Fully-Echoed Q-Routing With Simulated Annealing Inference for Flying Adhoc Networks

Arnau Rovira-Sugranes, Fatemeh Afghah, Junsuo Qu, and Abolfazl Razi

***Abstract*—Current networking protocols deem inefficient in accommodating the two key challenges of Unmanned Aerial Vehicle (UAV) networks, namely the network connectivity loss and energy limitations. One approach to solve these issues is using learning-based routing protocols to make close-to-optimal local decisions by the network nodes, and Q-routing is a bold example of such protocols. However, the performance of the current implementations of Q-routing algorithms is not yet satisfactory, mainly due to the lack of adaptability to continued topology changes. In this paper, we propose a full-echo Q-routing algorithm with a self-adaptive learning rate that utilizes Simulated Annealing (SA) optimization to control the exploration rate of the algorithm through the temperature decline rate, which in turn is regulated by the experienced variation rate of the Q-values. Our results show that our method adapts to the network dynamicity without the need for manual re-initialization at transition points (abrupt network topology changes). Our method exhibits a reduction in the energy consumption ranging from 7% up to 82%, as well as a 2.6 fold gain in successful packet delivery rate, compared to the state of the art Q-routing protocols.**

***Index Terms*—UAV networks, learning-based routing, Q-routing, adaptive networking, energy efficiency.**

I. Introduction

FLYING Adhoc Networks (FANETs), especially those composed of Unmanned Aerial Vehicles (UAVs), are becoming increasingly popular in many sensing, monitoring, and actuation applications due to their key features such as free mobility, faster speeds, less human hazards in harsh and risky environments, autonomous operation, larger coverage areas, lower costs, and flexible imaging capabilities. The range of applications is countless and includes but not limited to transportation [1], traffic control [2], fire monitoring [3], [4], human action recognition [5], surveillance [6], border patrolling [7], search and rescue [8], disaster management [9], wireless network connectivity [10], smart agriculture, and forestry [11]. However, many communication and control protocols, which are primarily developed for ground networks with somewhat stationary infrastructures deem inefficient for UAV networks. Even the communication protocols designed for vehicular networks do not accommodate the key issues of UAV networks like their limited communication range, limited energy, limited processing power, faster speed, and structure-free mobility. In [12], distributed UAV networks are thoroughly studied with reviewing FANET structures and utilized networking protocols to highlight the current networking challenges and issues. They conclude that optimal routing and maintaining connectivity remain as two key challenging issues [13], which have been studied in several subsequent papers.

Machine Learning (ML) algorithms support efficient parameter estimation and interactive decision-making in wireless networks by learning from data and past experience. A review of using ML methods for different aspects of wireless networking is provided in [14]. Also, ML algorithms facilitate the abstraction of different networking tasks from network topology prediction and channel status estimation by enabling embedded learnability features. For instance, Reinforcement Learning (RL)-based routing involves finding the most convenient path for any source-destination pair through the network based on different optimization criteria without directly monitoring and incorporating the network topology. Typically, the nodes' local information is used to take optimal communication decisions to minimize the overall energy consumption and enhance network connectivity [15]. Reinforcement learning has also been used for other aspects of UAV networks, including spectrum management [16] and intelligent jamming defense [17].

A. Contributions

In this paper, we introduce a promising Q-learning-based routing protocol that is suitable for highly dynamic UAV networks. Low complexity, low overhead requirements, local forwarding decision, and no need for initial route setup are some of the key characteristics of the proposed method to improve the probability of successful packet delivery in UAV networks. The major contributions of this paper are summarized as follows:

- First, we introduce a trajectory creation approach, which uses a piece-wise linear mobility model to produce node trajectories. It consists of a hierarchical generative model that defines random parameters for each

Manuscript received March 9, 2021; revised April 26, 2021; accepted May 22, 2021. Date of publication June 1, 2021; date of current version September 16, 2021. This material was based upon the work supported by the National Science Foundation under Grants 1755984 and 2008784. Recommended for acceptance by Dr. Liang Zhao. *(Corresponding author: Arnau Rovira-Sugranes.)*

Arnau Rovira-Sugranes, Fatemeh Afghah, and Abolfazl Razi are with the School of Informatics, Computing and Cyber Systems, Northern Arizona University, Flagstaff, AZ 86011 USA (e-mail: ar2832@nau.edu; Fatemeh.Afghah@nau.edu; abolfazl.razi@nau.edu).

Junsuo Qu is with the School of Automation, Xi'an University of Posts and Telecommunications, Xi'an 710021, China (e-mail: qujunsuo@xupt.edu.cn).

Digital Object Identifier 10.1109/TNSE.2021.3085514

UAV class, representing each node's motion profile. This model is suitable for UAV networks with heterogeneous mobility levels (e.g., networks of quadcopters, mini-drones, and fixed-wing UAVs) since it is easy to infer the motion profile of each node by sampling its motion trajectory.

- Secondly, we propose a full-echo Q-routing with an adaptive learning rate controlled by Simulated Annealing (SA) optimization, where the *temperature* parameter captures the influence of the nodes' mobility on the update rates of Q-value. The soft variation of the exploration rate with the re-initiation feature not only optimizes the exploration rate, but also accommodates abrupt changes in the network dynamicity. The criteria we used for path selection minimizes the packet transmission energy.
- Lastly, we performed extensive simulations to assess the performance and the complexity of the proposed algorithms, compared with previous Q-routing algorithms and Q-routing with other heuristic optimization methods using different network scenarios. The quantitative results confirm a considerable reduction in energy consumption and an increase in the packet delivery rate for the proposed algorithm.

The rest of this paper is organized as follows. In Section II, we provide a comprehensive review of related work, especially the recently developed Q-Learning-based routing protocols. In Section III, we introduce the system and the utilized mobility model. In Section IV, we describe the proposed fully-echoed Q-routing algorithm equipped with Simulated Annealing inference. Section V investigates the properties of the proposed method in comparison with similar methods. In Section VI, we present the simulation results with quantitative analysis. Finally, the main findings of this work are reviewed in Section VII.

II. Related Work

RL-based routing was first introduced in [18], where Q-routing is utilized as an application of packet routing based on Q-learning. This method demonstrated superior performance, compared to a non-adaptive algorithm based on precomputed shortest paths [19]. The essence of Q-routing is gauging the impact of routing strategies on a desired performance metric by investigating different paths in the *exploration* phase and using the discovered best paths in the *exploitation* phase. Exploration imposes an overhead to the system but is critical for finding newly emerged optimal paths, especially when the network topology undergoes substantial changes. An essential challenge is to solve the trade-off between the exploration and exploitation rate constantly to accommodate the level of dynamicity of the network topology. An extension of the conventional Q-routing, known as Predictive Q-routing (PQ-routing) [20], attempted to address this issue and fine-tunes the routing policies under low network loads. Their approach was based on learning and storing new optimal policies under decreasing load conditions and reusing the best learned experiences by predicting the traffic trend.

Their idea was to re-investigate the paths that remain unused for a while due to the congestion-related delays. They considered probing frequency as an adjustable parameter that should be tuned based on the path recovery rate estimate. Results showed that PQ-routing outperformed the Q-routing in terms of both learning speed and adaptability. However, PQ-routing requires large memory for the recovery rate estimation. Also, it was not accurate in estimating the recovery rate under varying topology change rates (e.g., when nodes start moving faster or slower). Furthermore, this method only works when delays arise from the queuing congestion, and not from the network topology change.

Another modification of the conventional Q-routing is Dual Reinforcement Q-routing (DRQ-routing) [21]. Their idea was to use forward and backward explorations by the sender and receiver of each communication hop by appending information to the data packets they receive from their neighbors. Simulation results prove that this method learns the optimal policy more than twice faster than the standard Q-routing. A comparative analysis of learning-based routing algorithms is provided in [22], where the performance of the self-adaptive Q-routing and dual reinforcement Q-routing algorithms is compared against the conventional shortest path algorithms. Their results showed that the Q-Learning approach outperforms the traditional non-adaptive approaches when increasing traffic causes more frequent node and link failures. However, Q-routing does not always guarantee finding the shortest path and does not explore multiple forwarding options in parallel.

Two improved versions of Q-routing, namely Credence-based Q-routing (CrQ-routing) and Probabilistic Credence-based Q-routing (PCrQ-routing), are proposed in [23] to capture the traffic congestion dynamically and to improve the learning process to select less congested paths. CrQ-routing uses variable learning rates based on the inferred confidence to make the Q-value updates more efficient. Q-value updates are monitored to assess the freshness and accuracy of the measurements. A higher learning rate is used for the old updates, and a lower learning rate is used for the newer updates. Probabilistic Credence-based Q-routing (PCrQ-routing) takes a random selection approach to select a less congested path. This algorithm reverts back to the optimal selection policy when the utilized confidence approach does not learn the traffic load accurately and takes decision merely based on the information freshness. Both methods adapt to the current network conditions much faster than the conventional Q-routing.

Another technique to accelerate the learning speed of conventional Q-routing is the *full-echo* approach introduced in [18]. In conventional Q-routing, each node only updates the Q-values for the selected next-node (i.e., the best neighbor), whereas in the *full-echo* routing, a node gets each neighbor's estimate of the total time to the destination to update the Q-values accordingly for each of the neighbors. A more recent work added adaptive learning rates to the *full-echo* Q-routing

TABLE I
Q-ROUTING-BASED ROUTING PROTOCOLS COMPARISON: PROVIDED FEATURES AND TARGET APPLICATIONS

Routing protocol	Connectivity	Predictive	Exploration	Energy efficiency	Fastly adapts to abrupt changes	Network application
ECaD [32]	✓	✓	✓	✓	✗	FANETs
PARRoT [33]	✗	✓	✓	✗	✗	UAV-aided networks
ARdeep [34]	✓	✓	✓	✓	✗	Mobile robot networks
QMR [35]	✗	✓	✓	✓	✓	FANETs
FLRLBR [36]	✓	✓	✓	✓	✗	FANETs
QAGR [37]	✗	✓	✓	✗	✗	UAV-assisted VANETs
Proposed	✓	✓	✓	✓	✓	**FANETs**

to improve the exploration performance [24]. The adaptive *full-echo* Q-routing uses two types of learning rates, one fixed (basic) rate for the neighbor to whom the packet is sent and another one (additional) for the rest of the neighbors. The additional learning rate changes dynamically according to the estimated average delivery time and allows to explore other possible routes. Their results show that this technique reduces the oscillations of the *full-echo* Q-routing for high-load scenarios. An extension of this work, Adaptive Q-routing with Random Echo and Route Memory (AQRERM) is introduced in [25], which improves the performance of the baseline method in terms of the overshoot and settling time of the learning process, as well as the learning stability.

Recently, more advanced routing algorithms are proposed to extend the baseline Q-routing into more complex scenarios with enhanced performance. Three successful algorithms include Poisson's probability-based Q-routing (PBQ-routing) [26], Delayed Q-routing (DQ-routing) [27] and Qos-aware Q-routing (Q^2-Routing) [28]. PBQ-routing uses forwarding probability and Poisson's probability for decision making and controlling transmission energy for intermittently connected networks. The results of this work show that the delivery probability of this method is almost twice bigger than that of the standard Q-routing while reducing the overhead ratio to half. DQ-routing updates Q-values with random delays to reduce their overestimation and improve learning rate. Q^2-Routing includes a variable learning rate based on the amount of variation in Q-values while meeting the Quality of Service (QoS) requirements for the offered traffic.

Unfortunately, all of these Q-routing methods suffer from one or more weaknesses, which negatively affect their performance in extremely dynamic UAV networks. First, the methods that require large memories to store the history of the Q-values or the history of experienced delay (or other performance metrics) for each decision become prohibitively restrictive when using for memory-constrained drones. The second issue is the Q-routing protocols' incapability in adapting their learning rate to varying network dynamicity rates.

Other types of routing protocols for UAV networks have been proposed in recent years, with many surveys exploring the characteristic of each method. In [29], the authors provide a complete analysis of the major routing protocols for FANETs, with a comparison based on a set of key performance indicators. Also, a recently published paper [30] offers a comprehensive discussion of routing protocols in Software-Defined Network (SDN) and Network Function Virtualization (NFV) for UAV-assisted networks, which can lead to research directions in the future. Lastly, a simulation-based comparison of various routing protocols for FANETs is presented in [31] to identify the best methods in real-time dynamic operations.

Another class of routing protocols is position-based algorithms, which use different tracking systems to exploit and monitor other nodes' positions. These algorithms select their path based on their inferred location information, i.e., by directly incorporating nodes' mobility into the path selection mechanism to address connectivity issues, which includes greedy distance-based methods [38]. A complete review of position-based routing protocols when applied to 3D networks is presented in [39]. However, these methods require sophisticated tracking systems and large computation overhead for timely estimation of the location of all surrounding nodes [40]. Also, these methods do not fully explore most of the connected and durable paths. For this reason, some authors incorporated the movement information and the residual energy level of each UAV to guarantee communication stability and predict future link breakages. For example, the Energy-efficient Connectivity-aware Data Delivery (ECaD) routing algorithm [32] exploits new routes while predicting the link failures prior to their occurrence to achieve a balanced energy consumption among UAVs. Nevertheless, these methods do not use the power of ML methods to indirectly learn the influence of the position information on performance metrics and realize more intelligent and independent decision making for routing protocols. The above-mentioned facts led the researchers to use RL-based routing methods. Some of the RL-based routing algorithms include predictive ad-hoc routing fueled by reinforcement learning and trajectory knowledge (PARRoT) [33], adaptive and reliable routing protocol with deep reinforcement learning (ARdeep) [34], traffic-aware Q-network enhanced routing protocol based on GPSR (TQNGPSR) [41] and Q-learning based multi-objective optimization routing protocol (QMR) [35], as well as RL-based routing protocols that use fuzzy logic for decision-making [36], [37], [42]. In Table I, we present some of the most recent learning-based routing protocols, and compared their features to the proposed method. Although the proposed provide elegant solutions for routing, they do not seem to be practical with current drone technology due to their high computational complexity. Also, they perform poorly in adapting to abrupt network changes.

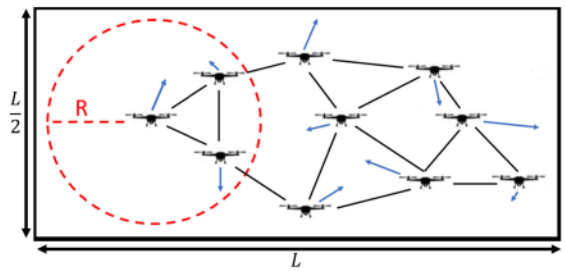

Fig. 1. Network topology with dynamic contact graph based on the nodes' communication range.

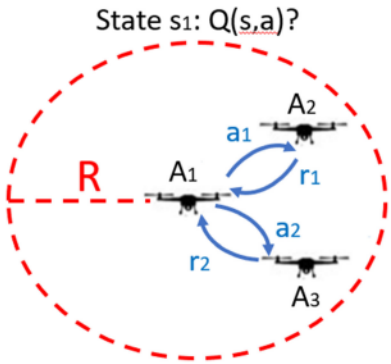

Fig. 2. Illustration of the RL-based routing.

III. System Model

We consider a wireless mesh network composed of N nodes $\mathcal{N} = \{n_1, n_2, \ldots, n_N\}$ distributed uniformly in a rectangular area of an arbitrary size $L \times L/2$, as depicted in Fig. 1. The communication range of each UAV is represented by a circular area of radius R. Therefore, the set of the neighbors for node n_i is defined as:

$$S_i(t) = \{n_j \in \mathcal{N} : d_{ij}(t) \leq R\} \quad (1)$$

where $d_{ij}(t) = \sqrt{(x_i(t) - x_j(t))^2 + (y_i(t) - y_j(t))^2}$ is the Euclidean distance between nodes n_i and n_j at time t. This realizes a dynamic contact graph where there exists a communication link between any pair of nodes within a given distance.

Here, we use Q-learning, a variant of model-free reinforcement learning that enables optimal decision making by evaluating the rewards of actions in an uncertain or unknown environment with no central supervisor [43]. Q-learning is a variant of the reinforcement learning algorithm, which provides agents A_i with the capability of directly learning the consequences of their actions a_i (which node to send the packet to) when they are at specific states s_i (e.g., location, traffic load, etc.) in terms of the achieved reward r_i. The reward is defined as the reduction of a desired performance metric, i.e., the transmission energy from the source to the destination, achieved by the action. The concept of reinforcement learning for optimized routing is shown in Fig. 2. In the initial environment represented by state s_1, node/agent A_1 has two candidate neighbors A_2 and A_3 to send its packet. The spirit of RL-based routing is selecting one of the actions a_1 or a_2 based on the reward expected for each action a at state s, defined as $Q(s, a)$. Once we establish an optimal forwarding decision policy (a_1 or a_2), the agent A_1 obtains an immediate reward from the environment, r_1 or r_2, respectively. Then, it transits to state s_2, where new decisions are made based on the new environmental conditions and the learned policy in terms of actions-rewards relations. The end goal is to find an optimal policy in which the cumulative reward over time is maximized by assigning optimal actions to each state.

The idea behind the full-echo routing equipped with SA optimization is to go one step beyond the conventional Q-routing paradigm and adapt the learning rate of the algorithm based on the rate of topology change, implied by the node velocities. The goal is to keep the overall energy consumption of the network at the minimum level possible while minimizing the packet drop rate for selecting unstable links.

Our approach is to regulate the exploration rate of Q-routing to identify potentially new optimal paths while not overusing exploration time. Some prior works intend to directly incorporate the predicted network topology into the routing algorithm [44], [45]. Despite their near-optimal performance, these methods require accurate tracking systems that can be restrictive in real scenarios. Here, we use simulating annealing optimization to adjust the exploration rate by indirectly learning the impact of node mobility on the communication energy consumption.

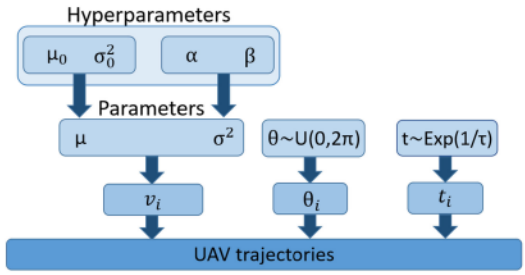

Fig. 3. A hierarchical generative model used to generate class-specific motion trajectories.

A. Piece-Wise Linear Mobility

To emulate networks with heterogeneous mobility parameters, we use a parametric generative model to produce node trajectories. The mobility model consists of piece-wise linear motions over time intervals whose duration is exponentially distributed as $t_i \sim Exp(1/\tau)$.[1] During each segment t_i, we use a constant velocity v_i and direction θ_i that vary for the next segment.

A hierarchical generative model is used to produce the parameters that ultimately define the motion trajectories (Fig. 3). Consider each node moves with a random but fixed velocity v_i in a random direction θ_i, at each interval t_i. The velocity v_i is a Random Variable (RV) with Gaussian

[1] This model can be viewed as a waypoint model with linear motions between the waypoints that has the flexibility of generating arbitrary trajectories when the interval between selected consecutive waypoints is small enough.

distribution $v_i \sim \mathcal{N}(\mu_{vi}, \sigma_{vi}^2)$. The direction θ_i follows a uniform distribution $\theta_i \sim \mathcal{U}(0, 2\pi)$. These distributions are used to simulate the worst-case scenario following prior works in [46]. More specifically, Gaussian distribution maximizes the entropy under limited energy, hence is appropriate for creating the most unpredictable node velocities. Likewise, the uniform distribution is the most uninformative distribution that maximizes entropy for RVs with limited range, like the direction θ_i, which is limited to the $[0, 2\pi]$ range. However, for the sake of completeness and to ensure that the results are generic, we examined our routing protocol using other widely adopted mobility models for UAV networks, including random waypoint [47], Gauss-Markov Mobility Model [48], and Paparazzi mobility model [49].

Another advantage of using a segment-wise mobility model with a symmetric distribution for θ, is to prevent the network from falling apart, as occurs for networks with linear motions at random directions.

The triplet (t_i, θ_i, v_i) forms the model parameters. To accommodate nodes with different velocity profiles, the mean μ_{vi} and variance σ_{vi}^2 of v_i are considered RVs controlled by hyper-parameters. In particular, we have $\mu_{vi} \sim \mathcal{N}(\mu_0, \sigma_0^2)$ and $\sigma_{vi}^2 \sim Inv-Gamma(\alpha, \beta)$, an inverse Gamma distribution with shape α and rate β. The hyper-parameters α, β, μ_0 and σ_0^2 are the same among all nodes, and are used to obtain node-specific model parameters μ_i and σ_i^2 for nodes $n_i = (1, 2, \ldots, N)$. The model parameters represent the motion profile of each node based on its class, and remain constant throughout the operation time. This hierarchical modeling with conjugate priors for the model parameters facilitates deriving closed-form posterior and predictive probabilities for the model parameters, to easily infer the motion profile of each node by sampling its motion trajectory. We use this model to generate the motion trajectories for all network nodes and to create the dynamic time-varying contact graph of the network using (1).

Another assumption that we considered is that the motion intervals are long enough to let the learning algorithm converge to an optimal solution with a reasonable learning rate. This assumption is relevant since the employed learning algorithm requires only around fifty transmission rounds for a full convergence that remains in the millisecond range while the motion changes for UAVs occur in the second range, if not minutes. However, the algorithm is flexible enough to recognize the change points and re-adapt to the new velocities with no human intervention, noting the re-initiation process at the beginning of each segment. The learning rate is determined by the temperature parameter T for the utilized SA algorithm, which quickly adapts to the network nodes' average velocities during each interval. This adaptation is realized without the need for directly inferring the node velocities using sophisticated tracking systems.

IV. Routing Protocol

The proposed routing protocol enables the nodes to make packet forwarding decisions based on their local experience with the ultimate goal of minimizing the end-to-end transmission energy. No prior information is required about the network nodes' mobility and traffic load distribution across the network.

We first review conventional Q-routing [18] and then indicate the modifications we made to develop our proposed method, namely the *fully-echoed Q-routing* with an adaptive learning rate using the inferred SA parameters. The Q-value $Q_x(d, y)$ is defined as the time-span it takes for node x to deliver a packet to the destination node d through neighbor node y. Then, after sending the packet from node x to node y, node y estimates the remaining time for the trip $t_{y\to d}$, defined by:

$$t_{y\to d} = \min_{z\in\mathcal{S}_y(t)} Q_y(d, z), \tag{2}$$

where $\mathcal{S}_y(t)$ is the set of the neighbors of y at current time t. Next, from the information that node x receives, we can update $Q_x(d, y)$ to:

$$Q_x(d, y)_{\text{new}} = Q_x(d, y)_{\text{old}} + \eta\big(q + s + t - Q_x(d, y)_{\text{old}}\big), \tag{3}$$

where q is the waiting time for node x and s is the transmission time from node x to node y. Also, η is an adjustable learning rate. To improve the learning speed, using the full echo Q-routing with adaptive learning rates, we update Q-tables for all neighbors by sending estimation packets to the neighbors. We define two learning rates: basic (η) and additional (η_2). Each node updates its Q-table using η if it refers to the neighboring node to which we sent the packet, and using η_2 otherwise. The basic learning rate (η) is fixed; however, the additional learning rate (η_2) is updated at each step using

$$\eta_2 = \frac{T_{est}}{T_{max}} \cdot \eta \cdot k \tag{4}$$

where T_{est} is the estimate of the average delivery time and T_{max} is the estimate of the maximum average delivery time. Also, k is a predefined parameter to be tuned by the experiments for optimal performance.

The exploration rate is controlled by the SA algorithm [50], as a natural optimization choice. The reason for selecting SA as the optimization algorithm compared to other heuristic optimization methods such as Gradient Descent (GD), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) is that SA's naturally embedded property of starting from more aggressive exploration rates (at higcth temperatures) and leaning gradually toward more conservative decisions over time by cooling down the temperature parameter T, makes it desirable for segment-wise routing decisions. This feature accommodates dynamic topology with abrupt changes. The temperature T changes exponentially from $T = (k_{max}/k)$ to $T = 1$, where k is the iteration and k_{max} is the maximum allowable number of iterations for exploration. Here, we control the temperature cooling based on the velocities of the network nodes captured by the changes in the selected links' performance. Once a velocity change is detected (at the beginning of an interval), the temperature automatically is increased to the highest value and cools down gradually during the interval.

Algorithm 1: Q-routing table update using SA

Initialize Q-values ($Q_x(d, y)$) for all neighbors y;
Initialize f, η;
for $k = 1$ to k_{max} **do**
  while node d has not been reached **do**
    T $\leftarrow k_{max}/k$
    T $\leftarrow T \times f$
    Select action a_r uniformly among neighbors;
    Select action a_p according to learned Q-values;
    $a \leftarrow a_p$;
    generate random variable $r \sim \mathcal{U}[0, 1]$
    if $(P(a_p, a_r, T) \geq r)$ **then**
      $a \leftarrow a_r$
    end if
    Execute action a
    Update Q-value:
    $\Rightarrow$ for selected neighbor:
    $Q_x(d, y)_{\text{new}} = Q_x(d, y)_{\text{old}} + \eta(q + s + t - Q_x(d, y)_{\text{old}})$
    $\Rightarrow$ for the rest of the neighbors:
    $Q_x(d, y)_{\text{new}} = Q_x(d, y)_{\text{old}} + \eta_2(q + s + t - Q_x(d, y)_{\text{old}})$
    (where $\eta_2 = \frac{T_{est}}{T_{max}} \cdot \eta \cdot k$)
  end while
  Evaluate f using (6)
end for

The summary of the operation of one full cycle of the SA optimization is provided in Algorithm 1, where we have:

$$\begin{cases} P(a_p, a_r, T) = 1, & \text{if } a_r < a_p, \\ P(a_p, a_r, T) = e^{\frac{-(a_r - a_p)}{T}}, & \text{otherwise.} \end{cases} \tag{5}$$

Here, $P(a_p, a_r, T)$ acts as the exploration probability. $P = 1$ means that the random action a_r is better than the previously identified best action a_p, and we select a_r. Otherwise, we select the next node based on this probability by taking a random action a_r with probability $P(a_p, a_r, T)$ and following the best action with probability $1 - P(a_p, a_r, T)$. Here, a random action a_r means the next node is selected uniformly among the available neighbor nodes.

The dynamicity of the network is indirectly inferred by the change of the consumed energy (or equivalently the variation of Q-values) during the last H iterations. More specifically, we define parameter f as:

$$f = \gamma|\Delta E| = \frac{1}{H}\sum_{i=1}^{H}|E_{k+1-i} - E_{k-i}|, \tag{6}$$

where H is the length of history, to be selected based on the velocity of the nodes and the length of each interval to identify significant variations. Here, we choose $H = 10$. Parameter γ is a scaling parameter to map the energy variation into the [0.5,10] range. The parameter f is used to regulate the temperature cooling in the SA algorithm by scaling the temperature parameter T depending on how fast the chosen path's energy changes over time.

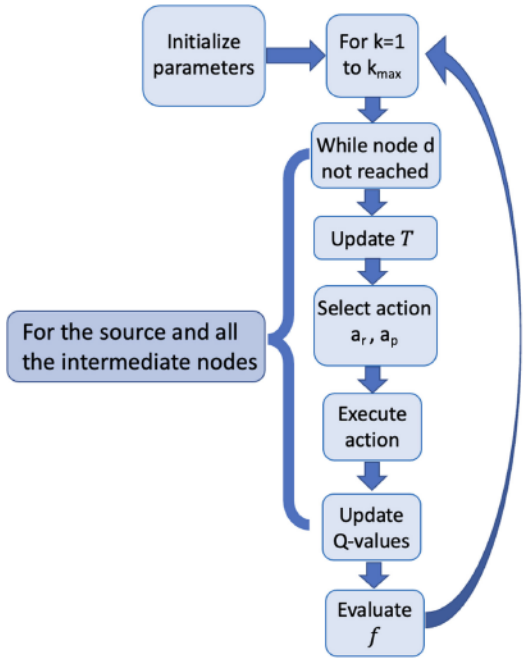

Fig. 4. Flowchart of the proposed Q-routing-based algorithm.

Lastly, we describe a full cycle of the proposed routing protocol in a visual manner in Fig. 4. We represent a source node s that wants to send k_{max} packets to the destination node d. After each packet is sent, we evaluate the factor f to subsequently update the temperature parameter T, which impacts the exploration rate.

V. Analysis of the Proposed Routing Protocol

In this section, we study some of the properties of the proposed routing protocol, and compare it to related methods previously offered for UAV networks.

A. Loop-Free Property

As described in Section IV, a direct implication of the operation of the proposed algorithm is its loop-free property, meaning that no intermediate node can be included in the end-to-end path more than once (as presented in Fig. 5). A loop-free property is crucial for dynamic networks to prevent data packets from being continually routed through the same nodes over and over. A routing loop can cause a packet never to reach its intended destination, which can substantially disrupt the operation of the network. This property is achieved based on the fact that each node maintains an updated Q-table with information regarding the visited nodes. For example, in Fig. 5, when node 4 has to choose the next node between its neighbors, it does not consider node 2 as a possible node. This happens because node 2 has been previously visited in the selected path. This way, we ensure that a loop-free path is always selected.

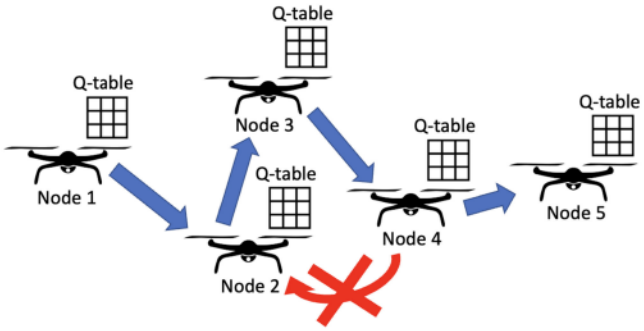

Fig. 5. Loop-free property of the proposed Q-learning-based routing protocol.

TABLE II
COMPLEXITY FOR DIFFERENT ROUTING PROTOCOLS

Routing protocol	Computation complexity
AODV [51]	$O(2N)$
OLSR [52]	$O(N)$
DSDV [53]	$O(N)$
DSR [54]	$O(2N)$
GPSR [55]	$O(N^3)$
Q-routing [18]	$O(N^2)$
Proposed	$O(N^2)$

B. Computation Complexity

Here, we include a comparative analysis of the computation complexity of the proposed method, compared to other state-of-art routing algorithms, including ad hoc on-demand distance vector (AODV) [51], optimized link state routing (OLSR) [52], destination sequenced distance vector (DSDV) [53], dynamic source routing (DSR) [54], greedy perimeter stateless routing (GPSR) [55], as well as the conventional Q-routing [18]. Computation complexity is defined as the number of operations required to execute one round of an algorithm. Results are shown in Table II, where N represents the number of nodes or network size. We can see that the proactive routing protocols (e.g., OLSR, DSDV) and reactive routing protocols (e.g., AODV, DSR) have low and average complexity, respectively. Position-based routing protocols (e.g., GPSR) and learning-based routing protocol (e.g., Q-routing and the proposed method) have a higher complexity. Our complexity is quadratic in N, which is higher than proactive and reactive methods, but still affordable for reasonable network sizes. It has been shown that learning-based decentralized methods that adapt to dynamic networks without the need for global knowledge and long route setup are more suitable for UAV networks since they eliminate the need for costly and sophisticated positioning methods. Our proposed method constantly adapts to both minor or abrupt changes, leading to a higher packet delivery ratio and energy efficiency, as well as retaining maximal connectivity. For this reason, the higher complexity, compared to proactive and reactive routing protocols, is justified by the increase in routing performance. Lastly, we can state that our method has lower complexity with respect to popular topology-aware routing protocols while providing better results.

C. Memory Requirements

In this section, we study the memory requirements and the storage efficiency for the proposed routing protocol. The proposed routing protocol needs Q-table resources for each node, with a short history to consider the adaptability to the network state at each step. In Section IV, we defined the exploration-exploitation approach that considers the history of the last 10 packets to reevaluate the exploration rate. Consequently, if we study computational complexity or memory requirements for Q-routing-based routing protocols, they suffer from the curse of dimensionality. The Q-table grows at $O(N^2)$ with the number of nodes.

However, compared to standard RL or deep-learning-based algorithms such as [33], [34], [56], [57], the memory requirements for our method are relatively low. RL algorithms require large memory for relatively large state spaces and close-to-one reward discount rate. Deep learning algorithms are also computationally extensive and require large memory to store training samples and the history of reward-action pairs. Therefore, their learning phase may take much longer than for the proposed methods. Consequently, it is fair to state that the computational complexity of our Fully-echoed Q-routing with SA inference method is doable with on-board memory capabilities in UAVs, in contrast to other more complex and memory-needy solutions that provide similar outcomes.

D. Overhead Analysis

Overhead is defined as the number of additional routing packets sent for route discovery, establishment, and maintenance. An advantage of our method is that we do not use exploration packets (like sending periodic *hello* packets in proactive routing protocols such as OLSR algorithm [52]) to find optimal paths; rather, it is learned by monitoring *data* packets in the exploration phase. Our method addresses the essential trade-off between exploration and exploitation based on the network's behavior. During the exploration phase, we learn the network's state by studying all neighbors' behavior, and these suboptimal transmissions can be considered exploration overhead. In the exploitation phase, the overhead is considerably low since only the best identified paths are utilized. Since our method addresses this trade-off using the SA optimization module based on the network's dynamicity level, the incurred overhead is much lower than other routing protocols.

We analyze how overhead impacts our routing protocol by investigating if the incurred overhead correlates with a better learning state or not, and what is the rate of taking non-optimal decisions. We expect to find a trade-off between the exploration rate (that brings more overhead) and the knowledge of the network. In Table III, we study the effect of exploration rate with reaching the optimal solution by counting the number of packets we send to reach the optimal solution. We can observe that our proposed adaptive method gives the best result in terms of optimality of the path selected and the number of packets needed to reach that solution. If we fix the exploration

TABLE III
EFFECT OF EXPLORATION RATE IN FINDING OPTIMAL SOLUTION

Exploration rate	Optimal selection	Packets until converged
Low	No	2
Medium	No	14
High	Yes	41
Proposed (adaptive)	**Yes**	**4**

rate to realize a low or medium exploration rate, the algorithm converges quickly to the final path, with 2 and 14 packets, respectively. However, the algorithm may not converge to an optimal solution since it may select some of the intermediate nodes inaccurately. On the other hand, if a fixed high exploration rate is selected, the optimal solution can be found faster, but it takes an average of 41 packets to converge. The proposed routing protocol finds the best path faster than fixed exploration rates (with an average of 4 packets) and quickly converges to the best solution. This means that the proposed algorithm with an adaptive exploration policy outperforms the fixed-rate algorithm both in terms of fast convergence and the optimality of the solution.

Concluding, we observe how the adaptive protocol has a lower overhead than high exploration methods, as it needs fewer packets to learn the state of the network.

VI. SIMULATION RESULTS

To assess the performance of the proposed method, we compare it against the state-of-the-art Q-routing algorithms discussed in Section I, including (i) Random Exploration-Exploitation Routing (REE-Routing), (ii) Probabilistic Exploration Routing (PE-Routing), (iii) Conventional Q-routing [18], (iv) Adaptive learning rates Full-Echo Q-routing (AFEQ-routing) [24], and (v) Simulated Annealing based Q-routing (SAHQ-routing) [58]. Methods (i) and (ii) are simulated for the sake of comparison only. We compare only against other Q-routing-based routing protocols, as previous works have shown that the learning-based algorithms outperform AODV, OLSR, and GPSR, among other well-known routing protocols [33], [59]. We simulate different network scenarios using the piece-wise linear mobility (Fig. 3). This model uses the entropy-maximizing Gaussian distribution for the seed and the most uninformative uniform distribution for the direction to simulate the worst-case scenario, although other mobility models for UAV networks could be used. To realize a fair comparison, we use the same set of trajectories to test different algorithms.

The first set of comparative results is presented in Table IV, in terms of the end-to-end transmission energy. We simulate networks with different sizes ($N = 10,\ N = 20$) and three velocity profiles of slow-speed ($\mu_0 = 10, \sigma_0^2 = 2.5$), medium-speed ($\mu_0 = 25, \sigma_0^2 = 5$), and fast-speed ($\mu_0 = 50, \sigma_0^2 = 10$) with $\alpha = 5, \beta = 1$ for all scenarios. The communication range is $R = 7500$ meters. The proposed algorithm considerably improves upon the performance of all algorithms consistently by reducing the average energy consumption. The achieved

TABLE IV
COMPARATIVE ANALYSIS: ENERGY CONSUMPTION OF DIFFERENT ROUTING ALGORITHMS INCLUDING THE PROPOSED METHOD UNDER DIFFERENT NETWORK SIZES AND VELOCITY PROFILES

	N = 10			N = 20		
	Slow	Medium	Fast	Slow	Medium	Fast
REER	95.3	103.7	127.1	135.1	170.3	163.9
PER	146.3	146.5	149.5	269.0	289.0	246.6
QR	83.7	93.0	97.1	86.0	104.7	104.5
AFEQR	75.8	79.3	96.4	76.9	90.4	91.3
SAHQR	70.6	85.4	98.0	70.6	98.2	99.2
Proposed	65.6	70.6	87.0	47.8	76.0	82.5
Gain	**7%-55%**	**11%-52%**	**10%-42%**	**32%-82%**	**16%-74%**	**10%-67%**

TABLE V
PACKET DELIVERY RATE VERSUS COMMUNICATION RANGE FOR DIFFERENT ALGORITHMS

	R = 5000		R = 7500		R = 10000	
	N=10	N=20	N=10	N=20	N=10	N=20
REER	24.5%	50.9%	82.0%	83.3%	94.2%	96.6%
PER	18.6%	41.1%	70.1%	73.2%	86.5%	91.8%
QR	8.9%	38.9%	77.2%	85.4%	99.9%	99.9%
AFEQR	32.4%	70.1%	90.3%	95.9%	99.9%	99.9%
SAHQR	28.9%	61.4%	90.3%	96%	99.1%	99.9%
Proposed	32.4%	70.6%	90.3%	96.3%	100%	99.9%
Gain (up to)	**264%**	**81%**	**29%**	**32%**	**16%**	**9%**

gain ranges from 7% to 82% depending on the reference method and the utilized network parameters. We observe that our method offers higher gains for larger networks ($N = 20$), and slower speeds ($\mu_0 = 10, \sigma_0^2 = 2.5$).

The communication range R plays an essential role in the performance of routing algorithms. It not only influences the sparsity of the graph by affecting the node degrees but also affects the connectivity of the network under the utilized routing algorithm. More specifically, a packet drop occurs when a node has no neighbors within the communication range or the selected nodes go beyond the communication range while transmitting. The impact of the communication range on the packet drop rate is presented in Table V for $R = 5\ km$, $R = 7.5\ km$, and $R = 10\ km$. It can be seen that our method has a higher successful packet delivery rate compared to the baseline conventional Q-routing (QR), and performs better or almost equal to the rest of the methods. The gain in the packet delivery ratio can go as high as 264% depending on the network size (N) and the communication range (R). For the lower communication range ($R = 5000\ m$), and smaller networks ($N = 10$), the achieved gain (the reduction in packet drop rate) is higher.

Next, we study the effect of the SA-based optimization on the evolution of the Q-values that represent the expected energy for a packet from the source node n_1 to the end destination n_2 through any of its seven neighbors ($n_3, n_4, \ldots, n_9$). The Q-value evolution of the seven neighbors is depicted in Fig. 6 for three algorithms, including (i) SAHQ-routing with no adaptability of T parameter, (ii) a Q-routing with high exploration rate, and (iii) the proposed method. Each line represents the evolution of the Q-values when each of the

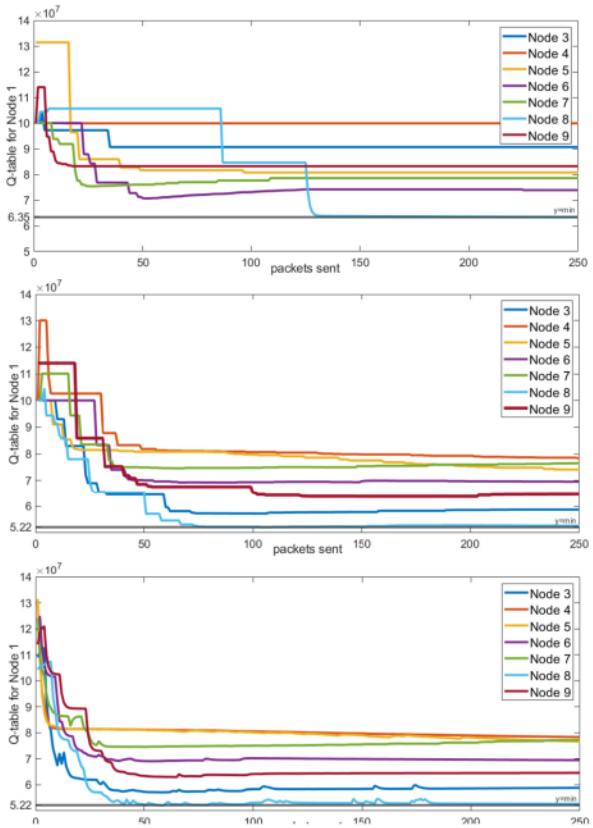

Fig. 6. Q-table for an exemplary source node n_1 to send a packet to destination n_2 through any of its seven neighbors $(n_3, n_4, \ldots, n_9)$ under different routing protocols including (top) SAHQ-routing with non-adaptive parameter T [58], (middle) high-exploration Q-routing, and (bottom) the proposed method with flexible exploration rate.

neighbors is selected as the next bode. It is shown how Q-values converge to the optimal solution over time when more packets are sent. It can be seen that the SAHQ-routing does identify the best next node (8) (after 125 rounds) but at a much lower rate compared to our method (after 40 rounds). Also, its recovered end-to-end path does not seem to be optimal (despite finding the best second node) since the minimum value in the Q-table is 63.5 Mega Joule (MJ) for SAHQ-routing, compared to the 52.2 MJ for our method. A similar gain is achieved for our method, compared to the Q-routing with high exploration. Nevertheless, our method converges to the optimal solution faster than the Q-routing with high exploration (40 rounds in contrast to 70 rounds, respectively). In short, our solution converges to the optimal value at a faster rate than the other two competitor methods. This gain comes from the capability of our method in adapting to the dynamicity of the network.

Fig. 7(a) (top row) illustrates the evolution of the temperature parameter T over time. As described previously, T is directly proportional to the exploration rate, and its natural behavior is exponential declining (i.e., $f(x) = e^{-x}$) iteration by iteration over time, as shown in the top-right most figure for the baseline method with non-adaptive T. However, our method adjusts the temperature cooling rate by examining the changes in the experienced consecutive Q-values. We observe that for a low-speed network, there are fewer exploration rate adjustment epochs compared to the fast-moving network. Likewise, Fig. 7(b) presents the packet transmission energy for different networking scenarios in each curve. Abrupt changes in the energy consumption are corresponding to selecting a different optimal path by the algorithm. We can see that for a network with slow-moving nodes, we experience fewer sharp transitions, compared to the network with faster nodes, as expected. This shows the reasonable operation of the proposed method. It is noteworthy that we observe fewer fluctuations for the baseline method with non-adaptive T, compared to the proposed method for high-speed networks. This is not necessarily a desirable behavior since it can lead to selecting non-optimal paths by missing the newly emerged optimal links when the network topology changes drastically. Therefore, extremely fast switching to choose the optimal path may go against the reliability and stability of the communication. In Fig. 8, we analyze the performance of the communication system to show that stability and reliability are not a concern. We use a slow network setup to see the effect of changing paths at the lowest dynamic rate. As shown, we observe that the proposed method with adaptive exploration rate selects the optimal path. In contrast, the routing protocol which uses SA with non-adaptive temperature misses the optimal links. Also, if we study the impact of switching paths on the stability and reliability of the network, we observe that in this case, the *non-adaptive* approach has more variability in choosing paths than the proposed method. Therefore, for our design, changing paths will depend on the variability of the network to not miss optimal paths when the network topology changes, and not compromising the stability of the communication.

In Fig. 9, it is seen that the average temperature (T) value for the SA algorithm increases for more dynamic networks with faster mobile nodes. It implies that we explore more often for faster networks to find newly emerged optimal paths since T is proportional to the exploration rate. We observe that the average temperature varies around 8.6% from low speed to medium speed networks and around 10.6% from medium to high-speed networks. Without the proposed method of controlling T based on the measured Q-value change rates, we have a T parameter that always declines over time and hence misses the opportunity of discovering new paths when the network topology undergoes abrupt changes.

Finally, we compare the performance of the proposed inferred SA method against other heuristic algorithms, such as GD, GA, and PSO, under different average node speeds. The results are presented in Table VI. All heuristic algorithms offer a similar outcome in terms of the average transmission energy. However, the proposed modified SA method, where the exploration rate is controlled by an adaptive temperature cooling process, achieves a much higher energy efficiency. Particularly, the results show that the proposed method performs around 20% better than the conventional optimization algorithms in terms of energy consumption. The improvement is slightly higher for bigger networks.

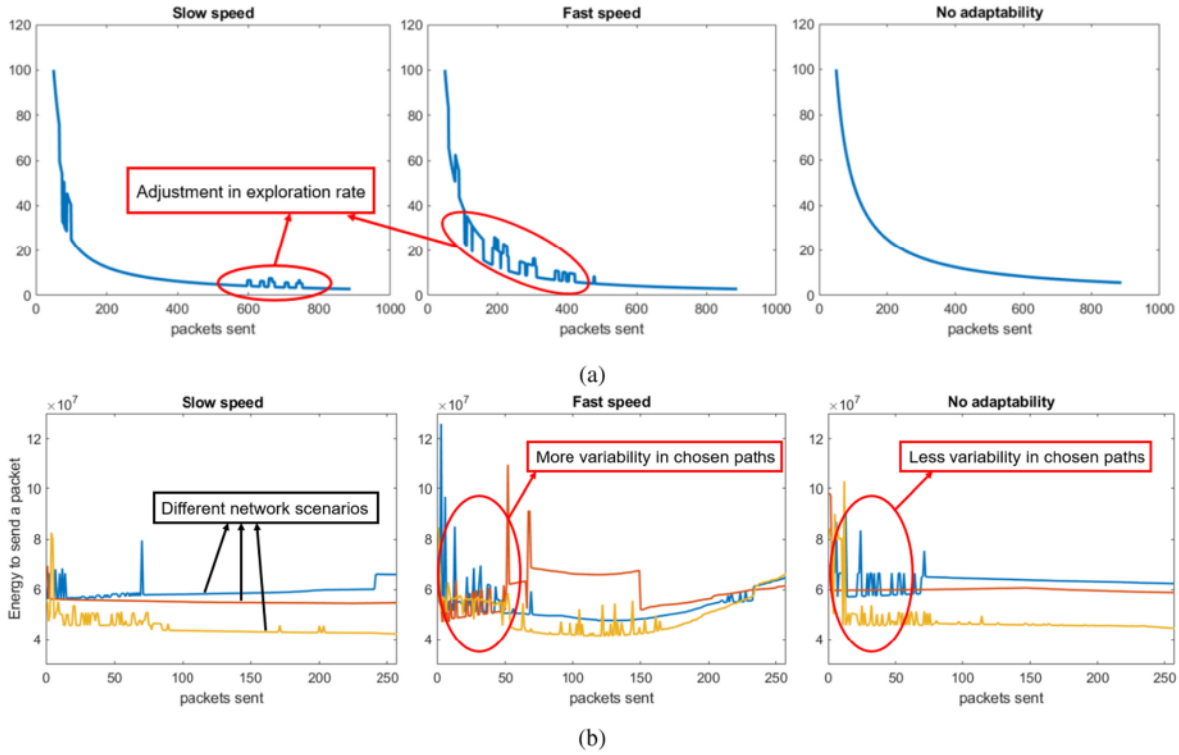

Fig. 7. a) The evolution of the temperature parameter T for different network scenarios with (left) slow-speed, (center) fast-speed, and (right) protocol with no temperature adaptability [58]; b) End-to-end energy consumption for different network scenarios.

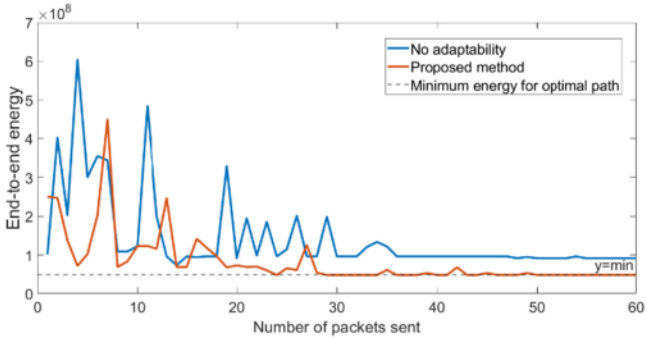

Fig. 8. End-to-end energy of the proposed method vs non-adaptive temperature method to show the stability and reliability of the proposed method.

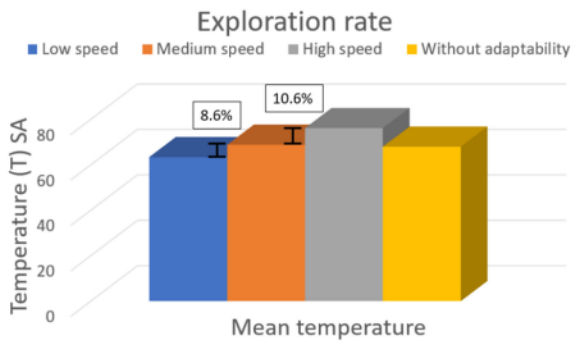

Fig. 9. The average of the temperature parameter T over time for the proposed method, as well as the baseline method with non-adaptive temperature [58] for networks with different average speeds.

TABLE VI
COMPARATIVE ANALYSIS FOR DIFFERENT HEURISTIC ALGORITHMS IN TERMS OF THE AVERAGE TRANSMISSION ENERGY

	N = 10			N = 20		
	Slow	Medium	Fast	Slow	Medium	Fast
PSO / GA / GD	83.28	79.67	79.70	76.98	78.95	72.64
Proposed (adaptive SA)	67.26	62.18	64.26	58.72	57.44	55.64
Gain	**19.23%**	**21.95%**	**19.37%**	**23.72%**	**27.25%**	**23.40%**

VII. CONCLUSION

In this work, we introduced a novel fully-echoed Q-routing protocol with adaptive learning rates optimized by the Simulated Annealing algorithm based on the inferred level of network dynamicity. Our method improves upon different implementations of Q-routing in terms of the convergence rate, the optimality of the end solution, and the adaptability to the network dynamicity. This gain is achieved by controlling the exploration rate of the Q-routing by regulating the temperature cooling rate of the utilized SA optimization based on the variation rate of the Q-values. Simulation results suggest that our algorithm achieves a reduction in the energy consumption between 7% to 82% and an increase of up to 264% as for a successful packet delivery ratio, compared to other Q-routing algorithms. The choice of SA is essential since the proposed SA-based method with adaptive temperature cooling process outperforms the same routing algorithm with other heuristic optimization methods, including GA, GD, and PSO. The proposed algorithm can solve the two key issues of UAV

networks, namely the limited energy consumption and the network connectivity loss. As a future extension of this work, we are working toward developing an aggressive reinforcement learning predictor, as a low complexity and scalable method capable of predicting non-linear changes in the link performance metrics under extreme dynamicity. Also, developing a more formal way of predicting per-node mobility and incorporating it into the optimization framework can be pursued as another future direction. A potential approach would be inferring the velocity of the nodes based on how fast the metrics change using a Bayesian inference method and using it to update Q-values in online Q-routing protocols.

REFERENCES

[1] E. N. Barmpounakis, E. I. Vlahogianni, and J. C. Golias, "Unmanned aerial aircraft systems for transportation engineering: Current practice and future challenges," *Int. J. Transp. Sci. Technol.*, vol. 5, no. 3, pp. 111–122, 2016

[2] K. Kanistras, G. Martins, M. J. Rutherford, and K. P. Valavanis, "A survey of unmanned aerial vehicles (UAVs) for traffic monitoring," in *Proc. Int. Conf. Unmanned Aircr. Syst.*, May 2013, pp. 221–234.

[3] Q. Huang, A. Razi, F. Afghah, and P. Fule, "Wildfire spread modeling with aerial image processing," in *Proc. IEEE 21st Int. Symp. World Wireless, Mobile Multimedia Netw.*, 2020, pp. 335–340.

[4] A. Shamsoshoara, F. Afghah, A. Razi, L. Zheng, P. Z. Fulé, and E. Blasch, "Aerial imagery pile burn detection using deep learning: The flame dataset," 2020, *arXiv:2012.14036*.

[5] H. Peng and A. Razi, "Fully autonomous UAV-based action recognition system using aerial imagery," in *Proc. Int. Symp. Vis. Comput.*, 2020, pp. 276–290.

[6] Z. Zaheer, A. Usmani, E. Khan, and M. A. Qadeer, "Aerial surveillance system using UAV," in *Proc. 13th Int. Conf. Wireless Opt. Commun. Netw.*, Jul. 2016, pp. 1–7.

[7] D. Bein, W. Bein, A. Karki, and B. B. Madan, "Optimizing border patrol operations using unmanned aerial vehicles," in *Proc. 12th Int. Conf. Inf. Technol. - New Generations*, Apr. 2015, pp. 479–484.

[8] S. Waharte and N. Trigoni, "Supporting search and rescue operations with UAVs," in *Proc. Int. Conf. Emerg. Secur. Technol.*, Sep. 2010, pp. 142–147.

[9] M. Erdelj, E. Natalizio, K. R. Chowdhury, and I. F. Akyildiz, "Help from the sky: Leveraging UAVs for disaster management," *IEEE Pervasive Comput.*, vol. 16, no. 1, pp. 24–32, Jan.–Mar. 2017.

[10] M. Messous, S. Senouci, and H. Sedjelmaci, "Network connectivity and area coverage for UAV fleet mobility model with energy constraint," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Apr. 2016, pp. 1–6.

[11] M. R. Brust and B. M. Strimbu, "A networked swarm model for UAV deployment in the assessment of forest environments," in *Proc. IEEE 10th Int. Conf. Intell. Sensors, Sensor Netw. Inf. Process.*, Apr. 2015, pp. 1–6.

[12] J. Wang, C. Jiang, Z. Han, Y. Ren, R. G. Maunder, and L. Hanzo, "Taking drones to the next level: Cooperative distributed unmanned-aerial-vehicular networks for small and mini drones," *IEEE Veh. Technol. Mag.*, vol. 12, no. 3, pp. 73–82, Sep. 2017.

[13] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Commun. Surv. Tut.*, vol. 18, no. 2, pp. 1123–1152, Apr.–Jun. 2016.

[14] J. Wang, C. Jiang, H. Zhang, Y. Ren, K. C. Chen, and L. Hanzo, "Thirty years of machine learning: The road to pareto-optimal wireless networks," *IEEE Commun. Surv. Tut.*, vol. 22, no. 3, pp. 1472–1514, Jul.-Sep. 2020.

[15] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *J. Artif. Int. Res.*, vol. 4, no. 1, pp. 237–285, May 1996. [Online]. Available: https://arxiv.org/abs/cs/9605103

[16] A. Shamsoshoara, F. Afghah, A. Razi, S. Mousavi, J. Ashdown, and K. Turk, "An autonomous spectrum management scheme for unmanned aerial vehicle networks in disaster relief operations," *IEEE Access*, vol. 8, pp. 58064–58079, 2020.

[17] N. I. Mowla, N. H. Tran, I. Doh, and K. Chae, "AFRL: Adaptive federated reinforcement learning for intelligent jamming defense in FANET," *J. Commun. Netw.*, vol. 22, no. 3, pp. 244–258, 2020.

[18] J. A. Boyan and M. L. Littman, "Packet routing in dynamically changing networks: A reinforcement learning approach," in *Proc. 6th Int. Conf. Neural Inf. Process. Syst.* San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993, pp. 671–678.

[19] S. Khodayari and M. J. Yazdanpanah, "Network routing based on reinforcement learning in dynamically changing networks," in *Proc. 17th IEEE Int. Conf. Tools Artif. Intell.*, 2005, pp. 362–366, doi: 10.1109/ICTAI.2005.91.

[20] S. P. M. Choi and D.-Y. Yeung, "Predictive q-routing: A memory-based reinforcement learning approach to adaptive traffic control," in *Proc. 8th Int. Conf. Neural Info. Process. Syst.*, Cambridge, MA, USA: MIT Press, 1995, pp. 945–951.

[21] S. Kumar and R. Mukkulainen, "Confidence based dual reinforcement Q-routing: An adaptive online network routing algorithm," in *Proc. 16th Int. Joint Conf. Artif. Intell.*, vol. 2, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999, pp. 758–763.

[22] F. Tekiner, Z. Ghassemlooy, and T.R. Srikanth, "Comparison of the Q-routing and shortest path routing algorithm," *School of Eng. & Technol.*, Newcastle upon Tyne: Northumbria University, 2004.

[23] N. Gupta *et al.*, "Improved route selection approaches using Q-learning framework for 2D NoCs," in *Proc. 3rd Int. Workshop Many-core Embedded Syst.*, 2015, pp. 33–40.

[24] Y. Shilova, M. Kavalerov, and I. Bezukladnikov, "Full echo Q-routing with adaptive learning rates: A reinforcement learning approach to network routing," in *Proc. IEEE NW Russia Young Researchers Elect. Electron. Eng. Conf.*, Feb 2016, pp. 341–344.

[25] M. Kavalerov, Y. Shilova, and Y. Likhacheva, "Adaptive Q-routing with random echo and route memory," in *Proc. 20th Conf. Open Innovations Assoc. FRUCT*. Helsinki, Finland, Finland: FRUCT Oy, 2017, pp. 20:138–20: 145.

[26] D. Sharma, D. Kukreja, P. Aggarwal, M. Kaur, and A. Sachan, "Poisson's probability-based Q-routing techniques for message forwarding in opportunistic networks," *Int. J. Commun. Syst.*, vol. 31, no. 11, p. e3593, 2018.

[27] F. Wang, R. Feng, and H. Chen, "Dynamic routing algorithm with Q-learning for internet of things with delayed estimator," *IOP Conf. Series: Earth Environ. Sci.*, vol. 234, 03 2019, Art. no. 012048.

[28] T. Hendriks, M. Camelo, and S. Latré, "Q2-routing : A QoS-aware Q-routing algorithm for wireless ad hoc networks," in *Proc. 14th Int. Conf. Wireless Mobile Comput., Netw. Commun.*, Oct. 2018, pp. 108–115.

[29] O. Oubbati, M. Atiquzzaman, P. Lorenz, H. Tareque, and M. S. Hossain, "Routing in flying ad hoc networks: Survey, constraints, and future challenge perspectives," *IEEE Access*, vol. 7, pp. 81057–81105 2019.

[30] O. Oubbati, M. Atiquzzaman, T. Ahanger, and A. Ibrahim, "Softwarization of UAV networks: A survey of applications and future trends," *IEEE Access*, vol. 8, pp. 98073–98125, 2020.

[31] A. Nayyar, "Flying adhoc network (FANETs): Simulation based performance comparison of routing protocols: AODV, DSDV, DSR, OLSR, AOMDV and HWMP," in *Proc. 2018 Int. Conf. Adv. Big Data, Comput. Data Commun. Syst.*, 2018, pp. 1–9.

[32] O. Oubbati, M. Mozaffari, N. Chaib, P. Lorenz, M. Atiquzzaman, and A. Jamalipour, "ECaD: Energy-efficient routing in flying ad hoc networks," *Int. J. Commun. Syst.*, vol. 32, no. 18, p. e4156, 2019.

[33] B. Sliwa, C. Schüler, M. Patchou, and C. Wietfeld, "PARRoT: Predictive ad-hoc routing fueled by reinforcement learning and trajectory knowledge," 2020, *arXiv:2012.05490*.

[34] J. Liu, Q. Wang, C. He, and Y. Xu, "ARdeep: Adaptive and reliable routing protocol for mobile robotic networks with deep reinforcement learning," in *Proc. IEEE 45th Conf. Local Comput. Netw.*, 2020, pp. 465–468.

[35] J. Liu *et al.*, "QMR: Q-learning based multi-objective optimization routing protocol for flying ad hoc networks," *Comput. Commun.*, vol. 150, pp. 304–316, 2020.

[36] C. He, S. Liu, and S. Han, "A fuzzy logic reinforcement learning-based routing algorithm for flying ad hoc networks," in *Proc. Int. Conf. Comput., Netw. Commun.*, 2020, pp. 987–991.

[37] S. Jiang, Z. Huang, and Y. Ji, "Adaptive UAV-assisted geographic routing with Q-learning in VANET," *IEEE Commun. Lett.*, vol. 25, no. 4, pp. 1358–1362, Apr. 2021.

[38] M. Khaledi, A. Rovira-Sugranes, F. Afghah, and A. Razi, "On greedy routing in dynamic UAV networks," in *Proc. IEEE Int. Conf. Sens., Commun. Netw. (SECON Workshops)*, 2018, pp. 1–5.

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

[40] A. Razi, "Optimal measurement policy for linear measurement systems with applications to UAV network topology prediction," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 1970–1981, Feb. 2020.

[41] Y. Chen *et al.* "A traffic-aware q-network enhanced routing protocol based on GPSR for unmanned aerial vehicle ad-hoc networks," *Front. Informat. Technol. Electron. Eng.*, vol. 21, no. 9, pp. 1308–1320, 2020.

[42] Q. Yang, S. Jang, and S. Yoo, "Q-learning-based fuzzy logic for multi-objective routing algorithm in flying ad hoc networks," *Wireless Pers. Commun.*, vol. 113, pp. 115–138, 2020.

[43] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," *IEEE Trans. Neural Netw.*, vol. 9, no. 5, pp. 1054–1054, Sep. 1998.
[44] A. Rovira-Sugranes and A. Razi, "Predictive routing for dynamic UAV networks," in *Proc. IEEE Int. Conf. Wireless Space Extreme Environments*, 2017, pp. 43–47.
[45] A. Razi *et al.*, "Predictive routing for wireless networks: Robotics-based test and evaluation platform," in *Proc. IEEE 8th Annu. Comput. Commun. workshop Conf.*, 2018, pp. 993–999.
[46] B. Keng, "Maximum entropy distributions," Jan 2017. [Online]. Available: https://bjlkeng.github.io/posts/maximum-entropy-distributions/
[47] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Commun. Mobile Comput.*, vol. 2, no. 5, pp. 483–502, 2002.
[48] K. Kumari and S. Maakar, "A survey: Different mobility model for FANET," *Int. Journal Adv. Res. Computer Sci. Softw. Eng.*, vol. 5, no. 6, 2015.
[49] O. Bouachir, A. Abrassart, F. Garcia, and N. Larrieu, "A mobility model for UAV ad hoc network," in *Proc. Int. Conf. Unmanned Aircr. Syst.*, May 2014, pp. 383–388.
[50] M. Guo, Y. Liu, and J. Malec, "A new Q-learning algorithm based on the metropolis criterion," *IEEE Trans. Syst., Man, Cybern., Part B*, vol. 34, no. 5, pp. 2140–2143, Oct. 2004.
[51] S. Murthy and J. J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks," *Mobile Netw. Appl.*, vol. 1, no. 2, pp. 183–197, Jun. 1996.
[52] T. H. Clausen and P. Jacquet, "Optimized link state routing protocol (OLSRP)," *The Internet Engineering Task Force, MANET working Group*, vol. 3626, no. 10, 2003.
[53] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *Proc. Conf. Commun. Architectures, Protoc. Appl.*, 1994, pp. 234–244.
[54] D. B. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Comput.*, pp. 153–181, 1996.
[55] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. 6th Annu. Int. Conf. Mobile Comput. Netw.*, 2000, pp. 243–254.
[56] W. Jung, J. Yim, and Y. Ko, "QGeo: Q-learning-based geographic ad hoc routing protocol for unmanned robotic networks," *IEEE Commun. Lett.*, vol. 21, no. 10, pp. 2258–2261, Oct. 2017.
[57] N. Lyu, G. Song, B. Yang, and Y. Cheng, "Qngpsr: A Q-network enhanced geographic ad-hoc routing protocol based on GPSR," in *Proc. IEEE 88th Veh. Technol. Conf.*, 2018, pp. 1–6.
[58] A. M. Lopez and D. R. Heisterkamp, "Simulated annealing based hierarchical Q-routing: A dynamic routing protocol," in *Proc. 8th ITNG Int. Conf. Inf. Technol.: New Generations*, Apr. 2011, pp. 791–796.
[59] 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.

Arnau Rovira-Sugranes received the Graduation degree in industrial electronics and automation engineering from Rovira i Virgili University, Tarragona, Spain, in 2016, coursing his senior year Electrical Engineering from Northern Arizona University, Flagstaff, AZ, USA, as an Exchange Student. He is currently working toward the Ph.D. degree in informatics and computing with the School of Informatics, Computing and Cyber Systems, Northern Arizona University. After graduating, he was with an automotive seating and electrical systems company before starting his Ph.D. program. His research interests include machine learning and data mining, communication and routing protocols and graph theory for flying ad-hoc wireless networks (FANETs), which produced peer-reviewed publications on predictive based solutions for Internet of Things. He has also served as an IEEE conference/journal paper reviewer and symposium moderator.

Fatemeh Afghah is currently an Associate Professor with the School of Informatics, Computing, and Cyber Systems, Northern Arizona University (NAU), Flagstaff, AZ, USA, where she is the Director of Wireless Networking and Information Processing Laboratory. Prior to joining NAU, she was an Assistant Professor with the Electrical and Computer Engineering Department, North Carolina A & T State University, Greensboro, NC, USA, from 2013 to 2015. She is the author or coauthor of more than 80 peer-reviewed publications. Her research interests include wireless communication networks, decision making in multi-agent systems, radio spectrum management, and artificial intelligence in healthcare. Her research has been continually supported by NSF, AFRL, AFOSR, and ABOR. She was the recipient of several awards, including the Air Force Office of Scientific Research Young Investigator Award in 2019, NSF CAREER Award in 2020, NAU's Most Promising New Scholar Award in 2020, and NSF CRII Award in 2017. She was served as the organized and the TPC chair for several international IEEE workshops in the field of UAV communications, including the IEEE INFOCOM Workshop on Wireless Sensor, Robot, and UAV Networks (WiSRAN'19) and IEEE WOWMOM Workshop on Wireless Networking, Planning, and Computing for UAV Swarms (SwarmNet'20).

Junsuo Qu received the B.S. degree in telecommunication engineering from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 1991 and the M.S. degree in communication and information systems from Xidian University, Xi'an, China, in 1998. He is currently a Full Professor with the School of Automation, Xi'an University of Posts and Telecommunications, Xi'an, China, and a Member of the China Institute of Communications. He is also the Director of the Xi'an Key Laboratory of Advanced Control and Intelligent Process. He is leading an IoT Research Team with the School of Automation.

Abolfazl Razi received the B.S. degree in electrical engineering from the Sharif University, in 1998, the M.S. degree in electrical engineering from Tehran Polytechnic, Tehran, Iran, in 2001, and the Ph.D. degree in electrical engineering from the University of Maine, Orono, ME, USA, in 2013. He is currently an Assistant Professor with the School of Informatics, Computing and Cyber Systems, Northern Arizona University (NAU), Flagstaff, AZ, USA. Prior to joining NAU, he held two Postdoctoral positions in the field of machine learning and predictive modeling with Duke University, Durham, NC, USA, during 2013–2014 and Case Western Reserve University, Cleveland, OH, USA, during 2014–2015. His current research interests include smart connected communities, biomedical signal processing, wireless networking, Internet of things, and predictive modeling. He was the recipient of several competitive awards, including the Best Research of MCI in 2008, Best Graduate Research Assistant of the Year Award from the College of Engineering, University of Maine in 2011, and the Best Paper Award from the IEEE/CANEUS Fly By Wireless Workshop in 2011.