Reinforcement Learning based Energy-Efficient Fast Routing for FANETs

Jieling Li, *Student Member, IEEE*, Liang Xiao, *Senior Member, IEEE*, Xuchen Qi, Zefang Lv, *Student Member, IEEE*, Qiaoxin Chen, and Yong-Jin Liu, *Senior Member, IEEE*

***Abstract*—Reinforcement learning (RL) based flying ad-hoc network (FANET) routing enables unmanned aerial vehicles (UAVs) to choose the next-hop to increase the packet delivery ratio, but the routing latency and energy consumption have to be further reduced over inaccurate feedback for large-scale networks. In this paper, we propose an RL based energy-efficient fast routing for each UAV to choose the forwarding decision and the power. Based on the state consisting of the battery level, channel conditions and forwarding decisions of the one-hop neighbors, the routing policy is chosen to enhance the utility as the weighted sum of the delivery success indicator, the latency and the energy consumption. The number of the latency violations and the learning parameters shared among the one-hop neighbors are exploited in the update of the routing policy distribution following the latency constraint with the reduced energy consumption. The deep neural networks address the state quantization error of the latency and the channel gain for UAVs with high mobility under large-scale networks. The performance bound regarding the end-to-end latency and the energy consumption is derived in terms of network topology and channel gain based on the packet forwarding game. The performance gain over the benchmark is provided via both simulation and experimental results.**

***Index Terms*—Unmanned aerial vehicle, flying ad-hoc network, routing, reinforcement learning, latency constraint.**

I. Introduction

Flying ad-hoc network (FANET) routing chooses the next-hop unmanned aerial vehicle (UAV) based on the route discovery and local topology to forward the sensing data to support applications such as live streaming, search and rescue, target tracking and data collection [2]–[6]. For example, the routing in [7] establishes and maintains the routing tables to identify the routing paths with the quality of service (QoS) requirement of less than 40 ms end-to-end routing latency for each 1-KB packet such as the target status to search and track multiple targets in large disaster areas.

The reinforcement learning (RL) based routing schemes enable UAVs to optimize the routing policy through the local observation and the exchange of states with one-hop neighbors such as the position, as the routing selection process can be regarded as a partially observable Markov decision process (MDP) [8], [9]. For example, the aerial-ground integrated network in [8] applies Q-learning to choose the next-hop based on the horizontal location and residual energy of the UAV and designs a central angle constraint to limit the scope of the next-hop search. However, the routing path suffers from a higher latency and energy consumption due to the partial observation and the inaccurate feedback in the next-hop selection under large-scale networks.

In this paper, an RL based energy-efficient fast routing scheme is proposed for FANETs under the state consisting of the battery level, the channel gain, the received signal-to-noise ratio (SNR), and the forwarding decision of the neighboring UAVs, as well as the hop count and the one-hop latency. Each UAV decides whether to forward the received packets and chooses the relay power in a hop-by-hop process with less route oscillations according to the carrier sense multiple access with collision avoidance (CSMA/CA). The utility is formulated as the weighted sum of the packet delivery success indicator, the average end-to-end latency and the routing energy consumption.

The channel states and the rebroadcast count of the previous packets in the data frame are shared among the one-hop neighbors via overhearing to perceive the contention status of the broadcast channel and mitigate the collision among the forwarders within the communication range. By combining the expected long-term utility (Q-value) indicating the path quality and the risk value that is the tolerance probability for the worst-case end-to-end latency, a safe routing exploration mechanism is designed to formulate and update the modified Boltzmann policy distribution for the forwarding decision and the relay power. Specifically, the Q-value is updated based on the routing experiences and the learning parameters of the one-hop neighbors to evaluate the path availability.

A deep RL based routing scheme is also proposed to further reduce the end-to-end latency with less routing energy consumption for large-scale FANETs. The neural networks with fully-connected layers address the state quantization error of the latency and the channel gain and compress the state space with a large number of one-hop neighbors through the state mapping of the hidden layers. The policy neural network and the risk neural network are designed to choose the forwarding decision and the relay power under rapidly changing channel and routing process. In addition, the weights of the policy neural network and the risk neural network

This work was supported in part by the National Natural Science Foundation of China under Grant U21A20444 and the National Key Research and Development Program of China under Grant 2023YFB3107603, and in part by the National Natural Science Foundation of China under Grants 62332019 and U2336214. (*Corresponding author: Liang Xiao*)

Jieling Li, Liang Xiao, Xuchen Qi, Zefang Lv and Qiaoxin Chen are with the Department of Informatics and Communication Engineering, Xiamen University, and also with the Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Xiamen University, Xiamen 361005, China. E-mail: lxiao@xmu.edu.cn.

Yong-jin Liu is with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China. E-mail: liuyongjin@tsinghua.edu.cn.

This paper was presented in part at IEEE GLOBECOM 2023 [1].

are periodically exchanged among the one-hop neighbors to improve the experience sampling efficiency.

The FANET routing is formulated into a packet forwarding game among the collaborative UAVs, in which each UAV chooses the routing policy to transmit the packet to the destination for reducing the end-to-end latency with less power. The Nash equilibrium of the packet forwarding game is analyzed to derive the performance bound of the proposed schemes under the specified network topology and channel gain. The computational complexity of the proposed schemes increases with the number of the routing experience samples, the relay power level and the number of the knowledge shared from the one-hop neighbors such as the channel gain and the battery level.

Simulation is performed with 20 UAVs based on the line-of-sight dominant radio channel and the Gauss-Markov mobility model to support FANET applications such as the monitoring and target tracking. Simulation results demonstrate that the proposed schemes reduce the average end-to-end latency and the routing energy consumption compared with the Q-learning based next-hop selection scheme (QNS) in [8]. Experimental results for the image transmission to support the target tracking application based on 5 UAVs equipped with Raspberry Pi-4B to send the packets to the control center that displays the image transmission performance in the graphical user interface (GUI) are provided to show the efficacy of the proposed schemes. The main contributions of this work are listed as follows:

- An energy-efficient fast routing based on RL is proposed to optimize both the forwarding decision and the relay power according to the end-to-end latency QoS requirement of FANET applications and the shared learning experiences of neighboring UAVs to improve the routing path exploration.
- A deep RL version is also proposed to compress the state space of the channel gain and the battery level of the one-hop neighbors in large-scale networks and address the state quantization error of the latency and the channel gain for UAVs with high mobility.
- The performance bound in terms of end-to-end latency and energy consumption is derived by analyzing the Nash equilibrium of the routing game among the collaborative UAVs. The performance gain of the proposed scheme is also verified via experiments based on a 5-UAV FANET equipped with Raspberry Pi-4B.

The rest of this paper is organized as follows. The related work is reviewed in Section II, and the system model is presented in Section III. An RL based routing scheme and a deep RL version are presented in Sections IV and V, and the performance is analyzed in Section VI. Simulation and experimental results are reported in Section VII and Section VIII. The conclusions of this paper is drawn in Section IX.

II. Related Work

Existing routing schemes explore the routing path by initiating the route discovery such as the table-driven and on-demand routing [10]–[13], as well as the utilization of a greedy forwarding technique that selects the next-hop according to the minimum distance to the destination [14]. For example, the modified optimized link-state routing in [10] uses improved expected transmission count according to the residual energy and position of the UAV to select multi-point relay nodes and thus improves the throughput and reduces the latency. The on-demand routing in [11] leverages certificate authority to detect the malicious vehicles and identify the routing path and thus increases the throughput. The secure reactive routing in [12] applies the trust-based approach to detect the malicious vehicles and discover the appropriate routing path and thus increases the detection accuracy and reduces the routing overhead. The energy-efficient routing in [13] initiates the route discovery based on the battery level, the connectivity degree and the lifetime of the links to reduce the path failure probability. The hybrid routing in [14] exploits a synchronization mechanism to maintain the virtual topology of the aeronautical ad-hoc networks and employs the location-based greedy forwarding technique to minimize synchronization overhead. However, the network topology such as the distance to the destination is rarely known in large-scale FANETs with high mobility.

Multi-path and opportunistic routing enable UAVs to forward the received packets to generate the path diversity and improve the end-to-end availability. For example, the intelligent forwarding protocol in [15] exploits handshake-less communication and forwarder selection mechanism to improve the transportation safety. The flooding routing in [16] uses the random network coding to transmit the packets without being aware of network topology and route discovery, thus reducing the latency violation probability. Another flooding routing in [17] utilizes the clustering method and Poisson cluster process to partition UAV networks and thus minimize the hop count and reduce the latency violation probability. The opportunistic routing in [18] applies ACK-based and timer-based coordination approaches depending on the service flow requirements and adjusts the control message sending interval dynamically to satisfy the service requirements. The probability prediction opportunistic routing in [19] predicts the packet backlog size and uses the prediction result to determine the utility of the relay node, which enhances the network throughput and decreases the delivery latency.

RL-based routing schemes have been applied to optimize the routing policy, thereby improving the QoS and reducing the communication overhead in the constrained environment of FANETs. For example, a three-layer network architecture in [8] applies Q-learning to determine the next-hop and uses the dynamic learning rate during the expected long-term utility update process to accelerate routing convergence. The self-learning routing protocol in [9] utilizes the eligibility trace function to optimize the next-hop selection through the exchanged position and global network utility, thus reducing the average path lifetime. The Q-learning based position routing in [20] applies two-hop neighbor discovery to maintain the UAV topology to improve the packet delivery ratio, which requires the trade-off between performance enhancement and routing complexity. The RL-based cluster routing in [21] enables the ground control station to optimize the cluster head

TABLE I: Summary of routing schemes for FANETs.

Reference	Mobility model	Link state	Energy-efficient	Neighbor discovery	Neighbor set	Route update
[8]	Gauss-Markov	Yes	Yes	Yes	One-hop	Dynamic
[10]	Gauss-Markov	Yes	Yes	Yes	One-hop	Periodically
[13]	Random Way Point	Yes	Yes	Yes	One-hop	On demand
[17]	Poisson point process	Yes	No	No	One-hop	N/A
[20]	Gauss-Markov	Yes	Yes	Yes	Two-hop	Dynamic
[21]	3D-waypoint	Yes	Yes	Yes	One-hop	Dynamic
[22]	Random Way Point	Yes	No	Yes	One-hop	Dynamic
[23]	Gauss-Markov	Yes	No	Yes	One-hop	Dynamic
[24]	Gauss-Markov	No	Yes	No	One-hop	Dynamic
Proposed	Gauss-Markov	Yes	Yes	No	One-hop	Dynamic

selection according to the link stability and the battery level, and thus reduce the latency and the topology construction time. The Q-learning based FANET routing in [22] utilizes a finite set of recent episodes and the transmission quality based on channel condition to choose the next-hop and update Q-values and thus avoid selecting the noisy channels, which reduces the routing latency and jitter.

Deep RL-based routing schemes have been utilized to extract intricate environment characteristics while compressing high-dimensional state and action spaces. For example, the RL-based routing in [23] assesses the link quality and builds the actor-critic networks to make the routing decision, which reduces the latency and enhances the transmission rate. The adaptive communication-based routing in [24] utilizes the multilayer perceptron algorithm to assess the requirement for flooding the route request and designs the actor-critic framework with compressed communication data to perform the routing strategy. The multi-hop relay network in [25] applies a mixing network to allocate the frequency resource, design the trajectories and choose the next-hop, which enhances the throughput and reduces the data transmission time. However, the performance degrades due to the inaccurate utility and the partial observation of the state under large-scale FANETs. The other comparison based on different parameters such as the mobility model is presented in Table I.

In our prior work in [1], an RL based routing scheme with experiences sharing is proposed to optimize both the routing and the power allocation based on the maximum tolerable end-to-end latency. In this paper, a deep RL version is further proposed for the UAV with sufficient computational resources to address the quantization error for both the latency and channel gain in the state formulation and compress the state space with a large number of one-hop neighbors through the state mapping of the hidden layers and thus enhance the routing performance for the large-scale networks with high moving speed. The influence of the network topology and the channel gain on the routing performance bound is analyzed based on the packet forwarding game among the collaborative UAVs. The validation of the proposed schemes is also conducted via experiments based on the transmission of UAVs equipped with Raspberry Pi-4B.

III. System Model

A. Network Model

As shown in Fig. 1, N UAVs equipped with Wi-Fi covering a radius of up to 250 m exchange the control messages such as the task instructions and the search area indication, as well as the sensing data such as the surveillance video streams to support applications such as the target tracking and the live streaming [26]. For example, each UAV is assumed to forward the packets containing the belief map and the target position via CSMA/CA every time slot to reduce uncertainty in the region, cover the new target area and search the long-distance targets in large areas.

At time slot k, UAV i moving at speed $v_i^{(k)} \in [0, \overline{V}]$, receives M packets from one-hop neighbors with ID $\mathcal{N}_i^{(k)}$, generates a packet $g_i^{(k)}$ and decides whether to route the $n_i^{(k)}$ packets to the destination that is an another UAV or the control center. Before forwarding the packets to the next-hops, the number of the one-hop neighbors $\varpi_i^{(k)}$ is evaluated based on the received signal via overhearing.

B. Communication Model

Upon receiving M packets, UAV i estimates the channel gain to the one-hop neighbors denoted by $\boldsymbol{h}_i^{(k)}$, evaluates the SNR $\xi_i^{(k)}$ and calculates the one-hop latency of the packet denoted by $t_{i,m}^{(k)}$, $1 \le m \le M$ and the rebroadcast count of the same packets $\varphi_i^{(k)}$. As shown in Fig. 2, the received packets containing the source ID $\rho_{i,m}^{(k)}$, the sequence number $q_{i,m}^{(k)}$, the one-hop latency $t_{i,m}^{(k)}$, the hop count $\varrho_i^{(k)}$, the battery level $b_i^{(k)}$, the SNR, and the rebroadcast count $\varphi_i^{(k)}$, are forwarded to the one-hop neighbors with relay power $p_i^{(k)} \in \left[\underline{P}, \overline{P}\right]$.

Each one-hop neighbor contends the shared channel with CSMA/CA and routes the respective generating packet and the received packets simultaneously. By overhearing the rebroadcast packet from one-hop neighbors, UAV i obtains the feedback information such as the SNR $\tilde{\xi}_i^{(k)}$, the one-hop latency $\tilde{t}_{i,m}^{(k)}$ and the rebroadcast count of the same packets within the communication range $\tilde{\varphi}_i^{(k)}$ with 1 octet, and then calculates the average one-hop latency $\tilde{t}_i^{(k)}$ and updates the

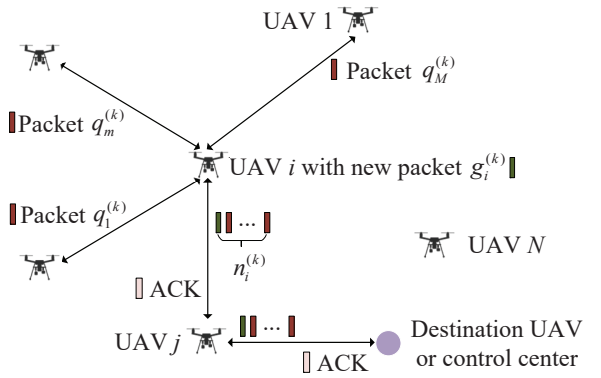
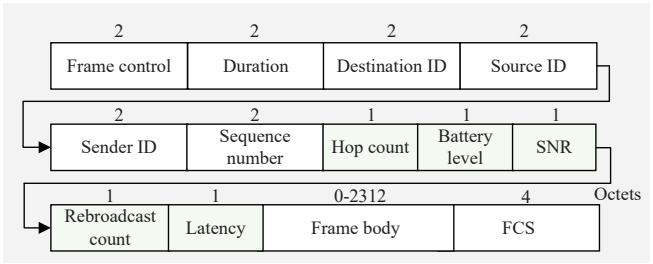
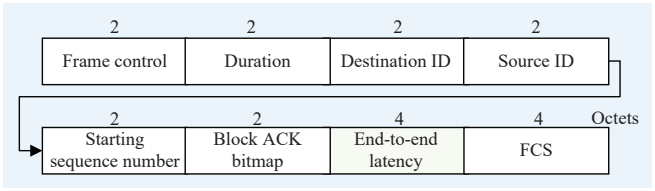


Fig. 1. Illustration of the routing in FANETs, in which UAV i moving with speed $v_i^{(k)}$ generates packet $g_i^{(k)}$ and decides whether to forward the $n_i^{(k)}$ packets to neighboring UAVs with relay power $p_i^{(k)}$ at time slot k.



(a) Data



(b) ACK

Fig. 2. Frame format of the routing scheme, in which the routing history are exchanged with neighboring UAVs via overhearing.

rebroadcast count $\varphi_i^{(k)}$, thereby reducing the communication overhead of per-hop ACKs at the feedback channel [15].

Upon receiving the packets, the destination calculates the end-to-end latency $\boldsymbol{\tau}^{(k)}$ and then forwards the modified ACK packet including the end-to-end latency with 4 octets to all UAVs along the path that the packet passed through via a reliable feedback channel. After receiving the ACK packet from the destination, the delivery success indicator $\kappa_i^{(k)}$ is set as 1, and 0 otherwise. The UAV energy consumption contains the communication energy determined by the relay power, the packet size, the one-hop latency and the hop count to the destination, as well as the propulsion energy that increases with the wind speed according to [27]. Without confusion, the superscript k and UAV ID i are omitted in the following for simplicity. For ease of reference, the main notations and symbols are listed in Table II.

IV. RL BASED FANET ROUTING

The RL based FANET routing scheme (RLFR) with low-latency and energy-efficient is proposed to enable each UAV as the learning agent to choose whether to forward the received packets and the relay power. Based on the one-hop latency, the channel conditions, and the observed network states shared by one-hop neighbors such as the battery level and the rebroadcast count, this scheme enhances the utility as a weighted sum of the delivery success indicator, the average end-to-end latency and the routing energy consumption. A safe routing exploration mechanism following the end-to-end latency constraint is utilized to improve the path quality. The learning parameters shared by one-hop neighbors such as the state value functions are also exploited to evaluate the end-to-end path availability and thus enhance the routing path exploration, as summarized in Algorithm 1.

At time slot k, each UAV evaluates the number of neighbors within the communication range denoted by ϖ, estimates the channel gain to these ϖ UAVs, resulting in the channel vector denoted by $\boldsymbol{h}$ and measures the battery level b_0. The UAV also receives packets from the next-hops and thus obtains the rebroadcast count of the previous same packets among the one-hop neighbors φ, the current hop count of the received packet ϱ, the battery level of the ϖ UAVs $\boldsymbol{b}$, the received signal SNR ξ, and the one-hop latency t_m, $1 \leq m \leq M$, consisting of both the contention and transmission latency.

The state $\boldsymbol{s}^{(k)}$ contains the signal SNR ξ, the average one-hop latency t, and the hop count ϱ in the received packet, the information of the one-hop neighbors, including the quantized channel states $\boldsymbol{h}$, the number of neighbors within the communication range ϖ, the battery level $\boldsymbol{b}$ and the rebroadcast packet count φ, and the previous routing performance that is the average end-to-end latency τ, i.e.,

$$\boldsymbol{s}^{(k)} = [\tau, \varrho, t, \boldsymbol{b}, \varpi, \varphi, \xi, \boldsymbol{h}] \in \mathbf{S}. \tag{1}$$

The source ID and the sequence number of the received packets denoted by $(\rho_m^{(k)}, q_m^{(k)})$ are stored in a cache Ω to eliminate repeatedly forwarding of the same packet and avoid the risk of the routing loop, in which the sequence number is increased by one for each packet dispatched by the source. The hop count ϱ is increased by one if the tuple is not found in the cache, and the packet is discarded otherwise. The routing policy $\boldsymbol{a}^{(k)} = [x^{(k)}, p^{(k)}] \in \mathbf{A}$ consists of the forwarding decision $x^{(k)} \in \{0, 1\}$ and the relay power $p^{(k)} \in \{\underline{P} + \epsilon\overline{P}/L | 0 \leq \epsilon \leq L-1\}$. The received packets are forwarded to the one-hop neighbors $\mathcal{N}$ with relay power $p^{(k)}$ if $x^{(k)} = 1$, and drop the packets otherwise, thus optimizing resource utilization and alleviating the broadcast storm.

According to the modified Boltzmann distribution [28], the routing policy distribution $\pi\left(\boldsymbol{s}^{(k)}, \cdot\right)$ depends on both the expected long-term utility (Q-value) denoted by $Q\left(\boldsymbol{s}^{(k)}, \cdot\right)$ and the risk value denoted by $R\left(\boldsymbol{s}^{(k)}, \cdot\right)$ that is the maximum

TABLE II: Summary of Symbols and Notations.

Symbol	Description
N	Number of UAVs
M	Number of the received packets
$\mathcal{N}_i^{(k)}$	Neighbor set of UAV i at time slot k
$v_i^{(k)}$	Moving speed of UAV i
$z_i^{(k)}$	Received packets size of UAV i
$\rho_{i,m}^{(k)}$	Source ID of the m-th packet received by UAV i
$q_{i,m}^{(k)}$	Sequence number of m-th packet received at UAV i
$g_i^{(k)}$	Packet sequence number of UAV i
$p_i^{(k)}$	Relay power of UAV i
$h_{i,j}^{(k)}$	Channel gain from UAV i to j
$t_{i,m}^{(k)}$	One-hop latency of the m-th packet received by UAV i
$\boldsymbol{\tau}^{(k)}$	End-to-end latency of the packets received at the destination
$b_i^{(k)}$	Battery level of UAV i
$\xi_i^{(k)}$	SNR of the signal received by UAV i
$n_i^{(k)}$	Number of the packets forwarded by UAV i

tolerable end-to-end latency, given by

$$\pi\left(\boldsymbol{s}^{(k)},\cdot\right)=\frac{\exp\left(Q\left(\boldsymbol{s}^{(k)},\cdot\right)-cR\left(\boldsymbol{s}^{(k)},\cdot\right)\right)}{\sum\limits_{\hat{\boldsymbol{a}}\in\mathbf{A}}\exp\left(Q\left(\boldsymbol{s}^{(k)},\hat{\boldsymbol{a}}\right)-cR\left(\boldsymbol{s}^{(k)},\hat{\boldsymbol{a}}\right)\right)}. \quad (2)$$

The risk value in the routing policy distribution acts as a temperature parameter that modulates the exploration of risky routes and the randomness of routing actions to enhance policy exploration efficiency [29]. The forwarding decision and relay power that result in higher expected utility and lower risk value are selected with a higher probability via $|\mathbf{S}|\times|\mathbf{A}|$-dimensional Q-table and R-table.

Upon receiving the ACK from the destination, each UAV obtains the delivery success indicator κ, calculates the average end-to-end latency of the received packets denoted by τ, and estimates the communication energy consumption w caused by itself. The global utility relies on the delivery success indicator and the average end-to-end latency, which are influenced by the status of the routing process completion. The local utility determined by the communication energy consumption is evaluated based on the relay power and the transmission duration of a packet that depends on the data rate and the packet size. In particular, the global and local utilities are balanced because the execution of routing action affects the end-to-end path availability and the cooperative cost. Thus, the utility $u^{(k)}$ is given by

$$u^{(k)}=\kappa-c_1\tau-c_2w, \quad (3)$$

where c_1 and c_2 are the coefficients of the average end-to-end latency and the communication energy consumption. A large global utility is regarded as the incentive of successful delivery on the utility to improve the probability of forwarding the packets to the destination.

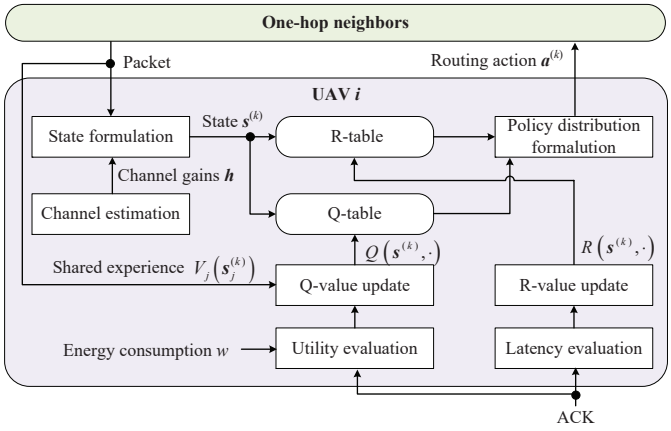


Fig. 3. Illustration of an RL based routing scheme for FANETs, in which UAV i chooses routing action $\boldsymbol{a}^{(k)}$ based on state $\boldsymbol{s}^{(k)}$ in each time slot.

As shown in Fig. 3, the state value function $V\left(\boldsymbol{s}^{(k)}\right)$ is leveraged to guide the routing policy learning to enhance the long-term utilities of the one-hop neighbors that traverse through various states in the learning process such as the battery level and the hop count. Specifically, the state value functions are exchanged among neighbors within the communication range to share the learning experience and coordinate the routing action based on the distributed value function approach. After receiving the ACK from the destination, the state value function $V\left(\boldsymbol{s}^{(k)}\right)$ is broadcast to provide learning parameters via control channel, and the corresponding state value function $V_j\left(\boldsymbol{s}_j^{(k)}\right), j\in\mathcal{N}$ is received from one-hop neighbors. In particular, the learning parameters on the same routing path are shared to utilize the experience that is closely related to routing success. In the absence of neighbors within the communication range along the routing path, the local policy is learned based on individual routing experiences.

The Q-value is updated using the Bellman iteration equation for distributed value function by learning a weighted sum of its own expected future utility $\max_{\hat{\boldsymbol{a}}\in\mathbf{A}}Q\left(\boldsymbol{s}^{(k+1)},\hat{\boldsymbol{a}}\right)$ and those of the one-hop neighbors $\sum_{j\in\mathcal{N}}V_j\left(\boldsymbol{s}_j^{(k)}\right)$, given by

$$Q\left(\boldsymbol{s}^{(k)},\boldsymbol{a}^{(k)}\right)\leftarrow(1-\alpha)\,Q\left(\boldsymbol{s}^{(k)},\boldsymbol{a}^{(k)}\right)+\alpha\Bigg(u^{(k)}+\lambda\Bigg(\max_{\hat{\boldsymbol{a}}\in\mathbf{A}}Q\left(\boldsymbol{s}^{(k+1)},\hat{\boldsymbol{a}}\right)+\upsilon\sum\nolimits_{j\in\mathcal{N}}V_j\left(\boldsymbol{s}_j^{(k)}\right)\Bigg)\Bigg), \quad (4)$$

where $\alpha\in(0,1]$ and $\lambda\in(0,1]$ indicate the weights of the current routing experience and the importance of future routing performance. The weight of the shared learning experience υ determines the impact of the state value function of the one-hop neighbors on the long-term utility.

By introducing the worst-case criterion constraint, the number of the latency violations denoted by $l^{(k)}$ indicates the maximum tolerable latency in different FANET applications.

Algorithm 1 RL based routing of UAV

Input: α, λ, υ, β, Υ, $\{\mu_j\}_{1\le j\le \upsilon}$, Γ, τ, ξ, Ω and φ
Output: Q and R

for $k = 1, 2, ..., K$ **do**
  Receive packets $q_{1\le m\le M}^{(k)}$
  Evaluate $\varpi^{(k)}$ and estimate $\boldsymbol{h}^{(k)}$
  Obtain $\xi^{(k)}$, $t^{(k-1)}$, $\boldsymbol{b}^{(k)}$ and $\varphi^{(k-1)}$ from $\mathcal{N}$
  Formulate $\boldsymbol{s}^{(k)}$ via (1)
  Select $\boldsymbol{a}^{(k)}$ via (2)
  if $x^{(k)} = 1$ **then**
    for $m = 1, 2, ..., M$ **do**
      if $\left(\rho_m^{(k)}, q_m^{(k)}\right) \in \Omega$ **then**
        Drop packet $q_m^{(k)}$
      else
        Calculate $\tilde{t}_m^{(k)}$ and $\tilde{\varphi}^{(k)}$
        Measure $\tilde{b}^{(k)}$ and $\tilde{\xi}^{(k)}$
        Add $\tilde{\xi}^{(k)}$, $\tilde{\varrho}^{(k)}$, $\tilde{t}_m^{(k)}$, $\tilde{b}^{(k)}$ and $\tilde{\varphi}^{(k)}$ to $q_m^{(k)}$
        Forward $q_m^{(k)}$ with relay power $p^{(k)}$
        $\Omega \leftarrow \left(\rho_m^{(k)}, q_m^{(k)}\right) \cup \Omega$
      end if
    end for
    Estimate $w^{(k)}$
  end if
  Obtain $\boldsymbol{\tau}^{(k)}$ from the feedback
  Calculate $u^{(k)}$ via (3)
  Share $V\left(\boldsymbol{s}^{(k)}\right)$ among $\mathcal{N}$
  Update $Q\left(\boldsymbol{s}^{(k)}, \boldsymbol{a}^{(k)}\right)$ via (4)
  Evaluate $l^{(k)}$ via (5)
  Update $R\left(\boldsymbol{s}^{(k)}, \boldsymbol{a}^{(k)}\right)$ via (6)
end for

The average end-to-end latency τ is compared with each of the Υ worst-case latency thresholds $\{\mu_j\}_{1\le j\le \Upsilon}$ to obtain the probability of satisfying the latency QoS requirement for service differentiation [30]. With the tolerance probability of attaining latency $\Gamma \in (0, 1]$, the number of the latency violations is defined as

$$l^{(k)} = \sum_{j=1}^{\Upsilon} \mathbb{1}\left(\Pr\left(\tau \le \mu_j\right) \le 1 - \Gamma\right), \tag{5}$$

where $\mathbb{1}(\cdot)$ is the indicator function. The risk value is updated with the risk learning rate $\beta \in (0, 1]$ and the number of the latency violations $l^{(k)}$, given by

$$R\left(\boldsymbol{s}^{(k)}, \boldsymbol{a}^{(k)}\right) \leftarrow (1-\beta)\, R\left(\boldsymbol{s}^{(k)}, \boldsymbol{a}^{(k)}\right) + \beta l^{(k)}. \tag{6}$$

V. Deep RL based FANET Routing

The deep RL based FANET routing (DRLFR) is proposed to address the state quantization error in terms of the latency and the channel gain under large-scale networks. The state is exchanged among one-hop neighbors to mitigate the estimation error of the network topology and channel conditions and increase the UAV perceptive field, thus improving the routing path exploration. The policy neural network and the risk neural network compress the state space with a large number of one-hop neighbors and evaluate the path quality and the latency violation, respectively. The shared learning parameters such as the neural network weights are exploited to mitigate the collision among the forwarders within the communication range, as summarized in Algorithm 2.

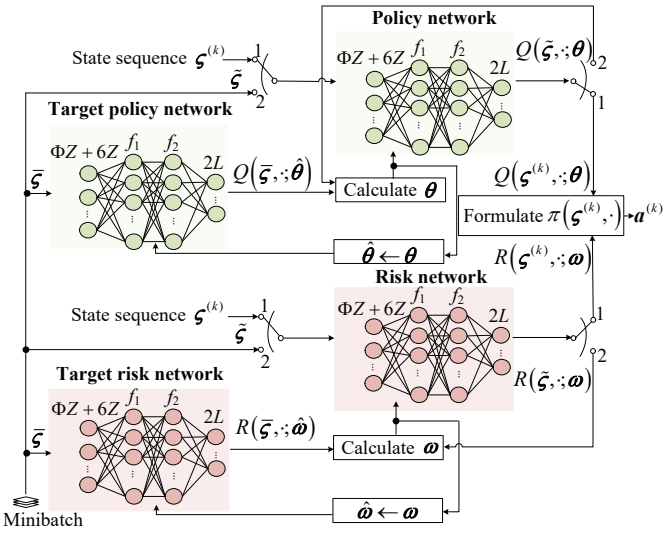


Fig. 4. Illustration of the neural networks in the deep RL based UAV routing, in which the policy neural network and the risk neural network are used to estimate the Q-value and the risk value, respectively.

At time slot k, the state $\boldsymbol{s}^{(k)}$ is formulated as in Algorithm 1 without quantization and exchanged among the neighbors within the communication range to share environment observations. The neighboring UAV has more chance to be selected to share the states such as the battery level and the number of the one-hop neighbors due to the radio channel similarity to the same receiving UAV compared with distant UAVs. With the state sequence $\boldsymbol{\varsigma}^{(k)} = [\boldsymbol{s}^{(k)}, [\boldsymbol{s}_j^{(k)}]_{1\le j\le Z}]$ including the first Z receiving states of the one-hop neighbors that contain more path quality information than $\boldsymbol{s}^{(k)}$ as the input, the policy neural network with weights $\boldsymbol{\theta}$ and the risk neural network with weights $\boldsymbol{\omega}$ estimate the Q-value $Q\left(\boldsymbol{\varsigma}^{(k)}, \cdot; \boldsymbol{\theta}\right)$ and the risk value $R\left(\boldsymbol{\varsigma}^{(k)}, \cdot; \boldsymbol{\omega}\right)$, respectively.

As shown in Fig. 4, each of the four neural networks with identical architecture, which consists of three fully-connected layers, i.e., an input layer with $(\Phi Z + 6Z)$ nodes, two hidden layers with f_1 and f_2 nodes and an output layer with $2L$ nodes. The routing action $\boldsymbol{a}^{(k)}$ is selected according to the routing policy distribution $\pi(\boldsymbol{\varsigma}^{(k)}, \cdot; \boldsymbol{\theta}, \boldsymbol{\omega})$ based on the Q-value and the risk value with modified Boltzmann distribution similar to [28]. The target neural networks with weights $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\omega}}$, respectively, are used to update the weights of the neural network to enhance the stability of the routing policy and thus mitigate the route oscillations.

Upon receiving the ACK from the destination, the routing utility $u^{(k)}$ and the number of the latency violations $l^{(k)}$ are evaluated via (3) and (5), respectively, to update the neural network weights $\boldsymbol{\theta}$ and $\boldsymbol{\omega}$ with the experience replay technique. The flag $f^{(k)}$ determines whether the destination

is one of the one-hop neighbors, guiding the forwarding decision towards a shorter routing path with fewer hops, thereby enhancing routing policy exploration for the large-scale networks. The experience denoted by $\boldsymbol{e}^{(k)}$ is stored in the memory pool $\mathcal{D}$, consisting of the current state sequence $\boldsymbol{\varsigma}^{(k)}$ and action $\boldsymbol{a}^{(k)}$, the routing utility $u^{(k)}$, the number of the latency violations $l^{(k)}$, the next state sequence $\boldsymbol{\varsigma}^{(k+1)}$ and the flag $f^{(k)}$, i.e.,

$$\boldsymbol{e}^{(k)} = \left\{\boldsymbol{\varsigma}^{(k)}, \boldsymbol{a}^{(k)}, u^{(k)}, \boldsymbol{\varsigma}^{(k+1)}, l^{(k)}, f^{(k)}\right\}. \tag{7}$$

The minibatch $\mathcal{B}$ is formulated via sampling J high priority routing experiences from $\mathcal{D}$, which is the input of both the evaluated and target neural networks. Specifically, the probability of choosing the ϵ-th routing experience sample $\mathrm{X}_{\epsilon}^{\tilde{\vartheta}} / \sum_{\tilde{\epsilon}} \mathrm{X}_{\tilde{\epsilon}}^{\tilde{\vartheta}}$ is used to update the neural network weights with a preference for routing experiences and improve the perception of delayed feedback from the destination, in which $\tilde{\vartheta}$ determines the degree of prioritization and X_{ϵ} represents the priority of each routing experience.

The neural network weights are updated to reduce the fitting error and thus enhance the estimation accuracy of the routing policy for fast learning. For each $\{\tilde{\boldsymbol{\varsigma}}, \tilde{\boldsymbol{a}}, \tilde{u}, \bar{\boldsymbol{\varsigma}}, \tilde{l}, \tilde{f}\} \in \mathcal{B}$, the policy neural network weights $\boldsymbol{\theta}$ is updated by minimizing the loss function between the estimated and the target Q-value, given by

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) =& \mathbb{E}_{\{\tilde{\boldsymbol{\varsigma}}, \tilde{\boldsymbol{a}}, \tilde{u}, \bar{\boldsymbol{\varsigma}}, \tilde{l}, \tilde{f}\} \in \mathcal{B}} \Bigg[\Bigg(\tilde{u} - Q\Big(\tilde{\boldsymbol{\varsigma}}, \tilde{\boldsymbol{a}}; \boldsymbol{\theta}\Big) \\ &+ \gamma Q\Big(\bar{\boldsymbol{\varsigma}}, \arg\max_{\boldsymbol{a} \in \mathbf{A}} Q\left(\bar{\boldsymbol{\varsigma}}, \boldsymbol{a}; \boldsymbol{\theta}\right); \hat{\boldsymbol{\theta}}\Big) \Bigg)^2 \Bigg]. \end{aligned} \tag{8}$$

Similarly, the risk neural network outputs the risk value of the routing policy that reduces the number of the latency violations, which is used to update the weights $\boldsymbol{\omega}$ via minimizing the loss function, given by

$$\begin{aligned}\mathcal{L}(\boldsymbol{\omega}) =& \mathbb{E}_{\{\tilde{\boldsymbol{\varsigma}}, \tilde{\boldsymbol{a}}, \tilde{u}, \bar{\boldsymbol{\varsigma}}, \tilde{l}, \tilde{f}\} \in \mathcal{B}} \Bigg[\Bigg(\tilde{l} - R\Big(\tilde{\boldsymbol{\varsigma}}, \tilde{\boldsymbol{a}}; \boldsymbol{\omega}\Big) \\ &+ \gamma R\Big(\bar{\boldsymbol{\varsigma}}, \arg\min_{\boldsymbol{a} \in \mathbf{A}} R\left(\bar{\boldsymbol{\varsigma}}, \boldsymbol{a}; \boldsymbol{\omega}\right); \hat{\boldsymbol{\omega}}\Big) \Bigg)^2 \Bigg]. \end{aligned} \tag{9}$$

The policy neural network weights $\boldsymbol{\theta}$ are broadcast to the one-hop neighbors $\mathcal{N}$, in which the neighboring UAVs $\mathcal{Z}$ on the same routing path are selected to updated $\boldsymbol{\theta}$ every C time slots through the distributed averaging consensus, given as

$$\boldsymbol{\theta} \leftarrow \frac{\boldsymbol{\theta} + \sum_{j \in \mathcal{Z}} \boldsymbol{\theta}_j}{|\mathcal{Z}| + 1}. \tag{10}$$

The risk neural network weights $\boldsymbol{\omega}$ are also updated with the weights of the neighboring UAVs $\mathcal{Z}$ similar to (10). In addition, the target network updates the weights $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\omega}}$ every C time slot by copying the weights of the policy neural network and the risk neural network. In particular, the parameter C is used to make a trade-off between the stability of the learning process and the computational cost under large-scale networks.

Algorithm 2 Deep RL based routing of UAV

Input: γ, τ, ξ, Z, J and C
Output: $\boldsymbol{\theta}$ and $\boldsymbol{\omega}$

for $k = 1, 2, ..., K$ **do**
  Same as Lines 3−12 in Algorithm 1
  Exchange $\boldsymbol{s}^{(k)}$ with $\mathcal{N}$
  $\boldsymbol{\varsigma}^{(k)} = \left[\boldsymbol{s}^{(k)}, \left[\boldsymbol{s}_j^{(k)}\right]_{1 \le j \le Z}\right]$
  Input $\boldsymbol{\varsigma}^{(k)}$ to the policy neural network and the risk neural network and obtain $\pi(\boldsymbol{\varsigma}^{(k)}, \cdot; \boldsymbol{\theta}, \boldsymbol{\omega})$
  Select $\boldsymbol{a}^{(k)}$ based on $\pi(\boldsymbol{\varsigma}^{(k)}, \cdot; \boldsymbol{\theta}, \boldsymbol{\omega})$
  Same as Lines 14−23 in Algorithm 1
  Evaluate $l^{(k)}$ via (5)
  Formulate $\boldsymbol{e}^{(k)}$ via (7)
  $\mathcal{D} \leftarrow \mathcal{D} \cup \boldsymbol{e}^{(k)}$
  Randomly sample $\mathcal{B}$ with J experience from $\mathcal{D}$
  Update $\boldsymbol{\theta}$ and $\boldsymbol{\omega}$ via (8) and (9)
  if k mod $C = 0$ **then**
    Share $\boldsymbol{\theta}$ and $\boldsymbol{\omega}$ among $\mathcal{N}$
    Update $\boldsymbol{\theta}$ and $\boldsymbol{\omega}$ via (10)
    $\hat{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta}$ and $\hat{\boldsymbol{\omega}} \leftarrow \boldsymbol{\omega}$
  end if
end for

VI. Performance Analysis

A. Computational Complexity

The computational complexities of the proposed schemes rely on the update of Q-value, risk value, routing policy distribution and four neural networks and the number of multiplications in the routing policy selection. According to [31], the computational complexity of the proposed RLFR is $\mathcal{O}(\Phi L)$, which mainly depends on the state and action space, i.e., the number of the knowledge shared from the one-hop neighbors Φ such as the channel gain and the battery level, as well as the quantized relay power level L.

The computational complexity required by DRLFR relies on the number of sampled routing experiences and multiplications in the policy neural network and risk neural network shown in Fig. 4. More specifically, the computational cost in training the neural networks with the fully-connected layers is determined by forward and backward propagation, and thus depending on the $(\Phi Z + 6Z)$ inputs, f_1 and f_2 neurons in the hidden layers and $2L$ outputs for choosing the forwarding decision and the L relay power levels. Thus, the number of multiplications in the forward propagation of the policy neural network is given by

$$X_1 = (\Phi Z + 6Z) f_1 + (f_1 + 1) f_2 + 2 (f_2 + 1) L. \tag{11}$$

The policy neural network performs X_2 multiplications in the backward propagation, given by

$$X_2 = 2 (\Phi Z + 6Z) f_1 + 2 (f_1 + 1) f_2 + 6 (f_2 + 1) L. \tag{12}$$

According to [32], the number of neurons in the hidden layers relies on the output dimension $2L$ and the number of

the experience sample K, i.e.,

$$f_1 = \sqrt{K(2L+2)} + 2\sqrt{\frac{K}{(2L+2)}}, \tag{13}$$

and that in the second hidden layer is

$$f_2 = 2L\sqrt{\frac{K}{(2L+2)}}. \tag{14}$$

Similarly, the risk neural network with identical network structure to the policy neural network performs X_1 and X_2 multiplications in the forward and back propagation to estimate the latency risk. The DRLFR scheme samples J experiences every time slot to update the neural network parameters. In particular, the update of the policy neural network weights $\boldsymbol{\theta}$ and the risk neural network weights $\boldsymbol{\omega}$ requires both the forward and backward propagation, while the target networks only execute forward propagation.

Theorem 1: The computational complexities of RLFR and DRLFR are given by $\mathcal{O}(\Phi L)$ and $\mathcal{O}(JKL)$.

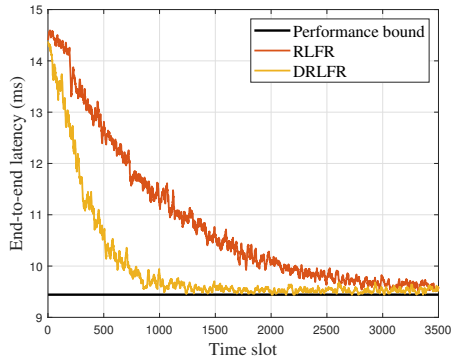
Proof. See Appendix A. □

Remark 1: The computational complexities of the proposed routing schemes increase with the quantized relay power level L, in which the RLFR also depends on the number of shared knowledge from the one-hop neighbors Φ, i.e., the channel gain and the battery level. The computational complexity of DRLFR increases with the number of experience samples K and the sampled experience size J, which is available for the UAV with sufficient computational resources under large-scale networks. Both the proposed schemes have been successfully implemented in Raspberry Pi-4B with low computation capacity that has the Cortex-A72 processor, 4-core CPU and 4 GB memory.

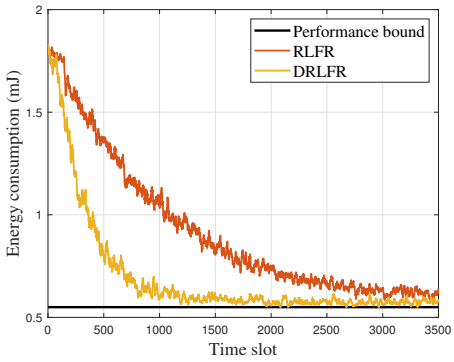
B. Performance Bound

The routing process is formulated as a packet forwarding game among the N UAVs to provide the bound regarding the end-to-end latency τ and the routing energy consumption of the FANET denoted by W. In this game, UAVs i choose the routing policy $\boldsymbol{a}_i$, including the forwarding decision $x_i \in \{0,1\}$ and the relay power $p_i \in [\underline{P}, \overline{P}]$, to maximize the utility u_i. The forwarding strategy in the routing process depends on the judgment of its own benefits and the source as the initiator of the routing game always tries to select cooperative strategy due to being task driven.

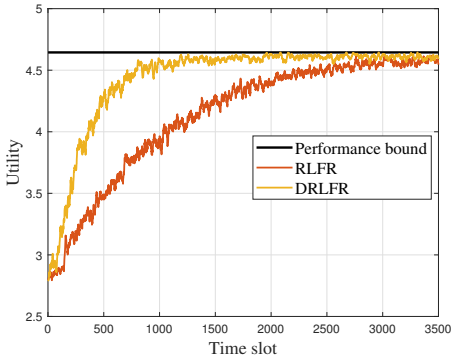
The z_i-bit packet is forwarded to the one-hop neighbors $\mathcal{N}_i$ with relay power p_i, resulting in the one-hop latency $t_i = T_i + z_i/r_i$, $1 \le i \le N$, where T_i is the contention latency and r_i is the data rate between UAV i and next-hop. With the bandwidth B and the receiver noise power σ^2, the data rate is given as $r_i = B\log_2\left(1 + p_i h_i/\sigma^2\right)$, where h_i is the channel gain from UAV i to next-hop. In particular, the channel gain h_i is assumed to rely on the adjacent distance d_i and the path-loss parameter ς. The utility depends on the delivery success indicator κ, the end-to-end latency with ϱ hops, and the communication energy consumption caused by itself according to (3), i.e.,



(a) End-to-end latency



(b) Routing energy consumption of the FANET



(c) Utility

Fig. 5. Upper performance bound of the 20-UAV FANET routing in Theorem 2.

$$u_i = \kappa - c_2 x_i p_i \frac{z_i}{r_i} - c_1 \sum_{j=1}^{\varrho} T_j + \frac{z_j}{r_j}. \tag{15}$$

If the packet is not successfully received, the end-to-end latency incurs a penalty, which is determined by either the packet expiration time or the maximum retransmission time.

Theorem 2: The performance bound of the proposed RL based routing schemes for UAV $i \in \{1, 2, \ldots, N\}$ with the same packet size z, contention latency T and adjacent distance d is given by

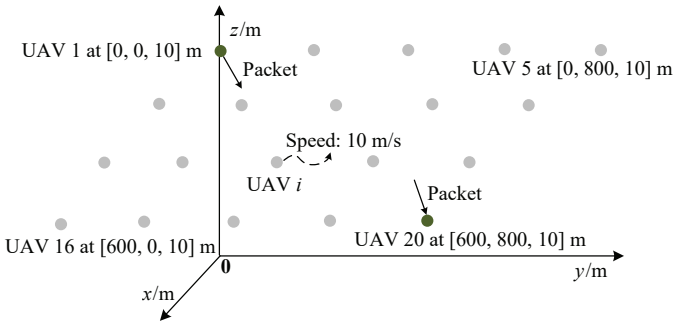


Fig. 6. Simulation setting for the UAV 1 as the source sends a 10 KB packet to the UAV 20 as the destination in each time slot in a FANET with each UAV moving in a direction randomly chosen with 10 m/s speed.

TABLE III: Parameter setting in the simulation.

Parameter	**Value**
FANET size N	20-60
UAV speed v	10-50 m/s
Maximum UAV transmit power p	100 mW
Communication range D	200 m
Bandwidth B	20 MHz
Packet size z	10 KB
Receiver noise power σ^2	-174 dBm
Learning rate α	0.7
Discount factor λ	0.9
Number of neurons $f_{1/2}$	64/128
Minibatch size J	64
Risk value weight c	0.5
Risk learning rate β	0.8
Utility function weight $c_{1/2}$	0.8/0.6

$$\tilde{\tau} = NT + \frac{Nz}{B\log_2\left(\underline{P}d^{-\varsigma}+1\right)}, \tag{16}$$

$$\tilde{W} = \frac{N\underline{P}z}{B\log_2\left(\underline{P}d^{-\varsigma}+1\right)}, \tag{17}$$

$$\tilde{u}_i = 1 - c_1\bar{\tau} - c_2\frac{\underline{P}z}{B\log_2\left(\underline{P}d^{-\varsigma}+1\right)}, \tag{18}$$

if

$$h_i > \sqrt{\frac{\overline{P}^3\left(c_1+c_2\overline{P}\right)}{c_2 B}}, \tag{19}$$

$$d < \underline{D} < \overline{D} < 2d, \tag{20}$$

where $\overline{D}$ and $\underline{D}$ are the communication distance of the UAV with maximum and minimal relay power $\overline{P}$ and $\underline{P}$, respectively.

Proof. See Appendix B. □

Remark 2: If the channel gain h_i is higher than a bound under network state as given in (19) and (20), all UAVs cooperatively choose to forward the packet with minimal relay power $\underline{P}$. Any unilateral adjustment in strategy leads to routing process failure, in which the cost of failure exceeds the gains of the energy consumption saving. In this case, the bound of the end-to-end latency given by (16) linearly increases with the FANET size N and logarithmically increases with the adjacent distance d. As shown in Fig. 5, the proposed schemes converge to the bound provided in (16)-(18). For example, DRLFR takes less than 1500 time slots to converge to 9.5 ms end-to-end latency for 10 KB packet transmission, which is less than 1.1% compared with the bound.

VII. Simulation Results

As shown in Fig. 6, 20 UAVs with 10 m minimal distance moving at 10 m/s circulate around the respective waypoints based on the Gauss-Markov mobility model. The UAV 1 as the source chosen on the border that transmits 10 KB packet generated from a constant bit rate source at rate of 50 packets per second with 100 mW transmit power to the destination aided by the other UAVs at 2.4 GHz frequency and with relay power up to 100 mW, i.e., $\{25j|0 \le j \le 4\}$ mW [7]. The time to live is set as 15 hops to ensure that the packets are routed through a limited hop count. The receiver noise power and the maximum communication range are set as -174 dBm and 200 m, respectively, with a channel bandwidth of 20 MHz and an SNR threshold of 20 dB according to [17]. The channel gain $\boldsymbol{h}$ is assumed to remain quasi-static for the duration of each time slot and rely on the path loss model in free-space propagation given by [33] as

$$PL(\tilde{d}) = 10\varsigma\log_{10}\tilde{r} + PL_0,\ \tilde{d} \ge 1 \text{ m}, \tag{21}$$

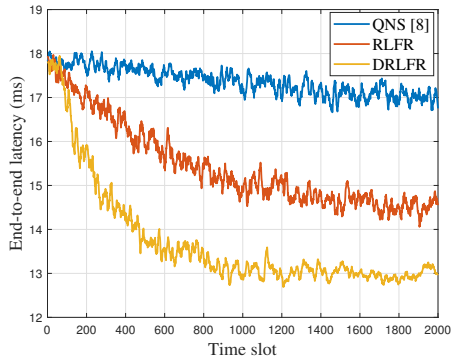
where $\tilde{d}$ is the hopping distance, $\varsigma = 2.05$ is the path-loss parameter and PL_0 represents the reference path loss.

The weight coefficient $c = 0.5$ is used to balance the risk value and the Q-value on the routing policy distribution, and the weights of long-term utility receiving from one-hop neighbors $\upsilon = 0.05$. The parameter $C = 50$ is fine-tuned to make a trade-off between the routing performance and the complexity. The main environment and learning parameters are shown in Table III.

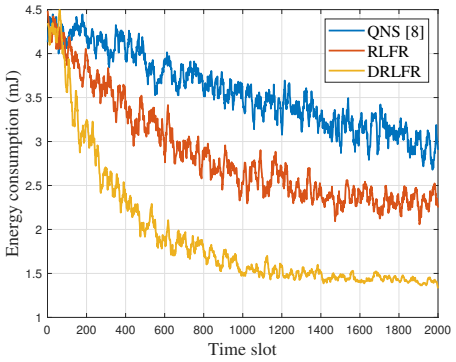
As shown in Fig. 7, the proposed routing schemes reduce the average end-to-end latency with less routing energy consumption of the FANET over time under the network topology and mobility model shown in Fig. 6. For example, the proposed RLFR decreases 17.7% average end-to-end latency and saves 48.8% routing energy consumption after 1500 time slots, because the safe policy exploration reduces the probability of routing action with high latency, and the learning experiences shared among one-hop neighbors coordinates the forwarding decision and power allocation.

The benchmark QNS in [8] applies Q-learning to choose the next-hop to transmit the packets along single-path according to one-hop neighbor information such as the channel condition and UAV position to support the typical FANET routing system, while the position-based routing in [20] considers two-hop neighbors as shown in Table I. Compared with QNS, the proposed RLFR reduces 12.9% average end-

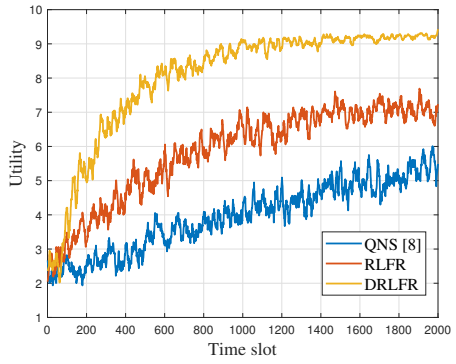
This article has been accepted for publication in IEEE Transactions on Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCOMM.2024.3409561



(a) End-to-end latency

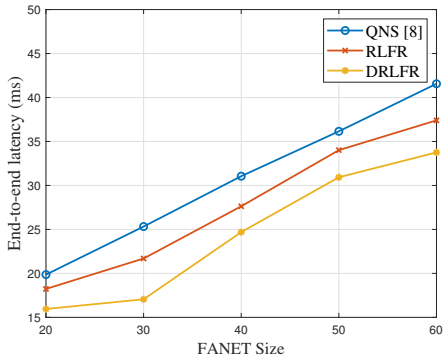


(b) Routing energy consumption of the FANET

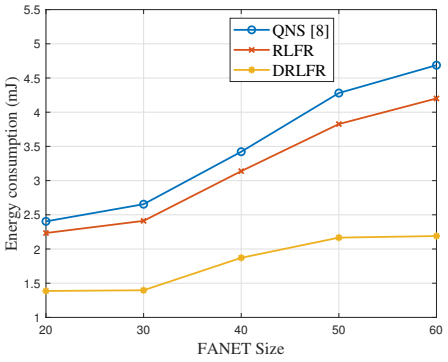


(d) Utility

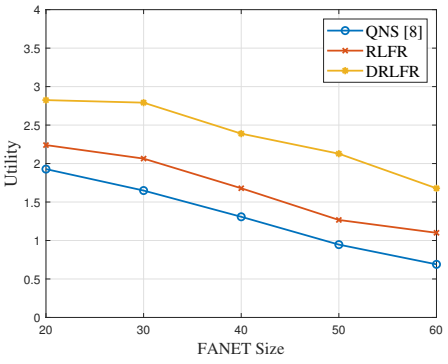
Fig. 7. Routing performance of a 20-UAV FANET as shown in Fig. 6, in which the source UAV aims to send a 10 KB packet to UAV 20 as the destination in each time slot.



(a) End-to-end latency



(b) Routing energy consumption of the FANET



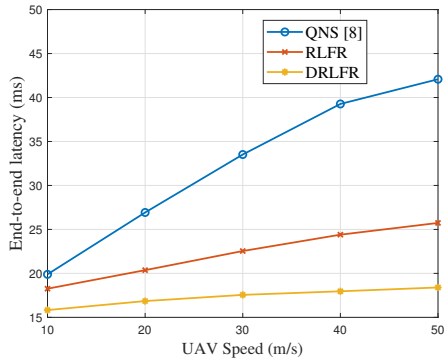
(c) Utility

Fig. 8. Average performance of the UAVs moving at 10 m/s over 30 runs each with 2,000 time slots, with the FANET size increasing from 20 to 60.

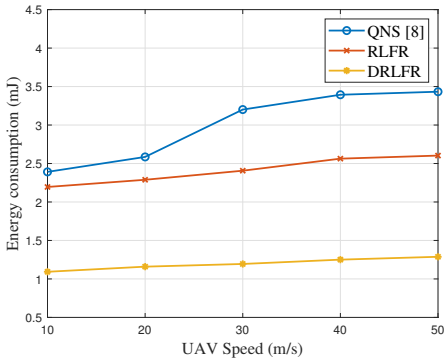
to-end latency and 24.1% routing energy consumption after 1500 time slots, because the benchmark scheme assigns only one neighbor as the forwarder, resulting in the routing path failure if the link to this neighbor is poor, despite the potential overhearing of the packets sent by other neighbors.

Through the utilization of deep learning, the policy neural network and the risk neural network address the state quantization errors of channel gains and latency to improve the routing performance. For example, our proposed deep version DRLFR further reduces 12.8% average end-to-end latency and 36.3% routing energy consumption compared with RLFR after 1200 time slots. Compared with QNS, the proposed deep version decreases 24.5% average end-to-end latency and 41.6% routing energy consumption, because the neural networks utilize the routing decision based selective forwarding to enhance the path diversity and compress the action space of the power allocation to enable the packets to be transmitted to the destination with reduced overhead.

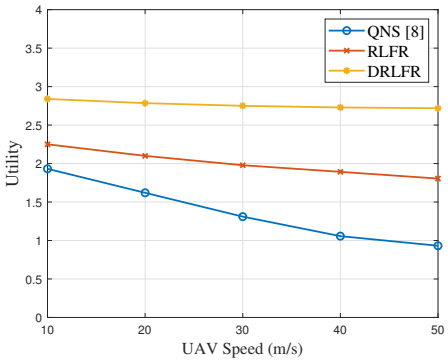
The average routing performance over 30 runs as depicted in Fig. 8 is validated for FANET with up to 60 UAVs. For example, compared with QNS, the proposed RLFR decreases 11.4% routing energy consumption and 11.2% average end-to-end latency for the 40 UAVs due to the safe policy exploration with latency constraint and the sharing of routing experience among one-hop neighbors. The performance gain of DRLFR in terms of average end-to-end latency increase



(a) End-to-end latency



(b) Routing energy consumption of the FANET



(c) Utility

Fig. 9. Average performance of a 20-UAV FANET over 30 runs each containing 2000 time slots, with the UAV speed changing from 10 m/s to 50 m/s.

with the FANET size to larger than 16.2%, because the neural networks compress the state space in terms of the number of the shared knowledge from one-hop neighbors such as the battery level and the channel gain through the state mapping of the hidden layers under large-scale networks.

As shown in Fig. 9, the average routing performance over 30 runs is verified for the UAV speed increasing from 10 to 50 m/s. For example, the proposed RLFR reduces 32.3% average end-to-end latency and saves 25.1% routing energy consumption at a speed of 30 m/s compared with QNS, owing to the utilization of path diversity that ensures end-to-end availability by mitigating the possibility of concurrent failure across all paths connecting the source and destination.

Fig. 10. Experimental setting of a FANET, in which each of the 5 UAVs sends the captured images to the control center with transmit power up to 10 mW.

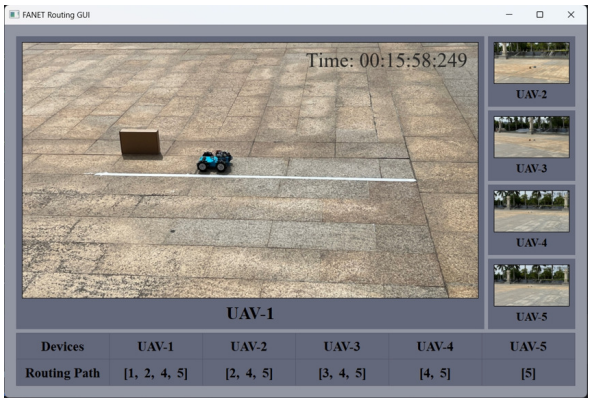


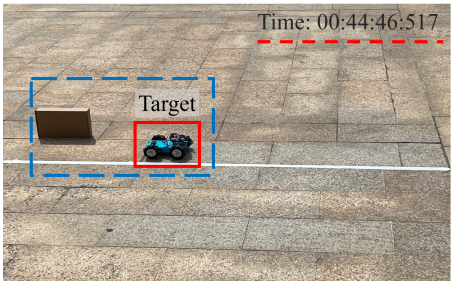
Fig. 11. GUI of the FANET routing system implemented at the control center.

DRLFR further reduces 51.4% average end-to-end latency and 63.9% routing energy and the performance gain increases with the speed to 57.1% and 65.1%, respectively, due to the shared observations of neighboring UAVs in the RL state to enhance the estimation accuracy of the channel gain and latency under high speed.

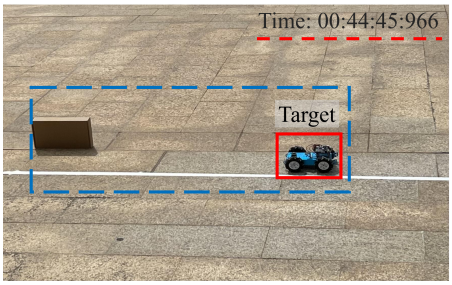
VIII. Experimental Results

Experiments have been performed for image transmission in the FANET with initial topology as shown in Fig. 10. Each of the 5 UAVs equipped with Raspberry Pi-4B that has a Cortex-A72 processor, 4-core CPU and 4 GB memory, which has less computation resources compared with UAV systems utilizing NVIDIA Jetson Xavier NX processor, 6-core CPU and 8 GB memory as presented in [34]. The captured images containing 100 packets, each of 2-KB, are sent to track the target vehicle that moved at a speed of 1m/s. The resulting packet was sent to a laptop with Intel i7-11700F as the control center in each time slot with transmit power up to 10 mW at 2.4 GHz following IEEE 802.11n. The GUI as shown in Fig. 11 displays the images sent from the 5 UAVs and received by the control center as well as the routing path.

As shown in Fig. 12, the image transmission performance regarding the end-to-end latency is compared with the benchmark scheme QNS, in which the proposed RLFR has lower latency by sending the latest captured image of the target



(a) RLFR



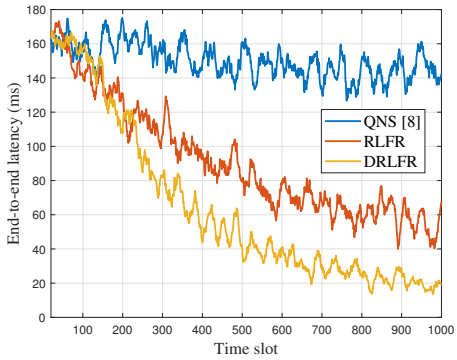
(b) QNS [8]

Fig. 12. FANET image transmission performance in the experimental setting as shown in Fig. 10, showing that the proposed RLFR has 551 ms less routing latency than QNS as proposed in [8].

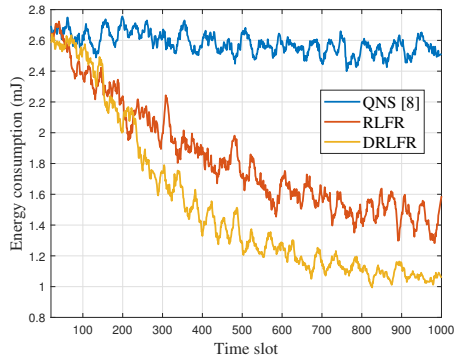
vehicle from UAV 1 to the control center. The results averaged by 20 repeated experiments presented in Fig. 13 show that the proposed schemes enhance the routing performance with lower routing energy consumption and end-to-end latency and higher utility. For example, the proposed RLFR reduces 59.2% average end-to-end latency and saves 44.0% routing energy consumption over 600 time slots compared with the benchmark QNS scheme, owing to the latency constraint and the avoidance of routing loops. The policy neural network and risk neural network based fully-connected layers in DRLFR further improve the routing performance, achieving an 84.6% reduction in average end-to-end latency and a 54.1% decrease in routing energy consumption.

IX. Conclusion

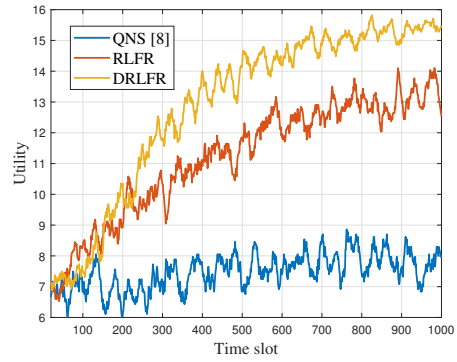
In this paper, an RL based energy-efficient fast routing for FANETs has been proposed to support applications such as target tracking that depend on the fast transmission of sensing data and flight commands. Based on the packet forwarding game among the collaborative UAVs, the bound of the proposed scheme has been provided, showing that the routing energy consumption and the end-to-end latency linearly increase with the FANET size and logarithmically increase with the average distance under the specified network topology and channel conditions. The bound has been verified via the simulation results based on up to 60 UAVs. Experiments results based on 5 UAVs equipped with Raspberry Pi-4B to track the target based on the received images at the control center show the performance gain over the benchmark QNS, with end-to-end latency and routing energy consumption reduced by 84.6% and 54.1%, respectively.



(a) End-to-end latency



(b) Routing energy consumption of the FANET



(c) Utility

Fig. 13. Experimental results for the 5-UAV FANET to support target tracking application, with topology as shown in Fig. 10.

The promising future work includes the anti-jamming routing technique that selects the routing path to bypass the jammed regions and improve the transmission reliability to support ultra-reliable applications such as emergency assistance and safety flight guidance. Another interesting topic is the routing recovery mechanism that rapidly identifies the alternative route to maintain the network stability of FANETs against the departure of specific UAVs. The impact of the propulsion energy consumption on the communication performance can also be investigated.

APPENDIX A
PROOF OF THEOREM 1

Proof: By (11)-(14), the computational complexity of DRLFR is given by

$$\begin{aligned}\Gamma =&\mathcal{O}\Big(\left(4J+2\right)X_1+2JX_2\Big) &(22)\\ =&\mathcal{O}\Big(8J\left(\Phi Z+6Z\right)f_1+8J\left(f_1+1\right)f_2\\ &+10J\left(f_2+1\right)2L\Big) &(23)\\ =&\mathcal{O}\left(J\Phi Z\sqrt{KL}+JKL+JL\sqrt{KL}\right) &(24)\\ =&\mathcal{O}\left(JKL\right), &(25)\end{aligned}$$

where (23) is obtained by (11) and (12), (24) is obtained by (13) and (14), and (25) is obtained as $K \gg \Phi Z$ in [32].

APPENDIX B
PROOF OF THEOREM 2

Proof: By (15) and (20), $\forall i \in \{1,2,\ldots,N\}$, $x_i \in \{0,1\}$ and $p_i \in \left[\underline{P},\overline{P}\right]$, we have

$$\begin{aligned}&u_i\left(\left[x_i,p_i\right],\left[1,\underline{P}\right]^{N-1}\right)\\ =&\kappa-c_1\varrho T-c_1\sum_{j=1}^{\varrho}\frac{z}{r_j}-c_2x_ip_i\frac{z}{r_i} &(26)\\ \le&1-c_1NT-c_1\sum_{j=1}^{N}\frac{z}{r_j}-c_2p_i\frac{z}{r_i} &(27)\\ =&u_i\left(\left[1,p_i\right],\left[1,\underline{P}\right]^{N-1}\right). &(28)\end{aligned}$$

By (19), $\forall p_i \in \left[\underline{P},\overline{P}\right]$, we have

$$\begin{aligned}&\frac{\partial u_i\left(\left[1,p_i\right],\left[1,\underline{P}\right]^{N-1}\right)}{\partial p_i}\\ =&\frac{\left(c_1+c_2p_i\right)\left(1+p_ih_i\right)^2}{B{h_i}^2}-\frac{c_2}{\log_2\left(1+p_ih_i\right)} &(29)\\ \le&\frac{\left(c_1+c_2\overline{P}\right)\left(1+\overline{P}h_i\right)^2}{B{h_i}^2}-\frac{c_2}{\log_2\left(1+\overline{P}h_i\right)}<0. &(30)\end{aligned}$$

Thus, by (28) and (30),

$$u_i\left(\left[1,p_i\right],\left[1,\underline{P}\right]^{N-1}\right)\le u_i\left(\left[1,\underline{P}\right],\left[1,\underline{P}\right]^{N-1}\right). \quad (31)$$

By (26), (27) and (30), $\forall x_i \in \{0,1\}$ and $p_i \in \left[\underline{P},\overline{P}\right]$,

$$u_i\left(\left[x_i,p_i\right],\left[1,\underline{P}\right]^{N-1}\right)\le u_i\left(\left[1,\underline{P}\right],\left[1,\underline{P}\right]^{N-1}\right). \quad (32)$$

By (26)-(30), $\forall j \in \mathcal{N}_{-i}$, $\boldsymbol{x}_{-j} \in \{0,1\}^{N-1}$ and $\boldsymbol{p}_{-j} \in \left[\underline{P},\overline{P}\right]^{N-1}$, we have

$$u_j\left(\left[1,\underline{P}\right],\left\{\boldsymbol{x}_{-j},\boldsymbol{p}_{-j}\right\}\right)\le u_j\left(\left[1,\underline{P}\right],\left[1,\underline{P}\right]^{N-1}\right). \quad (33)$$

Thus, by (32) and (33), $\left(\left[1,\underline{P}\right],\left[1,\underline{P}\right]^{N-1}\right)$ is an Nash equilibrium of the routing game. According to [35], by (19), (20), and (26)-(30), the performance bound of the proposed routing scheme is given by (16)-(18).

REFERENCES

[1] X. Qi, J. Li, Z. Lv, *et al.*, "Reinforcement learning based energy-efficient routing with latency constraints for FANETs," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, pp. 2638–2643, Kuala Lumpur, Malaysia, Dec. 2023.

[2] D. S. Lakew, U. Saad, N.-N. Dao, *et al.*, "Routing in flying ad-hoc networks: A comprehensive survey," *IEEE Commun. Surv. Tutor.*, vol. 22, no. 2, pp. 1071–1120, 2th Quart. 2020.

[3] O. S. Oubbati, M. Atiquzzaman, P. Lorenz, *et al.*, "Routing in flying ad-hoc networks: Survey, constraints, and future challenge perspectives," *IEEE Access*, vol. 7, pp. 81057–81105, Jun. 2019.

[4] T. Bouzid, N. Chaib, M. L. Bensaad, *et al.*, "5G network slicing with unmanned aerial vehicles: Taxonomy, survey, and future directions," *Trans. emerg. telecommun. technol.*, vol. 34, no. 3, p. e4721, Mar. 2023.

[5] O. S. Oubbati, A. Lakas, F. Zhou, *et al.*, "A survey on position-based routing protocols for flying ad-hoc networks (FANETs)," *Veh. Commun.*, vol. 10, pp. 29–56, Oct. 2017.

[6] A. Bujari, C. E. Palazzi, and D. Ronzani, "A comparison of stateless position-based packet routing algorithms for FANETs," *IEEE Trans. Mob. Comput.*, vol. 17, no. 11, pp. 2468–2482, Nov. 2018.

[7] X. Qiu, S. Zhang, Z. Wang, *et al.*, "Integrated host-and content-centric routing for efficient and scalable networking of UAV swarm," *IEEE Trans. Mobile Comput.*, vol. 23, no. 4, pp. 2927–2942, Apr. 2024.

[8] L. Zhang, X. Ma, Z. Zhuang, *et al.*, "Q-Learning aided intelligent routing with maximum utility in cognitive UAV swarm for emergency communications," *IEEE Trans. Veh. Technol.*, vol. 72, no. 3, pp. 3707–3723, Mar. 2023.

[9] Z. Zheng, A. K. Sangaiah, and T. Wang, "Adaptive communication protocols in flying ad-hoc network," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 136–142, Jan. 2018.

[10] S. Gangopadhyay and V. K. Jain, "A position-based modified OLSR routing protocol for flying ad-hoc networks," *IEEE Trans. Veh. Technol.*, vol. 72, no. 9, pp. 12087–12098, Sep. 2023.

[11] S. Safavat and D. B. Rawat, "On the elliptic curve cryptography for privacy-aware secure ACO-AODV routing in intent-based internet of vehicles for smart cities," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 8, pp. 5050–5059, Aug. 2021.

[12] H. Fatemidokht, M. K. Rafsanjani, B. B. Gupta, *et al.*, "Efficient and secure routing protocol based on artificial intelligence algorithms with UAV-assisted for vehicular ad hoc networks in intelligent transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4757–4769, Jul. 2021.

[13] O. S. Oubbati, M. Mozaffari, N. Chaib, *et al.*, "ECaD: Energy-efficient routing in flying ad-hoc networks," *Int J Commun Syst*, vol. 32, no. 18, p. e4156, Dec. 2019.

[14] J. Yang, K. Sun, H. He, *et al.*, "Dynamic virtual topology aided networking and routing for aeronautical ad-hoc networks," *IEEE Trans. Commun.*, vol. 70, no. 7, pp. 4702–4716, Jul. 2022.

[15] H. I. Abbasi, R. C. Voicu, J. A. Copeland, *et al.*, "Towards fast and reliable multihop routing in VANETs," *IEEE Trans. Mobile Comput.*, vol. 19, no. 10, pp. 2461–2474, Oct. 2020.

[16] H. Song, L. Liu, S. M. Pudlewski, *et al.*, "Random network coding enabled routing protocol in unmanned aerial vehicle networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 12, pp. 8382–8395, Dec. 2020.

[17] H. Song, L. Liu, B. Shang, *et al.*, "Enhanced flooding-based routing protocol for swarm UAV networks: Random network coding meets clustering," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, pp. 1–10, Vancouver, BC, Canada, May 2021.

[18] R. Sanchez-Iborra and M.-D. Cano, "Joker: A novel opportunistic routing protocol," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1690–1703, May 2016.

[19] N. Li, J.-F. Martinez-Ortega, V. H. Diaz, *et al.*, "Probability prediction-based reliable and efficient opportunistic routing algorithm for VANETs," *IEEE/ACM Trans. Netw.*, vol. 26, no. 4, pp. 1933–1947, Aug. 2018.

[20] M. Y. Arafat and S. Moh, "A Q-learning-based topology-aware routing protocol for flying ad-hoc networks," *IEEE Internet Things J.*, vol. 9, no. 3, pp. 1985–2000, Feb. 2022.

[21] S. Swain, P. M. Khilar, and B. R. Senapati, "A reinforcement learning-based cluster routing scheme with dynamic path planning for multi-UAV network," *Veh. Commun.*, vol. 41, p. 100605, Jun. 2023.

[22] L. A. L. da Costa, R. Kunst, and E. P. de Freitas, "Q-FANET: Improved Q-learning based routing protocol for FANETs," *Comput. Networks*, vol. 198, p. 108379, Aug. 2021.

[23] X. Qiu, L. Xu, P. Wang, *et al.*, "A data-driven packet routing algorithm for an unmanned aerial vehicle swarm: A multi-agent reinforcement learning approach," *IEEE Wireless Commun. Lett.*, vol. 11, no. 10, pp. 2160–2164, Oct. 2022.

[24] Z. Wang, H. Yao, T. Mai, *et al.*, "Learning to routing in UAV swarm network: a multi-agent reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 14, no. 8, pp. 1–14, May 2022.

[25] R. Ding, J. Chen, W. Wu, *et al.*, "Packet routing in dynamic multi-hop UAV relay network: A multi-agent learning approach," *IEEE Trans. Veh. Technol.*, vol. 71, no. 9, pp. 10059–10072, Sep. 2022.

[26] S. Hayat, E. Yanmaz, and R. Muzaffar, "Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint," *IEEE Commun. Surv. Tutor.*, vol. 18, no. 4, pp. 2624–2661, 4th Quart. 2016.

[27] Y. Zhang, J. Lyu, and L. Fu, "Energy-efficient trajectory design for UAV-aided maritime data collection in wind," *IEEE Trans. Wireless Commun.*, vol. 21, no. 12, pp. 10871–10886, Dec. 2022.

[28] X. Lu, L. Xiao, G. Niu, *et al.*, "Safe exploration in wireless security: A safe reinforcement learning algorithm with hierarchical structure," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 732–743, Feb. 2022.

[29] E. Parisotto, J. L. Ba, and R. Salakhutdinov, "Actor-mimic: Deep multitask and transfer reinforcement learning," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, pp. 1–16, San Juan, PR, May 2016.

[30] W. Zhuang, Q. Ye, F. Lyu, *et al.*, "SDN/NFV-empowered future IoV with enhanced communication, computing, and caching," *Proc. IEEE*, vol. 108, no. 2, pp. 274–291, Feb. 2020.

[31] Z. Lv, L. Xiao, Y. Du, *et al.*, "Multi-agent reinforcement learning based UAV swarm communications against jamming," *IEEE Trans. Wireless Commun.*, vol. 22, no. 12, pp. 9063–9075, Dec. 2023.

[32] G.-B. Huang, "Learning capability and storage capacity of two-hidden-layer feedforward networks," *IEEE Trans. Neural Netw.*, vol. 14, no. 2, pp. 274–281, Mar. 2003.

[33] Y. Chen, N. Zhao, Z. Ding, *et al.*, "Multiple UAVs as relays: Multi-hop single link versus multiple dual-hop links," *IEEE Trans. Wireless Commun.*, vol. 17, no. 9, pp. 6348–6359, Sep. 2018.

[34] H. Guo, Y. Zheng, Y. Zhang, *et al.*, "Global-local MAV detection under challenging conditions based on appearance and motion," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–13, Apr. 2024.

[35] Y. Xu, J. Liu, Y. Shen, *et al.*, "QoS-aware secure routing design for wireless networks with selfish jammers," *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 4902–4916, Aug. 2021.

Jieling Li (Student Member, IEEE) received the M.S. degree from the College of Computer and Data Science, Fuzhou University, China, in 2022. He is currently pursuing a Ph.D. degree with the Department of Informatics and Communication Engineering, Xiamen University. His research interests include network security, wireless communications and reinforcement learning.

Liang Xiao (Senior Member, IEEE) received the B.S. degree in communication engineering from the Nanjing University of Posts and Telecommunications, China, in 2000, the M.S. degree in electrical engineering from Tsinghua University, China, in 2003, and the Ph.D. degree in electrical engineering from Rutgers University, NJ, USA, in 2009. She was a Visiting Professor with Princeton University, Virginia Tech, and the University of Maryland, College Park. She is currently a Professor with the Department of Information and Communication Engineering, Xiamen University, Xiamen, China. She was a recipient of the Best Paper Award for 2016 INFOCOM Big Security WS and 2017 ICC. She has served as an Associate Editor for IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY and a Guest Editor for IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING.

Xuchen Qi received the B.S. degree from Tongji University, China, in 2020. He is currently pursuing a M.S. degree with the Department of Informatics and Communication Engineering, Xiamen University. His research interests include network security and reinforcement learning.

Zefang Lv (Graduate student Member, IEEE) received her B.S. degree in statistics from Shandong University in 2016 and her M.S. degree in applied statistics from North China Electric Power University in 2020. She is currently pursuing a Ph.D. degree with the Department of Information and Communication Engineering, Xiamen University. Her research interests include network security, wireless communications and reinforcement learning.

Qiaoxin Chen received the B.S. degree in electronic and information engineering from Fuzhou University, in 2020, the M.S. degree from Fujian Normal University, China, in 2023. He is currently pursuing a Ph.D. degree with the Department of Informatics and Communication Engineering, Xiamen University. His research interests include network security and wireless communication.

Yong-Jin Liu (Senior Member, IEEE) is a full professor with the Department of Computer Science and Technology, Tsinghua University, China. He received the BEng degree from Tianjin University, China, in 1998, and the PhD degree from the Hong Kong University of Science and Technology, Hong Kong, China, in 2004. His research interests include machine learning, cognitive computation, computer graphics and computer-aided design. For more information, visit https://cg.cs.tsinghua.edu.cn/people/Yongjin/Yongjin.htm.