

Journal Pre-proof

Q-learning driven routing for aeronautical Ad-Hoc networks

Tuğçe Bilen, Berk Canberk

PII: S1574-1192(22)00137-7
DOI: <https://doi.org/10.1016/j.pmcj.2022.101724>
Reference: PMCJ 101724

To appear in: *Pervasive and Mobile Computing*

Received date: 6 August 2022
Revised date: 17 October 2022
Accepted date: 7 November 2022

Please cite this article as: T. Bilen and B. Canberk, Q-learning driven routing for aeronautical Ad-Hoc networks, *Pervasive and Mobile Computing* (2022), doi: <https://doi.org/10.1016/j.pmcj.2022.101724>.

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2022 Published by Elsevier B.V.



Revised manuscript (clean version)

Q-Learning Driven Routing for Aeronautical Ad-Hoc Networks

Tuğçe BILEN^a, and Berk CANBERK^{b,c}^aComputer Engineering Department, Istanbul Technical University, Istanbul-Turkey^bArtificial Intelligence and Data Engineering Department, Istanbul Technical University, Istanbul-Turkey
e-mails:{bilent, canberk}@itu.edu.tr^cSchool of Computing, Engineering and Built Environment, Edinburgh Napier University
e-mail:B.Canberk@napier.ac.uk

Abstract

The aeronautical ad-hoc network (AANET) is one of the critical methodologies to satisfy the Internet connectivity requirement of airplanes during their flights. However, the ultra-dynamic topology and unstable air-to-air link characteristics increase the need for AANETs to a particular routing algorithm compared to the terrestrial networks. This need is mainly because these AANET-specific characteristics increase the delays, packet losses, and network load with accuracy reduction by continuously changing topology and breaking air-to-air links during routing. The works in the literature do not satisfy the ultra-dynamic topology and unstable air-to-air link characteristics of AANETs during routing. On the other hand, the routing algorithm can adapt to the dynamic conditions of AANETs by utilizing Artificial Intelligence (AI) based methodologies. For adaptation to this dynamic environment, we aim to let the airplanes find their routing path through exploration and exploitation by mapping the AANET environment to QLR. Clearly, this article proposes an updated Layered Hidden Markov Model (updated-LHMM) estimation-based Q-learning (QLR) routing for AANETs to solve the delay, packet loss, network load, and accuracy problems. For this aim, the Bellman Equation is adapted to the AANET environment by proposing different methodologies for its related QLR components. Results reveal that the proposed strategy mainly reduces the routing delay and packet losses by 30% and 33% compared to the methods in the literature.

Keywords: AANETs, Routing Management, Reinforcement Learning, Q-Learning, Hidden Markov Model

1. Introduction

According to the International Air Transport Association (IATA) expectations, there will be a massive increase in passenger numbers, and it will reach 7.8 million in 2036, as shown in Fig. 1 [1]. The enormous increase in passenger numbers with developing technology transforms the in-flight connectivity (IFC) into an essential requirement instead of a luxury.

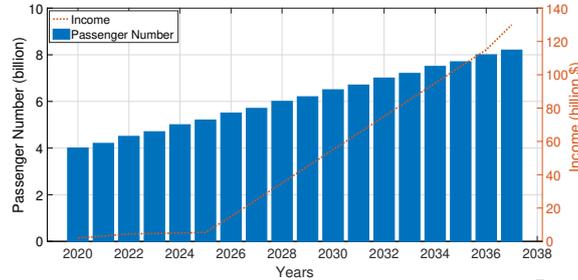


Figure 1: Passenger and income changes by year.

The passengers want to access the Internet in a high quality similar to that they receive on a terrestrial connection during flight without observing interruption. These changes have increased the number of people who need and use IFC. Correspondingly, the importance of IFC for passengers increases airlines' motivations to satisfy the passenger demand, attract and hold passengers by generating revenue. More specifically, it is expected that the total revenue obtained from IFC will increase from \$700 million in 2015 to nearly \$5.4 billion by 2025 with a 23% compound annual growth rate (CAGR) over the ten years, as shown in Fig. 1 [2]. These reasons have made IFC an essential field of research, and different technologies have been proposed to enable its infrastructure. Although the satellite and air-to-ground connectivities constitute the leading technologies for IFC, the high cost, latency, and reduced coverage generally deteriorate these key enablers' efficiency [3].

One of the novel solutions is the AANETs to satisfy the IFC's huge demand by also solving the defects of satellite and air-to-ground connectivities. The AANETs are based on creating air-to-air links and transmission of packets over these connections to enable IFC for airplanes [4]. The advantage of routing protocols and management models are taken to transfer the packets over AANET. However, traditional routing protocols designed for stationary and terrestrial networks do not satisfy the requirements for AANETs [5]. The upcoming section will explain why special routing management is required for AANETs and the problems encountered during this management design.

1.1. Problem Statement for AANET Routing Management

The ad hoc networks are a decentralized network type where nodes communicate directly with each other. Firstly, these nodes are considered as mobile users and this consideration is the basis for the Mobile Ad Hoc Networks (MANETs). Then, these mobile nodes become the vehicles and the Vehicular Ad Hoc Networks (VANETs) are created. With the evolving technology, the vehicles are replaced by the UAVs and this ad hoc network is called Flying Ad Hoc Networks (FANETs). At the

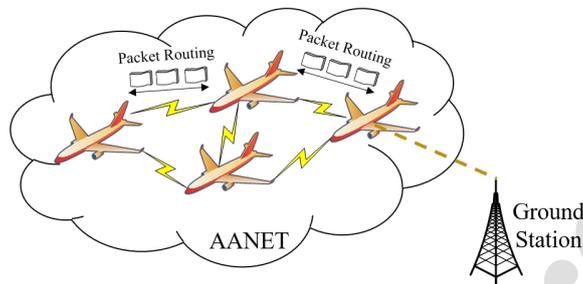
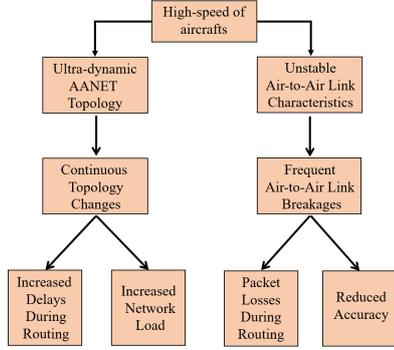


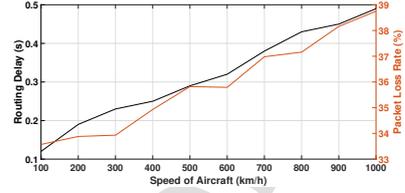
Figure 2: AANET routing mechanism.

final stage, the ad hoc networks are created with the airplanes to form the AANETs. In AANET, the aircraft's packets are routed through air-to-air links until reaching the destination aircraft that have Internet connectivity, and each aircraft could be considered as a router during this procedure, as shown in Fig. 2. However, the requirements of AANET are not satisfied by the traditional routing protocols designed for terrestrial networks since it has specific characteristics compared to other ad-hoc networks. This paper investigates these characteristics from two different aspects as ultra-dynamic AANET topology and unstable air-to-air link characteristics. These two characteristics also lead to further problems during AANET routing management, as explained follows:

- Ultra-dynamic AANET topology: The high speed of aircraft makes the AANET environment ultra-dynamic compared to the terrestrial vehicular networks, as shown in Fig. 3a. Due to this dynamic environment, the AANET topology changes continuously. Accordingly, the dynamic characteristic of AANET topology makes it challenging to determine the route and increases the routing delay and network load as given in Fig. 3a. More specifically, a new route must be found for each aircraft during continuous topology changes. This situation increases the cost of routing in terms of delays, as shown in Fig. 3b. The ultra-dynamic AANET topology also reduces the stability of air-to-air links, as detailed in the following item.
- Unstable Air-to-Air Link Characteristics: The main reason for the unstable air-to-air link characteristics is based on the high speed of airplanes in AANET topology, as shown in Fig. 3a. Here, the high-velocities cause frequent breakages of air-to-air links by continuously changing the connected aircraft and links as given in Fig. 3a. These frequent breakages reduce the accuracy of routing operation, and this situation also increases the packet losses during routing, as shown in Fig. 3b.



(a) Routing management problems in AANET.



(b) Relation of aircraft speed with routing delay and packet loss rate.

Figure 3: Routing management challenges in AANET.

Based on these two characteristics and their possible problems, the routing management model determination constitutes the critical design criteria to increase the packet delivery success and accuracy in an AANET by reducing routing latency with lower network load. There are numerous studies proposing routing management algorithms in the literature. These algorithms will be detailed in upcoming section. Although there are numerous AANET routing algorithms in the literature, the ultra-dynamic environment and unstable air-to-air link characteristics are not considered simultaneously by any work. More specifically, the packet routes should be calculated dynamically according to the current status of network topology due to these two AANET-specific characteristics. However, this dynamism has not been handled dramatically to reduce packet loss, delay, and network load with increased accuracy.

Moreover, if the AI-based solutions are utilized, airplanes can adapt to the dynamic conditions of AANETs by finding routes according to the instant status of AANETs. However, the AI-driven methods for AANET routing management are not considered in these works, as shown in Table 1. On the other hand, reinforcement learning enables airplanes to make their own routing decisions sequentially through exploration and exploitation different from the usual AANET routing mechanism as given with Fig. 2. More specifically, each aircraft can be considered as an agent having a state. This agent takes action (route packet), gets a reward, and switches to a new state during the routing procedure as detailed in upcoming section. Furthermore, the reinforcement learning-based routing algorithms in literature (i.e.[6],[7], [8], [9], [10]) are proposed for other ad-hoc networks as explained above. Therefore, these algorithms do not satisfy the two main specific characteristics of AANETs as explained in above. At that point, the deep learning based algorithm is proposed for AANET

routing in [11]. However, the algorithm in this work is first trained with historical flight data and then stored on airplanes for routing decision. At that point, ultra-dynamic and unstable characteristics of AANETs can reduce the efficiency of this algorithm.

This article proposes a routing management model for AANETs by utilizing Q-learning, one of the main reinforcement learning approaches. The main aim of the reinforcement learning utilization is to let the airplanes find their own routing paths in the dynamic AANET environment through exploration and exploitation without any central node or entity. Also, another selection reason for reinforcement learning is that the decision-making is sequential in AANET routing, and this situation strongly matches the principal characteristic of reinforcement learning. For these reasons, the Bellman equation is adapted by proposing different methodologies for each of its elements according to AANETs.

The main contributions could be listed as follows:

- The Bellman equation is utilized by proposing different methodologies, each of its related QLR components according to the AANET environment.
- The updated-LHMM is proposed for the maximum state-action determination component in QLR. Here, the routing path of an aircraft is divided into layers and find the best airplanes for layers that give the maximum possible states.
- The G/G/1 system-based queuing load is used to determine the aircraft states in the updated LHMM. The decomposition-based two-parameter is also utilized characterization to determine air-to-air link states that correspond to the state-transition probabilities in the updated-LHMM. Accordingly, the packet transfer procedure between two aircraft is divided into three phases: superposition, queuing, and splitting by considering accepted, queued, and transferred packet streams.
- A quadratic reward function is defined by considering the air-to-air link quality and distance parameters for the dynamic reward determination component. Here, the link longevity and G/G/1 system-based queue waiting time metrics is used for the link quality. Also, the Yen's N-shortest path algorithm is used for the distance parameter.

The rest of the paper is organized as follows: Section 3 analyzes the literature in terms of routing algorithms for AANETs. Section 3 describes the QLR for AANETs. Section 4 gives the proposed QLR-driven AANET routing management framework. The performance of the proposed method is evaluated in Section 5. The paper is concluded in Section 6. Also, Section 7 gives the future directions.

2. Related Works

As explained in Section 1, there are numerous studies proposing routing management algorithms in the literature. The Geographic Load Share Routing (GLSR) forwards the packets to the geographically closest neighbor destination based on greedy forwarding in [12]. The Multipath Doppler Routing (MUDOR) aims to find a more stable path by utilizing mobility, and link duration parameters in routing decisions in [13]. The multi-point relaying is used to reduce the number of redundant transmission messages by the Path Link Availability Protocol (PLAR) in [14]. AeroRP proposes the per-hop basis routing decision without considering source to destination end-to-end route [15]. The Hierarchical Space Routing Protocol (HSRP) presents a routing algorithm by changing the HELLO beacon frequency with flight flow rate, flight speed, and air vehicle density parameters [16]. The Ad-hoc Routing Protocol Aeronautical Mobile Ad-Hoc Networks (ARPAM) uses different parameters like distance and the number of hops between nodes to discover the shortest routes in [17]. The work offers the Delay aware Multipath Doppler Routing (DMDR) by utilizing the Doppler shift, expected queuing delay of packets, and relative velocities to select the stable and efficient paths during routing in [18]. To reduce the congestion, end-to-end delay, and packet loss rate, the Multiple QoS Parameters-based Routing Protocol (MQSPR) utilizes the path availability period, available path load capacity, and path latency metrics during routing in [19]. The actual aircraft densities are considered to maximize the packet delivery with Node Density Trajectory Based Routing (NoDe-TBR) in [20]. The Reactive-Greedy-Reactive (RGR) routing algorithm is proposed to combine the Greedy Geographic Forwarding and reactive routing mechanisms by employing location information, and end-to-end paths [21]. The Trunk-Branch Cooperation aided Routing (TBCR) is proposed based on the geographic greedy forwarding strategy in [22]. This algorithm considers the geographic locations and the virtual locations of airplanes. The AODV-LD proposes the utilization of expected residual path duration during routing [23]. It also includes two additional approaches as stochastic and deterministic strategies. To minimize the total delay during routing, the shortest path algorithm is utilized with a weighted digraph formulation in [24]. In addition to these, multi-objective genetic algorithm is proposed to optimize the end-to-end latency, the end-to-end spectral efficiency, and the path expiration time during routing in [25]. Similarly, the classic NSGA-II that is a kind of multi-objective evolutionary algorithm is proposed to optimize the routing path [26]. This methodology generates the approximation for the Pareto optimal set.

Table 1: Existing Techniques for AANET Routing Management

Routing Algorithm	Parameters	Performance Criteria	Methodology
GLSR [11]	Queue Size Geographic Distance	Network Throughput Packet Delivery Ratio End-to-End Delay	Greedy Forwarding
MUDOR [12]	Doppler Value	Link Down Number of Handoff	Route Request Route Reply
PLAR [13]	Node Density Link Availability	Path Availability Rate	Multipoint Relaying
AeroRP [14]	Time to Intercept Speed	Average Accuracy Average Delay Average Overhead	Speed-based Heuristic Geographic Routing
HSRP [15]	Air Vehicle Density Flight Flow Rate Aircraft Velocity	Delivery Ratio End-to-End Delay Routing Overhead	HELLO Beacon
ARPAM [16]	Distance Number of Hops	Routing Overhead	Route Request Route Reply
DMDR [17]	Relative Velocity Expected Queuing Delay	Throughput Routing Overhead	Doppler Shift
MQSPR [18]	Path Availability Period Residual Path Load Capacity Path Latency	End-to-End Delay Packet Delivery Ratio Handoff per Hour	Internet gateway (IGW) Advertisements Forward Best Advertisement
Node-TBR [19]	Delay Reachability	Average Delay Normalized Reachability Aircraft Density	Geopath Computation Algorithm Forwarding Algorithm
RGR [20]	Location End-to-End Paths	Delivery Ratio Average Delay Average Routing Packets	Greedy Geographic Forwarding Reactive Routing
TBCR [21]	Geographic Locations Virtual Locations	End-to-End Transmission Delay Synchronization Overhead	Geographic Greedy Forwarding
AODV-LD [22]	Uniform Motion Link Duration	End-to-End Delay Packet Delivery Ratio	Expected Residual Path Duration
Minimum Delay Routing [23]	Delay	Routing Delay	Shortest Path Algorithm

3. Q-Learning Routing (QLR) for AANETs

3.1. AANET Topology

This paper considers a two-layered AANET topology consisting of ground and aircraft planes, as shown in Fig. 4. In this topology, we aim to implement a QLR-driven AANET routing management framework given in Fig. 4. The details of these layers and the proposed framework could be summarized as follows:

- **Ground Plane:** In this plane, the special ground stations (GSs) on terrestrial areas are deployed to utilize cellular communication, as shown in Fig. 4. Correspondingly, a direct air-to-ground link between aircraft and the closest ground station could be established to enable broadband

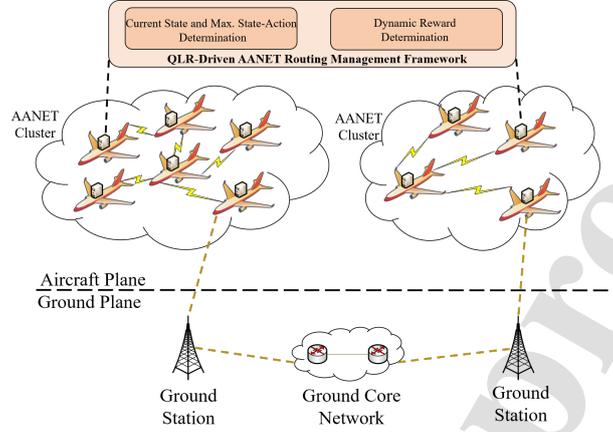


Figure 4: The AANET topology.

connectivity. More clearly, the advantage of the cellular communication model is taken thanks to these ground stations to satisfy the Internet connectivity requirement of airplanes.

- **Aircraft Plane:** This plane consists of the airplanes connected through air-to-air links to form an AANET, as shown in Fig. 4. Here, the packets of an aircraft are transferred through other airplanes and air-to-air links to reach the aircraft having Internet connectivity. Accordingly, airplanes without ground station connectivities can utilize the AANET to access the Internet [27]. In this article, the primary focus is routing the packets over an already established AANET. For this reason, we do not focus on the physical characteristics and channel models of air-to-air links in this layer in the rest of the article.
- **QLR-Driven AANET Routing Management Framework:** This framework is implemented on airplanes to take their own routing decisions based on the Q-learning without any central node or entity. This framework is composed of two main parts current state& maximum state-action determination and dynamic reward determination, as shown in Fig. 4. The details of this framework are explained in Section 4.

3.2. Mapping for Q-Learning to AANET

This paper adapts the Q-Learning, which is one of the basic reinforcement algorithms for routing management. Here, each aircraft is considered as an agent having a state, then this agent takes action (route packet), gets a reward, and switches to a new state during the routing as shown in Fig. 5. Based on this, the AANET routing scenario is mapped to the reinforcement learning concept, and the main components of this system could be explained as follows:

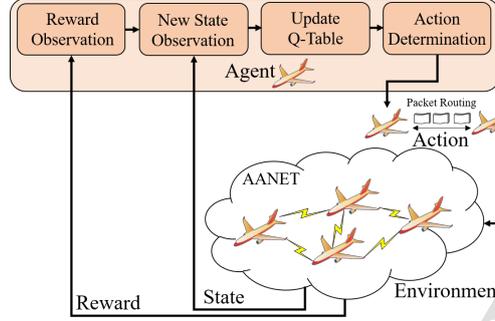


Figure 5: Q-learning for AANET routing.

- *Environment*: The AANET in the aircraft layer constitutes the environment for Q-learning. Here, we specifically create the topology of AANETs in the form of clusters since the airplanes in the same cluster have the same movement and position characteristics. However, AANET is an ultra-dynamic and unstable environment, as detailed above [28]. For this reason, the common clustering algorithms do not satisfy the requirements of the AANET environment. Therefore, we should utilize a clustering algorithm that considers the AANET-specific characteristics. At that point, we use our already proposed three-phased clustering algorithm for topology formation since this is a highly complex procedure. This algorithm considers the cluster creation, air-to-air link determination, and head selection procedures at the same time to create more stable clusters by also considering their aircraft-specific characteristics. The details about the three-phased topology formation algorithm can be found in [29]. Therefore, in the rest of the article, we consider that the AANET topology is created in the form of more stable clusters based on the three-phased formation algorithm. With this clustering algorithm, we can deal with the ultra-dynamic AANET topology and unstable air-to-air link characteristics.
- *Agent*: Each aircraft that routes the packet in an AANET is defined as an agent. Therefore, each cluster includes agent i , $i = 1, 2, \dots, N$. Here, N shows the maximum aircraft number in the corresponding cluster. Also, each cluster in the AANET can have a different aircraft number.
- *Action (a)*: Every packet transfer done by an aircraft through air-to-air links is defined as an action. The action set includes these actions, and this set is different for each aircraft depending on its air-to-air links. Also, each action causes a change in the current state of airplanes.
- *State (s)*: Each cluster in an AANET is modeled according to the G/G/1 queuing system and decomposition-based two-parameter characterization. Here, the states of airplanes are repre-

Table 2: Mapping of QLR components to AANET

QLR Component	AANET Mapping
Environment	AANET
Agent	Aircraft
Action	Packet Transfer
State	G/G/1 based queuing load for aircraft states Decomposition-based two parameter characterization for air-to-air link states
Reward	Air-to-air link quality Air-to-air link distances
Reward Table	Table of air-to-air link rewards
Q-Table	Table of routing decisions

sented with G/G/1 system-based queuing load. Also, the states of air-to-air links are represented with decomposition-based two-parameter characterization by considering the accepted, waited, and transferred packet streams. Accordingly, each air-to-air link state is represented with two parameters as the arrival rate (λ_i) and squared coefficient of variation of arrival requests (C_{Ai}^2) to aircraft i in the form as $s_i(\lambda_i, C_{Ai}^2)$, $i = 1, 2, \dots, N$. These queuing-based models increase the suitability of the Markovian characteristic of reinforcement learning to the AANET environment.

- *Reward (R)*: The aircraft accepts a reward for each packet transfer, and it is used for measuring the success of this transfer. This article utilizes the air-to-air link quality and distance metrics for reward definition. Here, the link quality is defined by the G/G/1-based queue waiting time and air-to-air link longevity parameters.
- *Reward Table*: The rows and columns of this table show the airplanes in a cluster. Also, the value of each cell denotes the reward of the corresponding air-to-air link. Therefore, each aircraft fills a row in the reward table with associated reward values belonging own links.
- *Q-Table*: The Q-table is filled through an iterative process by using Bellman Equation as given with Eq. 1 for routing decisions. The aircraft is learned through the experience without any guide or controller by filling this table. These mappings are also summarized in Table 2.

The aircraft having Internet connectivity through a ground station is also defined as IGW.

Table 3: Elements of QLR-Driven AANET Routing Management

Element	Aim	Proposed Methodology
$Q_{t-1}(s, a)$	Current state-action determination	G/G/1-Based Queuing Model
α	Learning rate	Constant
$R(s, a)$	Dynamic reward determination	Polynomial function definition based on air-to-air link qualities and distances
γ	Discount factor	Constant
$\max_{a'} Q(s', a')$	Maximum state-action determination	G/G/1-Based Queuing Model Updated Layered-Hidden Markov Model Decomposition-based two parameter characterization

4. QLR-Driven AANET Routing Management Framework

As explained above section, the air-to-air links in a cluster are determined through the three-phased clustering. According to its characteristics, we can establish more long-lasting links between airplanes. Therefore, the longevity of air-to-air links allows the clustered AANET topology could be taken as stable. Accordingly, we can apply the QLR-driven routing to these stable clusters. In this QLR-based methodology, we aim to enable airplanes in an AANET with memory, and accordingly, they can determine their own routing path dynamically. Thus, we study in a clustered topology whose links do not change as much as possible, and we handle dynamism on it with QLR. To achieve this, we adapt the Bellman Equation as given with Eq. 1. With Eq. 1, the aircraft in a particular state considers all the possible routing directions and states of airplanes in these directions to select one of them, maximizing reward value.

$$Q_t(s, a) = Q_{t-1}(s, a) + \alpha(R(s, a) + \gamma \max_{a'} Q(s', a') - Q_{t-1}(s, a)) \quad (1)$$

In Eq. 1, s and a show a particular state of an aircraft and an action performed in that state, respectively. The $R(s, a)$ is a reward function outputting a reward value by taking action a at state s . Also, the $Q_{t-1}(s, a)$ equals the value of state-action pair at time t and the s' is the state coming after from s by taking action a . Additionally, the a' is the action that could be taken at state s' . Here, γ is a discount factor that shows the importance of upcoming states with α learning rate. This paper adapts Eq. 1 to AANETs by determining its components with proposed methods as given in Table 3. The main aim of the component adaptations is to make QLR compatible with the ultra-dynamic and unstable characteristics of the AANET environment. Each component of the Bellman Equation is executed through QLR-driven AANET routing management framework as given with Fig. 4. This

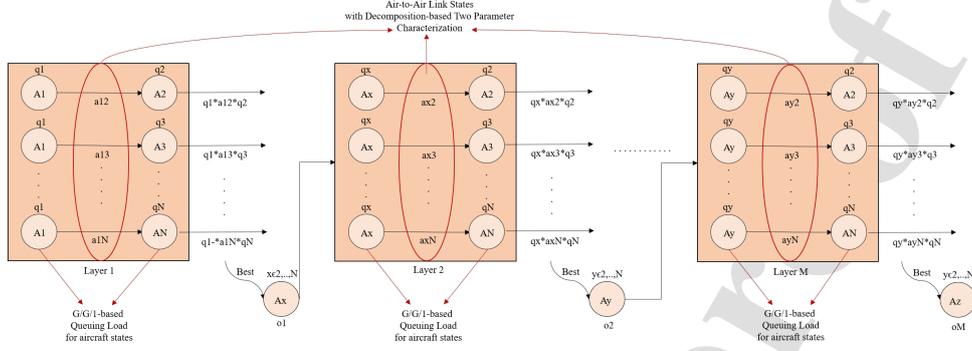


Figure 6: Updated-LHMM for AANET routing management.

framework and its components are detailed in the following subsections.

4.1. $(Q_{t-1}(s, a))$ and $\max_{a'} Q(s', a')$ Determination

The queuing theory could be used in many different systems that take input from outside, provide a certain service and produce output. In this way, any estimation could be possible about the number of services that can be received from the system, and its efficiency could be increased. Also, the sequential decision making of reinforcement learning could be supported by the queuing theory. For these reasons, this article takes advantage of the $G/G/1$ queuing system to find the current state-actions $(Q_{t-1}(s, a))$ of airplanes. Accordingly, each aircraft are represented with a $G/G/1$ queue due to the general distribution of arrival and service times. This article relates the states of airplanes with the load conditions. For this reason, the queuing load parameter as given with Eq. 2 belonging to the $G/G/1$ system becomes the $Q_{t-1}(s, a)$ parameter for aircraft i during the current state determination.

$$L_{q_i} = \frac{\rho_i^2(1 + C_{S_i}^2)(C_{A_i}^2 + \rho_i^2 C_{S_i}^2)}{2(1 - \rho_i)(1 + \rho_i^2 C_{S_i}^2)}, i = 1, 2, \dots, N \quad (2)$$

In the queuing load definition given with Eq. 2, $C_{A_i}^2$ and $C_{S_i}^2$ correspond to the squared coefficient of variation of arrival and service times of aircraft i . Here, $C_{A_i}^2 = \sigma_{A_i}^2 / \lambda_i$, $i = 1, 2, \dots, N$ is the equation for the squared coefficient of variation of inter-arrival time. Similarly, $C_{S_i}^2 = \sigma_{S_i}^2 / \lambda_i$, $i = 1, 2, \dots, N$ gives the squared coefficient of variation of service time. In these equations, $\sigma_{A_i}^2$ and $\sigma_{S_i}^2$ correspond to the variance of the inter-arrival and service time, respectively. Also, λ_i becomes the mean arrival rate while μ_i represents the service rate on aircraft i . Here, these two parameters are also used to calculate the ρ_i in Eq. 2 as λ_i / μ_i . This ρ_i shows the utilization of single aircraft according to the mean arrival and service rates.

The current state-action pair is calculated as given with Eq. 2. However, the it is not sufficient to determine the routing path according to the QLR as given with Eq. 1. At that point, the Layered Hidden Markov Model is updated to shape the packet routing process of airplanes and determine the $\max_{a'} Q(s', a')$. Accordingly, the $\max_{a'} Q(s', a')$ could be found for each step of the routing path from a source aircraft to IGW. Here, we conduct the next aircraft determinations in one step after from a current layer. Therefore, the result of current layer gives $\max_{a'} Q(s', a')$ observation to the next layer as shown in Fig. 6. By applying the same procedures with updated-LHMM, we can find $\max_{a'} Q(s', a')$ for each step of the routing path to fill the Q-table.

Therefore, the whole routing path is divided into layers through the updated-LHMM and calculate possible aircraft probabilities in these layers to find the $\max_{a'} Q(s', a')$. Here, each layer corresponds to the $\max_{a'} Q(s', a')$ decision (that is also the next aircraft) and gives this decision as an output for the following layer, as shown in Fig. 6. The row number of a layer in updated-LHMM depends on the neighbor number of an aircraft having a packet for transfer. Also, each row consists of two columns that represent the current and neighbor airplanes, as shown in Fig. 6. The neighbor airplanes become hidden states, while output observation of a layer is called the observable state. The observations correspond to the $\max_{a'} Q(s', a')$ for that layer and could be represented as $O = o1, o2, \dots, oX, \dots, oM$ for M layers. Also, the hidden states are not directly used, and they only show the possible route for the corresponding aircraft. Additionally, current states of airplanes are shown with $q1, q2, \dots, qx, \dots, qN$ as shown in Fig. 6 and here, $q1 = L_{q1}, q2 = L_{q2}, \dots, qx = L_{qx}, qN = L_{qN}$ as given with Eq. 2. The transition probabilities between two airplanes in a layer are shown with $a_{ij}, i, j = 1, 2, \dots, N$. These transition probabilities are found by utilizing the decomposition-based two-parameter characterization as will be explained in the upcoming part. Therefore, the G/G/1 system-based queuing load and decomposition-based two-parameter characterization are utilized to find the observations for different neighbor airplanes in a corresponding layer. Then, the aircraft having the best observation among them with a greater value ($\max_{a'} Q(s', a')$) is selected and transferred to the next layer to find the maximum state-action pair for that layer.

These procedures are applied through the M layer and fill Q- and reward tables by considering the next steps during the movements. With the updated-LHMM concept, all of the possible routing probabilities could be considered for each step to find the $\max_{a'} Q(s', a')$. Also, by utilizing updated-LHMM, we aim to represent the routing probability distributions in AANET over sequences of different airplanes, as shown in Fig. 6. Additionally, as explained above, the transition probabilities between two airplanes are found through the decomposition-based two-parameter characterization. This concept will be detailed in the following part of the paper.

4.1.1. Decomposition-based Two Parameter Characterization

To determine the transition probabilities in updated-LHMM, we take advantage of the $G/G/1$ queuing system with decomposition-based two-parameter characterization. At that point, we consider each aircraft as a router that executes packet transfer through the existed links in an AANET. Here, the accepted, waited, and transferred packets affect the transition probabilities between airplanes. Accordingly, in this article, we take characteristics of accepted, waited, and transferred streams into consideration to determine the transition probabilities between airplanes. On the other hand, the ultra-dynamic environment constitutes one of the critical AANET characteristics, which makes the modeling of AANET challenging. More specifically, the direct analytical results cannot be reached for these streams, and at that point, the approximation-based models help to determine the transition probability metric. This article uses decomposition-based two-parameter characterization as an approximation method that divides the network into smaller parts and analyzes these parts as separate systems. This type of decomposition strongly supports the AANET environment, where the airplanes are the elements of topological clusters. Then, we can investigate the air-to-air links in aircraft clusters separately.

As explained above, a relation is defined between the accepted, queued, and transferred packets of airplanes to approximate the transition probabilities in two-parameter characterization. Here, we divide the packet transfer procedures of airplanes into three phases: superposition, queuing, and splitting. Accordingly, we can establish a relationship with the accepted, queued, and transferred packets of airplanes for transition probability approximation. In these relations, the main aim is to determine the arrival rate (λ_i) and the squared coefficient of variation of the inter-arrival times (C_{Ai}^2) to aircraft i . Here, we aim to find more precise results for C_{Ai}^2 different from the $Q_{t-1}(s, a)$ since we should reach the air-to-air link states instead of a single aircraft. To execute these targets, two-parameter characterization starts with the superposition phase. Here, the packets that come into an aircraft i are first combined as a total arrival stream to find C_{Ai}^2 as given with Eq. 3.

$$C_{Ai}^2 = \frac{\gamma_i}{\lambda_i} C_{0i}^2 + \frac{1}{\lambda_i} \sum_{j=1}^k \lambda_j C_{ji}^2, \quad i = 1, 2, \dots, k \quad (3)$$

In Eq. 3, C_{0i}^2 and C_{ji}^2 are the squared coefficient of variation of the inter-arrival times for the accepted packets from the directly connected neighbors and other airplanes, respectively. As shown in Fig. 7, each aircraft can receive packets from its directly connected neighbors, or these packets could be originated by others (aircraft j , $j = 1, \dots, N$). Accordingly, the total arrival rate to an aircraft could be formulated as $\lambda_i = \gamma_i + \sum_{j=1}^N \lambda_j r_{ji}$, $i, j = 1, \dots, N$. In this formula, the γ_i represents the arrivals of

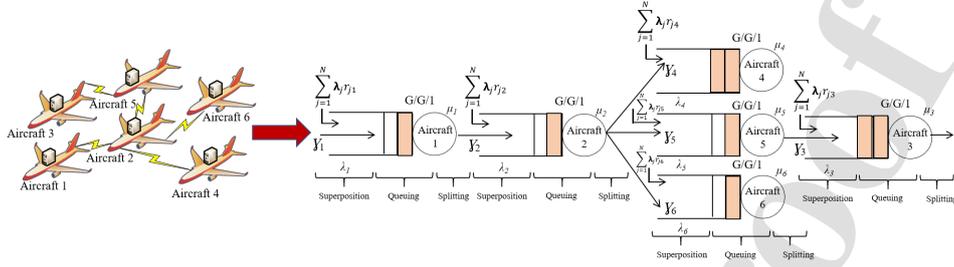


Figure 7: Decomposition-based Two Parameter Characterization-aided queuing model of AANET topology.

aircraft i from the directly connected neighbors, and it could be found as $1/E[A_i]$. Here, A_i represents the arrival time of packets to aircraft i . Also, r_{ji} is the transfer probability of packets from aircraft j to i , and the arrival rate of aircraft j is the λ_j as shown in Fig. 7.

After determining the behavior of accepted requests, we should investigate their subsequent paths. Here, the arrival packets wait in the queue of airplanes, and then they are transferred to other air-to-air links as shown in Fig. 7. Based on this path of packets, we should consider the queuing effects first. For this aim, we define a relation between the arrived packets into an aircraft and the left packets from the queue as given with Eq. 4. This relation is defined by calculating the squared coefficient of variation between the departures of the aircraft queue ($C_{D_i}^2$) as given with Eq. 4. In this equation, $\rho_i = \lambda_i/mu_i$ is the utilization rate of the aircraft i as explained in Section 4.1. Also, $C_{A_i}^2$ is obtained from Eq. 3. Finally, $C_{S_i}^2$ is the squared coefficient of variation of the service distribution for the aircraft i , and it is calculated as explained in Section 4.1. Therefore, in the first step, we highlight the input behaviors of an aircraft queue by defining Eq. 3. Then, to underline the departures from this queue, we define a relation as given with Eq. 4. Thus, in the last step, we should consider the departures forwarded to other queues from the current one, and this forwarding corresponds to the splitting procedure as shown in Fig. 7. More clearly, the queued packets are transferred to other airplanes, and for this reason, the third phase corresponds to the splitting procedures to digitize these transfers. Accordingly, the splitting phase establishes a relation between the transfers to other airplanes and queuing departures as given with Eq. 5. In this relation, $C_{D_i}^2$ is obtained from Eq. 4 and r_{ji} is the transfer probability of packets from aircraft j to i as explained above.

$$C_{D_i}^2 = \rho_i^2 C_{S_i}^2 + (1 - \rho_i^2) C_{A_i}^2, i = 1, 2, \dots, k \quad (4)$$

$$C_{j_i}^2 = r_{ji} C_{D_i}^2 (1 - r_{ji}), i = 1, 2, \dots, k \quad (5)$$

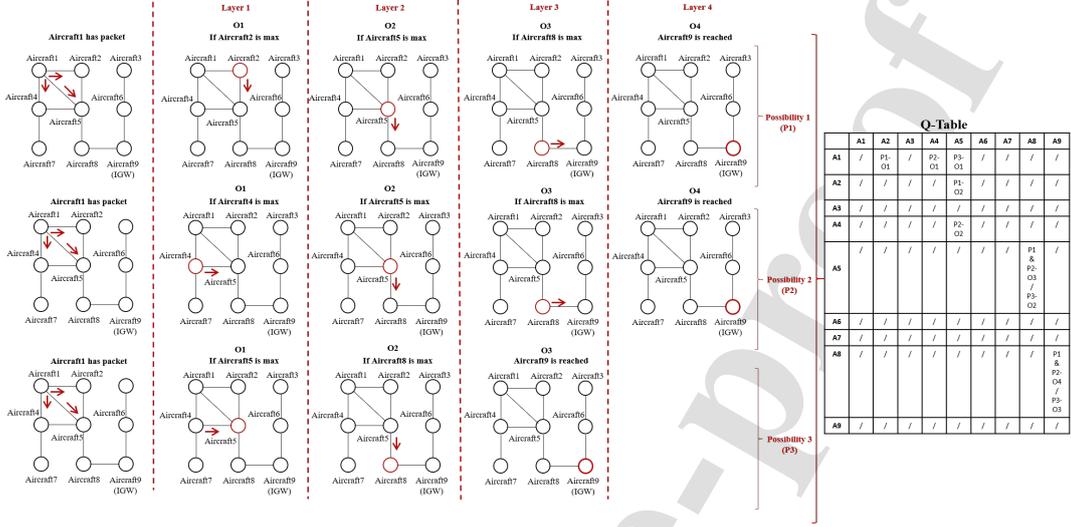


Figure 8: Illustrations of a sample packet transfer using the updated-LHMM strategy

Additionally, transfers from a queue (C_{ji}^2) mean input for another queue of the aircraft to which it is directly attached, as seen on Eq. 3 and Eq. 5. For this reason, the accepted, queued, and transferred packets should be reflected simultaneously for all airplanes, as shown in Fig. 7. In this way, we can obtain a complete result for each air-to-air link. Based on this, we should consider Eq. 3, Eq. 4, and Eq. 5 under one formulation as given with Eq. 6. Therefore, this equation gives the squared coefficient of variation between the arrival packets of aircraft i by considering accepted, queued, and transferred packets. This obtained result becomes the main aim to find the transition probability for the corresponding air-to-air as given with a_{ij} in Fig. 6. More clearly, we utilize this method to approximate the state transition probabilities for air-to-air links in arrival streams.

$$C_{Ai}^2 = \frac{\gamma_i}{\lambda_i} C_{0i}^2 + \sum_{j=1}^k \frac{\lambda_j}{\lambda_i} r_{ji} (r_{ji} [\rho_j^2 C_{Sj}^2 + (1 - \rho_j^2) C_{Aj}^2] + 1 - r_{ji}) \quad (6)$$

Moreover, the additional details about the mathematical foundations for two-parameter characterization could be found in [30] and [31].

4.2. Dynamic Reward ($R(s, a)$) Determination

As explained in Section 3.2, we have a Reward Table to show the possible rewards of all air-to-air links. The rows and columns of this table correspond to airplanes in a cluster. The value of each cell denotes the reward of related air-to-air links between corresponding airplanes. Accordingly, this table can also illustrate the topology of a cluster. More specifically, if there is no connection between two

airplanes, we should assign zero to the value of this link on the Reward Table. However, we cannot assign fixed values to the existed air-to-air links since the packet transfer does not take place with the same quality on every air-to-air link in an AANET. Also, the quality of a specific air-to-air link does not always remain at the same value due to the ultra-dynamic AANET environment. For these reasons, in this paper, we aim to determine the rewards dynamically instead of assigning fixed values. Clearly, we aim to set a reward value to an air-to-air link by considering the *quality* of packet transfer executed through this link with its *distance* value.

In this paper, we define two main parameters to highlight the air-to-air link *quality*: stability (link longevity) and queue waiting time. Here, we assume that the *quality* of an air-to-air link firstly depends on its stability since the ultra-high velocities of airplanes in AANET reduce this stability. Here, the longevity of links becomes the crucial criteria for digitizing link stability. More specifically, this parameter gives how long an air-to-air link lasts between two aircraft. Accordingly, the longevity of a link between aircraft i and j is calculated with $T_{Lij} = \frac{-\vec{D}_{ij}\vec{V}_{ij} + \sqrt{(\vec{D}_{ij}\vec{V}_{ij})^2 + V_{ij}^2(R^2 - D_{ij}^2)}}{V_{ij}^2}$. In this formula, \vec{D} and \vec{V} show the relative position and velocity vector between the next aircraft j and current ones i [32]. Additionally, the R offers the transmission range of an aircraft.

The second parameter for the *quality* becomes the waiting times of the transferred packets in the queue of corresponding airplanes. More specifically, the packets of an aircraft can wait longer in the queue of the nearest neighbor compared to the farther one. For this reason, we also add the queue waiting time parameter to the reward determination. To find the queue waiting time of airplanes, we again utilize the proposed G/G/1 based queuing model as explained in Section 4.1. According to G/G/1 model, the queue waiting time of airplanes (T_{q_i}) could be found by utilizing Eq. 7. In this equation, $C_{A_i}^2$ and $C_{S_i}^2$ correspond to the squared coefficient of variation of arrival and service times of aircraft i . The definitions of other parameters are the same with Eq. 2

$$T_{q_i} \approx \frac{\rho_i(1 + C_{S_i}^2)(C_{A_i}^2 + \rho_i^2 C_{S_i}^2)}{2(1 - \rho_i)(1 + \rho_i^2 C_{S_i}^2)\lambda_i}, i = 1, 2, \dots, N \quad (7)$$

In addition to the *quality*, the *distances* between airplanes affect the reward of the links. Clearly, the packets that are transferred to the destination aircraft through the shortest paths can receive higher rewards compared to the longer ones. This situation leads that *distance* of the aircraft to IGW through different air-to-air links is an important criterion during reward assignment based on the *quality*. Therefore, we should find the shortest distance from the aircraft to IGW without edge repetition. On the other hand, we have many state-action pairs in the QLR concept to select the most optimal one with a higher reward value, as shown in Fig. 8. This situation increases the number

Algorithm 1 Yen's Algorithm for N-Shortest Path**Data:** AANET topology, Source aircraft s , IGW, N **Result:** N-Shortest paths from source aircraft to IGW**Initialize** A and B

1. Determine A^1 from s to IGW with Dijkstra
2. Store $n - 1$ paths ($n = 1, 2, \dots, N$) in A and candidates in B
3. To find A^k , get A^{k-1} with path $j_s, a_1^{k-1}, a_2^{k-1}, \dots, a_l^{k-1}, IGW_i$
Set of vertices $P = \{j_s, a_1^{k-1}, a_2^{k-1}, \dots, a_l^{k-1}, i\}$ for analysis
4. **for** s to a_l^{k-1} in P **do**
 - if** $A^i \in A$ with $j_s, a_1^{k-1}, a_2^{k-1}, \dots, a_l^{k-1}$ **then**
 - Set edge weight from a to its immediate neighbor to ∞ for A^i
 - Set $R^k = \{j_s, a_1^{k-1}, a_2^{k-1}, \dots, a_l^{k-1}, i\}$ as root path
 - Set the path from a_l^{k-1} to IGW as S^k spur path
 - Remove vertices of R^k from AANET topology
 - Find shortest path from a to IGW with Dijkstra
 - if** Path is found and returned **then**
 - Add R^k and S^k to form candidate path
 - Add candidate path into B
 - end**
- end**

end

5. Choose path from B with shortest distance, add into A

6. Go to step 3 and continue until N shortest path determined

of possible routing paths for reward evaluation, and finding only one shortest path would not be a sufficient assessment. For this reason, we should find additional shortest path options to fill the reward and Q-table by considering all possibilities. At that point, we utilize the *Yen's Algorithm* to find the N possible shortest path on AANET [33]. Here, we aim to find the N shortest path from source aircraft to IGW with N iterations as A^1, A^2, \dots, A^N and keep them in container A. Here, the first shortest path A^1 is found by using Dijkstra's algorithm and other potential paths are held in container B. Then, a path with a lower cost is selected from container B to insert A. The pseudo-code of Yen's algorithm is given with Algorithm 1. As utilized in Algorithm 1, the path $\langle s, a_1^{k-1}, a_2^{k-1}, \dots, a \rangle$ from $\langle s, a_1^{k-1}, a_2^{k-1}, \dots, a_l^{k-1}, IGW \rangle$ is called as root path while the a_l^{k-1} to IGW is the spur path.

After conducting the *quality* and *distance* based operations, we should combine them to determine the final reward values. Here, by defining the link longevity and queue waiting time parameters for each shortest path separately, we obtain a reward function between aircraft j and i ($R_{ji}(s, a)$) in the polynomial form as given with Eq. 8. In this function, if there is a link between two airplanes and this link is located on the shortest paths, then the value of X is taken as X_1 . Conversely, if the existed

link between the aircraft i and j is not on shortest path, in this case the X equals to X_2 ($X_1 > X_2$). Finally, the value of X is taken as 0 if there is no air-to-air link between aircraft i and j . Moreover, the reward of a packet transfer through an air-to-air link connecting an aircraft to an IGW should be greater than other links given with Eq. 8 since the goal node has a higher reward according to the QLR principle. Additionally, in Eq. 8, $T_{L_{ij}}$ and T_{q_j} are found with Eq. 2 and Eq. 7. Here, $T_{L_{ij}}$ could be directly utilized while T_{q_j} at the denominator since we prefer links having higher longevity with a lower queue waiting times. More specifically, we can obtain a higher reward for an aircraft having a lower queue waiting time (T_{q_j}). Therefore, we can fill the reward table by considering all of the state-action pair and shortest path possibilities correspondingly, as shown in Fig. 8. Also, the flow of the proposed QLR-based AANET routing is summarized in Fig. 9.

$$R_{ji}(s, a) = T_{L_{ji}}X^2 + \frac{1}{T_{q_i}}X + c, \quad i, j \in 1, 2, \dots, N$$

$$c = \begin{cases} 100, & \text{if } i = \text{IGW} \& x \neq 0 \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

Finally, the learning rate (α) shows how often we modify the estimations. The number of iterations to reach the solution could be increased if this rate is selected as too small. Additionally, we cannot control the dynamic changes in the environment if it is taken as a more excellent value. Also, the discount factor (γ) shows the importance of future rewards, and we relate it with the change rate of the cluster. Clearly, if there are too many agent replacements in a cluster, then the immediate rewards become essential since the reward table should be updated for every change. These values should be selected from the range $0 < \alpha, \gamma < 1$ according to the QLR principles. By utilizing the proposed methodologies with Eq. 1, each aircraft can fill the Q-learning table according to its state and maximum state-action pair determinations as given with Eq. 8.

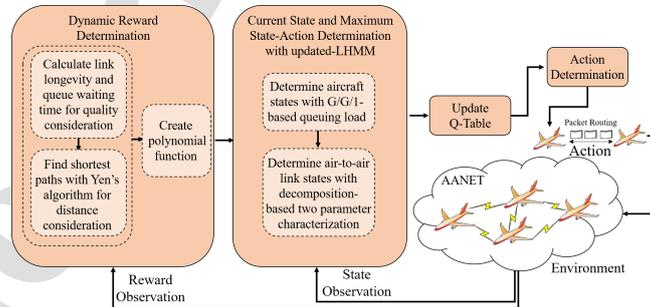
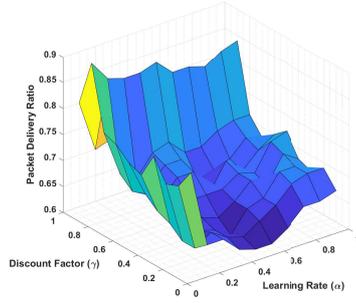
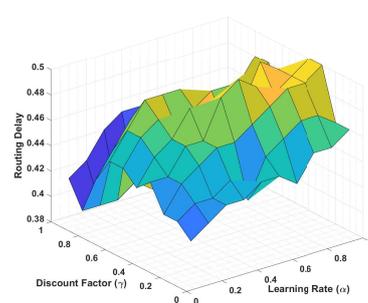


Figure 9: Flow of the proposed QLR-based AANET routing



(a) Packet Delivery Ratio



(b) Routing Delay

Figure 10: The effects of α and γ on (a) Packet Delivery Ratio and (b) Routing Delay.

Table 4: Simulation Parameters and Configurations

Scenario Parameters		Traffic Parameters		Channel Configurations	
Parameter	Value	Parameter	Value	Parameter	Value
Simulator	OMNET++	Packet Rate	4 per second	Channel Capacity	2 Mbit/sec
Region	1000 * 1000 nmi	Packet Size	512 Kbytes	RF Power	125 w
Aircraft Number	100-250	Queue Type	G/G/1	Bandwidth	10 Mbps
Aircraft Altitude	35000ft	Queue Size	30 packets	A2A Distance	450 nmi
Aircraft Speed	900-1800 km/h	Traffic Distribution	General	A2G Distance	200 nmi
Simulation Time	1000 s	Confidence Interval	95%	MAC Protocol	802.11p

5. Performance Evaluation

This part of the article aims to evaluate the proposed approach's performance and compare it with some routing algorithms in the literature. For this aim, this section first detail the simulation environment and utilized parameters. Then, the results belonging to these simulations will be evaluated.

5.1. Simulation Environment

The *OMNET++* framework is utilized to create the AANET in the form of clusters to simulate the routing management procedures. In these simulations, a 1000 * 1000 nautical mile (nmi) air traffic area is created with four ground stations at each corner of this area in these simulations. Here, nmi is selected since it is a unit of length used in air and space navigation. As explained in Section 3.2, the airplanes are connected to these ground stations as IGW, and the Internet connectivity is enabled thanks to these IGWs through an established AANET. Here, the distance between aircraft and

ground station is taken as 200 nmi for connection establishment [34]. If this length is exceeded, then the connection cannot be established as in terrestrial cellular networks. Also, the channel capacity of each ground station is taken as 2 Mbit/sec with 125 w transmit radio frequency power. These are the generally utilized values during aircraft to ground station (A2G) connectivity for extended performance. Moreover, the air-to-air links have line-of-sight (LOS) characteristics on the U/VHF band and a relatively high Signal to Noise Ratio due to the battery power is no limitation for the airplanes. For this reason, the minimum distance are taken to establish air-to-air link (A2A) as 450 nmi at 35000ft flight altitude with equal capacities and 10 Mbps air-to-air link bandwidth [35]. Here, the packet intensity of links are normalized according to the given capacities of them. Also, the idealized communication link model is used with an omnidirectional transmission range [36]. This omnidirectional antennas can execute the signal transmission in all directions; for this reason, it could be considered more suitable for AANETs. Additionally, we consider each aircraft as a router with the capacity to receive, queue and route packets to its neighbors according to the AANET concept and we model these queues according to the G/G/1 system with buffer size of 30 packets [37]. This maximum size is selected to reduce the drops caused by queuing overflow. Nevertheless, the received packets could be dropped if the queues of airplanes are full. Also, the packets are generated at random airplanes according to the general distribution and the packet production rate of airplanes are limited to 4 per second with 512 Kbytes. These values are also selected according to the queuing capacities to reduce the drops caused by drops. Additionally, 10 traffic intensity levels are considered on OMNET++ ranging from 10% to 100% of total network capacity. Correspondingly, we normalize the network traffic intensity according to the capacity. Here, we also generate 100 traffic matrices for each traffic intensity according to the general distribution to obtain more exact results. Moreover, the error bars will be given in Section 5.3 indicates the 95% confidence interval. We summarize all of these simulation parameters on Table 4.

In these simulations, we also utilize the TensorFlow as a backend with Phyton 2.7 to implement the QLR-driven AANET routing management framework of the topology. Here, the discount factor of future rewards is selected as 0.9 with a 0.1 learning rate. We determine the discount factor and learning rate values according to their effects on routing delay and packet delivery ratio. Accordingly, we select $\alpha = 0.1$ and $\gamma = 0.9$ to enable a good trade-off between the routing delay and packet delivery ratio, as shown in Fig. 10. Also, to utilize Yen's N-shortest path algorithm, we use the Dijkstra Algorithm in the first step as given with Algorithm 1.

Moreover, we evaluate the reward metric to tackle the success of the exploration and exploitation phases. In this paper, the exploration and exploitation phases follow a greedy approach, as shown

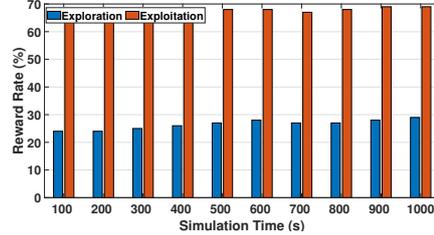


Figure 11: Reward performance of exploration-exploitation phases.

in Fig. 8. More clearly, the airplanes do not initially know the current topology and routing paths. For this reason, the aircraft explores the AANET cluster to improve the current information and next rewards in the exploration phase. On the other hand, in exploitation, the currently estimated knowledge is utilized to get a higher reward based on the Greedy approach.

In this article, we highlight the rewards of the exploration and exploitation phases based on simulation time, as shown in Fig. 11. Here, we first execute the exploration phase to explore the clustered AANET topology. Accordingly, we aim to improve the following rewards by continuously exploring the environment. For this reason, as shown in Fig. 11, we can perform the proposed methodology with a lower reward compared to the exploitation. More clearly, the exploitation phase utilizes the obtained knowledge based on the Greedy model to get higher rewards. For this reason, as shown in Fig. 11, we can receive roughly 50% better reward compared to the exploration case. Based on these, we evaluate the performance evaluation metrics by considering the exploitation phase. The details of these metrics will be explained in the upcoming part.

5.2. Investigated Evaluation Metrics

The performance of the proposed method is evaluated in terms of routing delay, packet loss rate, accuracy, and network load. The details of these metrics could be summarized as follows:

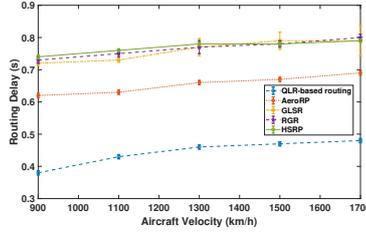
- *Routing Delay* is computed in terms of the waiting times of the routing packets in the queues of the airplanes. Here, we add the waiting times of all airplane queues from the first aircraft that started the routing until the ground station to reach a total routing delay. More specifically, it is calculated using Eq. 7 for all airplanes from the first aircraft to the ground station. This metric is crucial as it indicates the operation speed of the routing process. Also, the main reason for waiting time consideration for routing delay is to reduce the packet drops by calculating the queue waiting times of them. In this way, the IFC's huge demand could be satisfied since AANETs is one of the novel solutions to handle it.

- *Packet Loss Rate* shows the ratio of packets that are not successfully accepted by the destination aircraft to the total sent. This metric is vital in showing the rate at which packets are successfully delivered.
- *Accuracy* is found by dividing the number of received packets by the airplanes by the number of packets sent at the MAC layer. This metric gives information about the accuracy of the determined route by the routing algorithm in the ultra-dynamic AANET environment.
- *Network Load Rate* is calculated by dividing the number of packets in the queues of airplanes by the total sum of maximum sizes of all the queues. Here, loads of the queues are calculated with Eq. 2. This metric is essential to show that packets are not waiting in queues of airplanes and are being routed.

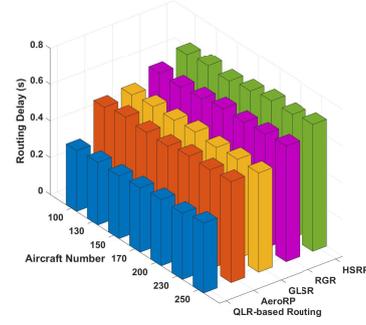
5.3. Simulation Results

The performance of the proposed QLR-based routing method is evaluated by comparing it with the GLSR [12], AeroRP [15], HSRP [16], RGR [21] algorithms as detailed in Section 2. The main reason for choosing these algorithms is that their purpose and performance criteria are almost the same as the proposed routing methodology. Also, the details of these routing methodologies are given in Section 2.

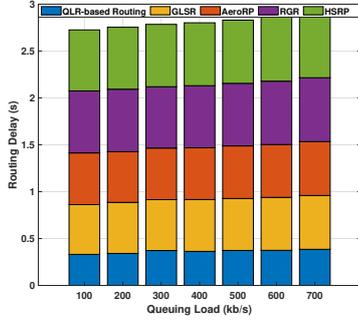
- **Routing Delay:** To calculate the routing delay, in our simulations, we train each methodology 100 times for each different x-axis value, and then, we take the average routing delay of each step as shown in Fig. 12. These average delays are summed to obtain a total routing delay for each x-axis episode. Here, the x-axis values correspond to the aircraft velocity, number, queuing load, and network traffic intensity, as shown in Fig. 12. At that point, we investigate the effects of topology and link change on performance by utilizing aircraft velocity and aircraft number metrics. The main reason of this utilization is that it is hard to digitize the topology and link related parameters during simulations. More clearly, we need traceable and modifiable metrics to see impact of topology and link related parameters on performance. Here, the increase in aircraft velocity and number means that the links and topology change rapidly. Also, queuing load and network traffic intensity shows the adaptability and endurance of the methodologies to different conditions. Accordingly, in all methodologies, the routing delay tends to increase with the growing x-axis values. Here, the AeroRP, GLSR, RGR, and HSRP observe the routing delay greater than 0.55s. The performance of the AeroRP is greater than the other three methodologies



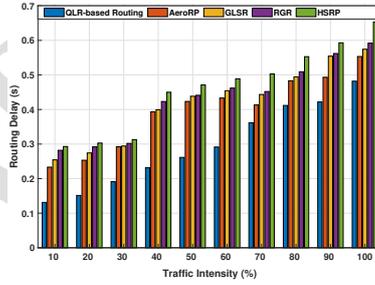
(a)



(b)



(c)



(d)

Figure 12: Routing Delay performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the (a) Velocity of Airplanes (b) Number of Airplanes (c) Queuing Load of Airplanes (d) Network Traffic Intensity

in literature as shown in Fig. 12a, Fig. 12b and Fig. 12d. The AeroRP takes time to intercept as a primary metric for routing decisions by including speed and relative velocity components. Accordingly, the source node in AeroRP can have an idea about the potential neighbors in the transmission range. This situation reduces the routing delay of the AeroRP compared to the other methodologies in the literature. However, as shown in Fig. 12c, the GLSR improves its performance compared to the other three methodologies in terms of queuing load. The main reason for this is that the GLSR also forwards the packets to the geographically closest neighbor with the additional speed of advance metric. Therefore, it can route the packet by considering the queuing load and waiting time situations thanks to this metric. Unlike the AeroRP and GLSR, the RGR utilizes the request/response messages to obtain the location information of neighbors. These periodic message exchanging procedures can increase the routing delay of

RGR compared to the other methodologies. Moreover, the HSRP works based on increasing or decreasing the spatial radius by observing the change of flight flow rate. More specifically, the update rate of HSRP is increased if the aircraft moves faster, aircraft number and flight flow rate are larger. The increased update frequency has a significant effect on the increase in routing delay of HSRP. Unlike, our proposed QLR-based routing strategy outperforms the other methodologies thanks to the delay consideration during reward determination as given with Eq. 8. By deploying the queuing delays of airplanes as provided with Eq. 7 in reward determination, we can enable airplanes to choose routes where their packets will wait less in the queue. Here, we can achieve the routing delay of less than 0.43s as shown in Fig. 12. Accordingly, we can reduce the routing delay by roughly 30% compared to the investigated methodologies in the literature.

- **Packet Loss Rate:** The average packet loss is expressed as the average of all losses observed at a certain point of the x-axis. Like the routing delay, we train each methodology 100 times for each certain point of x-axis values. Also, as in the routing delay, we select the x-axis values of packet loss metric as aircraft velocity, number, queuing load, and network traffic intensity as shown in Fig. 13. Again in all methodologies, the packet loss tends to increase with the growing x-axis values. Here, the AeroRP, GLSR, RGR, and HSRP observe packet loss rates higher than the 24% level. The AeroRP has a slightly better performance compared to the other three methodologies. It adds location information to each per-hop packet instead of periodic beacon messages. All neighboring node nodes in the transmission area can receive this location information from the packet header and keep it on the neighbor table to utilize during routing. Accordingly, the AeroRP can transfer packets with accurate routes, which affects reducing packet losses compared to the HSRP, GLSR, and RGR. Also, the HSRP, GLSR, and RGR have approximately the same performance as shown in Fig. 13. The change of spatial radius cannot effectively increase the packet delivery ratio of HSRP compared to the other two methodologies, as shown in Fig. 13. The requests, replies, and hello messages carry source location, destination location, and neighbor location information, respectively, in RGR. These could be utilized to update the node tables for reducing the packet losses compared to the HSRP. However, the GLSR has a reduced packet loss in terms of queuing load compared to the HSRP, RGR, and also airplanes, as shown in Fig. 13c. The main reason for this situation is that the GLSR can perform load sharing between the neighbors, thanks to the speed of advance metric. Unlikely, as shown in Fig. 13, we observe less packet loss in our proposed QLR-based routing strategy thanks to the updated-LHMM utilization. In the updated-LHMM, we consider the corresponding and neighbor aircraft

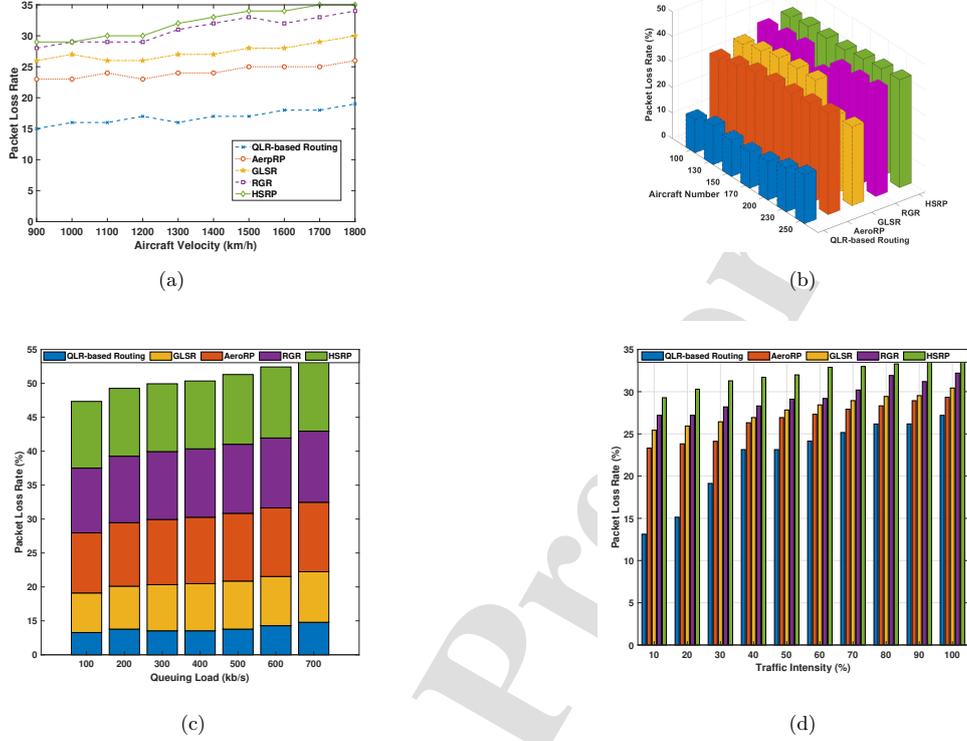
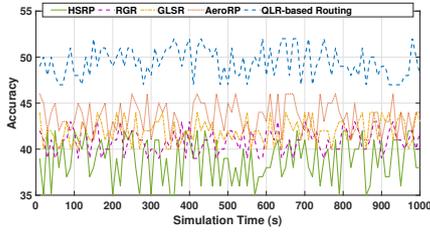


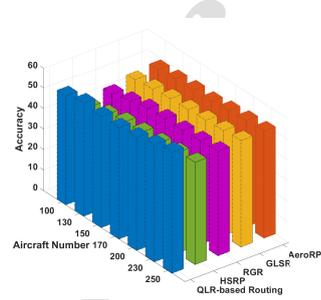
Figure 13: Packet loss rate performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the (a) Velocity of Airplanes (b) Number of Airplanes (c) Queuing Load of Airplanes (d) Network Traffic Intensity

states in terms of queuing load as given with Eq. 2. Also, we can consider the link states by observing accepted, queued, and transferred packets as given in Eq. 6. Therefore, we can route the packets by considering less loaded airplanes and links during $\max_{a'} Q(s', a')$ determination as summarized in Fig. 6. Accordingly, we can achieve packet losses of less than 16%, and we can reduce the packet losses roughly 33% compared to the investigated methodologies in the literature.

- Accuracy: We evaluate the accuracy of investigated methodologies according to the simulation time and number of aircraft parameters as shown in Fig. 14. To analyze the accuracy according to the simulation time, we take average accuracies of 100 aircraft in 100s intervals and sum them for each corresponding episode as shown in Fig. 14a. Here, the AeroRP, GLSR, and RGR observe an accuracy of less than 46% with very close accuracy values due to the consideration of geographic effects during routing. The AeroRP uses a prediction mechanism by utilizing the last



(a)

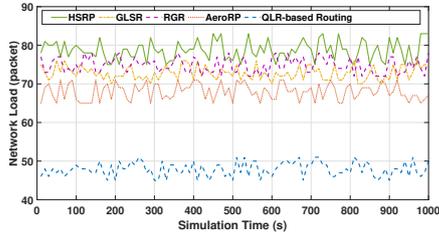


(b)

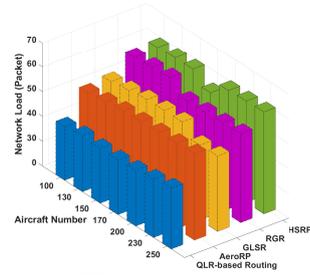
Figure 14: Accuracy performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the (a) Simulation Time (b) Number of Airplanes

known distance and velocity parameters to determine the neighbors in the transmission range of an aircraft. Clearly, the aircraft can accept hello beacons with coordinate and velocity data to select the time-to-intercept for each neighbor. Similarly, in GLSR, the packets are forwarded to the geographically closest neighbor to the destination. Also, the RGR uses both reactive and greedy approaches during the routing of data packets. Differently, the HSRP updates the neighbor node information table based on flight flow rate. Accordingly, they change the update frequency of hello beacons according to the flight flow rate, and this reduces the accuracy of HSRP by roughly 10% compared to the other three methodologies. However, our proposed QLR-based routing methodology outperforms these works by enabling an accuracy of higher than 50%. In our method, the aircrafts can take more precise routing decisions based on their learned experience, thank Eq. 1 and Q-table as given in Fig. 8. Also, we utilize the link longevity parameter during reward determination as provided with 8. This parameter increases accuracy by selecting more stable paths during routing. Similarly, we can improve the accuracy 26% compared to the investigated works in literature according to the increased aircraft number as shown in Fig. 14b.

- **Network Load:** Like the accuracy metric, we evaluate the load on the network according to the simulation time and aircraft number parameters as shown in Fig. 15. Again, we calculate the load on the network according to the simulation time by taking the average of results for 100 aircraft in 100s intervals and sum them for each corresponding episode as shown in Fig. 15a. Also, we measure the load on the network by increasing aircraft number 100 to 250 as shown in Fig. 15b. Here, we take the average network loads of airplanes for each corresponding aircraft number and sum them. The high load on the network shows that routing is not executed



(a)



(b)

Figure 15: Network load performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the (a) Simulation Time (b) Number of Airplanes

effectively, and the number of packets waiting in the queue is growing. The investigated four routing methodologies experience higher than 65% load on the network as given with Fig. 15a. The AeroRP makes its routing decisions per-hop basis based on speed-based heuristics different from the other geographic routing protocols as GLSR and RGR. This situation reduces its network load by roughly 10% compared to the other two algorithms. The GLSR and RGR work based on greedy geographic forwarding, and this means that they aim to find the nearest neighbor during routing. Here, the GLSR utilizes the speed of advance metric to consider the congestion and load balancing issues during routing. Also, the RGR keeps the location information of each neighbor and updated them via periodic hello messages. Accordingly, these situations increase the network load on these algorithms. The HSRP observes higher network load due to the dynamic correction of spatial radius. Unlikely, our proposed QLR-based routing strategy outperforms these four methodologies achieving network load less than 50% as given with Fig. 15a. In the proposed QLR-based routing, we aim to select the most optimal paths by considering the states of airplanes and links through updated-LHMM as shown with Fig. 6. To determine these states, we consider the load of the airplanes and links as given with Eq. 2 and Eq. 6. Also, we can achieve these determinations for all neighbors at the same hop to determine the optimal one as given with Fig. 8. Additionally, as shown in Fig. 14b, we can reduce the network load 24% compared to the investigated works in literature according to the increased aircraft number.

As explained throughout the article, the proposed QLR-driven routing management is proposed for AANETs by considering its ultra-dynamic topology and unstable air-to-air link characteristics. Additionally, this proposed approach could be utilized by other ad-hoc networks like FANETs and VANETs.

In these ad-hoc networks, the proposed methodology achieve a lower routing delay and packet loss rate compared to the AANETs. The main reason of this performance efficiency is that their topology and link characteristics are more stable compared to the AANET environment.

6. Conclusion

This article proposed QLR-driven routing management for AANETs to reduce the delays, packet losses, and network load with higher accuracy. In this methodology, our main aim was to adapt the Bellman equation in QLR to the AANET environment by proposing different methods for each of its elements. To obtain the maximum state-action determination component of the Bellman equation, we updated the layered hidden Markov model. In the updated-LHMM, we determined the states of airplanes by modeling them according to G/G/1 queuing system. Also, we utilized the decomposition-based two-parameter characterization to obtain the air-to-air link states in updated-LHMM. The decomposition-based two-parameter characterization takes characteristics of accepted, waited, and transferred streams into consideration to determine the air-to-air link states. Finally, to obtain the reward value in the Bellman equation, we defined a quadratic reward function based on the air-to-air link quality and distance metrics. Here, we used the link longevity and G/G/1-based queue waiting time parameters to define the link quality. Also, we used Yen's N-shortest path algorithm to reach the distance metric in the reward function definition. According to the simulation results, the proposed strategy reduces the routing delay and packet losses by 30% and 33% compared to the works in the literature. Also, we can reduce the network load by 24% with 26% higher accuracy.

7. Future Directions

In this article, we proposed a routing management model for AANETs based on QLR. Accordingly, we can handle the routing procedure in an ultra-dynamic and unstable AANET environment. Also, in our previous works, we proposed the topology formation, sustainability, and mapping management models for AANETs. Based on these, as future work, we aim to propose a digital twin-based approach to conduct these management procedures through a more proactive and real-time approach. In this way, we aim to improve the compatibility of these management models with the ultra-dynamic and unstable AANET environment.

References

- [1] IATA, 2036 forecast reveals air passengers will nearly double to 7.8 billion, Tech. rep. (Oct 2017).
- [2] Euroconsult, Passenger connectivity services to surpass \$5 billion by 2025, Tech. rep. (2016).
- [3] T. Bilen, B. Canberk, Learning-vector-quantization-based topology sustainability for clustered-aanets, *IEEE Network* 35 (4) (2021) 120–128. doi:10.1109/MNET.011.2000688.
- [4] T. Bilen, H. Ahmadi, B. Canberk, T. Q. Duong, Aeronautical networks for in-flight connectivity: A tutorial of the state-of-the-art and survey of research challenges, *IEEE Access* 10 (2022) 20053–20079. doi:10.1109/ACCESS.2022.3151658.
- [5] T. Bilen, B. Canberk, V. Sharma, M. Fahim, T. Q. Duong, Ai-driven aeronautical ad hoc networks for 6g wireless: Challenges, opportunities, and the road ahead, *Sensors* 22 (10) (2022). doi:10.3390/s22103731.
URL <https://www.mdpi.com/1424-8220/22/10/3731>
- [6] J. Wu, M. Fang, H. Li, X. Li, Rsu-assisted traffic-aware routing based on reinforcement learning for urban vanets, *IEEE Access* 8 (2020) 5733–5748. doi:10.1109/ACCESS.2020.2963850.
- [7] R. A. Nazib, S. Moh, Reinforcement learning-based routing protocols for vehicular ad hoc networks: A comparative survey, *IEEE Access* 9 (2021) 27552–27587. doi:10.1109/ACCESS.2021.3058388.
- [8] X. Guo, H. Lin, Z. Li, M. Peng, Deep-reinforcement-learning-based qos-aware secure routing for sdn-iot, *IEEE Internet of Things Journal* 7 (7) (2020) 6242–6251. doi:10.1109/JIOT.2019.2960033.
- [9] V. Di Valerio, F. Lo Presti, C. Petrioli, L. Picari, D. Spaccini, S. Basagni, Carma: Channel-aware reinforcement learning-based multi-path adaptive routing for underwater wireless sensor networks, *IEEE Journal on Selected Areas in Communications* 37 (11) (2019) 2634–2647. doi:10.1109/JSAC.2019.2933968.
- [10] L. A. L. da Costa, R. Kunst, E. Pignaton de Freitas, Q-fanet: Improved q-learning based routing protocol for fanets, *Computer Networks* 198 (2021) 108379. doi:<https://doi.org/10.1016/j.comnet.2021.108379>.
URL <https://www.sciencedirect.com/science/article/pii/S1389128621003595>

- [11] D. Liu, J. Cui, J. Zhang, C. Yang, L. Hanzo, Deep reinforcement learning aided packet-routing for aeronautical ad-hoc networks formed by passenger planes, *IEEE Transactions on Vehicular Technology* 70 (5) (2021) 5166–5171. doi:10.1109/TVT.2021.3074015.
- [12] J. Jiang, P. Zhang, R. Zhang, S. Chen, L. Hanzo, Aperture selection for aco-ofdm in free-space optical turbulence channel, *IEEE Transactions on Vehicular Technology* 65 (8) (2016) 6089–6100. doi:10.1109/TVT.2015.2471809.
- [13] N. Tadayon, G. Kaddoum, R. Noumeir, Inflight broadband connectivity using cellular networks, *IEEE Access* 4 (2016) 1595–1606. doi:10.1109/ACCESS.2016.2537648.
- [14] X. Fan, W. Cai, J. Lin, A survey of routing protocols for highly dynamic mobile ad hoc networks, in: *IEEE 17th International Conference on Communication Technology (ICCT)*, 2017, pp. 1412–1417. doi:10.1109/ICCT.2017.8359865.
- [15] Y. Cao, K. Wei, G. Min, J. Weng, X. Yang, Z. Sun, A geographic multicopy routing scheme for dtms with heterogeneous mobility, *IEEE Systems Journal* 12 (1) (2018) 790–801. doi:10.1109/JSYST.2016.2563519.
- [16] D. Zhong, Y. Wang, Y. Zhu, T. You, An Aeronautical Ad Hoc Network Routing Protocol Based on Air Vehicles Movement Features, in: *22nd IEEE International Conference on Applied Electromagnetics and Communications (ICECOM)*, 2016, pp. 1–6. doi:10.1109/ICECom.2016.7843898.
- [17] B. Du, R. Xue, L. Zhao, V. C. Leung, Coalitional graph game for air-to-air and air-to-ground cognitive spectrum sharing, *IEEE Transactions on Aerospace and Electronic Systems* 56 (4) (2020) 2959–2977. doi:10.1109/TAES.2019.2958162.
- [18] X. Li, J. Yan, Lepr: Link stability estimation-based preemptive routing protocol for flying ad hoc networks, in: *IEEE Symposium on Computers and Communications (ISCC)*, 2017, pp. 1079–1084. doi:10.1109/ISCC.2017.8024669.
- [19] Q. Luo, J. Wang, Multiple qos parameters-based routing for civil aeronautical ad hoc networks, *IEEE Internet of Things Journal* 4 (3) (2017) 804–814. doi:10.1109/JIOT.2017.2669993.
- [20] Q. Vey, S. Puechmorel, A. Pirovano, J. Radzik, Routing in aeronautical ad-hoc networks, in: *IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*, 2016, pp. 1–10. doi:10.1109/DASC.2016.7777989.

- [21] M. Y. Arafat, S. Moh, Routing protocols for unmanned aerial vehicle networks: A survey, *IEEE Access* 7 (2019) 99694–99720. doi:10.1109/ACCESS.2019.2930813.
- [22] J. Yang, K. Sun, H. He, X. Jiang, S. Chen, Dynamic virtual topology aided networking and routing for aeronautical ad-hoc networks, *IEEE Transactions on Communications* 70 (7) (2022) 4702–4716. doi:10.1109/TCOMM.2022.3177599.
- [23] K. Fuger, C. Petersen, A. Timm-Giel, Aodv-ld: Link duration based routing for multi-hop aircraft-to-ground communication, in: 2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring), 2022, pp. 1–5. doi:10.1109/VTC2022-Spring54318.2022.9860831.
- [24] J. Cui, D. Liu, J. Zhang, H. Yetgin, S. X. Ng, R. G. Maunder, L. Hanzo, Minimum-delay routing for integrated aeronautical ad hoc networks relying on real flight data in the north-atlantic region, *IEEE Open Journal of Vehicular Technology* 2 (2021) 310–320. doi:10.1109/OJVT.2021.3089543.
- [25] J. Zhang, D. Liu, S. Chen, S. X. Ng, R. G. Maunder, L. Hanzo, Multiple-objective packet routing optimization for aeronautical ad-hoc networks, *IEEE Transactions on Vehicular Technology* (2022) 1–15doi:10.1109/TVT.2022.3202689.
- [26] J. Cui, H. Yetgin, D. Liu, J. Zhang, S. X. Ng, L. Hanzo, Twin-component near-pareto routing optimization for aanets in the north-atlantic region relying on real flight statistics, *IEEE Open Journal of Vehicular Technology* 2 (2021) 346–364. doi:10.1109/OJVT.2021.3095467.
- [27] T. Bilen, P. J. Aydemir, A. E. Konu, B. Canberk, Customized k-means based topology clustering for aeronautical ad-hoc networks, in: 2021 IEEE 26th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 2021, pp. 1–5. doi:10.1109/CAMAD52502.2021.9617810.
- [28] B. Du, X. Di, D. Liu, H. Ning, Coalitional dynamic graph game for aeronautical ad hoc network formation, *IEEE Internet of Things Journal* 9 (8) (2022) 5773–5784. doi:10.1109/JIOT.2021.3076087.
- [29] T. Bilen, B. Canberk, Three-phased clustered topology formation for aeronautical ad-hoc networks, *Pervasive and Mobile Computing* 79 (2022) 101513. doi:https://doi.org/10.1016/j.pmcj.2021.101513.
URL <https://www.sciencedirect.com/science/article/pii/S1574119221001358>

- [30] T. Bilen, B. Canberk, Deliver the content over multiple surrogates: A request routing model for high bandwidth requests, *Computer Communications* 146 (2019) 39 – 47. doi:comcom.2019.07.009.
- [31] T. Bilen, D. S. Kurnaz, S. Sevim, B. Canberk, Parametric-decomposition based request routing in content delivery networks, in: *Wired/Wireless Internet Communications*, Springer International Publishing, 2018, pp. 323–335.
- [32] Nguyen Thi Xuan My, Y. Miyanaga, C. Saivichit, Link longevity-based routing mechanisms for aviation ad hoc network, in: *IEEE 16th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 2011, pp. 46–50. doi:10.1109/CAMAD.2011.5941115.
- [33] X. Zhang, P. Li, J. Hu, M. Liu, G. Wang, J. Qiu, K. W. Chan, Yenâ€™s algorithm-based charging facility planning considering congestion in coupled transportation and power systems, *IEEE Transactions on Transportation Electrification* 5 (4) (2019) 1134–1144. doi:10.1109/TTE.2019.2959716.
- [34] P. Jacob, R. P. Sirigina, A. S. Madhukumar, V. A. Prasad, Cognitive radio for aeronautical communications: A survey, *IEEE Access* 4 (2016) 3417–3443. doi:10.1109/ACCESS.2016.2570802.
- [35] J. Jiang, P. Zhang, R. Zhang, S. Chen, L. Hanzo, Aperture selection for aco-ofdm in free-space optical turbulence channel, *IEEE Transactions on Vehicular Technology* 65 (8) (2016) 6089–6100. doi:10.1109/TVT.2015.2471809.
- [36] K.-D. BÄ¼chter, Availability of airborne ad-hoc communication network in global air traffic simulation, in: *IEEE 10th International Symposium on Communication Systems, Networks and Digital Signal Processing*, 2016, pp. 1–4. doi:10.1109/CSNDSP.2016.7573927.
- [37] T. Bilen, B. Canberk, Handover-aware content replication for mobile-cdn, *IEEE Networking Letters* 1 (1) (2019) 10–13. doi:10.1109/LNET.2018.2873982.

Title of Manuscript: Q-Learning Driven Routing for Aeronautical Ad-Hoc Networks

The authors do not have any conflict of interests

Authors: Tuğçe BILEN and Berk CANBERK

Affiliation: T. BILEN was with the Department of Computer Engineering, Faculty of Computer and Informatics, Istanbul Technical University, Istanbul-Turkey and B. CANBERK was with the Artificial Intelligence and Data Engineering, Faculty of Computer and Informatics, Istanbul Technical University, Istanbul-Turkey