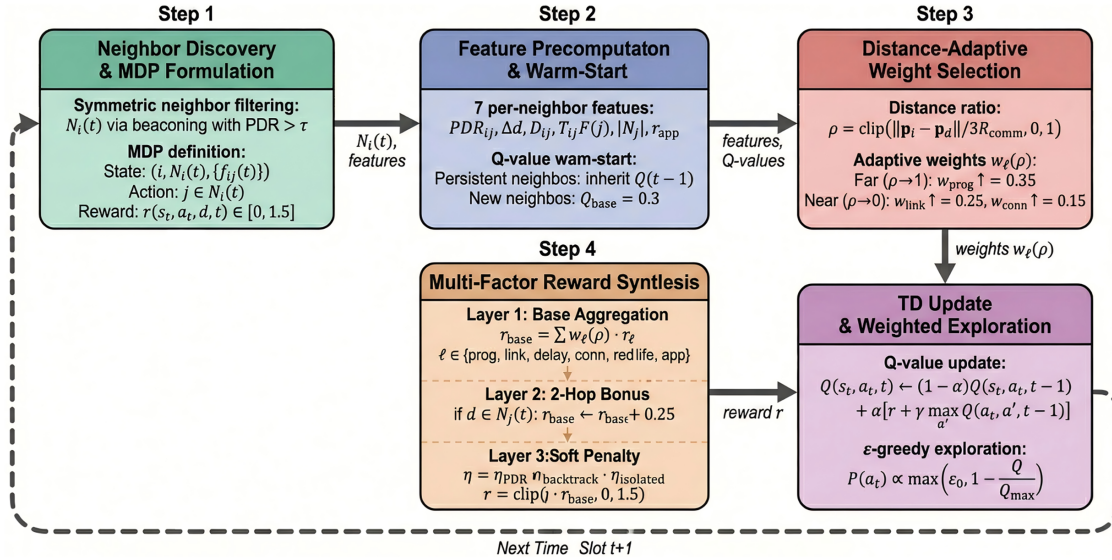


Figure 2: Predictive-Q learning flowchart.

4.1 Neighbor Discovery and MDP Formulation

At each time slot, node i first updates the feasible action set $\mathcal{N}_i(t)$ through symmetric-neighbor discovery (Section 3.3). Based on this local candidate set, the next-hop selection problem is cast as a Markov decision process (MDP):

- State: $s_t = (i, \mathcal{N}_i(t), \{f_{ij}(t)\}_{j \in \mathcal{N}_i(t)})$, where i is the current forwarding node and $f_{ij}(t)$ collects per-neighbor features, such as bidirectional PDR, distance, relative velocity, forward connectivity, and neighbor count. The state is time-varying due to topology and link evolution.
- Action: $a_t = j \in \mathcal{N}_{s_t}(t)$, selecting a next-hop from the current neighbor set. The action space changes dynamically with neighbor arrivals and departures.
- Transition: The packet is forwarded to j with possible retransmissions (Section 3.4). Upon successful delivery, the successor state is $s_{t+1} = (j, \mathcal{N}_j(t'), \{f_{jk}(t')\})$, where t' is the decision epoch at node j . The transition is stochastic due to link-level success probability $\text{PDR}_{ij}(t)$ and topology evolution between t and t' .
- Reward: A local multi-factor function $r(s_t, a_t, d, t) \in [0, 1.5]$ combining geographic progress, link quality, retransmission delay, forward connectivity, neighbor redundancy, predicted link lifetime, and approach alignment (detailed in Section 4.4).

The objective is to learn a policy π maximizing the expected discounted return $G_t = \sum_{k \geq 0} \gamma^k r_{t+k+1}$, where $\gamma \in (0, 1)$ is the discount factor.

4.2 Feature Precomputation and Warm-Start Initialization

For each candidate action $j \in \mathcal{N}_i(t)$, the node precomputes local features required by reward design, including reliability, progress, delay proxy, predicted lifetime, forward reachability, neighborhood size, and approach tendency. These correspond to PDR_{ij} , Δd , D_{ij} , T_{ij} , $F(j)$, $|\mathcal{N}_j|$, and r_{app} , respectively.

In parallel with feature extraction, the Q-table itself must remain continuous under fast topology changes. Because fast neighbor churn can otherwise reset learned values and delay adaptation, Predictive-Q inherits Q-values for persistent neighbors and initializes new neighbors with a small optimistic baseline

$$Q_{\text{pred}}(s_t, a_t, t) = \begin{cases} Q_{\text{pred}}(s_t, a_t, t-1), & a_t \in \mathcal{N}_{s_t}(t) \cap \mathcal{N}_{s_t}(t-1), \\ Q_{\text{base}}, & a_t \in \mathcal{N}_{s_t}(t) \setminus \mathcal{N}_{s_t}(t-1). \end{cases} \quad (8)$$

where $Q_{\text{base}} = 0.3$ is a small optimistic baseline to give new neighbors exploration opportunity.

This warm-start mechanism enables fresh neighbors to participate immediately in max-Q bootstrapping. It also preserves useful history for stable neighbors while avoiding zero-initialization delay.

4.3 Distance-Adaptive Weight Selection

Predictive-Q uses a normalized distance ratio to control reward weights. This design shifts emphasis from long-range progress toward reliability and stability as a packet approaches the destination:

$$\rho = \text{clip} \left(\frac{\|p_i - p_d\|}{3R_{\text{comm}}}, 0, 1 \right). \quad (9)$$

Table 2 shows the adaptive weight design.

Table 2: Distance-adaptive weights in predictive-Q.

Component	Weight $w_\ell(\rho)$	Range
Progress	$0.20 + 0.15\rho$	[0.20, 0.35]
Link (PDR)	$0.25 - 0.10\rho$	[0.15, 0.25]
Delay	0.15	[0.15]
Connectivity	$0.15 - 0.05\rho$	[0.10, 0.15]
Redundancy	0.10	[0.10]
Lifetime	0.10	[0.10]
Approach	0.05	[0.05]

The weight design follows a principle of diminishing progress priority: when far from the destination ($\rho \rightarrow 1$), geographic progress receives the highest weight (up to 0.35) to minimize hop count; as $\rho \rightarrow 0$, link reliability increases to 0.25 to avoid last-hop loss. The remaining components are kept constant at moderate magnitudes (0.05–0.15). A sensitivity analysis perturbing the three main weights by $\pm 20\%$ shows that PDR

deviates by at most 6.7% under weight perturbation, while the remaining five variants are within 3.2%, confirming robustness.

4.4 Multi-Factor Reward Synthesis

Direct end-to-end delay or energy optimization would require global path enumeration, so Predictive-Q uses seven normalized local reward components as proxies for long-term path quality.

4.4.1 Reward Components

For candidate next-hop j toward destination d , let $d_i = \|p_i - p_d\|$, $d_j = \|p_j - p_d\|$, $\Delta d = d_i - d_j$, and $F(j) = |\{k \in \mathcal{N}_j(t) : \|p_k - p_d\| < \|p_j - p_d\|\}|$ be the forward-neighbor count. Table 3 summarizes the reward components used in the weighted reward function.

Table 3: Reward components in predictive-Q.

Component	Formula	Description
Progress	$r_{\text{prog}} = \text{clip}\left(\frac{\Delta d}{0.8R_{\text{comm}}} + 0.5, 0, 1\right)$	Geographic advance toward d
Link quality	$r_{\text{link}} = \text{clip}(\text{PDR}_{ij}(t), 0, 1)$	Bidirectional PDR (Eq. (3))
Delay	$r_{\text{delay}} = \frac{t_q + t_p}{D_{ij}(t)}$	Inverse normalized delay (Eq. (5))
Fwd. connectivity	$r_{\text{conn}} = \min\left(1, \frac{F(j) + 0.5}{3}\right)$	Penalizes dead-end next hops
Redundancy	$r_{\text{red}} = \min\left(1, \frac{ \mathcal{N}_j(t) + 0.5}{4}\right)$	Path diversity at j
Approach	$r_{\text{app}} = \frac{1}{2} \left(\frac{\mathbf{v}_j^\top \mathbf{w}}{\ \mathbf{v}_j\ \ \mathbf{w}\ } + 1 \right)$	Velocity alignment toward d

As illustrated in Fig. 3, when $F(j) = 0$ and $d \notin \mathcal{N}_j(t)$, node j is a potential dead-end, yielding $r_{\text{conn}} \approx 0.17$.

The seventh component, link lifetime reward, requires a closed-form derivation. The predicted link lifetime T_{ij} is obtained by solving $\|\mathbf{r} + \mathbf{u}t\|^2 = R_{\text{comm}}^2$, where $\mathbf{r} = \mathbf{p}_i - \mathbf{p}_j$ and $\mathbf{u} = \mathbf{v}_i - \mathbf{v}_j$. Let $a = \|\mathbf{u}\|^2$, $b = 2(\mathbf{r}^\top \mathbf{u})$, $c = \|\mathbf{r}\|^2 - R_{\text{comm}}^2$, and $\Delta = b^2 - 4ac$. Then

$$T_{ij} = \begin{cases} 30, & a < 10^{-8} \text{ or } \Delta < 0, \\ \min(30, \frac{-b + \sqrt{\Delta}}{2a}), & \text{otherwise,} \end{cases} \quad (10)$$

where the cap of 30 s treats longer-lived links as effectively stable. The normalized reward is $r_{\text{life}} = \min(1, T_{ij}/15)$.

The constant relative-motion assumption underlying Eq. (10) is valid for the short prediction horizons typical of UAV routing decisions. In our simulations, the Gauss–Markov mobility model uses a correlation parameter of 0.5, ensuring smooth velocity transitions, and the 1 s time step keeps T_{ij} within the window

where relative motion can be treated as approximately linear. For scenarios with more aggressive maneuvering, prediction uncertainty can be incorporated by discounting r_{life} with a confidence factor based on historical velocity fluctuations, as outlined in the concluding discussion of this paper.

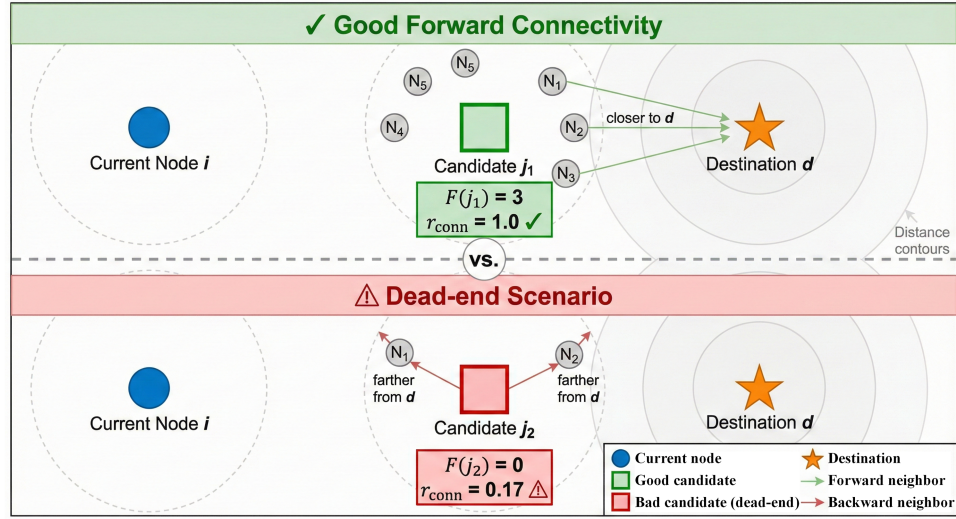


Figure 3: Forward connectivity example: candidate j_1 has multiple closer neighbors (large $F(j_1)$, high r_{conn}) vs. j_2 with no forward progress (dead-end, low r_{conn}).

4.4.2 Reward Aggregation and Penalty

The seven components are combined via distance-adaptive weighting:

$$r_{\text{base}} = \sum_{\ell \in \mathcal{L}} w_{\ell}(\rho) r_{\ell}, \quad (11)$$

where $\mathcal{L} = \{\text{prog, link, delay, conn, red, life, app}\}$. When the destination lies within the neighbor set of j (i.e., $d \in \mathcal{N}_j(t)$), a two-hop bonus is added:

$$r_{\text{base}} \leftarrow r_{\text{base}} + 0.25. \quad (12)$$

A multiplicative soft penalty $\eta \in (0, 1]$ suppresses risky actions without destabilizing learning:

$$\eta = \eta_{\text{PDR}} \cdot \eta_{\text{backtrack}} \cdot \eta_{\text{isolated}}, \quad (13)$$

where

$$\eta_{\text{PDR}} = \begin{cases} 0.6 + \text{PDR}_{ij}, & \text{PDR}_{ij} < 0.35, \\ 1.0, & \text{otherwise,} \end{cases} \quad (14)$$

$$\eta_{\text{backtrack}} = \begin{cases} 0.5, & \Delta d < 0 \wedge F(j) = 0 \wedge d \notin \mathcal{N}_j, \\ 0.75, & \Delta d < 0 \wedge \text{else,} \\ 1.0, & \Delta d \geq 0, \end{cases} \quad (15)$$

$$\eta_{\text{isolated}} = \begin{cases} 0.8, & |\mathcal{N}_j| \leq 1 \wedge d \notin \mathcal{N}_j, \\ 1.0, & \text{otherwise.} \end{cases} \quad (16)$$

The final reward is

$$r = \text{clip}(\eta \cdot r_{\text{base}}, 0, 1.5), \quad (17)$$

where the upper bound 1.5 accommodates the two-hop bonus. By keeping $r \geq 0$, soft penalties prevent negative Q-values from dominating the max-Q bootstrap.

4.5 Temporal-Difference Update and Weighted Exploration

Predictive-Q maintains $Q_{\text{pred}}(s, a, t)$ for each node-neighbor-time triple and performs temporal-difference updates as

$$Q_{\text{pred}}(s_t, a_t, t) \leftarrow (1 - \alpha)Q_{\text{pred}}(s_t, a_t, t - 1) + \alpha \left[r(s_t, a_t, t) + \gamma \max_{a' \in \mathcal{N}_{a_t}(t)} Q_{\text{pred}}(a_t, a', t - 1) \right], \quad (18)$$

where α is the learning rate and γ the discount factor. The bootstrap term references a_t as the successor state (since $s_{t+1} = a_t$), and using the previous time index $t - 1$ for the max-Q term stabilizes updates under concurrent learning across nodes.

To prevent premature convergence in non-stationary link conditions, we employ a decaying ϵ -greedy policy. Specifically, with probability $\epsilon(t)$ we explore by sampling the next hop from the biased distribution in Eq. (19), which favors lower-Q actions to improve coverage for under-explored neighbors; with probability $1 - \epsilon(t)$ we select the greedy action $a_t = \arg \max_{a \in \mathcal{N}_{s_t}(t)} Q_{\text{pred}}(s_t, a, t)$.

$$\Pr(a_t | \text{explore}) \propto \max \left(\epsilon_0, 1 - \frac{Q_{\text{pred}}(s_t, a_t, t)}{\max_{a' \in \mathcal{N}_{s_t}(t)} Q_{\text{pred}}(s_t, a', t) + \delta} \right), \quad (19)$$

where $\epsilon_0 = 0.1$ lower-bounds the sampling weight and $\delta = 0.01$ prevents division by zero. In our implementation, we use $\alpha = 0.55$, $\gamma = 0.50$, and $\epsilon(t) = \max(0.12, 0.30 - 0.005t)$.

4.6 Single-Path Forwarding Baseline

For comparative evaluation, we implement a single-path forwarding variant that directly uses the learned Q_{pred} as the primary next-hop selection criterion. This baseline incorporates lightweight runtime safeguards—loop prevention via visit counts, PDR-augmented scoring, and geographic fallback—that depend on per-packet trajectory history and therefore cannot be pre-encoded in the Q-table.

5 Dual-Path Forwarding Mechanism

5.1 Design Overview

Predictive-Q forwarding achieves robust performance under topology changes by selecting next hops based on the learned Q_{pred} table. However, in highly dynamic UAV networks, even a high-Q primary link may temporarily suffer from low PDR or abrupt link interruption, triggering repeated retransmissions, excessive delay, or premature route failure. To enhance reliability under such performance degradations without incurring heavy overhead of conventional multipath routing, we design a conditional dual-path forwarding scheme with three design goals:

1. Activate redundant path only when the primary link is unreliable;
2. Bound the redundant path to at most one extra replica (two concurrent branches);
3. Terminate unproductive backup path early to reduce cost.

Because each packet is allowed to split at most once, the number of concurrent replicas is upper-bounded by two, which provides a simple worst-case bound on replication overhead.

The dual-path mechanism operates at the packet level. Each forwarding node maintains a lightweight branch state for every packet replica, comprising the current node, remaining time-to-live (TTL), the path trace, and a per-branch visit-count map. One branch is designated as primary and follows the best-Q decision; a backup branch is created only once when the splitting rule is satisfied. Each branch independently proceeds hop by hop until it reaches the destination or meets a failure/termination condition. At the destination, the successful branch with minimum end-to-end delay is selected, while the energy cost is accounted over all replicas. The system-level architecture is illustrated in Fig. 4.

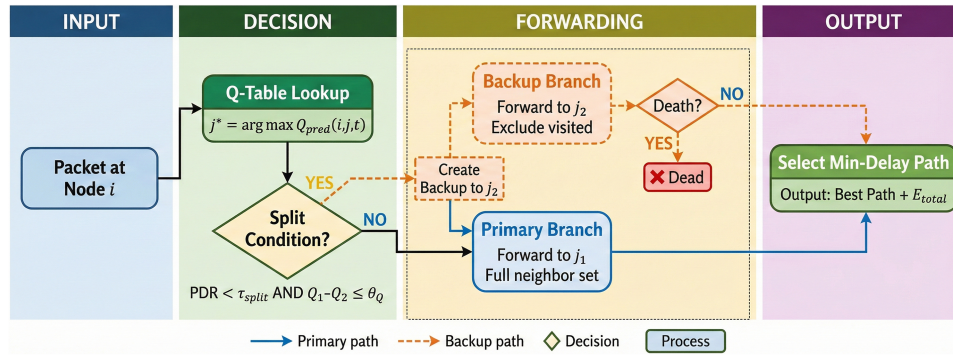


Figure 4: System-level architecture of the dual-path extension, from packet input and Q-table lookup to split decision, branch forwarding, death checking, and minimum-delay output selection.

5.2 Conditional Splitting Rule

At node i , let j_1 and j_2 denote the best and second-best next-hop candidates according to $Q_{\text{pred}}(i, \cdot, t)$, with Q-values $Q_1 = Q_{\text{pred}}(i, j_1, t)$ and $Q_2 = Q_{\text{pred}}(i, j_2, t)$. The backup branch is created only when the primary link reliability is below a threshold *and* the second-best candidate has competitive quality

$$\text{Split} \iff \text{PDR}_{ij_1}(t) < \tau_{\text{split}} \wedge Q_2 \geq \theta_Q Q_1, \quad (20)$$

where τ_{split} is the PDR threshold and $\theta_Q \in (0, 1)$ is a relative Q-quality threshold. This two-condition design ensures that (i) splitting occurs only when the primary link is genuinely unreliable, and (ii) the backup path has reasonable quality rather than a random alternative. In our implementation, we set $\tau_{\text{split}} = 0.75$ and $\theta_Q = 0.5$. The split decision is illustrated in Fig. 5.

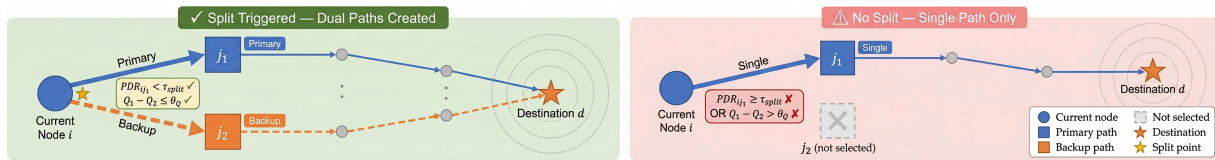


Figure 5: Split decision illustration. When both conditions are satisfied ($\text{PDR}_{ij_1} < \tau_{\text{split}}$ and $Q_2 \geq \theta_Q Q_1$), primary and backup branches are created; otherwise, only the primary path is used.

5.3 Loop Handling via Visit Counts

To control routing loops while maintaining forwarding flexibility, each branch maintains a visit-count map extracted from its path trace. At each hop, neighbors with visit count below a preset maximum are

prioritized; if no such neighbors exist, the primary branch falls back to using all neighbors, while the backup branch first checks the death rule and then either terminates or falls back to using all neighbors. TTL is decremented at every hop. When TTL reaches zero, the branch is marked as failed. In our implementation, the maximum visit count is set to 2.

5.4 Backup-Branch Early Termination (Death Rule)

A key difference from existing multipath schemes is that the backup branch is allowed to self-terminate when continued forwarding is unlikely to be beneficial. When no neighbor remains below the maximum visit-count limit, the backup branch compares the best available option against previously revisited neighbors. Let $Q_{\text{max-visited}}$ denote the maximum Q-value among neighbors that have reached the maximum visit count—these are nodes already exhausted by the branch. Let $Q_{\text{best-avail}}$ denote the maximum Q-value among the remaining neighbors with the smallest current visit count (ties broken by Q). If $Q_{\text{max-visited}} > 0$, the backup branch terminates when the best available option is sufficiently worse than those already revisited:

$$\text{Terminate} \iff Q_{\text{best-avail}} < \theta_{\text{death}} Q_{\text{max-visited}}, \quad (21)$$

where $\theta_{\text{death}} = 0.3$ in our implementation. The condition $Q_{\text{max-visited}} > 0$ ensures that termination only occurs after at least one neighbor has been exhausted, preventing premature termination when no node has reached the visit limit.

The rationale is as follows: if the backup branch has explored a region and the only forward options have much lower Q-values than nodes it has already passed through, then the branch is likely trapped in a suboptimal area. Continuing would waste energy without improving delivery probability. This per-hop check enables early termination as soon as the backup branch enters a dead-end region, rather than waiting until all options are exhausted.

5.5 Analysis of Complexity and Overhead

The per-node Q-table update involves feature precomputation with two-hop connectivity scanning costing $\mathcal{O}(\bar{d}^2)$ and up to K_{eff} Q-learning iterations each costing $\mathcal{O}(K_{\text{eff}}\bar{d})$, where $\bar{d}$ is the average neighbor degree. With the early convergence criterion ($|\Delta Q| < \sigma$), the effective number of iterations satisfies $K_{\text{eff}} \ll K_{\text{max}}$ in practice. Per-packet dual-path forwarding costs $\mathcal{O}(\bar{d}^2)$ per hop for multi-factor scoring. With at most two branches of T_{max} hops, the per-packet cost is $\mathcal{O}(2T_{\text{max}}\bar{d}^2)$, i.e., only a constant factor over single-path routing.

Let P denote the packet generation rate. The overall per-slot computational complexity is

$$\mathcal{O}(N^2 + N(\bar{d}^2 + K_{\text{eff}}\bar{d}) + P T_{\text{max}} \bar{d}^2), \quad (22)$$

where the first term corresponds to neighbor discovery via beacon exchanges and the remaining terms account for Q-table updates and packet forwarding, respectively.

For memory, the Q-table requires $\mathcal{O}(N^2)$ storage per node; per-packet state (path trace and visited map) requires $\mathcal{O}(T_{\text{max}})$ auxiliary memory, with no inter-branch signaling needed.

Overhead-wise, the dual-path extension incurs additional energy and bandwidth only when a backup branch is activated. Let p_{split} denote the probability of backup activation. The expected number of additional transmissions per packet is bounded by

$$\mathbb{E}[\text{Extra Tx}] \leq p_{\text{split}} \cdot T_{\text{backup}} \cdot E\bar{T}X, \quad (23)$$

where T_{backup} is the average number of hops traversed by the backup branch before termination or delivery, and $E\bar{T}X$ is the average expected transmission count per hop. The death rule truncates T_{backup} as soon as $Q_{\text{best-avail}} < \theta_{\text{death}} Q_{\text{max-visited}}$, ensuring that the backup branch does not persist in dead-end regions. In sparse topologies where p_{split} is higher due to fewer primary-path options, higher neighbor sparsity simultaneously lowers $\bar{d}$ and the death rule triggers earlier, naturally bounding the overhead.

Overall, the computational cost is dominated by the Q-table update and remains of the same order as standard Q-routing. The dual-path extension adds only a constant factor of two over single-path forwarding, without introducing exponential branching or cross-branch signaling. Empirically, under the default configuration on a standard laptop (AMD Ryzen AI 9 H 365, 32 GB RAM, MATLAB R2024b), per-slot processing completes in 120.3 ms on average, well within the 1 s simulation step, and the Q-table occupies only 79.7 KB per time slice. Per-packet routing takes 1.62 ms for Predictive-Q Dual and 1.00 ms for its single-path variant. These measurements, together with the theoretical bounds derived above, confirm that the proposed method is sufficiently lightweight for online execution on resource-constrained MEC-enabled UAV platforms.

6 Performance Evaluation

This section presents simulation results to evaluate the proposed Predictive-Q-Dual algorithm against four baseline methods under varying interference intensity, node density, and mobility speed.

6.1 Simulation Setup

We evaluate the proposed algorithm in MATLAB R2024b on a PC equipped with an AMD Ryzen 7 6800H CPU and 16 GB RAM. The simulation models a 3D UAV swarm operating in a $1000 \text{ m}^3 \times 1000 \text{ m}^3 \times 50 \text{ m}^3$ airspace with 3D Gauss–Markov mobility. External interference is modeled as mobile sources affecting receivers within radius $R_I = 300 \text{ m}$. Each simulation run spans 50 s. All reported results are averaged over 100 independent runs. The detailed parameters are summarized in [Table 4](#).

Table 4: Simulation settings.

Parameter	Value	Unit
Simulation area	$1000 \times 1000 \times 50$	m^3
Mobility model	3D Gauss–Markov	1
Simulation time/step	50/1	s
Communication range R_{comm}	250	m
Interference radius R_I	300	m
Neighbor threshold τ	0.2	1
Packet generation rate	16	packets/s
TTL (max hops)	30	hops
Per-hop delays t_q/t_p	10/4	ms
Split threshold τ_{split}	0.75	1
Number of UAV nodes N	50–100	1
Interference sources	0–5	1
Average node speed	20–70	m/s

Three performance metrics are evaluated: (i) Packet Delivery Ratio (PDR)—the fraction of packets successfully delivered; (ii) End-to-End Delay—average delay for delivered packets including retransmissions; and (iii) Energy Consumption—total transmission energy per packet including all branches.

Five algorithms are compared: GPSR [6] forwards to the geographically closest neighbor without link-quality awareness; QGeo [10] combines geographic routing with Q-learning but only rewards hop progress; Q-learning with Interference-Aware Routing (QIAR) is obtained by adapting the link-quality-aware Q-learning framework of QLR-FANET [18] to our interference scenario, incorporating instantaneous PDR into the reward but retaining fixed weights without mobility prediction; Predictive-Q-Single (Pred-Q-S) serves as the single-path ablation of our multi-factor Q-learning framework; and Predictive-Q-Dual (Pred-Q-D) adds the conditional dual-path extension.

6.2 Performance vs. Interference Intensity

Fig. 6 shows performance as the number of interference nodes increases from 0 to 5.

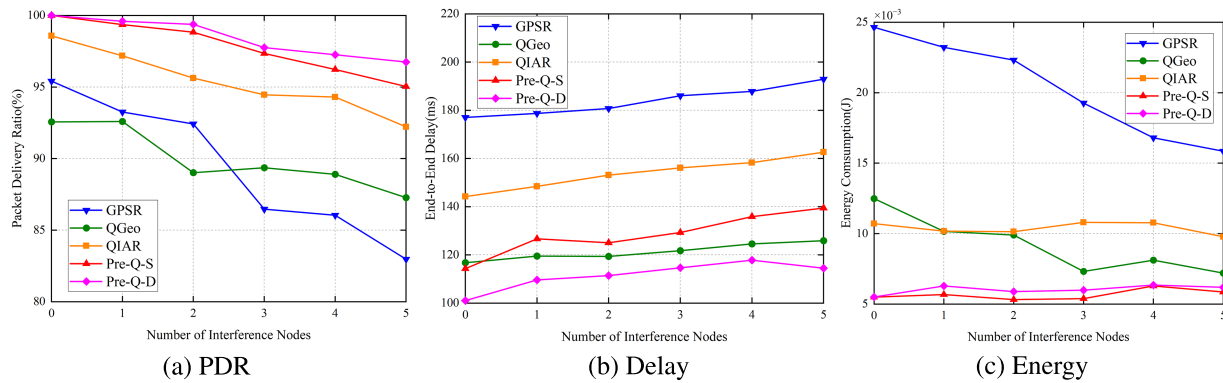


Figure 6: Performance vs. number of interference nodes ($N = 100$, speed 30 m/s).

For PDR (Fig. 6a), GPSR degrades most severely, from 95.4% to 83.0%, because its purely distance-based forwarding persistently selects interference-affected links without reliability feedback. QGeo mitigates this somewhat through Q-learning, yet its reward considers only geographic progress, so it still favors links with large distance advances despite low PDR. QIAR, adapted from QLR-FANET for interference awareness, benefits from instantaneous PDR feedback but its fixed weighting cannot adapt to changing interference intensity and offers no path-level redundancy for packets already committed to a degraded route. Pred-Q-S maintains PDR above 95.0% by combining multi-factor reward with distance-adaptive weighting that shifts priority toward link reliability near the destination. Pred-Q-D achieves the highest PDR, sustaining 96.7% even under 5 interference sources, because conditional splitting activates backup branches precisely when the primary link's PDR drops below τ_{split} .

For delay (Fig. 6b), GPSR suffers the highest delay, 177–193 ms, due to blind retransmissions on low-PDR links. Pred-Q-D achieves the lowest delay at 101–115 ms, outperforming QGeo by 9%–13%. This advantage arises from two mechanisms: r_{delay} penalizes high-retransmission links, and the dual-path output selection exploits the faster of two concurrent branches.

For energy (Fig. 6c), GPSR's consumption notably decreases from 24.6 to 15.9 mJ because more packets are dropped early, reducing total transmission attempts. Both Predictive-Q variants achieve the lowest energy, attributable to r_{delay} steering packets away from low-PDR links. The overhead of Pred-Q-D over

Pred-Q-S remains bounded at 0%–12%, confirming that the death rule and conditional splitting effectively prevent unnecessary backup transmissions.

6.3 Performance vs. Node Density

Fig. 7 shows the effect of varying the number of UAV nodes from 50 to 100.

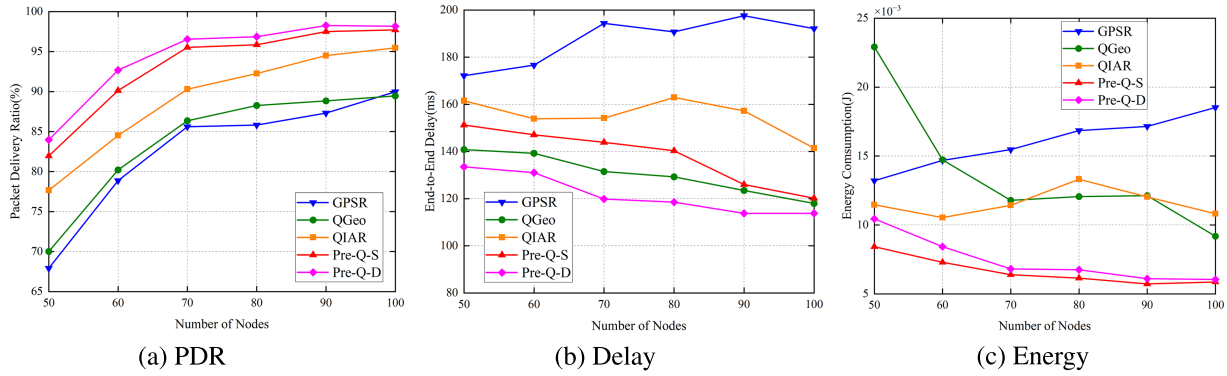


Figure 7: Performance vs. number of UAV nodes (speed 30 m/s, 2 interference sources).

For PDR (Fig. 7a), GPSR and QGeo improve sharply with density (67.9%→90.0% and 70.0%→89.5%, respectively), revealing that sparse topologies expose their core weaknesses—dead-ends for GPSR and indistinguishable link quality for QGeo. In contrast, Pred-Q-S starts at 82.0% even in sparse settings because r_{conn} and r_{red} explicitly penalize dead-end and isolated next hops. Pred-Q-D further reaches 98.2% at $N = 100$ by providing a backup path that hedges against limited forwarding options.

For delay (Fig. 7b), GPSR maintains the highest delay across all densities, while QGeo's delay decreases more rapidly as Q-learning benefits from richer neighbor sets. Pred-Q-D achieves the lowest delay throughout, outperforming QGeo by 4%–5%, because r_{delay} directly penalizes high-retransmission hops and the dual-path minimum-delay selection picks the faster branch.

For energy (Fig. 7c), GPSR's energy increases with density because more successful deliveries result in longer routes, while Q-learning-based algorithms reduce consumption as density provides shorter, more reliable paths. The overhead gap between Pred-Q-D and Pred-Q-S narrows from 24% at $N = 50$ to 3% at $N = 100$, because denser networks yield higher primary-path PDR and trigger the conditional splitting rule in Eq. (20) less frequently.

6.4 Performance vs. Mobility Speed

Fig. 8 evaluates robustness under mobility speeds ranging from 20 to 70 m/s.

For PDR (Fig. 8a), GPSR fluctuates between 67.1% and 81.1%, and QGeo ranges from 68.3% to 83.8%, as neither considers link stability. QIAR suffers the steepest degradation, from 87.4% to 65.9%, because its fixed weights lack mobility prediction, causing Q-values to become outdated as faster topology changes invalidate learned policies. Pred-Q-S mitigates this through r_{life} derived from relative velocity in Eq. (10), proactively steering packets away from links about to break. Pred-Q-D further improves PDR to 83.0%–95.1% by providing a backup branch that hedges against unpredicted link breaks.

For delay (Fig. 8b), GPSR suffers the highest delay because link breaks force perimeter-mode recovery with longer detours. Pred-Q-D achieves the lowest delay, outperforming QGeo by 4%–7%. The advantage

grows with speed because the dual-path mechanism provides a pre-established alternative when the primary link breaks, eliminating the recovery latency of single-path schemes.

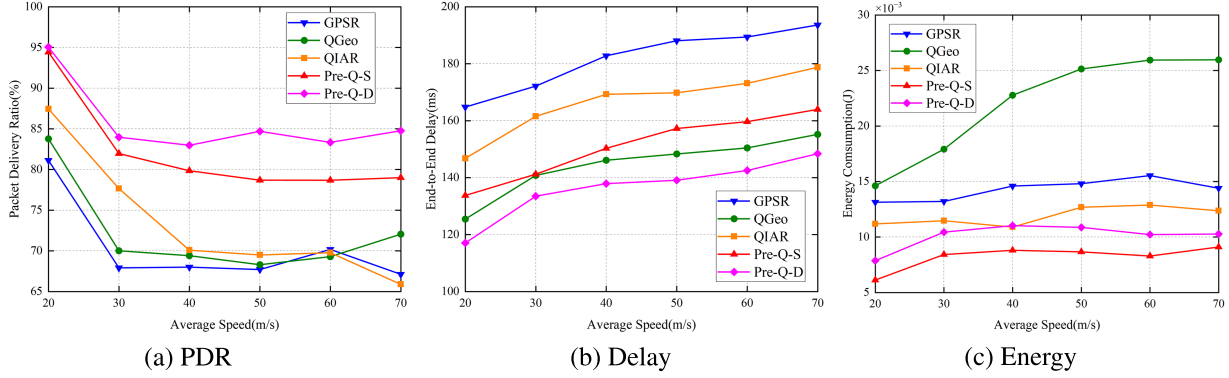


Figure 8: Performance vs. average node speed ($N = 50$, 2 interference sources).

For energy (Fig. 8c), QGeo shows a disproportionate increase from 14.6 to 26.0 mJ because its Q-learning struggles to converge under rapidly changing topology. Both Predictive-Q variants maintain the lowest energy. The dual-path overhead of 13%–29% is higher than in the interference experiment because faster mobility triggers more split events; nevertheless, the death rule in Eq. (21) bounds the cost by terminating backup branches once their best available Q-value falls below that of previously revisited neighbors.

7 Conclusion

This paper develops a Predictive-Q-Dual learning framework for reliable multi-hop routing in MEC-enabled interference-impaired high-mobility UAV swarm networks. The core Predictive-Q algorithm constructs a multi-factor reward that jointly captures link reliability, mobility-induced stability, forwarding progress, and neighborhood topology, combined with distance-adaptive weighting and warm-start Q-value inheritance for stable online learning. To handle transient link degradations, a lightweight dual-path extension conditionally splits a backup branch when the primary link becomes unreliable and terminates it early via a Q-value guided death rule, achieving reliability gains with bounded overhead. Simulations across varying interference, node density, and mobility speed show that Predictive-Q-Dual achieves more than 15% PDR improvement over GPSR. Compared with its single-path variant, the dual-path extension provides an additional 3%–5% PDR gain through controlled redundancy while bounding energy overhead at 5%–15%.

Looking beyond the present work, several avenues remain open. The Predictive-Q framework naturally extends to deep reinforcement learning, where Deep Q-Networks (DQN) or actor-critic architectures could learn Q-functions from raw link states, improving generalization without handcrafted feature engineering. A multi-agent extension is equally compelling: UAVs sharing learned Q-values or adopting federated training could accelerate convergence across larger swarms with partial observability. Cross-layer integration offers a further direction—jointly optimizing routing with power control and trajectory planning may yield additional energy gains on resource-constrained MEC platforms. In terms of open challenges, the constant relative-motion assumption in the link lifetime predictor should be relaxed for aggressive maneuvering scenarios, potentially through variance-based confidence factors or Kalman-filter-based relative-state estimation. Robustness to adversarial interference and the integration of semantic communication priors also remain important targets for future investigation.

Acknowledgement: Not applicable.

Funding Statement: This work was supported by National Natural Science Foundation of China (No. 62271402) and the Laboratory Construction and Management Research Project of Shandong University (No. sy20253502).

Author Contributions: Conceptualization, Zhihao Dong, Huakui Sun and Daosen Zhai; Methodology, Zhihao Dong and Huakui Sun; Software, Zhihao Dong; Validation, Zhihao Dong and Yueyue Tao; Formal Analysis, Zhihao Dong and Huakui Sun; Investigation, Zhihao Dong and Huakui Sun; Data Curation, Zhihao Dong and Yueyue Tao; Writing—Original Draft, Zhihao Dong; Writing—Review & Editing, Huakui Sun and Daosen Zhai; Visualization, Zhihao Dong; Supervision, Huakui Sun and Daosen Zhai; Project Administration, Huakui Sun; Funding Acquisition, Huakui Sun and Daosen Zhai. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Hasan SA, Mohammed MA, Sulaiman SK. Flying Ad-Hoc Networks (FANETs): review of communications, challenges, applications, future direction and open research topics. ITM Web Conf. 2024;64(5):01002. doi:10.1051/itmconf/20246401002.
- Mittal P, Shah S, Agarwal A, Mishra D, Debnath S. Interference aware joint power control and routing optimization in Multi-UAV FANETs. Ad Hoc Netw. 2023;150(4):103280. doi:10.1016/j.adhoc.2023.103280.
- Zheng S, Zhu X, Qin Z, Dong C. Cross-layer UAV network routing protocol for spectrum denial environments. Ad Hoc Netw. 2025;167(2):103702. doi:10.1016/j.adhoc.2024.103702.
- Huang Z, Sheng Z, Nasir AA, Yu H. Energy efficiency maximization for UAV-assisted full-duplex communication in the presence of multiple malicious jammers. IEEE Syst J. 2024;18(2):1257–68. doi:10.1109/jsyst.2024.3390554.
- Li B, Chen J. Radio map-assisted approach for interference-aware predictive UAV communications. IEEE Trans Wirel Commun. 2024;23(11):16725–41. doi:10.1109/twc.2024.3446240.
- Karp B, Kung HT. GPSR: greedy perimeter stateless routing for wireless networks. In: Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom); 2000 Aug 6–11; Boston, MA, USA. p. 243–54.
- De Couto DSJ, Aguayo D, Bicket J, Morris R. A high-throughput path metric for multi-hop wireless routing. In: Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom); 2003 Sep 14–19; San Diego, CA, USA. p. 134–46.
- Cui Y, Tian H, Chen C, Ni W, Wu H, Nie G. New geographical routing protocol for three-dimensional flying ad-hoc network based on new effective transmission range. IEEE Trans Veh Technol. 2023;72(12):16135–47. doi:10.1109/tvt.2023.3296082.
- Singh V, Sharma KP, Verma HK. CLGR: connectivity and link quality aware geographical routing using AHP-TOPSIS for FANETs. Peer-to-Peer Netw Appl. 2024;18(1):63. doi:10.1007/s12083-024-01879-5.
- Jung WS, Yim J, Ko YB. QGeo: Q-learning-based geographic ad hoc routing protocol for unmanned robotic networks. IEEE Commun Lett. 2017;21(10):2258–61.
- Li J, Xiao L, Qi X, Lv Z, Chen Q, Liu YJ. Reinforcement learning based energy-efficient fast routing for FANETs. IEEE Trans Commun. 2024;72(11):7063–76. doi:10.1109/tcomm.2024.3409561.
- Long H, Hu F, Kong L. Enhanced link prediction and traffic load balancing in unmanned aerial vehicle-based cloud-edge-local networks. Drones. 2024;8(10):528. doi:10.3390/drones8100528.
- Khawaja W, Jacobsen RH, Hussain S, Guvenc I. Predicting lifespan of ground-to-air multipath components in mmWave UAV channels. In: Proceedings of the 2025 IEEE Wireless Communications and Networking Conference (WCNC); 2025 Mar 24–27; Milan, Italy. p. 1–6.

14. Pieska M, Brunstrom A, Kassler A. On the feasibility of reliable tunneling: modelling delay imposed by re-transmissions. In: Proceedings of the 2023 13th International Workshop on Resilient Networks Design and Modeling (RNDM); 2023 Sep 20–22; Hamburg, Germany. p. 1–8.
15. Zheng Y, Chen J. Geography-aware optimal UAV 3D placement for LOS relaying: a geometry approach. *IEEE Trans Wirel Commun.* 2024;23(8):9301–14. doi:10.1109/twc.2023.3301613.
16. Yang Z, Fu D, Han L, Jhang ST. Improved route discovery based on constructing connected dominating set in MANET. *Int J Distrib Sens Netw.* 2015;2015(4):612102. doi:10.1155/2015/612102.
17. Sutton RS, Barto AG. Reinforcement learning: an introduction. 2nd ed. Cambridge, MA, USA: MIT Press; 2018.
18. Mai CT, Nguyen THL, Le Huu B, Vo TT. QLR-FANET: a Q-learning and rate control-based routing protocol for flying ad hoc network. *ETRI J.* 2025;47(6):1015–27. doi:10.4218/etrij.2024-0298.
19. Nam VH, Jeon G, Chehri A, Thanh BT, Quy VK. An improved reinforcement learning approach for sustainable 6G UAV communications. *Exp Syst.* 2026;43(2):e70192. doi:10.1111/exsy.70192.
20. Nam VH, Hue CTM, Anh DV. An improved reinforcement learning-based 6G UAV communication for smart cities. *Comput Mater Contin.* 2026;86(1):1–15. doi:10.32604/cmc.2025.070605.
21. Li X, Liu H, Wang H. Data transmission optimization in edge computing using multi-objective reinforcement learning. *J Supercomput.* 2024;80(14):21179–206. doi:10.1007/s11227-024-06213-7.
22. Zhan C, Hu H, Liu Z, Wang J, Fan R. Interference-aware online optimization for cellular-connected multiple UAV networks with energy constraints. *IEEE Trans Mob Comput.* 2024;23(12):13804–20. doi:10.1109/tmc.2024.3438759.
23. Li H, Zhai D, Zhang R, Liu L, Niyato D, Zhang Y. Robust position and power optimization for full-duplex UAV relay-assisted cellular network enhanced by NOMA. *IEEE Trans Wirel Commun.* 2026;25:7955–69. doi:10.1109/twc.2025.3634617.
24. Zhai D, Li H, Tang X, Zhang R, Cao H. Joint position optimization, user association, and resource allocation for load balancing in UAV-assisted wireless networks. *Digit Commun Netw.* 2024;10(1):25–37. doi:10.1016/j.dcan.2022.03.011.