Computers, Materials &
Continua

Tech Science Press

Doi:10.32604/cmc.2025.070605

ARTICLE

An Improved Reinforcement Learning-Based 6G UAV Communication for Smart Cities

Vi Hoai Nam[1], **Chu Thi Minh Hue**[2] **and Dang Van Anh**[1,*]

[1]Faculty of Information Technology, Hung Yen University of Technology and Education, HungYen, 170000, Viet Nam
[2]Visiting Lecturer, Faculty of Software Technology, FPT University, Ha Noi, 100000, Viet Nam

*Corresponding Author: Dang Van Anh. Email: dangvananh@utehy.edu.vn

Received: 20 July 2025; Accepted: 22 September 2025; Published: 10 November 2025

ABSTRACT: Unmanned Aerial Vehicles (UAVs) have become integral components in smart city infrastructures, supporting applications such as emergency response, surveillance, and data collection. However, the high mobility and dynamic topology of Flying Ad Hoc Networks (FANETs) present significant challenges for maintaining reliable, low-latency communication. Conventional geographic routing protocols often struggle in situations where link quality varies and mobility patterns are unpredictable. To overcome these limitations, this paper proposes an improved routing protocol based on reinforcement learning. This new approach integrates Q-learning with mechanisms that are both link-aware and mobility-aware. The proposed method optimizes the selection of relay nodes by using an adaptive reward function that takes into account energy consumption, delay, and link quality. Additionally, a Kalman filter is integrated to predict UAV mobility, improving the stability of communication links under dynamic network conditions. Simulation experiments were conducted using realistic scenarios, varying the number of UAVs to assess scalability. An analysis was conducted on key performance metrics, including the packet delivery ratio, end-to-end delay, and total energy consumption. The results demonstrate that the proposed approach significantly improves the packet delivery ratio by 12%–15% and reduces delay by up to 25.5% when compared to conventional GEO and QGEO protocols. However, this improvement comes at the cost of higher energy consumption due to additional computations and control overhead. Despite this trade-off, the proposed solution ensures reliable and efficient communication, making it well-suited for large-scale UAV networks operating in complex urban environments.

KEYWORDS: UAV; FANET; smart cities; reinforcement learning; Q-learning

1 Introduction

In recent years, the smart city concept has emerged and attracted interest from both academic and industry communities. Smart cities incorporate cutting-edge technologies to improve citizens' quality of life, enhance the efficiency of urban governance, and enable rapid response to emergencies [1]. Within this context, UAVs have emerged as versatile tools with a broad range of applications, including security surveillance [2], disaster response [3], traffic management [4], logistics and delivery services [5], and emergency medical delivery [6] as well as the provision of mobile network connectivity in critical scenarios as presented in Fig. 1. However, ensuring the efficient operation of UAVs in complex urban environments poses significant challenges, foremost among them being the maintenance of continuous and reliable communication links. The high density of skyscrapers, the presence of unexpected obstacles, and the prevalence of radio frequency interference considerably increase the risk of connection loss, leading to communication disruptions and elevated energy consumption by the UAVs. Conventional routing protocols often lack the agility and adaptability required to cope with the rapidly changing conditions characteristic of urban settings, thereby imposing substantial limitations on both performance and system reliability. To address these limitations, recent research has increasingly explored the application of Reinforcement Learning (RL) to develop more adaptive and flexible routing protocols [7]. RL enables UAVs to learn and dynamically adjust their routing strategies in real-time, thereby enhancing their ability to cope with continuously evolving urban environments. However, most existing studies have primarily evaluated RL approaches under idealized conditions or simplified models, which fall short of capturing the real-world complexities inherent in smart city scenarios.

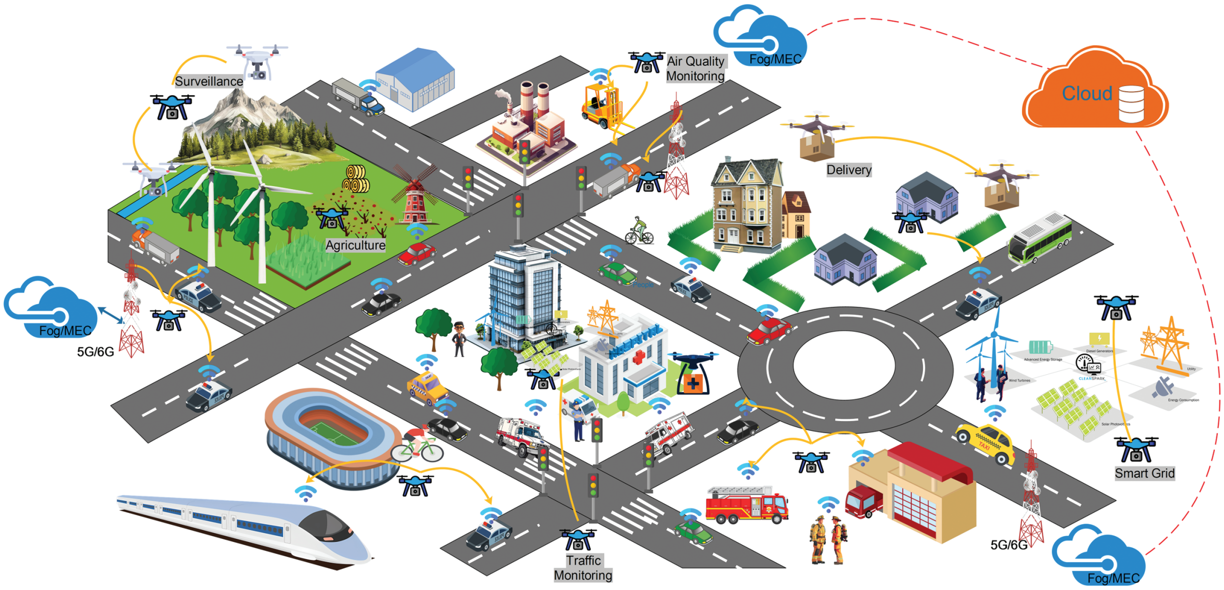

Figure 1: An illustration of UAV-based applications in smart cities

In this study, we propose an enhanced Q-learning routing scheme for UAV networks. Our method focuses on developing an optimized reward function that is better suited for the highly dynamic environments characteristic of UAV operations, addressing the limitations of traditional Q-learning techniques. The key contributions of this work can be summarized as follows:

- Propose an enhanced Q-learning-based routing protocol for UAV ad hoc networks in smart city environments.
- Design an optimized reward function for adaptive relay selection.
- Integrate a Kalman filter to predict UAV mobility and improve routing stability under dynamic conditions.
- Perform thorough simulations under realistic scenarios to assess packet delivery ratio, delay, and energy efficiency, proving to be superior to GEO and QGEO protocols.

The rest of this paper is organized as follows. Section 2 shows related works. Section 3 discusses reinforcement learning, Q-learning techniques, and introduces the proposed protocol. Section 4 presents the results with detailed discussions. Finally, Section 5 summarizes key findings and their implications and outlines future research directions.

2 Related Works

In this section, we review a selection of representative works that utilize deep reinforcement learning, multi-agent coordination, and hybrid optimization frameworks to improve strategies for UAV deployment, routing protocols, and collaborative decision-making. The studies are organized by their focus areas, which include UAV base station positioning, geolocation-based routing, distributed task assignment, and adaptive multi-hop communication in FANETs.

The studies summarized in Table 1 were conducted under heterogeneous simulation settings, physical assumptions, and evaluation criteria (e.g., number of UAVs, channel models, and traffic patterns). Therefore, directly reporting quantitative values across these works would not provide a fair or meaningful comparison. For this reason, Table 1 presents the information qualitatively, focusing on the performance metrics considered in each study rather than comparing absolute numerical results.

Table 1: Comparison of existing reinforcement learning-based approaches for UAV routing

Work	Year	Approach	PDR	Delay	Energy	Bandwidth	Privacy/Other
[8]	2023	ACDQL	x	x		x	
[9]	2023	DQN	x			x	
[10]	2023	DT & DQN		x	x		x
[11]	2023	Actor critic	x		x		
[12]	2024	MRL	x	x	x	x	x
[13]	2024	MADRL			x		x
[14]	2025	DRL	x	x			x
[15]	2025	Q-ANT	x	x		x	
[16]	2025	Q-learning	x	x			
[17]	2025	Q-learning			x	x	
[18]	2025	CNN & Actor-Critic		x	x		x
[19]	2025	Q-learning	x	x	x	x	

The research [8] proposes a continuous actor-critic deep Q-learning (ACDQL) algorithm to optimize the deployment of UAV-mounted base stations (UAV-BSs) towards 6G small cells in smart city environments. The framework addresses the complex challenge of continuously optimizing the placement of UAV base stations (BS) while considering mobile endpoints to maximize network performance. Simulation results indicate that the proposed ACDQL model increases the sum data rate to 45 Mbps, surpassing traditional Q-learning (35 Mbps) and DQL methods (42 Mbps), while also accelerating convergence by up to 85%.

This study [9] proposes AGLAN, an adaptive geolocation-based routing protocol for UAV networks, enhanced with reinforcement learning. By combining geolocation-derived IP addressing with a learned forwarding region (FR), AGLAN eliminates traditional routing tables and reduces overhead. A Deep Q-Network with pseudo-attention dynamically adjusts the FR angle to optimize data forwarding under UAV mobility. The learning model considers UAV mobility errors and dynamically updates routing rules in a 3D simulation environment. AGLAN ensures routing adaptability without requiring constant neighbor state exchanges, making it well-suited for decentralized UAV swarms. Simulation results show that AGLAN reduces packet loss by up to 6.8%, lowers jitter by 61%, and improves bandwidth efficiency by 9% compared to baseline protocols.

The research [10] proposes a task assignment framework for multi-UAV systems utilizing Digital Twins (DT) and a Deep Q-Learning Network (DQN) to manage tasks with random arrivals and strict time constraints efficiently. The proposed framework consists of two distinct stages. The first stage is the task assignment phase, in which tasks are subdivided and allocated to UAVs based on the shortest flying distance using a Genetic Algorithm (GA). The second stage is the dynamic task reassignment phase. UAVs utilize deep learning models that have been pretrained and deployed on airships to quickly adjust their assignments using a DQN.

To address the challenge of efficient routing in highly dynamic UAV swarm networks, the work [11] proposes a routing scheme based on Multi-Agent Reinforcement Learning (MARL), incorporating zone-based partitioning and adaptive inter-UAV communication. Specifically, the authors formulate the routing problem as a decentralized partially observable Markov decision process (Dec-POMDP) and integrate an actor–critic neural architecture, comprising a Multilayer Perceptron and a Gate Recurrent Unit (GRU), to enable cooperative routing strategy learning.

In research [12], the authors propose a distributed routing optimization algorithm tailored for FANETs, leveraging multi-agent reinforcement learning (MARL) through a DE-MADDPG framework integrated with an adaptive multi-protocol routing mechanism (AMAHR). Each UAV functions as an independent agent, autonomously adjusting routing protocols and parameters based on local network information to enhance overall network performance.

To optimize the positioning of unmanned aerial vehicle base stations (UAV-BSs) for aerial Internet service provisioning in urban environments, the study [13] proposes a cooperative positioning algorithm based on multi-agent deep reinforcement learning (MADRL), integrated with a CommNet architecture under the centralized training and distributed execution (CTDE) framework. Simulation results demonstrate that the proposed algorithm significantly improves quality of service (QoS), enhances user connectivity, reduces coverage overlap, and maintains superior energy efficiency compared to other MADRL-based approaches.

The study [14] presents a deep reinforcement learning-based framework to enhance routing performance in UAV-aided Vehicular Ad Hoc Networks (VANETs), integrating a multi-agent soft actor–critic (MASAC) algorithm with an Adaptive Dual-Model Routing (ADMR) protocol. The MASAC-based trajectory planner enables UAVs to autonomously adjust their flight paths in 3D space for optimal communication coverage. Simulation results demonstrate that the proposed approach improves the packet delivery ratio (PDR) by up to 15%, reduces end-to-end delay by approximately 20%, and achieves a network reachability ratio consistently above 90%, outperforming benchmark methods such as MADDPG, SZLS-GPSR, and AFR-OLSR.

To optimize routing in FANETs, the research [15] proposes the Q-ANT framework, which integrates Q-learning with Ant Colony Optimization. Q-ANT enhances adaptability in dynamic UAV environments by dynamically adjusting routing weights and filtering unstable links based on residual time and signal quality. Simulations show that Q-ANT significantly outperforms existing methods (QRF, FAnt-HocNet), achieving higher packet delivery rates, lower latency, and stable throughput up to 45 Mbps under high traffic, making it well-suited for high-speed, dense UAV networks.

To enhance data forwarding efficiency in FANETs, the work [16] proposes an RL-based framework approach that integrates a Q-learning-enabled Data Forwarding Agent (DFA) deployed on UAVs. This model allows UAVs to dynamically determine optimal next-hop decisions for packet transmission based on real-time environmental states, including neighbor positions, energy levels, and data volume. The simulation results show that the proposed method reduces end-to-end delay by 45 ms, lower than Greedy Forwarding (105 ms) and Random Forwarding (60 ms). To enhance multi-hop routing in UAV-assisted networks, research [17] proposes an improved Q-learning-based algorithm (IQLMR). Unlike traditional Q-routing, IQLMR integrates link stability, node energy, and hop count into the Q-value update process, allowing UAVs to select more reliable and energy-efficient paths dynamically.

The research [18] proposes a distributed inference framework for resource-constrained UAV swarms using multi-agent meta reinforcement learning (Meta-RL) to ensure reliable and low-latency performance. Simulation results demonstrate that the Meta-RL method significantly reduces inference latency by approximately 29% and transmission energy by around 23%, outperforming traditional reinforcement learning algorithms and closely matching theoretical optimization solutions. The study [19] presents a new routing protocol called Traj-Q-GPSR, which is designed to enhance data delivery in mission-oriented FANETs. This protocol builds upon the traditional Greedy Perimeter Stateless Routing (GPSR) by incorporating a two-hop trajectory-aware neighbor table and a Q-learning agent at each UAV node. Simulation results using NS-3 demonstrate that Traj-Q-GPSR reduces end-to-end delay by over 80%, lowers packet loss rate by approximately 23%, and increases routing efficiency by up to 52% compared to baseline GPSR.

The analyses above highlight the numerous RL techniques developed to improve the efficiency of UAVs. Some notable methods include deep Q-learning, actor-critic models, and multi-agent reinforcement learning, as well as CNN, which are utilized to manage mobility, privacy, security, offloading, and routing. While there have been attempts to integrate Q-learning into routing strategies, comprehensive evaluations under realistic smart city scenarios are still lacking. In this study, we propose a novel RL-based routing protocol aimed at optimizing data forwarding for UAVs in smart city environments. The following sections will discuss detailed technical issues related to this protocol.

3 Proposed Solution

3.1 Reinforcement Learning and Q-Learning

Reinforcement Learning is a typical AI technique that focuses on sequential decision-making problems in uncertain environments. In RL, an agent continuously interacts with the external environment through actions, aiming to optimize an objective function known as the expected cumulative reward. The agent's primary goal is to learn a policy-a function that maps the state of the environment to an action-such that the total reward accumulated in the long run is maximized. The learning process in RL is described using the Markov Decision Process (MDP) model or its extensions, such as Partially Observable Markov Decision Process (POMDP), when the environment's state cannot be fully observed. Key constituents within the Reinforcement Learning paradigm include:

- **Agent:** The decision-making entity that selects actions predicated on its current policy and perceived state.
- **Environment:** The external system with which the agent interacts, providing feedback through new observations and rewards.
- **State (S):** A representation of the environment's current configuration or the observations accessible to the agent.
- **Action (A):** The choices available to the agent for interacting with and influencing the environment.
- **Reward (R):** An immediate feedback signal from the environment, evaluating the consequence of the agent's most recent action.
- **Policy (π):** The agent's strategy or set of rules to determine its action selection based on the prevailing state.

The interplay among these elements unfolds: At a given time step t, the agent perceives the environment's state, denoted as S_t. Guided by its policy π, the agent executes an action A_t. This action subsequently perturbs the environment, causing a transition to a new state S_{t+1} and eliciting a reward R_{t+1}. This iterative sequence persists until a predefined termination criterion is met or an episode concludes, such as Fig. 2.

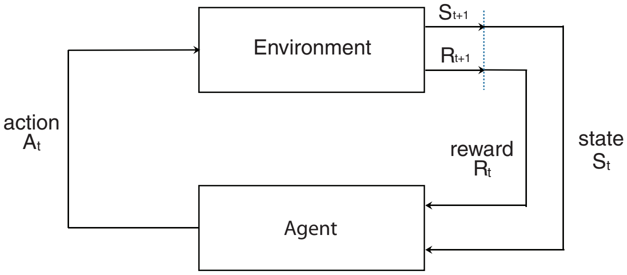

Figure 2: End-to-end delay & number of UAVs

Q-learning is a reinforcement learning algorithm that enables an agent to determine the optimal policy by interacting with the environment and learning the action-value function [20]. As an off-policy algorithm, it learns the optimal policy independently of the agent's current policy. This approach balances exploration, where actions are chosen randomly to discover new strategies, and exploitation, where actions are selected based on estimated optimal values. In Q-learning, the Q-value represents the expected future reward for taking a specific action in a given state and subsequently following the optimal policy. The Q-value is updated using the Eq. (1), as follows.

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[r + \gamma \max_{a'} Q(s',a') - Q(s,a) \right] \tag{1}$$

where,

- s is the current state.
- a is the action taken.
- r is the immediate reward received.
- s' is the next state after taking action a.
- $\alpha \in (0,1)$ is the learning rate (determines the step size of updates).
- $\gamma \in [0,1]$ is the discount factor (controls the importance of future rewards).
- $\max_{a'} Q(s',a')$ represents the maximum estimated future reward.

3.2 Optimize Q-Learning Proposed

This section proposes a solution by designing and storing Q-tables on UAVs, using location prediction features, establishing novel reward determination functions, and optimizing route selection policies to enhance sustainable and efficient UAV communication.

3.2.1 Q-Table Structure in Q-Proposed

In conventional Q-learning approaches, the Q-table is typically implemented as a fixed-size matrix of dimension $n \times n$, where n represents the number of UAVs (nodes). Each entry encodes the expected utility of forwarding a packet from a given UAV to another, assuming a fully enumerated set of discrete states and actions. However, in FANETs, where topology and network conditions change rapidly, maintaining a full Q-table becomes impractical due to the exponential growth of the state-action space and the sparsity of functional interactions. The Q-Proposed algorithm introduces a compact and adaptive Q-table structure to address this scalability issue. Instead of maintaining Q-values for all possible node pairs, Q-proposed discretizes the state space based on the UAV's distance to the depot into a fixed number of distance bins. The Q-table is reformulated as a nested dictionary in Eq. (2):

$$\text{Q-table} = \{b \in \mathcal{S} : \{a \in \mathcal{A} : Q(b,a) \in \mathbb{R}\}\} \tag{2}$$

where, $\mathcal{S} = 0, 1, \ldots, B-1$ is the set of state bins, each representing a discretized range of distances to the depot. $\mathcal{A} = 0, 1, \ldots, N-1$ is the set of available neighbor UAVs (actions). $Q(b,a)$ denotes the expected Q-value when selecting neighbor UAV a under state bin b. B is the total number of distance bins. N is the number of UAVs in the network.

This binning strategy significantly reduces memory overhead while preserving sufficient granularity for decision-making under mobility and link dynamics. Moreover, this structure facilitates the integration of additional routing metrics such as link quality, delay, and residual energy, without expanding the Q-table dimensionality. As a result, Q-Proposed achieves better generalization and adaptability in highly dynamic FANET scenarios, making it a practical reinforcement learning solution for real-time routing optimization.

3.2.2 Predict Feature Position

The Kalman Filter is employed to predict the future position of drones in the Q-Proposed algorithm, enhancing routing decisions in dynamic UAV environments. The filter recursively integrates noisy GPS measurements with a constant-velocity model to estimate trajectories over multiple time steps by modeling UAV motion with a state vector comprising position and velocity. The prediction, detailed in Eq. (3), accounts for process noise N and measurement noise R, yielding robust position estimates in units of meters (m), thus improving relay selection accuracy under mobility-induced uncertainties.

$$\begin{aligned} &\text{Predict:} \quad \mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1}, \quad \mathbf{P}_k = \mathbf{F}\mathbf{P}_{k-1}\mathbf{F}^T + \mathbf{N} \\ &\text{Update:} \quad \mathbf{K}_k = \mathbf{P}_k\mathbf{H}^T(\mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R})^{-1}, \quad \mathbf{x}_k = \mathbf{x}_k + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}\mathbf{x}_k) \end{aligned} \tag{3}$$

where,

- $\mathbf{x}_k$: State vector $[x, v_x, y, v_y, z, v_z]^T$, with x, y, z being the coordinates and v_x, v_y, v_z being the velocities on the x, y, z axes,
- $\mathbf{F}$: State transition matrix,
- $\mathbf{P}_k$: Covariance matrix,
- $\mathbf{N}$: Process noise covariance, $\mathbf{N} = \sigma_q^2\mathbf{I}_4$,
- $\mathbf{H}$: Measurement matrix,
- $\mathbf{R}$: Measurement noise covariance, $\mathbf{R} = \sigma_r^2\mathbf{I}_2$,
- $\mathbf{K}_k$: Kalman Gain,
- $\mathbf{z}_k$: Measurement vector $[x, y, z]^T$.

3.2.3 Reward Function

Indeed, the reward function in Q-learning not only provides feedback signals to the agent about its actions but also plays a vital role in shaping the agent's action strategy. A reasonable design of the reward function will help the agent learn optimal actions in the environment. The proposed solution uses three factors to calculate the reward function, including *energy*, *link quality*, and *end-to-end delay*. These metrics are calculated as follows.

Energy Consumption: This metric is an essential factor in considering sustainable UAV links. We propose accounting for this metric in the reward function to balance the energy consumption of nodes and enhance the UAV's network lifetime. The symbol E_i is is the energy consumption of i node, the unit is %, we have,

$$E_i = 1 - \frac{E_{re_i}}{E_{ini_i}} \tag{4}$$

where E_{re_i} and E_{ini_i} is the remaining energy and the initial energy of node i, respectively. The smaller the value of E_i, the route, and the higher the priority.

Delay: Defined as the period needed for a packet to transfer between two nodes at the network layer. Delay also reflects the effectiveness and sustainability of the link. In this study, when a node forwards data, the packet deadline will be updated by subtracting the forwarding time at the next node. The delay between a pair of nodes (i, j) is determined in Eq. (5), including queuing delay $d_{que_{ij}}$, MAC layer delay $d_{mac_{ij}}$, propagation delay $d_{pro_{ij}}$ and channel delay $d_{tran_{ij}}$, the unit is second (s).

$$d_{ij} = d_{que_{ij}} + d_{mac_{ij}} + d_{pro_{ij}} + d_{tran_{ij}} \tag{5}$$

Link Quality (*LQ*): This parameter reflects the sustainability of the connection between two UAV nodes. The link quality between two UAVs depends on the main factors such as distance, relative speed, and packet loss rate. The symbol LQ_{ij} is the link quality of a node pair (i, j), determined in Eq. (6), the unit is %, as follows. The link quality LQ_{ij} between a transmitting UAV i and a candidate neighbor j is computed as a weighted combination of multiple metrics:

$$LQ_{ij} = \alpha \cdot \left(1 - \frac{D_{ij}}{R}\right) + \beta \cdot \exp\left(-\frac{v_{ij}}{10}\right) + \gamma \cdot \left(1 - \mathrm{PLR}_{ij}\right) + \delta \cdot \left(\frac{\mathrm{RSSI}_{ij} + 100}{100}\right) \tag{6}$$

where,

- D_{ij}: Euclidean distance between the UAV_i and the UAV_j. Larger distances degrade the link quality,
- R: Maximum communication range of the UAV_i and the UAV_j. Used for normalization of distance,
- PLR_{ij}: Packet loss rate of the link between the UAV_i and the UAV_j,
- $RSSI_{ij}$: The signal strength that UAV_j receives from UAV_i during wireless communication,
- $\alpha, \beta, \gamma, \delta$: Weight factors balancing distance, relative velocity, packet loss rate, and RSSI contributions to link quality. In our simulation, we set $\alpha = 0.3$, $\beta = 0.2$, $\gamma = 0.3$, $\delta = 0.2$ after parameter tuning.

Aiming to enhance adaptation to high-dynamic environments, our reward function accounts for latency, energy consumption, and link quality factors between two nodes in Eq. (7).

$$R(s_t, a_t) = \begin{cases} r_{\max}, & \text{when } s_{t+1} \text{ is destination,} \\ r_{\min}, & \text{when } s_t \text{ is unreachable,} \\ w_{delay} \cdot e^{-\mathrm{d}_{ij}} + w_{energy} \cdot e^{-\mathrm{E}_j} + w_{LQ} \cdot LQ_{ij}, & \text{otherwise.} \end{cases} \tag{7}$$

where

- r_{max}, r_{min}: Upper and lower bounds of reward values, ensuring numerical stability during Q-value updates.
- w_d, w_e, w_{LQ}: Weight factors that balance the trade-off between minimizing delay, saving energy, and maximizing link stability.

3.2.4 Route Selection Strategy

In Q-Proposed, the relay selection process is guided by a reinforcement learning mechanism that adapts to the mobility of the UAV and the conditions of the communication links. At each decision step, the UAV assesses its distance to the depot and categorizes it into a discrete state bin, which simplifies the state space. It then evaluates a group of neighboring UAVs that are within its communication range. Q-Proposed computes a relay score for each candidate neighbor by combining the learned Q-value with physical network metrics, including normalized proximity and estimated link quality. The relay score is defined in Eq. (8):

$$\text{Score}_j = Q(b, j) \cdot \left(1 - \frac{D_{ij}}{R}\right) \cdot LQ_{ij} \tag{8}$$

where $Q(b, j)$ is the Q-value associated with selecting neighbor j under state bin b, D_{ij} is the Euclidean distance between the current UAV and neighbor j, R is the communication range, and LQ_{ij} is the estimated link quality in Eq. (6). The UAV then selects a relay using an exploration-exploitation strategy: it chooses the neighbor with the highest score or samples from a softmax distribution over scores, depending on the current exploration rate (ε). This hybrid strategy ensures a balance between optimal path discovery and adaptation to environmental dynamics as follows.

- **Softmax Exploration:** This strategy selects actions based on their Q value. Actions with high Q values will have a higher probability of being chosen. It's noted that actions with lower Q-values can still be explored.

$$P\left(a_t = a \mid s_t\right) = \frac{e^{Q(s_t, a)/\tau}}{\sum_{a'} e^{Q(s_t, a')/\tau}} \tag{9}$$

  where $P\left(a_t = a \mid s_t\right)$ is the probability to select action a at state s_t and τ is a parameter to adjust the level of exploration.
- **Epsilon-Greedy:** This strategy will randomly select actions with probability ε to explore and choose actions that have the highest Q-value with probability $1 - \varepsilon$ to exploit.

$$a_t = \begin{cases} argmax_a Q(s_t, a), & \text{probability } 1 - \varepsilon \\ \text{any action (a)}, & \text{probability } \varepsilon \end{cases} \tag{10}$$

The two strategies are implemented as follows: In the initial stage of learning, the algorithm explores many actions to gather information, so a high epsilon (ε) value is used with the Epsilon-Greedy strategy. As time progresses and the epsilon value decreases to a low level, the algorithm switches to the Softmax strategy. This approach allows the UAVs to exploit actions with high Q-values while still retaining the ability to discover other actions through Softmax probabilities. We present the pseudocode of the proposed algorithm in Algorithm 1.

Algorithm 1: Q-Proposed routing algorithm

```
Initialize: Q-table $Q[i, j]$ based on distance to depot use Eq. (2); Kalman Filters for each UAV with state $[x, v_x, y, v_y]$
Set parameters: $\epsilon_{\text{min}}, \epsilon, \epsilon_{\text{decay}}, \tau$
for each UAV $u_i$ do
    if $u_i$ has a packet to forward then
        Get current state $s$ (position, neighbors)
        Determine the set of available neighbors $A$
        Initialize candidate set $C \leftarrow \varnothing$
        for each neighbor $u_j \in A$ do
            Predict $u_j$'s Future Position using Kalman Filter
            Estimate required velocity $v_{\text{req}}$
            Compute actual velocity $v_{\text{actual}}$ based on predicted position
            Compute link quality $LQ(i,j)$ use Eq. (6)
            Compute $k_j = Score_j$ use Eq. (8)
            Add $u_j$ to $C$
        end for
        if $C \neq \varnothing$ then
            if $\epsilon > \epsilon_{\text{min}}$ then
                $a \leftarrow \arg\max_j k[j]$
            else
                $P(a) \leftarrow \text{softmax}(k_j/\tau)$
                Sample action $a$ from distribution $P(a)$
            end if
        else
            if $A \neq \varnothing$ then
                $a \leftarrow \arg\min_{u_j \in A} \text{dist}(u_j, \text{depot})$
            else
                Report Routing Hole Problem (RHP) and drop packet
            end if
        end if
        Execute action $a$: Send packet to selected neighbor UAV $u_j$
        Compute reward $r$ based on Eq. (7)
        Update Q-table based on Eq. (1)
        Update $\varepsilon \leftarrow \max(\varepsilon_{\text{min}}, \varepsilon \cdot \varepsilon_{\text{decay}})$
    end if
end for
```

4 Results and Discussion

4.1 Simulation Parameters

To illustrate the effectiveness of the proposed solution, we use the DRONET[1] simulation software to perform simulations. We establish the number of UAVs variable in the range [20–70]. UAVs are randomly distributed in an area of 1800 × 1800 m^2 with detailed simulation parameters described in Table 2. We

[1] https://github.com/Andrea94c/DroNETworkSimulator (accessed on 20 February 2025).

assume the UAV scenarios for smart agriculture and search and rescue in smart cities, so the flight speed of the UAV is set at about 12 m/s, and the communication distance between the UAVs is about 220 m. We assess the performance of three routing protocols: GEO (GPSR [21]), QGEO [22], and the proposed Q-Proposed protocol.

Table 2: Experimental parameter settings

Parameter	Value
Simulation area	1800 m × 1800 m × 50 m
Number of nodes	20, 30, 40, 50, 60, 70
UAV movement speed	12 m/s
Transmission Ranges	220 m
Protocol	GEO, QGEO, Q-Proposed
Battery model	Energy linear
Energy initial	1000 KJ
MAC layer	802.11n

4.2 Performance Metrics

Average Packet Delivery Ratio: PDR is defined as the percentage of total received packets (P_r) to total sent packets (P_s) during the simulation process, as determined in the following Eq. (11).

$$PDR = \frac{P_r}{P_s} \times 100\% \tag{11}$$

Average End-to-End Delay: The delay is defined as the subtraction of the received time and the sent time of a data packet from source UAV (T_s) to destination UAV (T_r). We symbolize D as the average end-to-end delay, which is the ratio of the total delay and the total received data packets during the simulation process. The unit is seconds (s) determined in Eq. (12).

$$D = \frac{\sum_{i=1}^{total}(T_t - T_s)}{P_s} \tag{12}$$

Energy Consumption: Defined as the subtraction of the initial energy and the remaining energy of each UAV. We symbolize E_{consum} as the total energy consumption of the entire UAV network, which is the energy consumption of all UAVs in the simulation process. The unit is KJ and is determined in Eq. (13).

$$E_{consum} = \sum_{i=0}^{n-1}(E_{init} - E_{remain}) \tag{13}$$

4.3 Simulation Results

Fig. 3 illustrates the end-to-end delay as the number of UAVs increases. The GEO method experiences the highest delay, starting at around 43 ms and rising to nearly 48 ms. This is due to its greedy forwarding approach, which does not optimize path selection in dense networks. Meanwhile, QGEO maintains a stable delay of approximately 37–38 ms by using Q-learning to improve forwarding decisions; however, it lacks mobility prediction and awareness of link quality. In contrast, the Q-Proposed method achieves the lowest delay, decreasing from about 36 to 35 ms as the node density increases. This improvement is a result of combining reinforcement learning with link quality metrics and mobility prediction based on Kalman filters, allowing efficient and reliable relay selection in dynamic FANET environments.

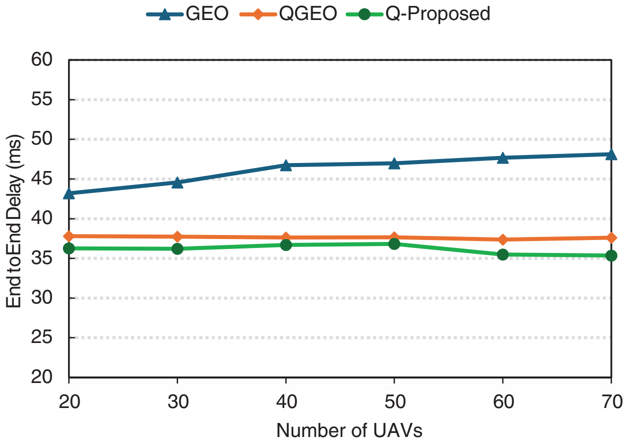

Figure 3: End-to-end delay & number of UAVs

Fig. 4 shows how the PDR changes with the number of UAVs for three routing protocols: GEO, QGEO, and the Q-Proposed approach. GEO exhibits the lowest PDR, ranging from approximately 76% to 78%, due to its reliance on a purely greedy geographic strategy that ignores link dynamics and node mobility, resulting in frequent routing failures under high mobility conditions. QGEO provides a moderate improvement, maintaining PDR between 81% and 83%. This improvement is due to the incorporation of Q-learning, enabling UAVs to learn from past outcomes instead of relying solely on distance-based heuristics. However, QGEO still does not adequately account for link quality and dynamic factors like mobility prediction, which restricts its adaptability in highly variable network environments.

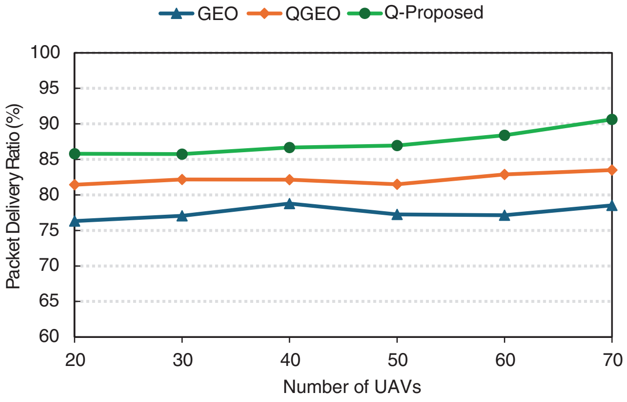

Figure 4: Packet delivery ratio & number of UAVs

In contrast, the Q-Proposed protocol consistently outperforms both alternatives, starting at about 85% and reaching over 90% as node density increases. This superior performance stems from combining reinforcement learning with link quality estimation (including RSSI, packet loss rate, and bitrate) and Kalman filter-based mobility prediction. These enhancements enable Q-Proposed to select reliable relay nodes and adapt quickly to topology changes, resulting in significantly improved delivery reliability in dynamic FANET scenarios.

Fig. 5 compares the total energy consumption for GEO, QGEO, and Q-Proposed as the number of UAVs increases. The GEO consumes the least energy, starting around 190 KJ and reaching 210 KJ at 70 UAVs, due to its simple geographic forwarding without additional control overhead. The QGEO requires more energy (390–490 KJ) because of Q-learning updates, which introduce extra computation and signaling. While the proposed Q-learning with Kalman prediction significantly improves reliability and delay performance, it also results in approximately 18%–20% higher energy consumption compared to QGEO. This overhead is mainly due to additional computations for link-quality estimation and mobility prediction. Nevertheless, these operations also reduce retransmissions, partially compensating for the extra energy cost.

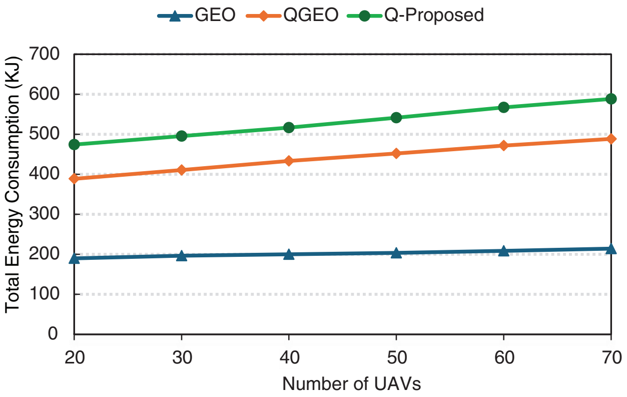

Figure 5: Total energy consumption & number of UAVs

While this leads to superior PDR and lower delay, it comes at the expense of higher energy consumption, highlighting a trade-off between routing performance and energy efficiency in dynamic FANET environments.

In summary, GEO shows the lowest delivery ratio and the highest delay since it relies only on greedy geographic forwarding. QGEO achieves moderate improvements by adding Q-learning, but it lacks mobility prediction and link-quality awareness. The Q-Proposed further enhances performance by combining reinforcement learning with Kalman filter-based mobility prediction and link-quality estimation, leading to higher delivery reliability and lower delay. GEO and QGEO were selected as baselines because they respectively represent a classical geographic routing protocol and its Q-learning extension, providing a fair reference for evaluating Q-Proposed.

The simulation results highlight distinct trade-offs among the three protocols. These results confirm that Q-Proposed provides superior routing performance for highly dynamic FANETs, with an expected trade-off in energy efficiency.

5 Conclusion

In this study, we introduced an enhanced routing protocol for FANETs based on reinforcement learning, which integrates mobility prediction and link awareness. Our proposed approach improves the relay selection process by considering various performance factors such as link quality, residual energy, and transmission delay within the Q-learning framework. Simulation results demonstrate that our method significantly enhances the packet delivery ratio by 12%–15% (from 78% to over 90%) and reduces end-to-end delay by 27.5% (from 48 to 35 ms) compared to conventional GEO and QGEO protocols. However, it does lead to higher energy consumption due to the more complex computations and control overhead involved. Nevertheless, the performance improvements confirm its suitability for highly dynamic and dense UAV networks. Future research will focus on integrating advanced RL techniques, such as Deep RL and Multi-Agent RL, as well as Federated Learning (FL) frameworks, to enhance scalability, privacy, and adaptability in UAV networks. These improvements aim to support real-world applications in smart city environments and 6G-enabled Internet of Things services such as intelligent transportation, disaster response, and large-scale urban monitoring.

Acknowledgement: We declare that we have used Grammarly (AI) to edit several complex sentences in related works.

Funding Statement: This research is funded by Hung Yen University of Technology and Education under grand number UTEHY.L.2025.62.

Author Contributions: The authors confirm contributions to the paper as follows: study conception and design: Vi Hoai Nam; data collection: Chu Thi Minh Hue; analysis and interpretation of results: Vi Hoai Nam, Chu Thi Minh Hue and Dang Van Anh; draft manuscript preparation: Vi Hoai Nam, Chu Thi Minh Hue and Dang Van Anh. All authors reviewed the results 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 to report regarding the present study.

References

1. Kirimtat A, Krejcar O, Kertesz A, Tasgetiren MF. Future trends and current state of smart city concepts: a survey. IEEE Access. 2020;8:86448–67. doi:10.1109/access.2020.2992441.
2. Yang T, Li Z, Zhang F, Xie B, Li J, Liu L. Panoramic UAV surveillance and recycling system based on structure-free camera array. IEEE Access. 2019;7:25763–78. doi:10.1109/access.2019.2900167.
3. Xu J, Ota K, Dong M. Ideas in the air: unmanned aerial semantic communication for post-disaster scenarios. IEEE Wireless Commun Lett. 2025;14(6):1598–602. doi:10.1109/lwc.2025.3530760.
4. Kumar VDA, Ramachandran V, Rashid M, Javed AR, Islam S, Al Hejaili A. An intelligent traffic monitoring system in congested regions with prioritization for emergency vehicle using UAV networks. Tsinghua Sci Technol. 2025;30(4):1387–400. doi:10.26599/tst.2023.9010078.
5. Yi J, Zhang H, Li S, Feng O, Zhong G, Liu H. Logistics UAV air route network capacity evaluation method based on traffic flow allocation. IEEE Access. 2023;11:63701–13. doi:10.1109/access.2023.3238464.
6. Chen J, Wan P, Xu G. Cooperative learning-based joint UAV and human courier scheduling for emergency medical delivery service. IEEE Transact Intell Transport Syst. 2025;26(1):935–49. doi:10.1109/tits.2024.3486789.
7. Minhas HI, Ahmad R, Ahmed W, Waheed M, Alam MM, Gul ST. A reinforcement learning routing protocol for UAV aided public safety networks. Sensors. 2021;21(12):4121. doi:10.3390/s21124121.
8. Parvaresh N, Kantarci B. A continuous actor-critic deep q-learning-enabled deployment of UAV base stations: toward 6G small cells in the skies of smart cities. IEEE Open J Communicat Soc. 2023;4:700–12. doi:10.1109/ojcoms.2023.3251297.
9. Park C, Lee S, Joo H, Kim H. Empowering adaptive geolocation-based routing for UAV networks with reinforcement learning. Drones. 2023;7(6):387. doi:10.3390/drones7060387.
10. Tang X, Li X, Yu R, Wu Y, Ye J, Tang F, et al. Digital-twin-assisted task assignment in multi-UAV systems: a deep reinforcement learning approach. IEEE Int Things J. 2023;10(17):15362–75. doi:10.1109/jiot.2023.3263574.
11. Wang Z, Yao H, Mai T, Xiong Z, Wu X, Wu D, et al. Learning to routing in UAV swarm network: a multi-agent reinforcement learning approach. IEEE Transact Vehic Technol. 2023;72(5):6611–24. doi:10.1109/tvt.2022.3232815.
12. Ke Y, Huang K, Qiu X, Song B, Xu L, Yin J, et al. Distributed routing optimization algorithm for FANET based on multiagent reinforcement learning. IEEE Sens J. 2024;24(15):24851–64. doi:10.1109/jsen.2024.3415127.

13. Kim J, Park S, Jung S, Cordeiro C. Cooperative multi-UAV positioning for aerial internet service management: a multi-agent deep reinforcement learning approach. IEEE Transact Netw Serv Manag. 2024;21(4):3797–812. doi:10.1109/tnsm.2024.3392393.
14. Chen J, Huang D, Wang Y, Yu Z, Zhao Z, Cao X, et al. Enhancing routing performance through trajectory planning with DRL in UAV-aided VANETs. IEEE Transact Mach Learn Communicat Netw. 2025;3:517–33. doi:10.1109/tmlcn.2025.3558204.
15. Wei X, Li N, Yang H, Qu B. Q-Learning enabled ant colony (Q-ANT) based navigation optimization framework for UAVs. IEEE Trans Consum Electron. 2025;71(2):3903–12. doi:10.1109/tce.2025.3570746.
16. Ibraheem Mohammed Y, Hassan R, Hasan MK, Islam S, Abbas HS, Khan MA, et al. Revolutionizing FANETs with reinforcement learning: optimized data forwarding and real-time adaptability. IEEE Open J Communicat Soc. 2025;6:4295–310. doi:10.1109/ojcoms.2025.3565471.
17. Sharvari NP, Das D, Bapat J, Das D. Improved Q-learning-based multi-hop routing for UAV-assisted communication. IEEE Transact Netw Serv Manag. 2025;22(2):1330–44. doi:10.1109/tnsm.2024.3522153.
18. Dhuheir M, Erbad A, Hamdaoui B, Belhaouari SB, Guizani M, Vu TX. Multi-agent meta reinforcement learning for reliable and low-latency distributed inference in resource-constrained UAV swarms. IEEE Access. 2025;13:103045–59. doi:10.1109/access.2025.3572036.
19. Wu M, Jiang B, Chen S, Xu H, Pang T, Gao M, et al. Traj-Q-GPSR: a trajectory-informed and Q-learning enhanced GPSR protocol for mission-oriented FANETs. Drones. 2025;9(7):489. doi:10.3390/drones9070489.
20. Watkins CJCH, Dayan P. Q-learning. Mach Learn. 1992;8(3):279–92.
21. Silva A, Reza N, Oliveira A. Improvement and performance evaluation of GPSR-based routing techniques for vehicular ad hoc networks. IEEE Access. 2019;7:21722–33. doi:10.1109/access.2019.2898776.
22. Jung WS, Yim J, Ko YB. QGeo: Q-learning-based geographic ad hoc routing protocol for unmanned robotic networks. IIEEE Communications Letters. 2017;21(10):2258–61. doi:10.1109/LCOMM.2017.2656879.