

Received July 24, 2021, accepted September 2, 2021, date of publication September 9, 2021, date of current version September 28, 2021. *Digital Object Identifier* 10.1109/ACCESS.2021.3111444

An Energy-Aware and Predictive Fuzzy Logic-Based Routing Scheme in Flying Ad Hoc Networks (FANETs)

SANG-WOONG LEE^{®1}, SAQIB ALI^{®2}, MOHAMMAD SADEGH YOUSEFPOOR³, EFAT YOUSEFPOOR³, POOIA LALBAKHSH^{®4}, DANIAL JAVAHERI^{®5}, (Member, IEEE), AMIR MASOUD RAHMANI⁶, AND MEHDI HOSSEINZADEH^{®1}

¹Pattern Recognition and Machine Learning Laboratory, Gachon University, Sujeonggu, Seongnam 13120, Republic of Korea

²Department of Information Systems, College of Economics and Political Science, Sultan Qaboos University, Al Khoudh, Muscat 123, Oman ³Department of Computer Engineering, Dezful Branch, Islamic Azad University, Dezful 57169-63896, Iran

⁴The College of Design and Social Context, School of Global, Urban and Social Studies, RMIT University, Melbourne, VIC 3000, Australia

⁵Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran 1477893855, Iran

⁶Future Technology Research Center, National Yunlin University of Science and Technology, Yunlin 64002, Taiwan

Corresponding authors: Amir Masoud Rahmani (rahmania@yuntech.edu.tw) and Mehdi Hosseinzadeh (mehdi@gachon.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea Government (MSIT) under Grant 2021R1F1A1053043.

ABSTRACT Today, unmanned aerial vehicles (UAVs), also known as drones, have become very popular in military applications, commercial applications, and academic research. Flying ad hoc network (FANET) is a new type of ad hoc network, which groups small drones into an ad hoc form. These networks have unique characteristics, including moving in a 3D space, high mobility, frequent topological changes, limited resources, low density of nodes, and so on, which impose various challenges when designing a proper and efficient routing scheme. In this paper, we present a fuzzy logic-based routing scheme for flying ad hoc networks. The proposed routing scheme has two phases: route discovery phase and route maintenance phase. In the first phase, we propose a technique for calculating the score of each node in the network to prevent the broadcast storm problem and control the flood of the control messages, which have been broadcast to discover a new route in the network. This score is calculated based on various parameters such as movement direction, residual energy of nodes, link quality, and node stability. Moreover, in the route selection process, we design a fuzzy system to select routes with more fitness, less delay, and fewer hops for data transfer. The second phase includes two steps: preventing route failure in order to detect and modify paths at the failure threshold, and reconstructing failed routes in order to recognize and quickly replace these routes. Finally, the proposed routing scheme is implemented in NS2 to evaluate its performance and determine its efficiency. The simulation results are compared with three routing methods, namely ECaD, LEPR, and AODV. These results show that the proposed routing method outperforms other routing schemes in terms of end to end delay, packet delivery rate, route stability, and energy consumption. However, it has slightly increased the routing overhead.

INDEX TERMS Flying ad hoc network (FANET), routing, fuzzy logic, unmanned aerial vehicle (UAV), artificial intelligence (AI).

I. INTRODUCTION

In the last decade, the use of unmanned aerial vehicles (UAVs) has growth significantly for various applications and services. When UAVs are organized as connected groups

The associate editor coordinating the review of this manuscript and approving it for publication was Nafees Mansoor¹⁰.

in the ad hoc form, they operate successfully in complex missions, and create a flying ad hoc network (FANET) [1], [2]. FANET is a new type of mobile ad hoc network (MANET), so that its nodes are drones. In these networks, all UAVs create an ad hoc network and only a subset of them connect to the ground station (GS) [3]. FANET is a multi-UAV system in which UAVs carry out missions collaboratively

and cooperatively. Therefore, it is faster and more efficient than single-UAV systems [4], [5]. Compared to single-UAV systems, multi-UAV systems are more useful in terms of reliability and survivability via redundancy. This means that if one of the UAVs fails in a mission, other drones can continue this mission. These networks have many applications in various fields, including military [6], forest fire monitoring, search and rescue operations [7], [8], reconnaissance operations, transportation, etc [9], [10]. Flying ad hoc networks are a subset of vehicular ad hoc networks (VANETs) and mobile ad hoc networks (MANETs). FANETs share some features with these ad hoc networks, such as the self-organization ability and communication between nodes without centralized infrastructure. However, they have their own characteristics [11]. In the following, some of these features are described briefly.

- The movement of nodes (UAVs) in FANET is different from their movement in MANET and VANET. Usually, their speeds are almost equal to 30-460 km/h in FANET. It means that their mobility degree and speed changes are really higher than those in MANET and VANET. Furthermore, UAVs move in a 3D space [12], [13]. In contrast, nodes in the MANET and VANET move in a 2D space. In FANET, links are quickly created and failed due to changing the position of UAVs, the distance between them, and their topology [14]. Moreover, the UAV addition and removal (due to energy loss and malfunction, etc.) affect the network topology. It has a negative effect on network connections and causes disconnected links when sending important information such as control and traffic messages. As a result, this dynamics causes frequent organizing of the network [4], [5].
- The number of nodes in FANET is less than those in MANET and VANET. It is due to the long distance between UAVs. Sometimes, this distance reaches several kilometers and requires a longer transmission range. As a result, it affects radio links, antenna structure, and physical layer behavior [5], [15].
- In real-time applications, data packets must be delivered to the destination with a slight delay in order to avoid collisions between UAVs [15].
- Compared to nodes in other ad hoc networks, UAVs in FANET have sufficient computational power and energy. This is because the energy required to fly a UAV is more than the energy required to process data. However, it can be a serious problem for small drones with limited energy capacity [4], [15].
- In FANET, it is very difficult to determine the exact position of UAVs due to their high mobility degree. As a result, their position information must be updated at the short time interval [15].

Considering the specific characteristics of these networks, it is very challenging to present an efficient routing scheme for FANETs [4]. Routing means sending and receiving information between source node and destination node in the network [16], [17]. Routing helps FANETs to maintain their applications and services stably and actively [18], [19]. In FANET, there are several types of routing methods according to their strategy in the routing process: topology-based routing schemes, geographical routing schemes, hybrid routing schemes (a combination of topology-based and geographical routing schemes) and nature-based routing schemes [5], [19]. In the following, we describe each of these routing schemes.

• *Topology-Based Routing Protocol:* This routing type uses topology information of nodes to send data packets in the network. In these protocols, it is necessary to create a proper route from the source to the destination before starting the data transmission process. Topology-based routing schemes are categorized as proactive and reactive [5], [20].

A. PROACTIVE ROUTING SCHEME

It is also known as table-driven protocol. In this routing type, each UAV exchanges the latest route information with other UAVs in the network, regardless of whether they have data packets to send or not. Route information is stored in the routing table of each UAV [20]. These tables is periodically updated and route information is shared between UAVs in the network. Proactive routing is not suitable for FANETs. Because it is weak against frequent topological changes. As a result, it has a high degree of disconnected links [21], [22].

B. REACTIVE ROUTING SCHEME

It is also known as on demand protocol. In this routing type, if an UAV is to send its data packets and there is no route to the destination node in its own routing table, then it initiates the route discovery process. In these routing protocols, only information about active routes is stored in the routing table. Therefore, a route maintenance mechanism is used to maintain valid routes and remove invalid routes [12], [23]. When the network topology is changed, the failed routes are deleted and the route discovery process is restarted. In this scheme, routing tables are periodically updated. As a result, bandwidth will be more efficient in reactive routing protocols compared to proactive routing protocols [20], [22].

- Geographical Routing Protocol (Location-Based): These protocols use spatial position information of flying nodes to perform routing operations on the network. In geographical routing protocols, it is important to estimate the location of the destination node using the information obtained from a positioning system or location prediction techniques to perform the routing process efficiently [5], [20].
- *Hybrid Routing Protocol:* These protocols integrate geographic and topology-based routing mechanisms to take the advantage of both [5], [20].
- Nature-Inspired Routing Protocol: These protocols originate from natural phenomena such as ant colony, bee colony, bird swarm and so on to improve the routing process. However, they have weaknesses in terms

of operational capability such as increased delay, high communication overhead, and increased energy consumption in the network [20]. In fact, nature-inspired routing schemes are a subset of topology-based routing methods, especially reactive routing [22], [24].

According to the content mentioned above, traditional routing algorithms cannot meet the requirements of FANETs. For this reason, many researchers focus on designing routing schemes for FANET. When designing routing schemes, it is very important to select an appropriate route. This is an important issue to communicate two node in FANET. However, it is always difficult to choose a benchmark to decide on the best path. When designing routing schemes in FANETs, they must consider criteria like, efficient use of network resources, energy saving, not having loop, reconstruction capability, and scalability [25]. Moreover, in FANET, routing protocols must be efficient, meaning that they should have low overhead, high reliability, low packet loss rate, acceptable delay, and appropriate stability. Although, achieving all the criteria mentioned in a routing method is very difficult or even impossible.

In this paper, we seek to address some of the mentioned problems by proposing an efficient routing scheme in FANET. In this paper, our focus is to solve two main problems, i.e. failing the communication routes and flooding the routing messages in the network. The first problem is due to high mobility of nodes in the network and can increase the packet loss rate (PLR) and delay in the data transmission process in the network. The second problem is due to the fact that most of the AODV-based routing algorithms flood routing messages in the network to find paths to the desired destination. This can lead to a broadcast storm. In this paper, we present a fuzzy logic-based routing scheme in FANETs. In our proposed method, we propose two main approaches to solve the first problem. The first is to create stable paths in the route discovery process and the second is to find the paths, which are at the failure threshold, and correct them rapidly. Also, we improve the AODV routing protocol and limit the nodes, which broadcast routing messages, to prevent the broadcast storm. In the proposed routing method, we consider various parameters, including link quality, node stability, and energy of UAVs in the routing process to establish stable routes and increase packet delivery rate (PDR). Furthermore, we seek to balance energy consumed by flying nodes in our scheme to increase the network lifetime. The proposed routing method has two phases: 1) the route discovery process 2) the route maintenance process.

• *Route Discovery Process:* This process starts its operations when the source node has a data packet, which must be sent to the destination node, and there is no route in its routing table. In this condition, the source node broadcasts a route request message (RREQ) to its neighboring nodes. Upon receiving the RREQ message, neighboring nodes calculate their score based on various parameters such as movement direction, residual energy, link quality, and node stability. If they obtain the desired score, then they rebroadcast the RREQ message. Otherwise, they delete this message. We believe that this solution improves the performance of our routing method and prevent the broadcast storm problem. Moreover, we have designed a fuzzy system to select routes with more fitness, less delay, and fewer hops for data transfer.

• *Route Maintenance Process:* This phase includes two steps: preventing the route failure and reconstructing the failed routes. In the first step, the purpose is to detect and correct routes at the failure threshold. This helps to avoid interruption in the data transmission process. In the second step, the purpose is to quickly recognize and replace failed routes to reduce delay in the data transmission process.

In the following, the paper is organized as follows: In Section II, some routing schemes in FANETs are reviewed. Section III introduces the basic concepts used in the proposed routing scheme. Furthermore, the system model applied in the proposed scheme is presented in Section IV. Section V details the formats of the control routing messages namely, RREQ and RREP, and data packet format. In Section VI, the proposed routing method is explained in detail. Section VII presents the simulation results of our proposed routing scheme. Finally, the conclusion and the future research direction are stated in Section VIII.

II. RELATED WORKS

Oubbati et al. [26] proposed a routing scheme called energyefficient connectivity-aware data delivery (ECaD) for FANETs. It is a desirable scheme based on AODV. However, ECaD has some weaknesses. We can solve these weaknesses to improve the performance of this scheme. In ECaD, UAVs with high energy levels participate in the route discovery process and less-energy nodes are forbidden from participating in the data delivery process. Its purpose is to create high-energy routes, but it has a weakness: if the energy of all neighbors of a node is less than the predetermined threshold, this node cannot establish any routes with other nodes. As a result, it is isolated in the network. It seems that choosing a predefined threshold value is not suitable in the network. Therefore, it is better to determine this value dynamically and based on network conditions. Moreover, in ECaD, decision-making for rebroadcasting the RREQ message is based on only one parameter i.e. energy of nodes. This is not sufficient. It must be done based on different factors. In addition, this scheme may be faced with a broadcast storm problem. The authors ignore this problem in their paper. We have considered these issues in our paper. This method takes into account the stability of the routes, balanced energy consumption, route failure prediction, and route connectivity expiration time. Moreover, ECaD has been implemented in a 3D space. Hence, it is compatible with FANET. However, its design does not consider the movement direction of UAVs in the route selection process. It is very important to consider the movement direction of flying nodes to establish more stable routes so that UAVs move in almost same direction in a route. In addition, the link quality parameter is not considered in the route discovery process. When this parameter is ignored, lowquality paths may be created and quickly failed. This makes interruptions in the data transmission process. Furthermore, ECaD does not consider the node stability in the route discovery process.

Perkins et al. [27] suggested the ad hoc on-demand distance vector (AODV) routing protocol. It has a very important feature, namely on-demand, meaning that the route discovery process is done only when there is a route request. Today, many AODV-based routing algorithms have been proposed for FANET. Therefore, AODV is still a valuable algorithm for evaluating many routing schemes. However, it is very challenging to use AODV algorithm in FANETs because this protocol is not compatible with the unique features of these networks, including high-speed flying nodes, frequent disconnection of communication links, and so on. Moreover, in this scheme, the route discovery process has high delay and may cause a broadcast storm problem that weakens the network performance. In AODV, when the source node is to send a data packet to the destination node and it does not find no valid route to the destination in its routing table, then the source node broadcasts a route request message (RREQ) to its neighbors. Then, neighboring nodes broadcast RREQ to their neighbors as well. This process continues until RREQ reaches the destination node or an intermediate node, which has a valid path to the destination node. After receiving RREQ by the destination node or the intermediate node, it unicasts a route reply message (RREP) to the neighboring node, which has sent RREQ for the first time. When RREP is sent back through the reverse route, nodes in this route update their routing tables and add a new entry to record the route information in this table, if a new route has been discovered. In AODV, a route maintenance mechanism is provided to detect failed routes. AODV has a major weakness: when the network size is large, the nodes may experience a high delay in the route discovery process. In addition, if a link failure occurs during route discovery process, then it causes the increased delay and bandwidth consumption.

Darabkh et al. [28] introduced an AODV-based routing scheme called the multi data rate mobility aware (MDRMA) protocol for FANETs. MDRMA is originated from the mobility aware dual phase ad-hoc on-demand distance vector with adaptive Hello messages (MADP-AODV-AHM) protocol. This scheme includes two algorithms: routing algorithm and power controlled algorithm. In MDRMA, the first algorithm aims to reduce the flood of RREQ messages and accelerate the data transmission process based on the distance between nodes in the network. Furthermore, it selects intermediate nodes in a route, based on their data transmission rate, speed, and movement direction to establish stable and reliable routes. The data transmission rate is selected according to a mathematical model, which is based on the pointto-point connectivity of a node with its neighboring nodes, the density of nodes in the network and their transmission

range. The power controlled algorithm aims to control the data transmission power on each wireless link in the route. MDRMA is a desirable method that can be modified in the future to improve its performance. It should be noted that this method does not consider the energy of UAVs in the route discovery process. It is a very important issue in small UAVs. Moreover, MDRMA does not design any mechanism for preventing route failure. Hence, it may increase the route failure probability. In addition, MDRMA ignores delay in the route selection process. It should also be noted that this routing method is implemented in a 2D space. Therefore, it is not compatible with the FANET environment.

Deshpande et al. [29] presented the stochastic multipath UAV routing (SMURF) for FANETs. SMURF is a centralized algorithm implemented by the control center, which sends the route information to UAVs through a softwaredefined networking (SDN) platform. SMURF is a multipath scheme, which increases fault tolerance and reliability because it always provides at least one stable route. In this method, the primary route is obtained based on the highest link existence probability. Then, a number of backup routes are determined to increase the route reliability if the primary route fails. Backup routes are selected using two parameters: the failure probability of the links in the first route and their reliability. This method has a high communication overhead. SMURF has a major disadvantage, meaning that it is a centralized algorithm, which is not suitable for FANET due to its characteristics such as high mobility, frequent topological changes, etc. SMURF is suitable for dense networks, while FANET is a sparse network. SMURF ignores the unique features of FANET because it does not consider parameters such as the energy of UAVs, movement direction, link quality, node stability, delay, and the number of hops during the route construction process between UAVs. Moreover, SMURF has no process to prevent route failure.

Li et al. [30] proposed a link stability estimation-based preemptive routing (LEPR) protocol for FANETs. LEPR is inspired by AODV. It introduces a new link stability scale based on location information of drones obtained from GPS. This scale has three parameters: link quality, safety degree, and mobility prediction factor. The link stability scale can determine the past, current, and future states of link stability. LEPR modifies the route discovery mechanism in AODV and calculates routes using the link stability scale. Moreover, a semi-proactive route maintenance process is applied when a link is disconnecting. This process reduces the number of failed routes and delay in the data transmission process through finding and switching to a reliable route. In LEPR, the main idea is to calculate several stable routes using the proposed link stability scale. LEPR is a suitable scheme, which takes into account features of FANET in the route discovery process. Furthermore, this scheme is implemented in a 3D space and is compatible with FANET. However, LEPR ignores important parameters such as delay, energy of UAVs, node stability, and the number of hops in the route discovery process.

Aadil et al. [31] offered a routing scheme called the energy-aware link-based clustering (EALC) model in FANETs. EALC seeks to minimize communication and computational overhead. In this scheme, a simple clustering mechanism is designed to reduce computational overhead and communication overhead and increase network lifetime. In this routing scheme, the optimal transmission range of drones is selected according to the network requirement, i.e. the minimum communication range of a node for efficient communication. This reduces the packet loss rate (PLR) and improves energy consumption. In EALC, nodes are first grouped using the K-means sorted fitness algorithm so that the communication between the nodes is done using cluster head nodes (CHs). In a cluster, cluster member nodes (CMs) transmit their data to CH, which is responsible for transferring data to the destination node. CHs are selected based on two parameters: their energy level and their distance from neighboring nodes. When the fitness of a CH is less than the threshold (i.e. it is 20% less than the fitness of CMs), then the clustering process is recalled. However, EALC does not consider the movement of UAVs and the quality of communication links in the routing process. Therefore, unstable paths may be established. As a result, route failure probability is very high. Moreover, the routes may have a high delay in this scheme. In addition, EALC does not design a mechanism to prevent route failure. Furthermore, this scheme is implemented in a 2D space and is not compatible with the FANET environment.

Gankhuyag et al. [32] proposed the robust and reliable predictive routing (RARP) in FANETs. It combines omnidirectional and directional transmission schemes and utilizes dynamic angle adjustment. RARP enhances AODV to be suitable for FANETs. In this scheme, when the source node wants to find a route to the destination node, it broadcasts an RREQ message to the neighboring nodes using the omnidirectional transmission scheme. After an intermediate node receives RREQ, it rebroadcasts this message until it reaches the destination node. When the destination node receives RREQ, it waits for a certain time interval. If the destination node receives several RREQs, it selects the best route based on a utility function. This function includes three parameters: flying risk, number of hops, and expected connection time. Then, the destination node unicasts the RREP message to the source node. It should be noted that in this scheme, RREP is sent back using a directional antenna with the angle θ in the direction of the estimated position of the receiver. RARP can determine the minimum expected connection time between two nodes in a route. Therefore, it can estimate the route lifetime. Thus, it calculates an alternative route before the minimum expected connection time expires. RARP is also implemented in a 3D space. Therefore, it is compatible with the FANET environment. However, it has a major drawback, meaning that it does not consider the quality of communication links, the energy of UAVs, and their movement direction, which may negatively affect the route stability.

Liu et al. [33] suggested a Q-learning based multiobjective optimization routing protocol (QMR) for FANETs. QMR uses a Q-learning based exploration and extraction mechanism to discover the optimal route and reduce delay and energy consumption. This scheme utilizes the current information to estimate future information about the relationships between neighboring UAVs to select the next-hop node reliably and stably. In the routing process, nodes use Hello messages to share information about their location, energy, and mobility model (speed and direction of flying nodes) as well as delay. QMR employs these parameters and the link quality information in the route discovery process to create stable routes. It should be noted that, Q-learning parameters, including learning rate and discount factor are adjustable in QMR, so that each link has a different learning rate and each node includes its own specific discount factor. They are determined based on the network condition. As a result, QMR has appropriately adapted with the dynamic FANET environment. If these parameters are considered constant, the accuracy of the routing process decreases and the route failure probability increases. In addition, QMR has designed a route failure prediction mechanism in the routing process. However, this scheme is implemented in a 2D space that is not compatible with the FANET environment. Also, QMR has not provided a proper mobility control mechanism for controlling swarm connectivity.

Table 1 presents the most important advantages and disadvantages of different routing schemes.

III. BASIC CONCEPTS

In the proposed scheme, we have used fuzzy logic in the route discovery process. Therefore, we briefly describe fuzzy logic in this section.

A. FUZZY LOGIC (FL)

According to research and studies, it can be deduced that real and complex processes cannot be accurately measured, modeled, and controlled because there are uncertainties such as incompleteness, randomness, and data loss. Fuzzy logic (FL) is a mathematical technique that approximately describes human thinking. In 1965, the FL theory was first presented by Zadeh in a paper entitled "Fuzzy Sets". In classical sets, membership is defined accurately and unambiguously that is, whether an element either belongs to a set or not. Fuzzy sets provide partial membership. In other words, an element may belong to a set to some extent. Therefore, the results are not absolutely limited to True or False, but they are partially True or False [34].

Assume that X (universal set) is a set of elements, which are generally represented by x, then fuzzy set A in X is defined based on Equation 1.

$$A = \{(x, \mu_A(x)) | x \in X\} = \sum_{i=1}^n \mu_A(x_i) / x_i$$
(1)

where, $\mu_A : X \to [0, 1]$ is called the membership function *A*, $\mu_A(x_i)$ is the membership degree of x_i in *A*.

TABLE 1. Comparison of advantages and disadvantages of different routing schemes.

Number	ber Scheme Advantages		Disadvantages			
1	ECaD [26]	Establishing stable routes, balanc-	Considering a fixed threshold value			
		ing energy consumption, designing	for the energy parameter, possibil-			
		a mechanism for preventing route	ity of a broadcast storm, broadcast-			
		failure, implementing in a 3D space	ing control messages only based on			
			the energy parameter, ignoring the			
			movement direction, link quality,			
			and distance in the route selection			
	4.0001/1071		process			
2	AODV [27]	On-demand routing process, de-	Not considering the specific fea-			
		signing a route maintenance mech-	tures of FANET, High delay and			
			route discovery process not de			
			signing a mechanism for preventing			
			route failure possibility of a broad-			
			cast storm			
3	MDRMA [28]	Creating stable routes, considering	Ignoring the energy of nodes in			
		data transmission rate, speed, and	the routing process, not designing			
		movement direction of nodes in the	a mechanism for preventing route			
		routing process, designing an al-	failure, not considering the delay			
		gorithm to control data transmis-	parameter in the route selection pro-			
		sion power, reducing the flooding of	cess, implementing in a 2D space			
		control messages				
4	SMURF [29]	Creating several routes between the	High communication overhead,			
		source node and the destination	proposing a centralized routing			
		roliability	method, not considering the			
		renability	not designing a mechanism for			
			preventing the route failure the			
			possibility of unstable paths.			
			implementing in a 2D space			
5	LEPR [30]	Designing a semi-proactive route	High routing overhead, Ignoring			
		maintenance process, creating sta-	delay, energy and hop count in the			
		ble routes, implementing in a 3D	routing process			
		space				
6	EALC [31]	Reducing communication overhead	Ignoring the movement direction			
		and computational overnead, ad-	and link quality in the routing pro-			
		justing the transmission range of	route and route failure not design			
		ments reducing packet loss rate	ing a mechanism for preventing the			
		improving energy consumption	route failure implementing in a 2D			
			space			
7	RARP [32]	Designing a mechanism for modi-	Not considering the link quality,			
		fying the routes, implementing in a	movement direction, and energy the			
		3D space	route discovery process, the possi-			
			bility of unstable paths			
8	QMR [33]	Reducing delay and energy	Implementing in a 2D space, not			
		consumption, creating stable and	designing a proper mobility control			
		reliable paths, adjusting the Q-	mechanism			
		Learning parameters dynamically,				
		designing a mechanism for				
		preventing route failure				



FIGURE 1. A fuzzy system.

In Equation 1, "/" does not refer to a division, but it is used to separate the membership degree of an element from the element itself. A fuzzy set is expressed using its membership function. The most common membership functions used to represent fuzzy sets are triangular, trapezoidal, and Gaussian functions.

Today, fuzzy inference mechanisms have many applications. The most common fuzzy systems are Mamdani fuzzy inference and Sugeno fuzzy inference (TSK). As shown in Figure 1, a fuzzy system includes four main modules: fuzzification, defuzzification, fuzzy rule base, and fuzzy inference engine. The fuzzification module maps the system inputs to the corresponding fuzzy sets and assigns one membership degree to each fuzzy set. The fuzzy inference engine implements these fuzzy values through fuzzy rules stored as IF-THEN rules. Its outputs are in the form of fuzzy variables. They must be converted to crisp values using a defuzzification module such as the averaging or centroid schemes [34].

IV. SYSTEM MODEL

In this section, we introduce the network model used in the proposed routing scheme. In our proposed routing scheme, we consider a homogeneous FANET. The network includes a number of UAVs distributed in a three-dimensional space. The IEEE 802.11a standard is considered as the wireless interface in the MAC layer of each UAV because it can efficiently support highly dynamic topologies and provide wide coverage of wireless communications. In the network, UAVs are moving and the distance between them changes over time. Each UAV has a unique identifier (ID_{IIAV_i}) . Furthermore, it is assumed that UAVs are equipped with the global positioning system (GPS). As a result, UAV_i is aware of its position (x_i, y_i, z_i) and its speed $(v_{x,i}, v_{y,i}, v_{z,i})$ at every moment. We assume that the speed of UAVs is limited to $[0, V_{MAX}]$; where, $V_{MAX} > 0$ is a constant value. In addition, our proposed routing scheme supports two communication types in the network:

 UAV-to-UAV Communication (U2U): In this communication type, UAVs communicate with each other to perform common missions, such as routing or tracking a target. The U2U communication may be single-hop or multi-hop. Moreover, this communication type may be short-range or long-range to improve FANET performance in terms of data rate and communication range. • UAV-to-GS Communication (U2G): In this communication type, the UAV communicates with the ground station (GS) to process the received information. In the proposed routing scheme, all UAVs are not directly connected to GS. Only UAVs, which are close to GS, can be directly connected to it.

It should also be noted that we have used the air-to-air (A2A) channel model in the proposed routing scheme [35]. It can be defined based on the free-space propagation model because the packet loss is high in less fading. As a result, path loss in A2A channel is expressed according to Equation 2 [35]:

$$PLAA(d_{ij}) = \beta 10 \log_{10}^{d_{ij}} + \alpha$$
⁽²⁾

where, β indicates the path loss exponent, so that $\beta = 2$ in the free-space propagation model. Furthermore, α is the path loss at the reference point. In the free-space propagation model, α is obtained using Equation 3:

$$r = 10\log_{10}^{\left(\frac{4\pi w}{l}\right)} \tag{3}$$

where, w indicates the carrier frequency and l represents the light speed so that $l = 3 \times 10^8 \ ^m/_s$. Also, d_{ij} is the distance between UAV_i and UAV_j . It is calculated using Equation 4:

0

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$$
(4)

where, (x_i, y_i, z_i) and (x_j, y_j, z_j) are spatial coordinates of UAV_i and UAV_j , respectively. The network model is shown in Figure 2.

V. TYPES OF MESSAGES IN THE PROPOSED SCHEME

In the proposed routing scheme, we modify the format of two control messages i.e. route request (RREQ) and route reply (RREP) in AODV. In the following, we introduce the format of these messages and their different fields in detail. Furthermore, the data packet structure is explained in this section.

A. ROUTE REQUEST (RREQ) MESSAGE

It is used in the route discovery process. Table 2 illustrates the RREQ format. In the following, we introduce the most important fields of this message:

• *Message Type:* If the *Message Type* field is equal to one. This means that this control message is an RREQ message.



FIGURE 2. Network model in our proposed scheme.

TABLE 2. The RREQ message format.

Message Type	Hop _{Count}	F _{Route}	Delay _{Route}		
RREQ Message ID					
Destination IP Address					
Destination Sequence Number					
Source IP Address					
Source Sequence Number					

- *Hop_{Count}*: When the source node sends the RREQ message, it adjusts the *Hop_{Count}* field on zero. Then, this field is added one unit at each hop. The purpose of this field is to prevent routing loops in the network.
- *RREQ Message ID*: It is a unique ID. Two fields, namely *RREQ Message ID* and *Source IP Address* are used to check RREQs and prevent duplicate messages.
- F_{Route} : This field indicates the route fitness, which has a value in [0, 1]. In fact, F_{Route} is equal to the lowest score of UAVs in a route. The source node sets F_{Route} to one. Then, its value is updated at each hop.
- *Delay_{Route}*: This field indicates the route delay. Initially, the source node sets the *Delay_{Route}* field to zero. Then, its value is updated in each hop and the time required to transfer the RREQ message from the current hop to the next hop is added to this field.
- *Source IP Address:* This field represents the source UAV's address, which sends the RREQ message.
- Destination IP Address: This field represents the destination UAV's address where a route must be discovered.

- *Source Sequence Number:* This field is used to ensure that the reverse route information to the source node is fresh.
- *Destination Sequence Number:* This field is used to ensure that the route discovered to the destination node is new before the source UAV selects a route.

B. ROUTE REPLY (RREP) MESSAGE

After the RREQ message reaches the destination node; this UAV generates an RREP message and sends back it to the source node. The format of this message is stated in Table 3. The RREP fields are similar to the RREQ message, which we explain each of them. Note that the *Message type* field must be equal to 2 in an RREP message.

TABLE 3. The RREP message format.

Message Type	Hop Count				
RREP Message ID					
Destination	IP Address				
Destination Sequence Number					
Source IP Address					
Lifetime					

C. DATA PACKETS

These packets include the data, which the source UAV wants to send to the destination UAV through a route. Table 4 presents the data packet format. In the following, we describe the fields of the data packet:

TABLE 4. Data packet structure.

Message ID	Source IP Address				
Data packet content					

- *Destination IP Address:* This field represents the destination UAV address where the data packet must be sent.
- Source IP Address: This field represents the source UAV address, which sends the data packet.
- *Message ID*: It is a unique ID. *Message ID* and *Source IP Address* are applied to control the data packet and avoid the duplicated data.

VI. PROPOSED ROUTING SCHEME

In this paper, we propose a fuzzy logic-based routing scheme for FANETs. Our method improves AODV. Note that AODV is one of the most efficient routing protocols introduced for ad hoc networks. In the proposed routing scheme, we seek to take into account various parameters, such as link quality, node stability, and energy of node, in the routing process so that stable routes are created and packet delivery rate (PDR) are improved. Furthermore, we try to enhance the network lifetime through balancing the energy consumption of the nodes. Our proposed routing scheme has two phases: 1) the route discovery process 2) the route maintenance process.

Phase 1, Route Discovery Process: This process starts its operations when the source node wants to send data packets to the destination node and there is no route to it in its routing table. In such cases, the source node broadcasts a route request (RREQ) message to its neighboring nodes. After receiving this message, neighboring nodes calculate their scores based on various parameters such as movement direction, residual energy, link quality, and node stability. Next, if they obtain the required score, then they can rebroadcast the RREQ message. We believe that this solution can improve the performance of our routing scheme and prevent the broadcast storm problem. Moreover, we designed a fuzzy system in the route selection step to choose routes with more fitness, less delay, and fewer hops for data transfer.

Phase 2, Route Maintenance Process: This phase includes two steps: preventing the route failure and reconstructing the failed routes. In the first step, the purpose is to detect and correct routes on the failure threshold. This helps to avoid interruption in the data transmission process in the network. In the second step, the purpose is to quickly recognize and replace failed routes to reduce delay in the data transmission process.

In the following, each of these phases is explained in detail. Also, Table 5 lists the symbols used in the proposed method.

A. ROUTE DISCOVERY PHASE

When the source node (UAV_S) is to communicate with the destination node (UAV_D) to send data packets, UAV_S first searches its routing table to find a valid route to UAV_D . If UAV_S does not find such route in its routing table, then



FIGURE 3. Calculating the movement direction of UAV_i relative to UAV_{prev}.

it should start the route discovery process. Therefore, it generates an RREQ message and broadcasts this message to its neighboring nodes. After a neighboring node receives RREQ, it must calculate its score before rebroadcasting the message. This score is calculated based on various parameters such as movement direction, residual energy, link quality, and node stability. We believe that this approach enhances the performance of our scheme and prevent the broadcast storm problem. In the following, we explain how to calculate the score of nodes in detail. Then, we provide an example to describe the route discovery process.

1) SCORE CALCULATION PROCESS FOR EACH NODE

In the route discovery process, it is necessary for each node to calculate its score, S_i relative to the previous-hop node (UAV_{prev}) . In this section, we describe how to calculate this parameter. S_i is obtained using four parameters: movement direction, residual energy, link quality, and node stability. It should be noted that all these parameters are normalized in [0, 1] to have the same effect on S_i . In the following, each of these parameters is described in detail.

• *Movement Direction* (λ_i) : The purpose of selecting this parameter is to select the next-hop node (UAV_{next}) from the UAVs, which received the RREQ message, so that the selected node moves in the same direction as the previous-hop node (UAV_{prev}) . In this case, UAV_{next} and UAV_{prev} can communicate with each other for more time. As a result, they create more stable routes. Therefore, this node earns a higher score than other UAVs for rebroadcasting the RREQ message. We present an example in Figure 3 to calculate the movement direction of UAV_{i} relative to UAV_{prev} . In this figure, the speed vector and movement direction of both nodes are represented. As shown in this figure, these two nodes move in different directions.

To calculate the movement direction of UAV_i relative to UAV_{prev} , we first obtain the angle between the speed

TABLE 5. The table of symbols.

Symbol	Description			
UAV	Unmanned aerial vehicle			
(x_i, y_i, z_i)	Spatial coordinates of UAV _i			
$(v_{x,i}, v_{y,i}, v_{z,i})$	Speed of <i>UAV</i> _i			
UAVS	Source node			
UAVD	Destination node			
RREQ	Route request message			
RREP	Route reply message			
<i>Hop_{Count}</i>	The number of hops in a route			
F _{Route}	Route fitness			
<i>Delay_{Route}</i>	Route delay			
S_i	The score of UAV_i			
UAV _{prev}	Previous-hop node			
UAV _{next}	Next-hop node			
λ_i	Movement direction			
$ec{V}_i$	Speed vector of <i>UAV_i</i>			
θ	The angle between UAV_i and UAV_{prev}			
E_{UAV_i}	Residual energy of UAV _i			
E_{Min}	Minimum residual energy of nodes			
E_{Max}	Maximum energy of nodes			
$Q_{Link_{i-prev}}$	Quality of the link between UAV_i and UAV_{prev}			
Q_{Min}	Minimum link quality			
Q_{Max}	Maximum link quality			
Stability _{UAVi}	Stability of <i>UAV</i> _i			
D _{trust}	A suitable distance between UAV_i and UAV_{prev}			
R	Communication range of UAVs			
Best _{Route}	A route with more fitness, less delay and fewer hops			
Tr_{UAV_i}	Data traffic in UAV_i			
Tr _{Threshold}	Maximum traffic in a node			
Warning Message	A message to determine that UAV_i is located on the failure threshold.			
Route Recovery Message	A message to correct a route located on the failure threshold			
Route Validation Message	A message to check a route			
ACK Message	A message to determine that a route is valid			
RERR Message	A message to determine that a route is invalid			



FIGURE 4. The angle between the speed vectors of two nodes.

vector of UAV_i (\vec{V}_i) and the speed vector of UAV_{prev} (\vec{V}_{prev}). This angle is called θ , which is also shown in Figure 4. Based on mathematical rules, we know

that the dot product (inner product) of two vectors is geometrically equal to:

$$\vec{V}_{i}.\vec{V}_{prev} = \left|\vec{V}_{i}\right| \left|\vec{V}_{prev}\right| \cos\theta \tag{5}$$

where,

$$\left|\vec{V}_{i}\right| = \sqrt{v_{x,i}^{2} + v_{y,i}^{2} + v_{z,i}^{2}}$$
 (6)

And,

$$\vec{V}_{prev} = \sqrt{v_{x,prev}^2 + v_{y,prev}^2 + v_{z,prev}^2}$$
(7)

On the other hand, the dot product of two vectors is algebraically calculated using Equation 8:

$$\vec{V}_{i}.\vec{V}_{prev} = v_{x,prev}v_{x,i} + v_{y,prev}v_{y,i} + v_{z,prev}v_{z,i}$$
(8)

Based on Equations 5, 8, the angle θ is obtained using Equation 9:

$$\theta = \cos^{-1} \left(\frac{v_{x, prev} v_{x,i} + v_{y, prev} v_{y,i} + v_{z, prev} v_{z,i}}{\left| \vec{V}_i \right| \left| \vec{V}_{prev} \right|} \right),$$

$$0 \le \theta \le \pi$$
(9)

Now, based on the angle θ between V_i and V_{prev} , we can calculate the movement direction of UAV_i relative to UAV_{prev} , that it is represented as λ_i . As a result, λ_i is calculated using Equation 10:

$$\lambda_{i} = \begin{cases} 1, & \theta = 0\\ \frac{\pi - \theta}{\pi}, & 0 < \theta < \pi\\ 0, & \theta = \pi \end{cases}$$
(10)

According to Equation 10, whatever λ_i is close to one, meaning that UAV_i and UAV_{prev} are moving in the same direction.

• Residual Energy (E_{UAV_i}) : Energy is an important parameter in the route discovery process. If low-energy nodes participate in the route construction, then unstable routes are established so that their failure probability is very high. On the other hand, this can lead to unbalanced energy consumption in the network. As a result, network lifetime is reduces. It should be noted that each node (i.e. UAV_i , where, i = 1, ..., N and N indicates the total number of nodes in the network) is aware of its energy (E_{UAV_i}) at any time. If E_{UAV_i} is less than E_{Min} , then UAV_i does not gain the desirable score to participate in the route discovery process. We consider E_{Min} equal to 20% of the initial energy of nodes. The parameter E_{UAV_i} is normalized according to Equation 11:

$$E_{norm-UAV_i} = \frac{E_{UAV_i} - E_{Min}}{E_{Max} - E_{Min}}$$
(11)

where, $E_{Min} \ge 0$ is the minimum energy and $E_{Max} > 0$ is the initial energy of UAVs in the network.

• Link Quality $(Q_{Link_{i-prev}})$: The purpose of this parameter is to select nodes, which have a higher link quality than other nodes, in the route discovery process. If the quality of the link between two nodes in a route is weak, the route failure probability will be very high. On the other hand, this can also increase the data packet loss rate (PLR). In the proposed routing scheme, the quality of communication links is determined based on the received signal strength indication (RSSI). RSSI can provide a precise and rapid estimation of the link quality [36]. In research, it has been proven that the higher values of RSSI will lead to a better PDR in the receiver and the transmitter [37]. In addition, research proves that RSSI is stable (i.e. standard deviation is less than 1dBm) for a short period of time (about 2 seconds) [38]. Therefore, RSSI information can be used to estimate the link quality. Often radio transceivers are equipped with a RSSI register, which provides the signal strength of the received packet. Therefore, nodes can

obtain RSSI information when receiving the RREQ message from their neighbor nodes. To estimate link quality $(Q_{Link_{i-prev}})$ between the two nodes, namely UAV_i and UAV_{prev} , we use the averaging technique on RSSI values to prevent the effect of temporary peaks. $Q_{Link_{i-prev}}$ is normalized based on Equation 12:

$$Q_{norm-Link_{i-prev}} = \frac{Q_{Link_{i-prev}} - Q_{Min}}{Q_{Max} - Q_{Min}}$$
(12)

where, $Q_{Min} \ge 0$ indicates the minimum quality of communication link between two nodes and $Q_{Max} > 0$ is the highest quality of communication link between two nodes in the network. According to [36], the RSSI value varies between 0 and R_{Max} , and the higher values of RSSI lead to a better PDR in the receiver and the transmitter. Based on [36], if $RSSI = 87 \ dBm$, then the PDR is approximately 99%, and when RSSI = 0, then PDR is zero. Therefore, we consider these values as Q_{Max} and Q_{Min} , respectively.

• *Node Stability (Stability_{UAV_i}):* The purpose of selecting this parameter is to select nodes, which are at a suitable distance from UAVprev, for participating in the route discovery process. This distance is called the trust distance (D_{trust}) . This distance is defined in $[d_{Min}, d_{Max}]$, where, $0 \le d_{Min} < d_{Max}, d_{Min} < d_{Max} \le R$, and R represents the communication radius of the nodes. UAVs, which are at this distance from UAVprev, are known as stable nodes. Because they can communicate with UAV_{prev} for an acceptable time interval. If the distance between UAV_i and UAV_{prev} is less than d_{Min} , they are very close to each other. As a result, it is not reasonable to select UAV_i as the next-hop node. Because the number of hops will be increased and it is practically useless. On the other hand, if the distance between UAV_i and UAV_{prev} is greater than d_{Max} , the distance between them is very high. Therefore, it is very likely that the two nodes will be out of the communication range of each other in the near future. This process is illustrated in Figure 5. As shown in this figure, the UAV_i is located in D_{trust} . Whereas, UAV_j and UAV_k are outside this range. As a result, UAV_i is the best node that can be selected as the next-hop node (UAV_{next}) to establish stable routes. The node stability (*Stability*_{UAV_i}) is calculated based on Equation 13:

$$Stability_{UAV_{i}} = \begin{cases} 1 - \frac{|d_{Min} - D_{ij}|}{d_{Min}}, & 0 \le D_{ij} < d_{Min} \\ 1, & d_{Min} \le D_{ij} \le d_{Max} & (13) \\ 1 - \frac{|D_{ij} - d_{Max}|}{R - d_{Max}}, & d_{Max} < D_{ij} \le R \end{cases}$$

where, D_{ij} is Euclidean distance between UAV_i and UAV_{prev} at the moment when the RREQ message is received by UAV_i , i.e. t_{RREO} .

$$D_{ij} = \sqrt{(x_i - x_{prev})^2 + (y_i - y_{prev})^2 + (z_i - z_{prev})^2}$$
(14)



FIGURE 5. Describing D_{trust} between nodes in space.

where, (x_i, y_i, z_i) and $(x_{prev}, y_{prev}, z_{prev})$ are the spatial coordinates of UAV_i and UAV_{prev} at the time t_{RREQ} , respectively. Note that UAV_i obtains the spatial coordinates of UAV_{prev} from the RREQ message.

After calculating the mentioned parameters, the score of each node (S_i) is calculated using Equation 15:

$$S_{i} = W_{1} (\lambda_{i}) + W_{2} (E_{norm-UAV_{i}}) + W_{3} (Q_{norm-Link_{i-prev}}) + W_{4} (Stability_{UAV_{i}})$$
(15)

where, W_1 , W_2 , W_3 , and W_4 are Weight coefficients, so that $W_1 + W_2 + W_3 + W_4 = 1$. In this paper, these weighted coefficients are considered similar i.e. $W_1 = W_2 = W_3 = W_4 = \frac{1}{4}$.

2) THE ROUTE DISCOVERY PROCESS USING AN EXAMPLE

In this section, we describe the route discovery process through an example to simplify its understanding (see Figure 6). This process starts when UAV_S wants to communicate with UAV_D to send its data packets. In this case, UAV_S searches its routing table to find a valid route to UAV_D . If it does not find such a route, then UAV_S must discover a valid route. Algorithm 1 presents pseudo-code related to the route discovery process. This process includes the following steps:

a: STEP 1 (GENERATING THE RREQ MESSAGE)

In this step, UAV_S generates an RREQ message and inserts its ID and the ID of UAV_D into this message. We introduced the format of the RREQ message in Section V. When generating this message, UAV_S sets the *Message Type* field to one. Furthermore, the Hop_{Count} field is adjusted to zero. Then, this field is added one unit in each hop. UAV_S sets the F_{Route} field to one. This field is updated at each hop. Moreover, UAV_S sets the $Delay_{Route}$ field to zero. Next, delay is calculated at each hop to update this field. Finally, UAV_S broadcasts

Algorithm 1 Route Discovery Process

Input: UAV_i , i = 1, 2, ..., N

N: The number of UAVs in the network.

Output: Discovering a valid route between UAV_S and UAV_D .

Begin

- 1: if UAV_S is to transfer its data packets to UAV_D then
- UAV_S: Search its routing table to find a valid route to UAV_D;
- 3: **if** there is a valid route to UAV_D **then**
- 4: **UAV**_S: Transfer its data packets to UAV_D through this route;
- 5: **else**
- 6: **Call** Algorithm 2;
- 7: end if
- 8: **end if**
- End

the RREQ message to its neighboring nodes. As shown in Figure 7, UAV_1 , UAV_2 , UAV_3 , and UAV_4 receive this message. In addition, UAV_5 adds an entry to its routing table, which is illustrated in Figure 7. It should be noted that in this routing table, the *Next-Hop* field is determined after completing the route discovery process and receiving the RREP message. Also, Algorithm 2 expresses the pseudo-code related to the first step.

Algorithm 2 Step 1 (Generating RREQ Message)

Input: UAV_S

Output: RREQ message

Begin

- 1: UAV_S: Generate an *RREQ message*;
- UAV_S: Insert its ID into Source IP Address field of the RREQ message;
- 3: **UAV**_S: Insert the ID of *UAV*_D into *Destination IP Address* field of the *RREQ message*;
- 4: UAV_S: Set *Hop_{Count}* field of the *RREQ message* to 0;
- 5: UAV_S: Set *F_{Route}* field of the *RREQ* message to 1;
- 6: **UAV_S**: Set *Delay_{Route}* field of the *RREQ message* to 0;
- 7: UAV_S: Broadcast the *RREQ message* to the neighboring nodes;
- 8: **UAV**_S: Add an entry to its routing table;
- 9: Call Algorithm 3;
- End

b: STEP 2 (REBROADCASTING THE RREQ MESSAGE)

When the neighboring nodes of UAV_S (i.e. UAV_1 , UAV_2 , UAV_3 , and UAV_4) receive the RREQ message, they first check its *RREQ Message ID* field to make sure that this message is not duplicated. Then, each neighboring node calculates its own score (S_i) relative to UAV_S . In section VI-A1, we explained how to calculate S_i . In this example, assume that the scores corresponding to UAV_1 , UAV_2 , UAV_3 , and UAV_4 are equal to $S_1 = 0.61$, $S_2 = 0.75$, $S_3 = 0.36$, and



FIGURE 6. An example for the route discovery process.



FIGURE 7. Route discovery process in step 1.

 $S_4 = 0.98$, respectively. Then, they broadcast a message, including S_i , to their single-hop neighboring nodes. After receiving this message, each node compares its score with the average score of neighboring nodes. If the score of each node (for example, UAV_i) is less than half the average score of the neighboring nodes, then this node does not get the required score to rebroadcast the RREQ message. This means when Equation 16 is met.

$$S_i < \frac{1}{2} \left(\frac{1}{n_{neighbor}} \sum_{i=1}^{n_{neighbor}} S_i \right)$$
(16)

where, $n_{neighbor}$ is the number of neighboring nodes.

As a result, UAV_i removes the RREQ message and cannot rebroadcast it. In the example provided in Figure 8, UAV_3 satisfies such condition (Equation 16). Whereas, other nodes (i.e. UAV_1 , UAV_2 , and UAV_4) achieve an acceptable score to rebroadcast the RREQ message. Therefore, they add an entry to their routing table to record the RREQ information. Because UAV_1 , UAV_2 , and UAV_4 have received the RREQ message only from UAV_S , they register its ID as the previous-hop node in their routing table, which is displayed in Figure 8. It should be noted that in the routing table, the *Next-Hop* field is determined when the route discovery is terminated and the RREP message is received. Then, UAV_1 , UAV_2 , and UAV_4 update some fields of the RREQ message, and rebroadcast it to their neighbors. For example, as shown in Figure 8, UAV_1 adds one unit to the Hop_{Count} field in the RREQ message. Also, the F_{Route} field is updated according to Equation 17:

$$F_{Roue} = Min\left(F_{Route}, S_{UAV_{Current}}\right) \tag{17}$$



FIGURE 8. Route discovery process in step 2.

where, F_{Route} indicates the route's fitness and $S_{UAV_{Current}}$ is the score of the node that has received the RREQ message at the current time. For example, UAV_1 updates F_{Route} according to Equation 18:

$$F_{Route} = Min(F_{Route}, S_1) = Min(1, 0.61) = 0.61$$
 (18)

Moreover, the $Delay_{Route}$ field is updated in each hop using Equation 19.

$$Delay_{Route} = Delay_{Route} + T_{prev,current}$$
 (19)

where, $T_{prev,next}$ is the time required to transfer the RREQ message from UAV_{prev} to $UAV_{current}$. According to [33], the one-hop delay from UAV_{prev} to $UAV_{current}$ is calculated using the medium access delay ($D_{mac_{prev,current}}$) and the queuing delay ($D_{que_{prev,current}}$). $D_{mac_{prev,current}}$ is the time required for the medium access protocol to deliver the packet successfully or remove it due to duplicate. $D_{que_{prev,current}}$ is the required time for the message to reach the head of the transmission queue. Note that it ignores the propagation delay because it is very small when data are exchanged at the light speed in wireless media. Therefore, $T_{prev,next}$ is expressed as follows:

$$T_{prev,next} = D_{mac_{prev,current}} + D_{que_{prev,current}}$$
(20)

Therefore, UAV_1 updates $Delay_{Route}$ based on Equation 21.

$$Delay_{Route} = Delay_{Route} + T_{S,1} = 0 + T_{S,1} = T_{S,1}$$
 (21)

1

where, $T_{S,1}$ indicates the time required to transfer the RREQ message from UAV_S to UAV_1 .

Next, UAV_1 rebroadcasts the RREQ message to its neighboring nodes. In addition, UAV_2 and UAV_4 perform a similar process. This process is illustrated in Figure 8. Furthermore, Algorithm 3 presents the pseudo-code related to the second step.

c: STEP 3 (SELECTING THE PREVIOUS-HOP NODE)

This step is almost similar to the second step. However, it has a different part. If a node receives an RREQ message from several nodes (for example, two nodes or more), it calculates its score relative to each of them. Then, it registers the node, which has the highest score relative to it, as the previous-hop node in its routing table. Note that in this process, nodes wait for a certain time period to receive all RREQ messages from their neighboring nodes to determine the Previous-Hop field. As shown in Figure 9, UAV₅ receives two RREQ messages from UAV_1 and UAV_2 . In this case, UAV_5 calculates its score relative to these two nodes, so that its score relative to UAV_1 is equal to $S_{5,1} = 0.63$; and its score relative to UAV_2 is equal to $S_{5,2} = 0.92$. UAV₅ selects UAV₂ as the previous-hop node and records it in its routing table because $S_{5,2} > S_{5,1}$. Also, in this step, UAV_6 receives the RREQ message from UAV_2 and inserts UAV_2 as the previous-hop node in its routing table.

Algorithm 3 Step 2 (Rebroadcasting RREQ Message)

Input: RREQ message

 $UAV_i, \quad i=1,2,\ldots,N$

N: The number of UAVs in the network.

Output: Broadcasting RREQ message

Begin

- 1: if UAV_i receives the RREQ message from one node then
- 2: UAV_i: Check *RREQ Message ID* field of the *RREQ message*;
- 3: **if** the *RREQ* message is not duplicated **then**
- 4: **UAV**_i: Calculate its score using Eq 15;
- UAV_i: Broadcast a message including its score to the single-hop neighboring nodes;

6: **if**
$$S_i < \frac{1}{2} \left(\frac{1}{n_{neighbor}} \sum_{i=1}^{n_{neighbor}} S_i \right)$$
 then

7: **UAV**_i: Remove the
$$RREQ$$
 message;

- 8: else
- 9: UAV_i: Updating some fields of the *RREQ message* (i.e. *Hop_{Count}*, *F_{Route}*, *Delay_{Route}*);
- UAV_i: Rebroadcast the *RREQ message* to the neighboring nodes;
- 11: **UAV**_i: Add an entry to its routing table;
- 12: end if
- 13: end if
- 14: **else**
- 15: Call Algorithm 4;
- 16: end if
- End

Furthermore, UAV_7 receives the RREQ message from UAV_4 and inserts UAV_4 as the previous-hop node in its routing table. In Figure 9, this process is displayed. Algorithm 4 expresses the pseudo-code related to Step 3.

d: STEP 4 (ROUTE SELECTION)

In this step, UAV_D receives the RREQ message from UAV_5 , UAV_6 , and UAV_7 . Next, it calculates its score relative to these nodes. As shown in Figure 10, the score of UAV_D relative to UAV_5 , UAV_6 , and UAV_7 is equal to $S_{D,5} = 0.91$, $S_{D,6} = 0.35$, and $S_{D,7} = 0.98$, respectively. Then, UAV_D updates Hop_{Count} , F_{Route} , and $Delay_{Route}$ in the RREQ messages received from these nodes, as shown in Figure 10. If UAV_D 's ID is similar to the *Destination IP Address* field in the RREQ message, it terminates the rebroadcasting process of this message because the RREQ message has reached its destination. Now, UAV_D must select one of the established routes. In the example given in Figure 10, three routes have been discovered to UAV_D . In the following, each of these routes and their attributes are introduced. It should be noted that UAV_D extracts these attributes from received RREQ messages.

• Route 1: Route#1 : $UAV_S \rightarrow UAV_2 \rightarrow UAV_5 \rightarrow UAV_D$; The characteristics of this route are: $F_{Route_1} = 0.75$, $Hop_{Count} = 3$, and $Delay_{Route_1} = T_{S,2} + T_{2,5} + T_{5,D}$.

Algorithm 4 Step 3 (Selecting the Previous Hop)

Input: RREQ message

 $UAV_i, \quad i=1,2,\ldots,N$

N: The number of UAVs in the network.

Output: *Previous hop node*

Begin

- 1: UAV_i: Calculate its scores relative to the previous hop nodes;
- 2: UAV_i: Select the node with highest score as the previous hop node;
- 3: UAV_i: Broadcast a message including its score to the single-hop neighboring nodes;

4: if
$$S_i < \frac{1}{2} \left(\frac{1}{n_{neighbor}} \sum_{i=1}^{n_{neighbor}} S_i \right)$$
 then

5: UAV_i: Remove the *RREQ message*;

```
6: else
```

- UAV_i: Updating some fields of the *RREQ message* (i.e. *Hop_{Count}*, *F_{Route}*, *Delay_{Route}*);
- UAV_i: Rebroadcast the *RREQ message* to the neighboring nodes;
- 9: UAV_i: Add an entry to its routing table;

10: end if

- 11: Call Algorithm 5;
 - End
 - Route 2: Route#2 : $UAV_S \rightarrow UAV_2 \rightarrow UAV_6 \rightarrow UAV_D$; The characteristics of this route are: $F_{Route_2} = 0.35$, $Hop_{Count} = 3$, and $Delay_{Route_2} = T_{S,2} + T_{2,6} + T_{6,D}$.
 - Route 3: Route#3 : $UAV_S \rightarrow UAV_4 \rightarrow UAV_7 \rightarrow UAV_D$; The characteristics of this route are: $F_{Route_3} = 0.55$, $Hop_{Count} = 3$, and $Delay_{Route_3} = T_{S,4} + T_{4,7} + T_{7,D}$.

Now, UAV_D selects the best route using the proposed fuzzy system, which is also represented in Figure 11. In this example, UAV_D selects the route 1 (i.e. $Route#1 : UAV_S \rightarrow UAV_2 \rightarrow UAV_5 \rightarrow UAV_D$) as the best route for data transfer. Algorithm 5 illustrates the pseudo-code related to Step 4. To design this fuzzy system, Mamdani fuzzy inference mechanism has been used. The proposed fuzzy system has three inputs (i.e. F_{Route} , Hop_{Count} , and $Delay_{Route}$), one output (*Best_{Route*), and a rule base. In the following, each of these parts is described in detail.

e: FUZZY INPUTS

In the proposed fuzzy system, there are three input parameters, including F_{Route} , Hop_{Count} , and $Delay_{Route}$. They are explained as follows:

• F_{Route} : UAV_D obtains fitness information of routes from the RREQ messages. As stated in this section, F_{Route} is determined based on the minimum score of the nodes in a route. It is calculated using Equation 17 and its value is in [0, 1]. Whatever F_{Route} is close to one, meaning that this route includes high-quality links, the nodes have more energy in this route and have located at a suitable



FIGURE 9. Route discovery process in step 3.

Algorithm 5 Step 4 (Selecting the Route)

Input: *RREQ message*

 UAV_D

Output: Best route

Begin

- 1: if UAV_D receives the RREQ message then
- 2: UAV_D: Calculate its score using Eq 15;
- UAV_D: Updating some fields of the *RREQ messages* (i.e. *Hop_{Count}*, *F_{Route}*, *Delay_{Route}*);
- 4: UAV_D: Select the best route based on the proposed fuzzy system;
- 5: UAV_D: Call Algorithm 6;
- 6: **else**
- 7: **Call** Algorithm 3;
- 8: end if
- End

distance from each other and are moving in almost the same direction. This ensures that this route can be stable for a more time interval. The fuzzy membership diagram corresponding to F_{Route} is represented in Figure 12. This fuzzy input has three modes: *low, medium,* and *high*.

• *Hop_{Count}*: *UAV_D* can achieve information about the number of hops in a route from received RREQ messages. Note that if the number of hops is few in a

route, there is less delay in the data transmission process. This can improve the performance of the proposed routing scheme. This fuzzy input is normalized based on Equation 22 to be in [0, 1]:

$$Hop_{Count-norm} = \frac{Hop_{Count}}{N-1}$$
(22)

where, N represents the total number of nodes in the network. The fuzzy membership diagram corresponding to $Hop_{Count-norm}$ is displayed in Figure 13. This fuzzy input includes three modes: *low, medium,* and *high*.

• *Delay_{Route}*: Delay is a very important issue in the routing process for FANET, because the speed of drones is very high. It causes frequent disconnection of communication links and instability of routes. If there is a long delay in a route, then routes may quickly fail during sending data to the destination. This can increase the packet loss rate (PLR) in the network. *UAV_D* can obtain information about *Delay_{Route}* from RREQ messages. This parameter is normalized using Equation 23:

$$Delay_{Route-norm} = \frac{Delay_{Route}}{Max_{Delay}}$$
 (23)

where, Max_{Delay} is the maximum delay in the routes established between UAV_S and UAV_D . The fuzzy membership diagram corresponding to $Delay_{Route}$ is



FIGURE 10. Route discovery process in step 4.







FIGURE 12. Fuzzy membership diagram corresponding to *F_{Route}*.

represented in Figure 14. This input includes three modes: *low*, *medium*, and *high*.

f: FUZZY OUTPUT (Best_{Route})

In the proposed fuzzy system, the output is the chance of choosing a route from the discovered routes (in the example, *Route* #1, *Route* #2, and *Route* #3) as the best route for data transfer. *Best_{Route}* is a route with the most fitness, the least number of hops, and the minimum delay. The fuzzy membership diagram corresponding to $Best_{Route}$ is illustrated in Figure 15. This fuzzy output has seven modes: *Very very low*, *Very low*, *Low*, *Medium*, *High*, *Very high*, and *Very very high*.



FIGURE 13. The fuzzy membership diagram corresponding to Hop_{Count-norm}.



FIGURE 14. The fuzzy membership diagram corresponding to Delay_{Route}.



FIGURE 15. The fuzzy membership diagram corresponding to Best_{Route}.

g: RULE BASE

Our proposed fuzzy system follows 27 rules. These rules are listed in Table 6. For example, rule 1 is stated as follows:

Rule 1: IF F_{Route} is Low AND Hop_{Count} is Low AND $Delay_{Route}$ is Low THEN $Best_{Route}$ is High.

h: STEP 5 (SENDING THE RREP MESSAGE)

After selecting the best route (i.e. $Route#1 : UAV_S \rightarrow UAV_2 \rightarrow UAV_5 \rightarrow UAV_D$), UAV_D sets its routing table and inserts the ID of UAV_5 into the *Previous-Hop* field. Next, it generates the RREP message and sends back it to UAV_5 according to *Route* #1. When UAV_5 receives the RREP message, it checks the ID of this message to find the corresponding entry in its routing table. Then, UAV_5 completes the *Next-Hop* field in this table and inserts the ID of UAV_D into it. It also checks the *Previous-Hop* field in its routing table and extracts the ID of UAV_2 from it. Then, UAV_5 sends back the RREP message to UAV_2 . This process continues until the RREP message reaches UAV_S . Finally, UAV_S uses the specified route for transmitting data to UAV_D . This process is depicted in Figure 16. Algorithm 6 presents the pseudo-code related to Step 5. Moreover, the flowchart of the route discovery process is shown in Figure 17.

B. ROUTE MAINTENANCE PHASE

In this phase, we describe the route maintenance process. Obviously, the routes established in the route discovery phase may be failed due to the unique features of FANET, such

	Fuzzy system inputs			Fuzzy system output
Fuzzy rules	F _{Route}	<i>Hop_{Count}</i>	Delay _{Route}	Best _{Route}
1	Low	Low	Low	High
2	Low	Low	Medium	Medium
3	Low	Low	High	Low
4	Low	Medium	Low	Medium
5	Low	Medium	Medium	Low
6	Low	Medium	High	Very low
7	Low	High	Low	Low
8	Low	High	Medium	Very low
9	Low	High	High	Very very low
10	Medium	Low	Low	Very high
11	Medium	Low	Medium	High
12	Medium	Low	High	Medium
13	Medium	Medium	Low	High
14	Medium	Medium	Medium	Medium
15	Medium	Medium	High	Low
16	Medium	High	Low	Medium
17	Medium	High	Medium	Low
18	Medium	High	High	Very low
19	High	Low	Low	Very very high
20	High	Low	Medium	Very high
21	High	Low	High	High
22	High	Medium	Low	Very high
23	High	Medium	Medium	High
24	High	Medium	High	Medium
25	High	High	Low	High
26	High	High	Medium	Medium
27	High	High	High	Low

TABLE 6. Fuzzy rule base in proposed fuzzy system.

as the high-speed UAVs and the dynamic network topology. The purpose of this phase is to quickly detect a route that is failing and replace it with a new one. In the proposed scheme, the route maintenance process includes two steps:

- Preventing route failure
- Reconstructing failed routes

In the following, each of these steps is explained in detail.

1) STEP 1, PREVENTING ROUTE FAILURE

In Step 1, the purpose is to detect and modify routes, which are at the failure threshold, to avoid disrupting the data transmission process in the network. If one of the following three modes occurs, the established route must be corrected to prevent it from being failing:

- *Mode 1 (Reducing Energy of a Node):* If the energy of UAV_i in a route is less than the threshold value (i.e. $E_{UAV_i} < E_{Threshold}$), then this node is at the failure threshold. As a result, the route must be modified because UAV_i cannot participate in the data transmission process.
- Mode 2 (Increasing Data Traffic in a Node): If the data traffic of UAV_i in a route exceeds the threshold value (i.e. $Tr_{UAV_i} > Tr_{Threshold}$), then the buffer capacity of

this node is at the overflow threshold. Therefore, delay will be increased in the data transmission process and the route will be blocked. As a result, the route must be modified because UAV_i cannot transmit data packets

• Mode 3 (Decreasing Link Quality Between Two Nodes): If the quality of the link between UAV_i and UAV_j in one route is less than the threshold value (i.e. $Q_{Link_{i-j}} < Q_{Threshold}$), then the link between these two nodes is at a failure threshold. As a result, the path must be modified.

When one of the three modes occurs, UAV_i checks its routing table to identify routes, which includes this node. Next, it sends the *Warning message*, which includes the ID of UAV_i , to its previous-hop nodes in these routes. For example, see Figure 18. In this figure, assume that UAV_5 is at the failure threshold. This node is in *Route* #1, which was established in Section VI-A2. As a result, it sends a *Warning message* to UAV_2 to correct the route.

Then, UAV_2 sends the *Route Recovery message* to its neighboring nodes (i.e. UAV_5 , UAV_1 , UAV_3 , UAV_5 , and UAV_6). After they receive this message, the neighboring nodes send their spatial coordinates to UAV_2 . Among these nodes, UAV_2 selects a node that is closest to UAV_5 . This node is called an alternative node (in this example, UAV_6), which is also shown in Figure 19.







FIGURE 17. Route discovery process flowchart.

Then, it is necessary to check whether UAV_6 can build a valid route to the destination node or not. If a valid route is created (For example, *Route* #2), then this new route replaces *Route* #1. It should be noted that the data transmission

process continues through *Route* #1 until *Route* #2 is created. This helps to ensure that the data transmission process is not disrupted. After the route is corrected, *Route* #2 will be used for data transfer. See Figure 20. Algorithm 7 presents

Algorithm o Step 5 (Sending KKEP Messa
--

Input: RREP message

 UAV_D

 $UAV_i, \quad i=1,2,\ldots,N$

N: The number of UAVs in the network.

Output: Best route

Begin

- 1: UAV_D: Set its routing table based on the selected route;
- 2: UAV_D: Generate an *RREP messages*;
- 3: **UAV_D**: Unicast the *RREP messages* to the previous hop node (*UAV_i*) based on the selected route;
- 4: while ID of UAV_i ≠ Source IP Address field of the RREP message do
- 5: UAV_i: Set the next hop in its routing table;
- 6: UAV_i: Unicast the *RREP messages* to the previous hop node based on the selected route;
- 7: end while
- 8: UAV_S: Set the next hop field in its routing table;
- 9: **UAV_S**: Transfer its data packets to *UAV_D* through this route;

pseudo-code related to Step 1 (i.e. preventing route failure). Also, Figure 21 shows the flowchart of this process.

2) STEP 2, RECONSTRUCTING THE FAILED ROUTES

The purpose of the second step is to quickly recognize and replace the failed routes to decrease interruptions in the data transmission process. Therefore, UAVs should periodically check the routes in their routing table to recognize failed routes. For this purpose, UAV_S periodically sends the *Route validation message* to UAV_D through the route in its routing table. For example, see Figure 22.

If this message reaches UAV_D successfully, it indicates that this path is still valid. As a result, UAV_D sends the ACK message to UAV_S . This process is shown in Figure 23.

Otherwise, if this message does not reach UAV_D correctly, it means that the route has been blocked. Next, the *RERR message* is transmitted to UAV_S as shown in Figure 24. In such cases, UAV_S must restart the route discovery process to create a new route to UAV_D . The pseudo-code related to the reconstruction process of failed routes is presented in Algorithm 8. Flowchart of this process is represented in Figure 25.

VII. SIMULATION AND EVALUATION OF RESULTS

In this section, our proposed routing scheme is implemented using NS-Allinone-2.35 simulator to evaluate its performance. Then, we compare the results obtained from this simulation with three routing methods, namely ECaD [26], AODV [27], and LEPR [30]. The RWP mobility model has been used to simulate the movement of drones in FANET. In RWP, the most important advantage is its simple implementation. For this reason, most routing methods use this model to simulate the movement of drones in FANET [4]. In the simulation process of the proposed routing scheme,

Algorithm 7 Preventing Route Failure

Input: UAV_i , i = 1, 2, ..., NN: The number of UAVs in the network. **Output:** *The modified route*

Begin

- 1: **for** i = 1 to *N* **do**
- 2: if $E_{UAV_i} < E_{Threshold}$ or $Tr_{UAV_i} > Tr_{Threshold}$ or $Q_{Link_{i-i}} < Q_{Threshold}$ then
- 3: **UAV**_i: Send a *Warning Message* to the previous hop node (*UAV*_i);
- 4: UAV_j: Send a *Route Recovery Message* to its neighboring nodes (UAV_{neighbor});
- 5: **UAV**_{neighbor}: Send its spatial coordinates to UAV_i ;
- 6: **UAV_j**: Select the neighbor node closest (that has not yet been selected) to UAV_i as $UAV_{\text{alternative}}$;
- 7: **if** *UAV*_{alternative} cannot create a valid route **then**
- 8: **Go to** Line 6;
- 9: **end if**
- UAV_j: Select UAV_{alternative} as its next hop node in the routing table;
- 11: end if
- 12: end for
- End

Algorithm 8 Reconstructing the Failed Route				
Input: UAV _S				
UAV_D				

Output: New route

Begin

- 1: UAV_S: Send a *Route Validation Message* to UAV_D;
- 2: if this route is valid then
- 3: **UAV**_D: Send an *ACK Message* to UAV_S ;
- 4: **Go to** Line 1;
- 5: **else**
- UAV_S: Receive a *RERR Message* from the intermediate node;
- 7: UAV_S: Call Algorithm 1;
- 8: end if
- End

we assume that the size of the FANET environment is equal to $2000 \times 2000 \times 1000 \ m^3$. Furthermore, the maximum number of UAVs is 100 nodes in the network. They are randomly and uniformly distributed in the network environment. The simulation process time is 300 seconds. The simulation parameters are summarized in Table 7. We evaluate the performance of the proposed scheme in terms of end-to-end delay (EED), packet delivery rate (PDR), routing overhead, route stability and energy consumption.

A. END TO END DELAY (EED)

The End to end delay (EED) is equal to the average time required from generating the data packet by the source node to reaching the destination node. Figure 26 compares end-to-end



	Unmanned aerial vehicle (UAV)		Routing Table- <i>UAVs</i>				
	Warning Message		UAV_2	UAVD	UAVD	RREQ Message ID	

FIGURE 18. Sending a warning message to UAV₂.





FIGURE 19. Sending route recovery message to neighboring nodes.



FIGURE 20. Route modification process.

delay in different routing schemes with respect to the density of nodes in the network. It should be noted that in this case, the speed of UAVs is a constant value i.e. 30 m/s. As shown in Figure 26, our proposed scheme has the lowest delay compared to other routing methods. On average, it can reduce delay by 20.01%, 48.68%, and 65.78% compared to





FIGURE 21. The flowchart of preventing route failure.



FIGURE 22. Sending the route validation message.

ECaD [26], LEPR [30], and AODV [27], respectively. This is because we take into account route delay in the route discovery process to select routes with the least delay for

data transmission. Whereas, this issue has been ignored in both AODV [27], and LEPR [30] schemes. As a result, they are very extremely delayed in the data transmission process.



FIGURE 23. Sending the ACK message.



FIGURE 24. Sending an RERR message.

Based on these results, it can be concluded that these two methods face a serious challenge in large networks, i.e. high delay during the data transmission process. This is a very important problem in FANET, because it negatively affects the stability of the routes. As shown in Figure 26, ECaD [26] has an acceptable delay because it has taken into account the route delay parameter in the routing process. However, in this method, the route failure probability is very high because ECaD [26] ignores the link quality parameter in the route discovery process and may select the low-quality links in a route. This can increase the delay in the data transmission process due to the modification or reconstruction of the failed routes.

B. PACKET DELIVERY RATE (PDR)

Packet delivery rate (PDR) means the ratio of total received data packets at the destination node to total data packets

130000

generated. Figure 27 illustrates the packet delivery rate (PDR) in different routing schemes with respect to the node density in the network. It should be noted that in this case, the speed of UAVs is a constant value, i.e. 30 m/s. As shown in Figure 27, our scheme has the highest PDR compared to other routing methods. On average, it improves the packet delivery rate by 14.04%, 60.92% and 54.63% compared to ECaD [26], LEPR [30] and AODV [27], respectively. According to Figure 27, when the number of nodes increases in the network, the packet delivery rate is almost constant in the proposed method. This means that our scheme is scalable. Moreover, Figure 28 compares the packet delivery rate (PDR) in different routing schemes with respect to the speed of UAVs in the network. It should be noted that in this case, the number of UAVs is fixed i.e. 50 nodes in the network. As shown in Figure 28, when the speed of UAVs is increased in the network, the performance of various routing schemes



FIGURE 25. Flowchart of the reconstruction process of failed routes.

Parameter	Value
Simulator	NS-2.35
Simulation environment size (m^3)	2000×2000×1000
The number of UAVs	100
Simulation time (s)	1500
Speed of UAVs (m/s)	[3,30]
Initial energy of UAVs (J)	2000
Mobility model	RWP
Mobility generator	MobiSim
Communication range (<i>m</i>)	300
Data packet size (<i>kbit</i>)	1
Path loss model	Free-space
Mac layer standard	IEEE 802.11a

TABLE 7. Simulation parameters.

are weakened and PDR is reduced. When the speed of UAVs is increased in the network, the routes created between UAVs become more unstable. As a result, the route failure probability increases and the packet loss rate increases. In Figure 28, the proposed method has the best PDR compared to other routing schemes. Compared to AODV [27], our proposed scheme improves PDR almost twice. Furthermore, on average, it increases PDR by 10.4% and 46.55% compared to ECaD [26] and LEPR [30], respectively. Because our proposed scheme designs a route modification mechanism that corrects a route quickly if the route is at the failure threshold. This mechanism reduces the packet loss rate because it can decrease the number of failed routes. On the other hand, in the proposed method, we consider the energy of the nodes in the route discovery process to create stable routes and reduce the route failure. However, this issue has been ignored in LEPR [30] and AODV [27]. Also, ECaD [26] does not



FIGURE 26. Comparison of different routing methods in terms of the end to end delay.

consider the quality of communication links. Hence, this issue can increase the probability of route failure and decrease PDR.

C. ROUTING OVERHEAD

Routing overhead represents the ratio of all messages generated in the data transmission process to messages received at the destination node. Figure 29 compares the routing overhead in different schemes with respect to the number of UAVs in the network. It should be noted that in this case, the speed of UAVs is a constant value i.e. 30 m/s. As shown in Figure 29, the routing overhead of the proposed scheme is higher than other routing methods. It increases the routing overhead by 12.68% and 69.15% compared to ECaD [26] and AODV [27], respectively. However, its overhead is 1.64% lower than LEPR [30]. Because our scheme requires to broadcast the



FIGURE 27. Comparison of different routing methods in terms of the packet delivery rate.



FIGURE 28. Comparison of PDR in different routing schemes with respect to speed of UAVs in the network.

route recovery message during the route modification process. Moreover, nodes broadcast their scores to the single-hop neighboring nodes in the route discovery process. This can increase communication overhead. Also, based on Figure 29, it can be found that routing overhead in low density of nodes in the network is high for most routing protocols. Also, it is reduced when the density of nodes is increased in network. The reason for this is that in low density of nodes, it is less likely to find appropriate paths. In this case, AODV has a better performance than others, because its route discovery mechanism is simple. This mechanism is only based on broadcasting the RREQ message between neighboring nodes to find the appropriate path. Whereas, in ECAD, only high-energy nodes can broadcast the RREQ message. This limitation reduces the ability to find the route between the two nodes, especially in low density of nodes in the network. As a result, this can increase its routing overhead. On the other hand, in our method, nodes should calculate their score and share it with their neighboring nodes. Then, only the nodes with the high score can broadcast the RREQ message.



FIGURE 29. Comparison of different routing methods in terms of the routing overhead.



FIGURE 30. Comparison of the number of broken paths in different routing schemes with respect to the density of nodes in the network.

This limitation in our method increases routing overhead, especially in low density of network nodes.

D. ROUTE STABILITY

Route stability is evaluated based on the number of broken routes in the network. Obviously, if a routing method reduces the number of failed paths in the network, it can establish more stable routes. Figure 30 compares the number of broken paths in different routing schemes with respect to the density of nodes in the network. It should be noted that in this case, the speed of UAVs is a constant value i.e. 30 m/s. Note that the density of nodes in the network and the number of broken paths are inversely related to each other, meaning that as the density of nodes in the network increases, the number of failed routes decreases. In Figure 30, this issue is evident in the various routing schemes. However, we know that the density of nodes is low in FANET. As a result, different routing schemes must be successful in low-density networks. As shown in Figure 30, the proposed scheme can create more



FIGURE 31. Comparison of the number of broken paths in different routing schemes with respect to speed of UAVs in the network.

stable routes compared to other routing methods. On average, it can reduce the number of broken paths by 17.82%, 44.68% and 58.49% compared to ECaD [26], LEPR [30] and AODV [27], respectively. Moreover, Figure 31 shows the number of broken paths in different methods with respect to speed of UAVs in the network. It should be noted that in this case, the number of UAVs is fixed i.e. 50 nodes in the network. Note that the route stability and speed of nodes in the network have a reverse relationship with each other. Therefore, if the speed of the nodes is high in the network, then the number of broken paths increases. This is also shown in Figure 31. In FANET, the speed of the nodes is high, so a routing scheme must be successful when the speed of the UAVs is high in the network. As shown in Figure 31, the proposed method has the least route failure compared to other schemes. On average, our method reduces the number of broken paths by 32.76%, 58.79% and 77.56% compared with ECaD [26], LEPR [30] and AODV [27], respectively. This is because we have used a fuzzy system in the proposed routing scheme to select the best route for data transmission. In this fuzzy system, three parameters i.e. route fitness, number of hops, and route delay, are considered as inputs to select a route with the least delay, the best fitness and the minimum number of hops. Route fitness is determined based on the quality of the links between UAVs, the energy of nodes, the distance between UAVs, and the movement direction of UAVs in a route. Therefore, this parameter can improve the performance of our proposed method and increase the route stability because routes with high energy and appropriate quality are selected.

E. ENERGY CONSUMPTION

Figure 32 compares different routing schemes in terms of energy consumption. It should be noted that in this case, the number of UAVs is equal to 100. As shown in this figure, the proposed method has the lowest energy consumption in comparison with other methods. On average,



FIGURE 32. Comparison of different routing methods in terms of energy consumption.

it decreases energy consumption by 10.42%, 11.89%, and 17.31% compared to ECaD [26], LEPR [30], and AODV [27], respectively. LEPR [30] and AODV [27] do not consider the residual energy of UAVs in their routing process. This increases energy consumption and imbalanced energy consumption in these schemes. Ultimately, this issue reduces the network lifetime. Our proposed method and ECaD [26] consider the residual energy of nodes in the routing process to establish high-energy routes. They don't allow low-energy nodes to participate in the route discovery process. This increases energy efficiency and improves the network lifetime. In addition, our proposed method seeks to improve the route discovery process by considering various parameters such as movement direction, residual energy of nodes, link quality, node stability, hop count, and delay to build stable routes. As a result, our method can reduce the number of broken routes. This can improve energy consumption.

VIII. CONCLUSION

In this paper, we presented an energy-aware and predictive fuzzy logic-based routing scheme in flying ad hoc networks. Our proposed routing scheme included two phases: the route discovery phase and the route maintenance phase. In this scheme, each node must first calculate a score based on four parameters: movement direction, residual energy, link quality, and node stability. If nodes obtain the required score, they can participate in the route discovery process. In the route selection process, we designed a fuzzy system to select a route with more fitness, less delay, and fewer hops for data transfer. In the second phase, we examined the two processes, namely preventing route failure and reconstructing failed routes. Finally, the proposed routing scheme was evaluated in terms of end-to-end delay, packet delivery rate, routing overhead, and route stability, and energy consumption. Then, the simulation results were compared with three routing methods, namely ECaD [26], LEPR [30], and AODV [27]. These results show that the proposed scheme has a successful performance in terms of delay in the data transmission

process. We also evaluated PDR in different routing methods with respect to both node density and speed of UAVs in the network. In both cases, our scheme had the highest PDR compared to other routing methods. Therefore, it is more compatible with FANET because it can form stable paths. According to the simulation results, we deduced that our proposed scheme has a successful performance in networks with different sizes. This indicates that our scheme is scalable. We compared routing overhead in different schemes with respect to the number of UAVs in the network. This experiment show that routing overhead in the proposed method is higher than other routing schemes. Therefore, we must try to reduce this weakness and provide a more efficient scheme in future. In addition, we tested our routing scheme for path stability with respect to both network density and speed of UAVs in the network. These results show that the proposed routing method outperforms others, especially in low-density networks. Therefore, it is more compatible with FANET. Moreover, the proposed method can reduce the route failure in the high speed of UAVs. In the future research direction, we try to focus on multi-path routing scheme in FANET to improve fault tolerance in these networks. In addition, we seek to use artificial intelligence (AI) and machine learning (ML) techniques to present more efficient routing algorithms in FANET.

REFERENCES

- O. K. Sahingoz, "Networking models in flying ad-hoc networks (FANETs): Concepts and challenges," J. Intell. Robot. Syst., vol. 74, nos. 1–2, pp. 513–527, 2014.
- [2] M. A. Khan, A. Safi, I. M. Qureshi, and I. U. Khan, "Flying ad-hoc networks (FANETs): A review of communication architectures, and routing protocols," in *Proc. 1st Int. Conf. Latest Trends Electr. Eng. Comput. Technol. (INTELLECT)*, Nov. 2017, pp. 1–9.
- [3] W. Zafar and B. M. Khan, "Flying ad-hoc networks: Technological and social implications," *IEEE Technol. Soc. Mag.*, vol. 35, no. 2, pp. 67–74, Jun. 2016.
- [4] D. S. Lakew, N.-N. Dao, W. Na, and S. Cho, "Routing in flying ad hoc networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1071–1120, 2nd Quart., 2020.
- [5] A. Chriki, H. Touati, H. Snoussi, and F. Kamoun, "FANET: Communication, mobility models and security issues," *Comput. Netw.*, vol. 163, Nov. 2019, Art. no. 106877.
- [6] A. Bujari, C. E. Palazzi, and D. Ronzani, "Fanet application scenarios and mobility models," in *Proc. 3rd Workshop Micro Aerial Vehicle Netw., Syst., Appl.*, 2017, pp. 43–46.
- [7] F. De Rango, G. Potrino, M. Tropea, A. Santamaria, and P. Fazio, "Scalable and lighway bio-inspired coordination protocol for FANET in precision agriculture applications," *Comput. Elect. Eng.*, vol. 74, pp. 305–318, Mar. 2019.
- [8] A. G. Orozco-Lugo, D. C. McLernon, M. Lara, S. A. R. Zaidi, B. J. González, and O. Illescas, "Monitoring of water quality in a shrimp farm using a FANET," *Internet Things*, vol. 2020, Jan. 2020, Art. no. 100170.
- [9] A. Bujari, C. T. Calafate, J.-C. Cano, P. Manzoni, C. E. Palazzi, and D. Ronzani, "Flying ad-hoc network application scenarios and mobility models," *Int. J. Distrib. Sensor Netw.*, vol. 13, no. 10, 2017, Art. no. 1550147717738192.
- [10] O. Chughtai, M. H. Rehmani, L. Musavian, S.-M. Senouci, S. Cherkaoui, and S. Mao, "IEEE access special section editorial: Flying ad hoc networks: Challenges, potentials, future applications, and way forward," *IEEE Access*, vol. 9, pp. 74189–74193, 2021.
- [11] V. Sharma and R. Kumar, "G-FANET: An ambient network formation between ground and flying ad hoc networks," *Telecommun. Syst.*, vol. 65, no. 1, pp. 31–54, 2017.

- [12] I. Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying ad-hoc networks (FANETs): A survey," *Ad Hoc Netw.*, vol. 11, no. 3, pp. 1254–1270, 2013.
- [13] I. U. Khan, I. M. Qureshi, M. A. Aziz, T. A. Cheema, and S. B. H. Shah, "Smart IoT control-based nature inspired energy efficient routing protocol for flying ad hoc network (FANET)," *IEEE Access*, vol. 8, pp. 56371–56378, 2020.
- [14] A. Nayyar, "Flying adhoc network (FANETs): Simulation based performance comparison of routing protocols: AODV, DSDV, DSR, OLSR, AOMDV and HWMP," in *Proc. Int. Conf. Adv. Big Data, Comput. Data Commun. Syst. (icABCD)*, Aug. 2018, pp. 1–9.
- [15] A. Mukherjee, V. Keshary, K. Pandya, N. Dey, and S. C. Satapathy, "Flying ad hoc networks: A comprehensive survey," *Inf. Decis. Sci.*, vol. 4, pp. 569–580, Oct. 2018.
- [16] A. Khan, F. Aftab, and Z. Zhang, "BICSF: Bio-inspired clustering scheme for FANETs," *IEEE Access*, vol. 7, pp. 31446–31456, 2019.
- [17] A. Sharma, P. Vanjani, N. Paliwal, C. M. W. Basnayaka, D. N. K. Jayakody, H.-C. Wang, and P. Muthuchidambaranathan, "Communication and networking technologies for UAVs: A survey," *J. Netw. Comput. Appl.*, vol. 168, Oct. 2020, Art. no. 102739.
- [18] Z. Guan, N. Cen, T. Melodia, and S. M. Pudlewski, "Distributed joint power, association and flight control for massive-MIMO self-organizing flying drones," *IEEE/ACM Trans. Netw.*, vol. 28, no. 4, pp. 1491–1505, Aug. 2020.
- [19] H. Zhang, L. Song, and Z. Han, Unmanned Aerial Vehicle Applications Over Cellular Networks for 5G and Beyond. Cham, Switzerland: Springer, 2020.
- [20] O. S. Oubbati, M. Atiquzzaman, P. Lorenz, M. 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.
- [21] A. Srivastava and J. Prakash, "Future FANET with application and enabling techniques: Anatomization and sustainability issues," *Comput. Sci. Rev.*, vol. 39, Feb. 2021, Art. no. 100359.
- [22] O. S. Oubbati, A. Lakas, F. Zhou, M. Günes, and M. B. Yagoubi, "A survey on position-based routing protocols for flying ad hoc networks (FANETs)," *Veh. Commun.*, vol. 10, pp. 29–56, Oct. 2017.
- [23] G. A. Walikara and R. C. Biradar, "A survey on hybrid routing mechanisms in mobile ad hoc networks," *J. Netw. Comput. Appl.*, vol. 77, pp. 48–63, Jan. 2017.
- [24] A. Yadav and S. Verma, "A review of nature-inspired routing algorithms for flying ad hoc networks," in *Applications of Artificial Intelligence in Engineering.* Singapore: Springer, 2021, pp. 197–203.
- [25] Q. Usman, O. Chughtai, N. Nawaz, Z. Kaleem, K. A. Khaliq, and L. D. Nguyen, "A reliable link-adaptive position-based routing protocol for flying ad hoc network," *Mobile Netw. Appl.*, vol. 4, pp. 1–20, May 2021.
- [26] O. S. 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, Dec. 2019.
- [27] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. 2nd IEEE Workshop Mobile Comput. Syst. Appl.*, Jan. 1999, pp. 90–100.
- [28] K. A. Darabkh, M. G. Alfawares, and S. Althunibat, "MDRMA: Multidata rate mobility-aware AODV-based protocol for flying ad-hoc networks," *Veh. Commun.*, vol. 18, Aug. 2019, Art. no. 100163.
- [29] A. A. Deshpande, F. Chiariotti, and A. Zanella, "SMURF: Reliable multipath routing in flying ad-hoc networks," in *Proc. Medit. Commun. Comput. Netw. Conf. (MedComNet)*, 2020, pp. 1–8.
- [30] X. Li and J. Yan, "LEPR: Link stability estimation-based preemptive routing protocol for flying ad hoc networks," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jul. 2017, pp. 1079–1084.
- [31] F. Aadil, A. Raza, M. F. Khan, M. Maqsood, I. Mehmood, and S. Rho, "Energy aware cluster-based routing in flying ad-hoc networks," *Sensors*, vol. 18, no. 5, p. 1413, 2018.
- [32] G. Gankhuyag, A. P. Shrestha, and S.-J. Yoo, "Robust and reliable predictive routing strategy for flying ad-hoc networks," *IEEE Access*, vol. 5, pp. 643–654, 2017.
- [33] J. Liu, Q. Wang, C. He, K. Jaffrès-Runser, Y. Xu, Z. Li, and Y. Xu, "QMR: Q-learning based multi-objective optimization routing protocol for flying ad hoc networks," *Comput. Commun.*, vol. 150, pp. 304–316, Jan. 2020.
- [34] S. N. Shahbazova, M. Sugeno, and J. Kacprzyk, *Recent Developments in Fuzzy Logic and Fuzzy Sets*, vol. 391. Berlin, Germany: Springer, 2020.

- [35] Y. Chen, N. Zhao, Z. Ding, and M.-S. Alouini, "Multiple UAVs as relays: Multi-hop single link versus multiple dual-hop links," *IEEE Trans. Wireless Commun.*, vol. 17, no. 9, pp. 6348–6359, Sep. 2018.
- [36] N. Baccour, A. Koubâa, L. Mottola, M. A. Zuniga, H. Youssef, C. A. Boano, and M. and Alves, "Radio link quality estimation in wireless sensor networks: A survey," ACM Trans. Sensor Netw. (TOSN), vol. 8, no. 4, pp. 1–33, 2012.
- [37] C. J. Lowrance and A. P. Lauf, "Link quality estimation in ad hoc and mesh networks: A survey and future directions," *Wireless Pers. Commun.*, vol. 96, no. 1, pp. 475–508, Sep. 2017.
- [38] A. Vlavianos, L. K. Law, I. Broustis, S. V. Krishnamurthy, and M. Faloutsos, "Assessing link quality in IEEE 802.11 wireless networks: Which is the right metric?" in *Proc. IEEE 19th Int. Symp. Pers., Indoor Mobile Radio Commun.*, Sep. 2008, pp. 1–6.



SANG-WOONG LEE received the B.S. degree in electronics and computer engineering and the M.S. and Ph.D. degrees in computer science and engineering from Korea University, Seoul, South Korea, in 1996, 2001, and 2006, respectively. From June 2006 to May 2007, he was a Visiting Scholar with the Robotics Institute, Carnegie Mellon University. From September 2007 to February 2017, he was a Professor with the Department of Computer Engineering,

Chosun University, Gwangju, South Korea. He is currently a Professor with the School of Computing, Gachon University, Seongnam, South Korea. His current research interests include face recognition, computational aesthetics, machine learning, medical imaging analysis, and AI-based applications.



SAQIB ALI received the M.Sc. degree in information systems and the Ph.D. degree in computer science from La Trobe University, Australia. He is currently an Associate Professor with the Department of Information Systems, Sultan Qaboos University, Muscat, Oman. His research interests include industrial informatics, and cyber security for cyber physical systems and business processes and automation.



MOHAMMAD SADEGH YOUSEFPOOR received the B.Sc. degree in computer science from Dezful Branch, Payame Noor University, Dezful, Iran, in 2010, and the M.Sc. degree in artificial intelligence (AI) computer engineering with the Department of Computer Engineering, Dezful Branch, Islamic Azad University, Dezful, in 2017. His research interests include wireless sensor networks (WSNs), the Internet of Things (IoT), machine learning, pattern recognition, cryptography, and network security.



EFAT YOUSEFPOOR received the B.Sc. degree in computer hardware engineering from Jundishapoor University, Dezful, Iran, in 2013, and the M.Sc. degree in artificial intelligence computer engineering from the Department of Computer Engineering, Dezful Branch, Islamic Azad University, Dezful, in 2019. Her research interests include wireless sensor networks (WSNs), cryptography, and network security.



POOIA LALBAKHSH received the Ph.D. degree in computer science and computer engineering from La Trobe University, Melbourne, VIC, Australia, in 2017. He has more than 15 years of experience in academia and industry working in the areas of decision support systems, simulation modeling, machine learning, swarm intelligence, and universal AI. He has completed several research and industrial projects in the areas of capacity analysis, adaptive intelligent trading,

virtual therapy, financial distress analysis, network monitoring, adaptive routing, and capital market prediction. He has collaborated with several multidisciplinary teams from The Defence Science and Technology Group (DSTG), The Australian National University (ANU), La Trobe University, RMIT University, and Euler Capital Pty Ltd. He currently focuses on decision support systems, optimization algorithms, and AI-enabled cyber security strategies.



DANIAL JAVAHERI (Member, IEEE) received the B.Sc., M.Sc. (Hons.), and Ph.D. (Hons.) degrees in computer software engineering from Islamic Azad University (IAU), in 2012, 2014, and 2018, respectively. He ranked first among graduates and completed his PhD with a GPA of 19.88/20 at the age of 28. He joined the Cyber Security Research Center, IAU, and the APA Specialized Center, IUMS, as a Security Specialist, in 2015, and has maintained his role as a University Lecturer in a

number of top universities in Iran. Over the past decade, he has conducted extensive research in the fields of software security, cloud computing, deep learning, intrusion detection, and malware analysis with reputable entities worldwide and has published several articles and books in relevant areas. Due to his exemplary performance in education and research, he was awarded Top Student and Top Graduate by Iran's National Elites Foundation in 2017–2018, including being awarded several honors by the university.



AMIR MASOUD RAHMANI received the B.S. degree in computer engineering from Amir Kabir University, Tehran, in 1996, the M.S. degree in computer engineering from Sharif University of Technology, Tehran, in 1998, and the Ph.D. degree in computer engineering from IAU University, Tehran, in 2005. He is currently a Professor of computer engineering. He is the author/coauthor of more than 350 publications in technical journals and conferences. His research interests include

distributed systems, the Internet of Things, and evolutionary computing.



MEHDI HOSSEINZADEH received the B.S. degree in computer hardware engineering from Islamic Azad University, Dezful Branch, Iran, in 2003, and the M.Sc. and Ph.D. degrees in computer system architecture from the Science and Research Branch, Islamic Azad University, Tehran, Iran, in 2005 and 2008, respectively. He is the author/coauthor of more than 150 publications in technical journals and conferences. His research interests include SDN, information tech-

nology, data mining, big data analytics, e-commerce, e-marketing, and social networks.

....