A local filtering-based energy-aware routing scheme in flying ad hoc networks

Mehdi Hosseinzadeh

Duy Tan University

Fatimatelbatoul Mahmoud Husari

Cihan University-Erbil

Mohammad Sadegh Yousefpoor

Lebanese French University

Jan Lansky

lansky@mail.vsfs.cz

University of Finance and Administration

Hong Min

Gachon University

Article

Keywords:

Additional Declarations: No competing interests reported.

A local filtering-based energy-aware routing scheme in flying ad hoc networks

Mehdi Hosseinzadeh[1,2]**, Fatimatelbatoul Mahmoud Husari**[3]**, Mohammad Sadegh Yousefpoor**[4]**, Jan Lansky**[5,*]**, and Hong Min**[6,*]

[1]Institute of Research and Development, Duy Tan University, Da Nang, Vietnam
[2]Department of Computer Science, University of Human Development, Sulaymaniyah, Iraq
[3]Department of Communication and Computer Engineering, Faculty of Engineering, Cihan University-Erbil, Kurdistan Region, Iraq
[4]Center of Research and Strategic Studies, Lebanese French University, Kurdistan Region, Iraq
[5]Department of Computer Science and Mathematics, Faculty of Economic Studies, University of Finance and Administration, Prague, Czech Republic
[6]School of Computing, Gachon University, Seongnam, Republic of Korea
[*]Corresponding authors: lansky@mail.vsfs.cz and hmin@gachon.ac.kr

ABSTRACT

Flying ad hoc network (FANET) is a new technology, which creates a self-organized wireless network containing unmanned aerial vehicles (UAVs). In FANET, routing protocols deal with important challenges due to limited energy, frequent failures in communication links, high mobility of UAVs, and limited communication range of UAVs. Thus, a suitable path is always essential to transmit data between UAVs reliably. In this paper, a local filtering-based energy-aware routing scheme (LFEAR) is proposed for FANETs. LFEAR improves the template of the route request (RREQ) packet by adding three other fields, namely the energy, reliable distance, and movement similarity of the relevant route to create stable and energy-efficient paths between UAVs. In the routing process, LFEAR presents a local filtering construction technique to avoid the broadcasting storm issue. This filter limits the broadcasting range of RREQs in the network. Accordingly, only UAVs inside this local filtered area can rebroadcast RREQ and other UAVs must eliminate this packet. Upon the end of the route discovery process, the destination begins the route selection phase and extracts information about each discovered route, including the number of hops, route energy, reliable distance, and movement similarity from the relevant RREQ. Then, the destination node calculates a score for each path based on the extracted information, selects the route with the highest score, and sends a route reply (RREP) packet to the source node through this route. Finally, the simulation process of LFEAR is performed using the NS2 simulator, and two simulation scenarios, namely change in network density and change in the speed of UAVs, are defined to evaluate network performance. In the first scenario, LFEAR improves energy consumption, packet delivery rate, network lifespan, and delay by 1.33%, 1.77%, 6.74%, and 1.71%, while its routing overhead is about 16.51% more than EARVRT. In the second scenario, LFEAR optimizes energy consumption and network lifetime by 5.55% and 5.67%, respectively. However, its performance in terms of routing overhead, packet delivery rate, and delay is 23%, 2.29%, and 6.67% weaker than EARVRT, respectively.

1 Introduction

Today, unmanned aerial vehicles (UAVs) have become advanced technology due to low cost, high strength, and various applications. Hence, they have attracted the attention of many researchers at universities and industry[1,2]. In addition, multi-UAV systems can play an effective role in various missions due to features such as high scalability, good stability, and high throughput. When multiple UAVs cooperate to carry out a particular mission, this cooperation leads to the formation of a flexible, dynamic, distributed, and strong network called a flying ad hoc network (FANET)[3,4]. An important application of FANET is precision agriculture. Newly, farmers find out that the use of a swarm of UAVs in an ad hoc form is very useful in agriculture. This increases production growth and improves the quality of agricultural products because FANET helps farmers manage crops and use water and fertilizer efficiently[5,6]. Additionally, the use of UAVs, equipped with cameras, sensors, and other data collection devices, allows farmers to control their product state and identify problems that are not visible on the ground surface[7,8]. Compared to other traditional methods, UAVs can be used in pesticide spraying automatically and accurately. Figure 1 depicts the application of FANETs in precision agriculture[9,10].

FANET does not rely on communication infrastructure. In FANET, the information, such as control commands, position, and data collected from the environment, is shared with UAVs through a wireless channel[11,12]. In FANETs, researchers have challenges such as flying UAVs in a 3D area, having an unstable network topology, limiting energy resources, moving UAVs at
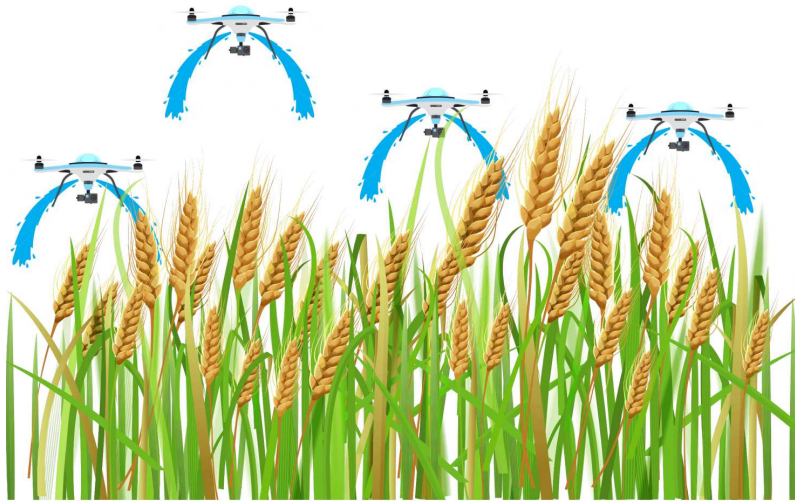
Figure 1. FANET in precision agriculture.

high speed, and applying a low-density network. These challenges should be taken into consideration to achieve a reliable and stable data transmission process[13, 14]. A suitable routing approach must provide extensive communication coverage, high reliability, and strong communication links in FANET. In FANETs, a routing protocol is responsible for discovering one or more routes and delivering data packets from the source to the destination via a multi-hop route[15, 16]. Many routing protocols such as DSDV[17], OLSR[18], DSR[19], AODV[20], TORA[21], and GPSR[22], which have been used on other ad hoc networks (i.e. mobile ad hoc networks (MANETs) or vehicular ad hoc networks (VANETs)) should be modified for FANETs.

Despite much research to solve this problem, research on routing protocols in FANET is still a hot scientific subject because the formation of stable paths in FANET is a major challenge due to the instability of communication links and the high mobility of UAVs. The AODV routing algorithm is a classic and successful protocol in FANETs[23, 24]. It utilizes an on-demand routing methodology. This means that AODV only finds suitable paths in the network when requesting a node to build a route and broadcasting a route request (RREQ) packet in the network. This feature is valuable because it reduces routing overhead and energy consumption and lowers the need for memory[25, 26]. Thus, AODV is a popular routing protocol in ad hoc networks, especially FANETs. Today, various AODV-based routing schemes have been presented for ad hoc networks, especially FANET. However, AODV suffers from some weaknesses that must be addressed in FANETs:

- **Lack of adaptability to the dynamic environment of FANET:** The AODV routing strategy relies only on one criterion, namely the number of hops, to find the shortest route to the destination. This strategy does not pay attention to the mentioned challenges in FANETs, including high-speed UAVs and the instability of network topology. This has led to the weak performance of AODV in FANET[27, 28].

- **The broadcasting storm issue:** In AODV, the route discovery strategy is performed by flooding RREQs in the network, but it results in high communication overhead and delay in the data transmission process. In addition, this flooding can cause a serious problem, namely the broadcasting storm, in the network. AODV tries to prevent this problem by inserting parameters such as sequence number, message ID, and number of hops into the RREQ packet, but it does not have a successful performance in this area, especially for FANET[29, 30].

- **Failure to use position information:** In AODV, the position information of UAVs is not used to find routes in the network. However, this information improves the performance of AODV and increases its adaptability to FANET. The position information of UAVs is obtained by connecting UAVs to a positioning system such as GPS[31, 32].

- **Lack of energy awareness:** Energy is a limited resource in FANET. Thus, each routing protocol must consider how to use this resource to improve network performance because UAVs are relatively far from each other in FANET. As a result, they consume a lot of energy for the data transfer process. This can quickly discharge their battery capacity. However, the routing strategy in AODV does not pay attention to this important challenge. Therefore, the selection of low-energy UAVs as intermediate nodes in routing paths can lead to instability of routes. This leads to an imbalance in energy consumption in the network and consequently reduces the lifetime of FANET[33, 34].

According to these facts, in this paper, a local filtering-based energy-aware routing method (LFEAR) is presented in FANETs to solve the challenges mentioned above. In this regard, LFEAR tries to improve the adaptability of AODV to FANET

and solve the broadcasting storm issue. To achieve these goals, LFEAR modifies the route discovery process in AODV, but the route maintenance process is similar to AODV. To solve the broadcasting storm issue, LFEAR uses local filtering to limit the broadcasting range of RREQ. This reduces routing overhead and delay in the route discovery procedure. To improve the adaptability to FANET, LFEAR applies information about the energy, position, and speed of UAVs in the route discovery process. In the following, the most important contributions are expressed in this paper.

- In LFEAR, the position and speed of each UAV are considered in the routing process to increase the adaptability of the proposed method to FANET. Therefore, UAVs regularly exchange this information through beacon messages to gain a better understanding of local network topology and make accurate decisions about UAVs in the route discovery process.

- In LFEAR, the template of two control packets used in the route discovery process, namely route request (RREQ) and route reply (RREP), are modified so that RREQ includes three new components, namely route energy, reliable distance, and movement similarity. The purpose of these three components is to increase the stability of paths, balance the energy consumption of UAVs, and improve network lifespan.

- In LFEAR, local filtering is employed to avoid the broadcasting storm issue by limiting the broadcasting range of RREQs. Accordingly, only UAVs inside the local filtering can rebroadcast the RREQ packet and other UAVs must remove this RREQ packet. Local filtering is a cylindrical zone whose central line is drawn from the source to the destination. Thus, the relevant equations are presented in the cylindrical coordinate system.

- In LFEAR, after the end of the route discovery procedure, the destination node must choose the best route from the found paths. Accordingly, the destination node extracts the information about each route, namely the number of hops, energy, reliable distance, and movement similarity from the RREQ packet to calculate its score. Lastly, the destination node selects the path with the highest score and sends RREP to the source node through this path.

The rest of this paper is as follows: Section 2 presents the related works in FANET. Section 3 describes the network and energy models in LFEAR. Section 4 elaborates on the proposed routing scheme accurately. The simulation results are shown in Section 5, and the conclusion is presented in Section 6.

2 Related works

In[35], the authors provided a utility function-based greedy perimeter stateless routing scheme (UF-GPSR) in FANETs. This scheme takes into consideration the essential components of UAVs, including distance, movement angle, velocity, the risk of communication links, and energy level to modify the greedy routing strategy. Furthermore, UF-GPSR computes the utility function based on the mentioned components to modify the routing procedure and designate the best UAV inside the communication area of the previous-hop UAV. The authors described their motivations for designing UF-GPSR: 1) The most suitable next-hop node is not always the UAV closest to the destination. 2) In environments with sparsely populated UAVs and high changes to network topology, routing holes are very likely, and the routing algorithm may converge to the local optimum. 3) The routing protocol must not rely only on the location coordinates of UAVs to choose the next-hop node. The results obtained from UF-GPSR illustrate that this scheme works well and satisfactorily.

In[36], the authors introduced an energy-efficient data transfer technique named ENSING for the Internet of Drones (IoD) to create three-dimensional connections in 6G networks. This approach addressed energy consumption in the swarm of UAVs because energy is an important issue, which affects UAVs in terms of their flight duration and their ability to do various missions. Additionally, the coordination and organization of UAVs in the network depend on communication links between UAVs in FANETs. ENSING is a new routing strategy, which focuses on energy consumption. It tries to lower the number of control packets to keep the energy of UAVs. In ENSING, the next-hop node is chosen based on the energy criterion. Thus, UAVs with an acceptable remaining energy level can participate in the data transfer procedure. The simulation results show that the performance of ENSING is better than other routing schemes.

In[37], the authors presented a position forecast-based greedy perimeter routing scheme named GPSR+ in FANET. GPSR+ utilizes a position forecast approach to gain the next position of UAVs in the FANET environment. Furthermore, the hello messaging period is modified to achieve better compatibility with FANET. It also designates a set of candidate UAVs based on the spherical removal strategy to decide on intermediate UAVs in the communication route. Finally, these selected UAVs create a stable route. The simulation results indicate the powerful performance of GPSR+ in comparison with other routing approaches.

In[38], the authors proposed a fuzzy-based trust-aware routing algorithm (FTSR) for FANETs. In FTSR, two trust formulations, namely local trust and path trust are defined. The first system, namely local trust, acts under a distributed process to unearth trustworthy neighboring UAVs and isolate untrustworthy neighboring UAVs in the network. Thus, only trustworthy nodes carry

Table 1. Comparison of related works.

Approach	Strengths	Weaknesses
UF-GPSR[35]	Obtaining high data delivery ratio and throughput, lowering latency in the routing process, achieving energy efficiency, improving routing overhead in the network	Not having enough scalability, not adjusting an adaptive time interval for broadcasting hello messages
ENSING[36]	Improving data delivery ratio, controlling and optimizing energy consumption, lowering latency in the routing process, managing routing overhead in the network, improving the reliability of communication links in the data transfer process	Not improving network scalability, failure to adaptability to FANET because of lack of attention to the speed and movement angle of UAVs, not addressing routing holes, this scheme may fall into local optimization
GPSR+[37]	Increasing data delivery ratio, optimal energy consumption in the network, improving reliability in the data transfer process, high compatibility with dynamic environment, especially FANET, increasing route stability	Increasing delay in the data transfer procedure, low network scalability
FTSR[38]	Enhancing data delivery ratio, achieving energy efficiency, improving reliability in the data transfer procedure, detecting untrustworthy nodes accurately, improving the security of communication links	Low compatibility with FANET, low network scalability
EARVRT[39]	Decreasing routing overhead, proper adaptability to the FANET environment, improving data delivery rate, decreasing latency in the data transfer procedure, optimal energy consumption	Low network scalability
LoCaL[40]	Decreasing routing overhead, achieving energy efficiency, increasing data delivery rate, improving network lifespan, reducing latency in the data transmission procedure	Low network scalability, lack of adaptability to FANET, instability of communication links
O-LAR[41]	Constructing routes with a low number of hops, decreasing latency in the data transfer procedure, increasing data delivery rate and throughput	Low network scalability, low compatibility with FANET, the low stability of communication links
PSO-GLFR[42]	Decreasing latency in the routing phase, increasing data delivery ratio, achieving energy efficiency	Low network scalability, lack of adaptability to FANET, instability of communication links, low reliability in the data transfer procedure

out the routing process. This lowers the number of unsafe paths in FANETs. On the other hand, the second trust system (i.e. path trust) recognizes untrustworthy nodes, which are not identified in the first trust system. This system calculates the trust of created routes in the network. To produce this system, the source UAV implements a fuzzy system to determine the safest route between source and destination. The results obtained from the simulation process confirm the performance of FTSR compared to other schemes.

In[39], the authors offered a virtual tunnel-based energy-aware routing technique named EARVRT for FANETs. In this scheme, a virtual relay tunnel (VRT) is designed to control the rebroadcasting range of RREQs when finding paths in the network. VRT decreases routing overhead because only UAVs, located in the tunnel, rebroadcast RREQ in the network. This tunnel prevents other UAVs from relaying RREQs to create the communication paths in the network. The goal of this tunnel is to avoid the broadcasting storm issue in FANET. UAVs construct the VRT tunnel very quickly and without extra overhead. Additionally, EARVRT takes into consideration three criteria, namely route energy, the number of hops, and a new criterion called route correlation, which depends on two elements, including the distance between UAVs and their cosine similarity. The simulation results confirm the successful performance of EARVRT in comparison with other algorithms.

In[40], the authors suggested a position-based cone-shaped routing technique called LoCaL for FANETs. The major motivation of LoCaL is to extend the lifespan of communication links in the network. In this scheme, the forwarding node is chosen based on several metrics, like link lifespan, remaining energy, and safety degree. Accordingly, LoCaL increases the stability of paths and decreases the number of broken paths. This route stability is achieved through a utility function. It assists LoCaL in selecting relay UAVs in the cone-shaped area. This reduces routing overload in the route-finding process. The simulation results show the successful performance of LoCaL compared to other schemes.

In[41], the authors offered a three-dimensional location-aware routing algorithm named O-LAR for propagating information in FANET. Additionally, O-LAR defines a weight function to construct the best route between UAVs. O-LAR utilizes this weight function to find intermediate UAVs in the communication path. It is dependent on multiple metrics, including movement angle, remaining energy, and distance. In addition, in O-LAR, a mathematical formula is presented to obtain parameters such as the link lifespan, remaining energy, and the average number of hops. The simulation results show that O-LAR works better than other schemes.

In[42], the authors offered a particle swarm optimization (PSO)-based greedy and limited flooding routing approach (PSO-GLFR) in FANET. In PSO-GLFR, two routing schemes, namely GPSR and AODV are merged. PSO-GLFR splits the data transfer procedure into two parts: 1) greedy routing and 2) flooding-based path-finding. Additionally, PSO-GLFR can solve the next-hop selection issue in the greedy routing strategy using the PSO algorithm. Simulation results represent the superiority of PSO-GLFR compared to other schemes.

Table 1 expresses the most important strengths and weaknesses of method mentioned in this section.

3 System model

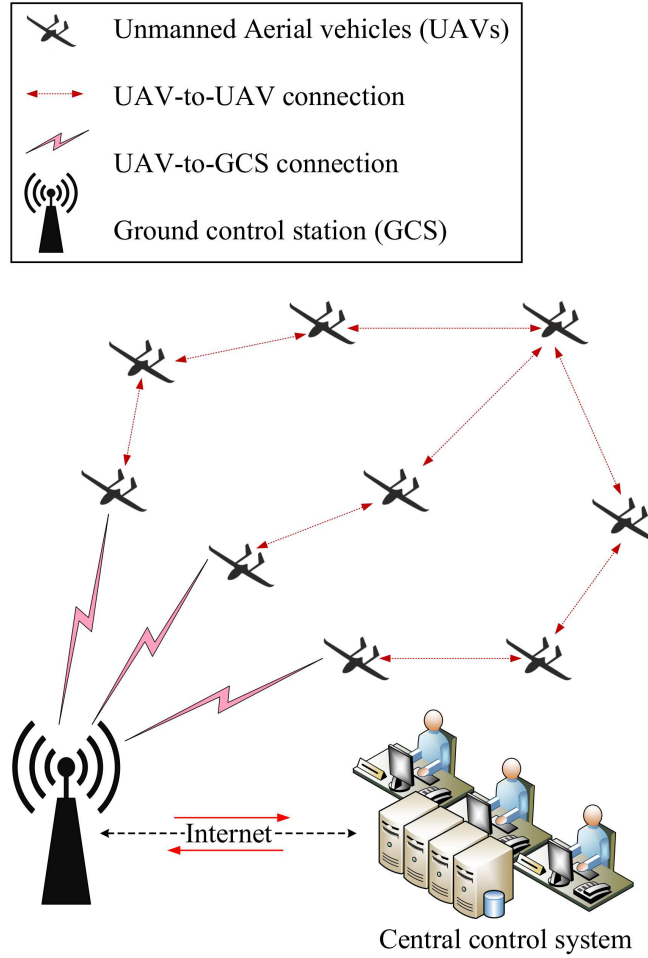Here, two network and energy models are described in LFEAR.

Figure 2. Network model in LFEAR.

3.1 Network model

In LFEAR, the network graph $G = (U, L)$ comprises two sets of vertices and edges i.e. $U = \{U_i | 1 \leq i \leq n\}$ and $L = \left\{ L_{ij} \big|_{\underset{i \neq j}{\forall}} U_i, U_j \in U \ and \ D_{ij} \leq R \right\}$, respectively. Here, U_i and n express i-th UAV and the total number of UAVs, respectively. In this network model, U_i uses a specific identifier (ID_i) to be distinguishable from other UAVs and R shows its communication radius in the network. Now, if U_i and U_j are in each other's communication area, their distance (i.e. $D_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$) is shorter than R, and U_i and U_j have a direct link L_{ij} in the set L. This link is used to communicate between the two UAVs. In addition, (x_i, y_i, z_i) and (x_j, y_j, z_j) show the spatial coordinates of U_i and U_j, respectively. Each UAV has access to a positioning system and can obtain its spatial coordinates at any moment. Now, if U_i and U_j are not in each other's communication area and do not have any direct link in the set L, their communication will be indirectly established. In this case, the network supports UAV-to-UAV communication, shown in Figure 2. UAVs employ IEEE 802.11g as a communication standard in the MAC layer. The network model also includes a ground control station (GCS) that is responsible for communicating with the central monitoring system and UAVs, transferring instructions to UAVs, receiving data collected by UAVs, and sending this data to the central system. This network also supports GCS-to-UAV communication so that GCS always connects to one or more UAVs and can communicate with other UAVs through these connected UAVs.

3.2 Energy model

In LFEAR, the energy model follows the first-order radio model[43], which supports the two energy models, namely the free space propagation model (FSPM) and the multi-path fading propagation model (MPFPM). These two models consider a direct

relationship between the energy consumption of the transmitter (U_i) and the recipient (U_j) and their distance (D_{ij}).

$$D_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \tag{1}$$

Furthermore, (x_i, y_i, z_i) and (x_j, y_j, z_j) show spatial coordinates U_i and U_j, respectively.

Accordingly, if U_i and U_j are far from each other, they need a lot of energy to exchange data. However, if U_i and U_j are close to each other, they need lower energy to exchange. According to FSPM and MPFPM, when U_i wants to send k bits to U_j, the energy consumption of U_i and U_j is obtained from Equations 2 and 3, respectively.

$$E_{TX}(k, d_{ij}) = \begin{cases} kE_{elec} + k\varepsilon_{fs}(d_{ij})^2 & d_{ij} \leq d_t \\ kE_{elec} + k\varepsilon_{mp}(d_{ij})^4 & d_{ij} > d_t \end{cases} \tag{2}$$

$$E_{RX}(k) = kE_{elec} \tag{3}$$

Here, E_{elec} denotes the energy used in the electrical circuit of U_i and U_j to exchange each bit. In addition, d_t indicates the distance threshold and is calculated through Equation 4.

$$d_t = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}} \tag{4}$$

If $d_{ij} \leq d_t$, the energy used in U_i follows the FSPM model. In this case, ε_{fs} is the power amplifier coefficient and the energy used in the amplifier is equal to $\varepsilon_{fs}(d_{ij})^2$. Otherwise, the energy used in U_i follows the MPFPM model. Furthermore, ε_{mp} indicates the power amplifier coefficient and the energy used in the amplifier is equal to $\varepsilon_{mp}(d_{ij})^4$.

4 Proposed method

In this section, a local filtering-based energy-aware routing scheme (LFEAR) is proposed in FANETs. In this regard, LFEAR seeks to improve the adaptability of AODV to FANET and solve the broadcasting storm problem. To achieve these goals, LFEAR modifies the route discovery process in AODV, but the route maintenance process in LFEAR is similar to it in AODV. To solve the broadcasting storm problem, LFEAR creates local filtering to limit the broadcasting range of RREQs in the network. This decreases routing overhead and latency in the path discovery procedure. To improve the compatibility with FANET, LFEAR uses the residual energy, position, and velocity of UAVs in the path discovery procedure. Thus, LFEAR includes two phases:

- Neighbor table construction phase

- Route discovery phase

In the following, each phase is described in detail. Figure 3 shows the schematic design of LFEAR.

4.1 Neighbor table construction phase

In LFEAR, the position and speed of UAVs are needed to enhance the adaptability of the proposed scheme to FANET. Hence, UAVs exchange this information regularly through beacon messages to obtain local network topology and decide on relay UAVs in the route discovery process. According to Figure 4, each beacon message includes an identifier (ID_i), spatial coordinates (x_i, y_i, z_i), and the speed vector (V_i, θ_i, φ_i). It is locally disseminated in the communication range of UAVs. In the speed vector, V_i, θ_i, and φ_i are the velocity length, the angle between the positive x-axis and the projection of the speed vector on the xy-plane, and the angle between the speed vector and the positive z-axis. Note that UAVs should not rebroadcast beacon messages received from other UAVs in the network.

When a UAV, like U_i, obtains a beacon message from a nearby UAV, like U_j, it adds an entry to its neighbor table (NT_i) and records the information of U_j in NT_i. See Figure 5. As soon as a new beacon message comes from U_j, its information in this entry will be updated immediately. To avoid holding out-of-date information in NT_i, U_i adds a validity time (VT_j) to the entry corresponding to U_j. VT_j is a timer, which is adjusted based on the beacon dissemination period. When U_i and U_j get away from each other's communication range, they do not receive each other's beacon messages, and in this case, VT_j is zero. As a result, the entry of U_j in NT_i is automatically removed. However, if U_i receives a new beacon message from U_j in the current beacon time, it will update VT_j again. As a result, the entry of U_j in NT_i remains valid at the next beacon period. Note that LFEAR regulates the beacon dissemination time based on the speed of UAVs to increase its adaptability to FANET. Accordingly, if UAVs move very quickly in the network, the beacon period will be shorter so that each UAV can keep its neighbor table up-to-date regarding the rapid changes in the network topology. However, if UAVs move very slowly in the network, LFEAR will prolong the beacon time because fewer changes occur in the network topology, and each UAV can update its neighbor table with low routing overhead. Algorithm 1 represents the pseudo-code of this phase.
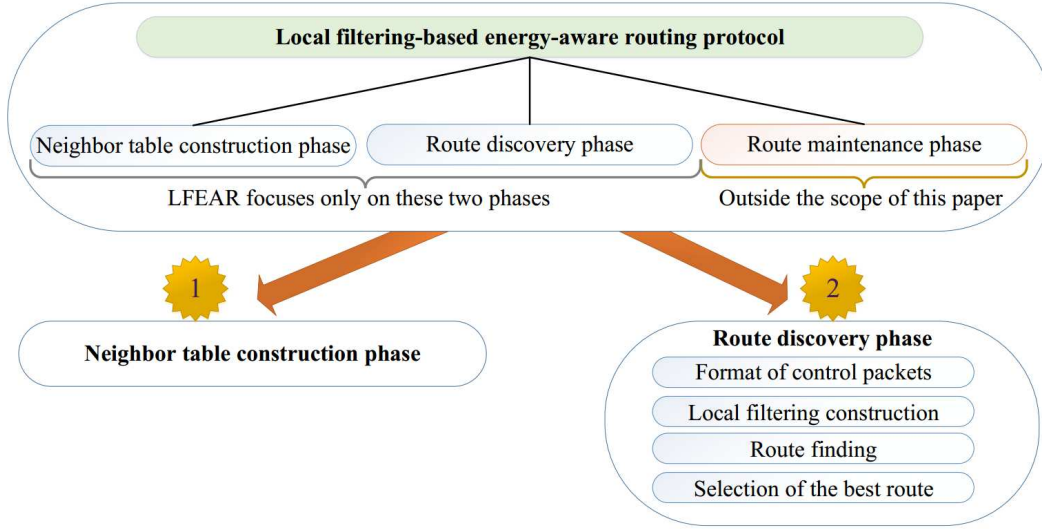
Figure 3. Schematic design of LFEAR.

Period of beacon message	ID of beacon message
ID_i	
(x_i, y_i, z_i)	
$(V_i, \theta_i, \varphi_i)$	

Figure 4. Beacon message format.

Identifier	Position	Speed	Validity time
ID_j	(x_j, y_j, z_j)	$(V_j, \theta_j, \varphi_j)$	VT_j

Figure 5. Structure of NT_i.

Algorithm 1 Neighbor table construction phase

Input: U_i: i-th UAV in the network
    n: Number of UAVs in the network.
    (x_i, y_i, z_i): Location of U_i
    ID_i: Identifier of U_i
    $(V_i, \theta_i, \varphi_i)$: Velocity information of U_i
    $Time_{Net}$: A counter for counting simulation time (ST).
Output: NT_i: Neighbor table related to U_i
    Begin
1: **U$_i$:** Determine the time period of beacon messages (PB) based on the velocity of UAVs;
2: **while** $Time_{Net} \leq ST$ **do**
3:     **if** $Time_{Net} \bmod PB = 0$ **then**
4:         **U$_i$:** Put ID_i, (x_i, y_i, z_i), and $(V_i, \theta_i, \varphi_i)$ in its beacon message in accordance with Figure 4;
5:         **U$_i$:** Transmit the beacon message to its adjacent UAVs;
6:     **end if**
7:     **if** U_i gets a beacon message from U_j **and** U_j has a particular entry in NT_i **then**
8:         **U$_i$:** Obtain the new information of U_j from the new beacon message;
9:         **U$_i$:** Replace the old and expired information of U_j with this information;
10:        **U$_i$:** Reset VT_j in the particular entry related to U_j in NT_i;
11:     **end if**
12:    **if** U_i gets a beacon message from U_j **and** U_j has not a particular entry in NT_i **then**
13:       **U$_i$:** Obtain the new information of U_j from this beacon message;
14:       **U$_i$:** Create a particular entry in NT_i for U_j;
15:       **U$_i$:** Record the information of U_j in this new entry;
16:       **U$_i$:** Tune VT_j in the particular entry related to U_j in NT_i;
17:    **end if**
18:    **for** $j = 1$ to n_i **do**
19:       **if** VT_j related to U_j in NT_i is equal to zero **then**
20:         **U$_i$:** Erase the particular entry related to U_j from NT_i;
21:       **end if**
22:    **end for**
23: **end while**
    End

4.2 Route discovery process

In this phase, suppose that U_S and U_D intend to communicate with each other to exchange their data packets, but the two UAVs have no direct link, and U_S cannot find any valid paths to U_D in its routing table. Therefore, U_S must execute the route discovery process by broadcasting the RREQ packet. In LFEAR, this process is explained in four steps:

- Format of control packets

- Local filtering construction process

- Path-finding process

- Selection of the best route

4.2.1 Format of control packets

Here, the formats of two control packets used in the route discovery process, namely route request (RREQ) and route reply (RREP), are explained. The formats of RREQ and RREP are presented in Figures 6 and 7, respectively. In the following, the most important fields in these control packets are described.

- **Message type:** The formats of RREQ and RREP are very similar. This field specifies the type of control packet to distinguish RREQs from RREPs. If this field is equal to one, the control packet is RREQ, while if it equals two, the packet is RREP.

- **Number of hops (HC_r):** When RREQ is initialized, U_S must set HC_r to zero. This field acts as a counter and counts the number of hops traveled in P_r (i.e. the route between U_S and U_D). This counter prevents routing holes in the network.

- **Route energy (E_r):** This field expresses the lowest residual energy of intermediate nodes in P_r and determines the energy crisis point (i.e. the UAV with the least energy capacity in P_r). Energy is an important criterion to evaluate P_r because if this route includes UAVs with inadequate energy levels, P_r is extremely unstable and may be cut quickly. As a result, the data transfer process will not be successful. If low-energy UAVs take part in the route construction process, it causes an imbalance in the energy consumption of nodes and reduces network lifespan. U_S determines the initial value of E_r based on its energy capacity. Then, E_r is updated in each hop based on Equation 5.

$$E_r = \min_{U_i \in P_r} \left\{ \frac{E_i}{E_{\max}} \right\} \tag{5}$$

where E_i and $E_{\max}$ represent the residual energy of U_i and the initial energy of UAVs in P_r, respectively.

- **Reliable distance (Ψ_r):** This field expresses the reliable distance between intermediate UAVs in P_r. The goal of Ψ_r is to choose two consecutive UAVs in P_r in a way that minimizes the likelihood of leaving each other's communication range. This means that the two UAVs are not close to each other's communication boundary. As a result, P_r remains valid for a longer time. This improves the data delivery ratio (PDR) and reduces delay in the routing process because LFEAR has less need to reconstruct the created paths between UAVs. Therefore, if P_r has a high reliable distance, it is more stable and suitable for transferring data packets. U_S sets the initial value of Ψ_r to one. Then, Ψ_r is calculated based on Equation 6 in each hop.

$$\Psi_r = \min_{U_j \, and \, U_i \in P_r} \left\{ \Psi_{ij} \right\} \tag{6}$$

So that Ψ_{ij} is the reliable distance between the two consecutive nodes U_i and U_j in P_r. Ψ_{ij} is calculated through Equation 23 in Section 4.2.3.

- **Movement similarity (ϑ_r):** This field indicates the least speed similarity of intermediate UAVs in P_r. The purpose of this field is to select the intermediate nodes with the similar speeds in P_r to form a stable path between U_S and U_D. In this case, intermediate nodes are in the communication range of each other for a longer time. However, if these intermediate UAVs have different speeds, the likelihood of leaving each other's communication range will be very high and the created routes must be rebuilt quickly. This needs a lot of communication and computational overheads. Thus, ϑ_r is calculated based on Equation 7.

$$\vartheta_r = \min_{U_j \, and \, U_i \in P_r} \left\{ CS_{\overrightarrow{V_{ij}}} \right\} \tag{7}$$

So that $CS_{\overrightarrow{V_{ij}}}$ shows the speed similarity of two consecutive nodes U_i and U_j in P_r. $CS_{\overrightarrow{V_{ij}}}$ is obtained from Equation 27 in Section 4.2.3.

Message type		Number of hops (HC_r)	
Route energy (E_r)	Movement similarity (ϑ_r)		Reliable distance (Ψ_r)
RREQ ID			
Destination IP Address			
Destination Sequence Number			
Source IP Address			
Source Sequence Number			

Figure 6. Format of RREQ.

Message type	Number of hops (HC_r)
RREP ID	
Destination IP Address	
Destination Sequence Number	
Source IP Address	
Time to live	

Figure 7. Format of RREP.

- **Message ID:** This field considers a unique identifier for each RREQ or RREP so that UAV identifies duplicated control packets and does not process these packets again.

4.2.2 Local filtering construction process

When U_S and U_D intend to communicate with each other, and U_S cannot find any valid path to U_D in its routing table, it must execute the route discovery process by broadcasting RREQs in its communication radius (R). This process is shown in Figure 8. According to this figure, the communication range of U_S is a circular area with the R radius, and U_S broadcasts RREQs for its neighboring UAVs, including U_1, U_2, U_3, U_4, U_5, U_6, and U_7.

To avoid the broadcasting storm issue, LFEAR uses local filtering to limit the broadcasting range of REEQs. Accordingly, only UAVs inside local filtering can rebroadcast the RREQ packet and other UAVs remove this packet. This local filtering is depicted in Figure 9. Local filtering is cylindrical, and its central line is drawn from U_S to U_D. According to Figure 9, U_1, U_2, and U_3 are outside the filtered area, and they must not rebroadcast the RREQ packet. But, U_4, U_5, U_6, and U_7 are inside the filtered zone, they can rebroadcast the RREQ packet.

Note that local filtering is cylindrical. Consequently, it is best to provide the relevant equations in the cylindrical coordinate system. As shown in Figure 10, if the Cartesian coordinates of a point, like P, are (x_p, y_p, z_p), then its cylindrical coordinates are (ρ_p, ϕ_p, z_p). Equation 8 is used to convert Cartesian coordinates into cylindrical coordinates.

$$
\begin{aligned}
\rho_p &= \sqrt{x_p{}^2 + y_p{}^2} \\
\phi_p &= \arctan\left(\frac{y_p}{x_p}\right) \\
z_p &= z_p
\end{aligned}
\tag{8}
$$

Here, ρ_p denotes the distance between the origin and the projection of P on the xy-plane. ϕ_p indicates the angle between the positive x-axis and ρ_p. z_p specifies the distance between P and the xy-plane. Furthermore, Equation 9 is used to convert cylindrical coordinates into Cartesian coordinates.

$$
\begin{aligned}
x_p &= \rho_p \cos\phi_p \\
y_p &= \rho_p \sin\phi_p \\
z_p &= z_p
\end{aligned}
\tag{9}
$$

To formulate the local filtering area, Figure 9 is simplified and converted to Figure 11. In Figure 11, local filtering is a cylindrical zone whose radius and height are R, and this area is rotated in the direction of U_D based on the angle α, so that the origin of the coordinate axes is U_S.

For constructing local filtering, three steps are considered. The pseudo-code of this process is presented in Algorithm 2.

- Transfer the three axes, namely X, Y, and Z, from $O(0,0,0)$ to $U_S(x_s, y_s, z_s)$. This operation converts the Cartesian coordinates of each point, like P, from (x_p, y_p, z_p) into (x'_p, y'_p, z'_p). The mathematical expression of this operation is
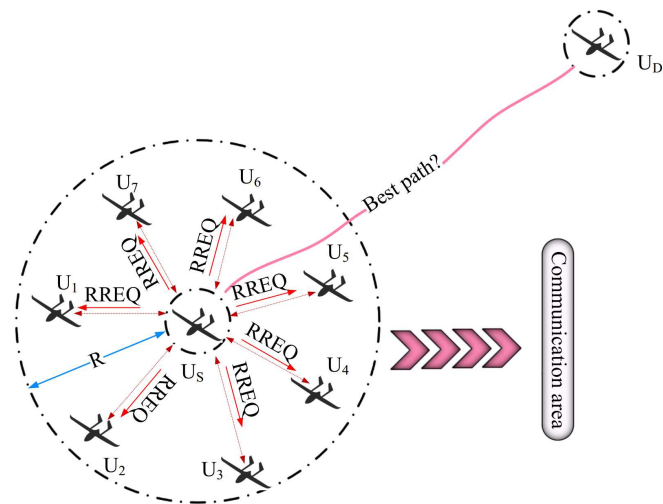
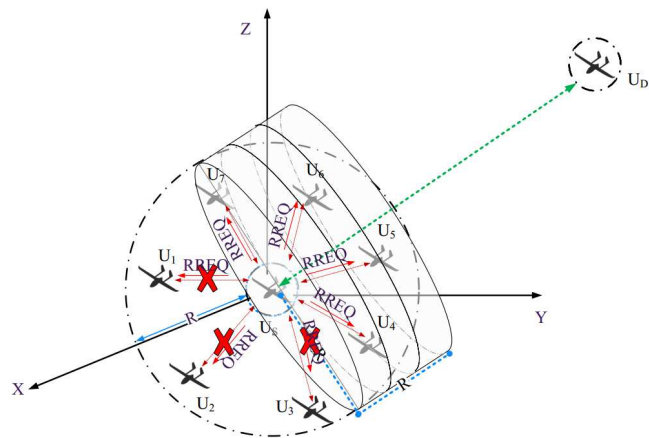**Figure 8.** Broadcasting RREQs in the route discovery process.
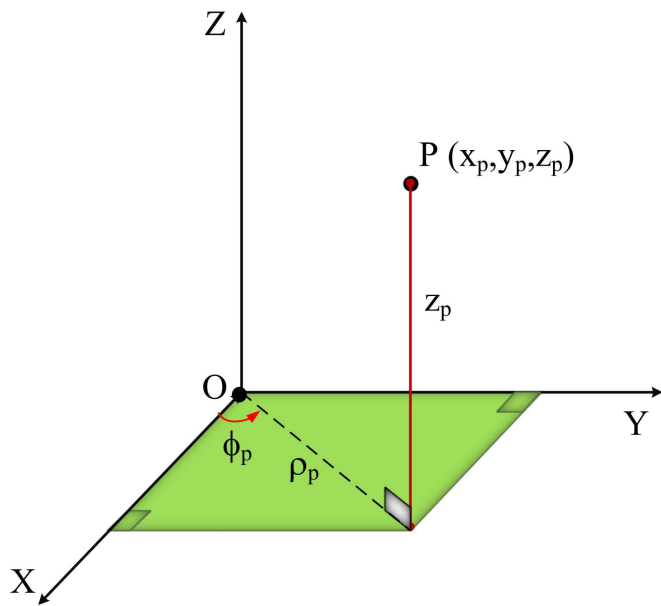

Figure 9. Filtering the relay UAVs.

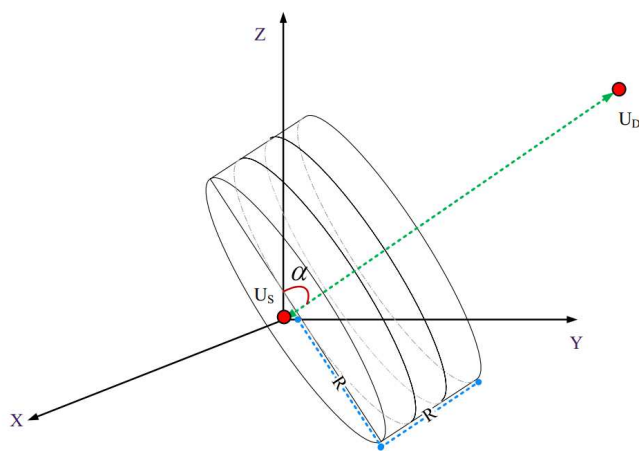**Figure 10.** Relationship between cylindrical coordinates and Cartesian coordinates.

Figure 11. Local filtering zone.

stated in Equation 10.

$$x'_p = x_p - x_s$$
$$y'_p = y_p - y_s$$
$$z'_p = z_p - z_s$$

(10)

- Rotate the transferred axes, namely X', Y', and Z' according to the rotation angle α in Figure 11. α is the angle between the z-axis and the line segment from U_S to U_D. This angle is obtained from Equation 11.

$$\alpha = \arccos\left(\frac{z_d - z_s}{\sqrt{(x_d - x_s)^2 + (y_d - y_s)^2 + (z_d - z_s)^2}}\right),$$
$$0 \le \alpha \le 2\pi$$

(11)

After rotating the coordinate axes, the Cartesian coordinates of each point, like P, is converted from $\left(x'_p, y'_p, z'_p\right)$ into $\left(x''_p, y''_p, z''_p\right)$. In LFEAR, the cylindrical coordinate system is used to calculate the filtered area, and the rotation matrix in this coordinate system is presented in Equation 12.

$$R_z(\alpha) = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(12)

The mathematical expression of this rotation operation is presented in Equation 13.

$$\begin{bmatrix} x''_p \\ y''_p \\ z''_p \end{bmatrix} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x'_p \\ y'_p \\ z'_p \end{bmatrix}$$

(13)

- Obtain the formulation of the cylindrical area with the radius and height of R in the rotated coordinate system through Equation 14.

$$0 \le \rho \le R, 0 \le z \le R, 0 \le \phi \le 2\pi$$

(14)

Algorithm 2 Local filtering construction

Input: U_S: Source UAV
  U_D: Destination UAV
Output: Local filtering zone
  Begin
1: **U_S:** Move the origin of Cartesian coordinate system from $O(0,0,0)$ to $U_S(x_s, y_s, z_s)$ using Equation 10;
2: **U_S:** Get the angle α between the line segment from U_S to U_D and z-axis based on Equation 11;
3: **U_S:** Rotate Cartesian coordinate system with the origin $U_S(x_s, y_s, z_s)$ using Equations 12 and 13;
4: **U_S:** Compute the local filtering zone based on Equation 14;
  End

4.2.3 Path-finding process

Here, assume that U_S and U_D intend to communicate with each other, and U_S cannot find any valid path in its routing table. Thus, U_S adjusts the RREQ packet according to the format presented in Figure 6 in Section 4.2.1 and broadcasts it within its communication radius (i.e. R). The path discovery process in LFEAR is explained based on the example presented in Figure 12. In this example, U_S disseminates RREQ for its neighboring UAVs, including U_1, U_2, U_3, U_4, U_5, U_6, and U_7. After receiving this RREQ, these neighbors execute the filtering steps to find out whether they are inside the filtered area of their previous-hop UAV or not. For the general expression of this filtering mechanism, assume that the previous-hop node is U_i, and its coordinates are equal to (x_i, y_i, z_i). In addition, the next-hop UAV is U_j, and its coordinates are equal to (x_j, y_j, z_j). Now, U_j carries out the four steps to determine whether it is inside the local filtering zone or not.

- **Step 1:** U_j transfers its spatial coordinates (x_j, y_j, z_j) to the coordinate system with the origin $U_i(x_i, y_i, z_i)$ according to Equation 15.

$$x'_j = x_j - x_i$$
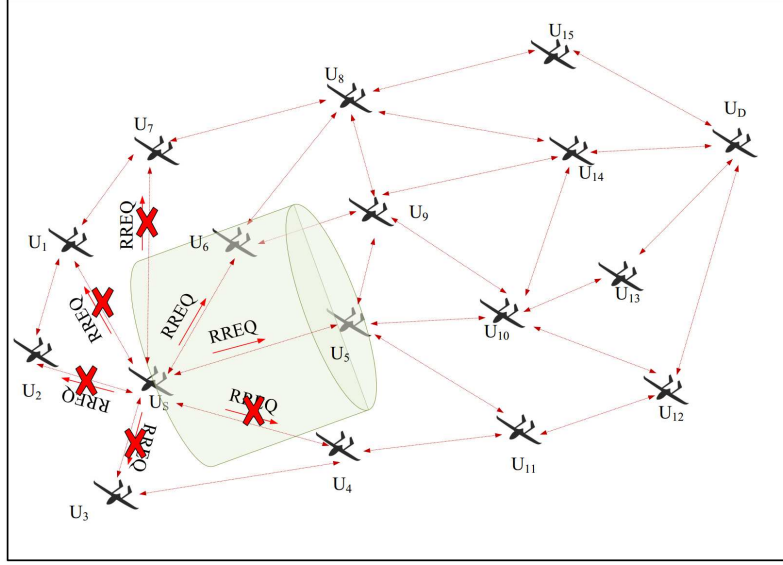$$y'_j = y_j - y_i$$
$$z'_j = z_j - z_i$$

(15)

Figure 12. Route discovery process between U_S and U_D.

- **Step 2:** U_j rotates its spatial coordinates (x'_j, y'_j, z'_j) in the coordinate system with the origin $U_i(x_i, y_i, z_i)$ based on the angle α (calculated based on Equation 11). This process is presented in Equation 16.

$$\begin{bmatrix} x''_j \\ y''_j \\ z''_j \end{bmatrix} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x'_j \\ y'_j \\ z'_j \end{bmatrix} \tag{16}$$

Hence,

$$\begin{aligned} x''_j &= x'_j \cos\alpha - y'_j \sin\alpha \\ y''_j &= x'_j \sin\alpha + y'_j \cos\alpha \\ z''_j &= z' \end{aligned} \tag{17}$$

- **Step 3:** U_j converts its spatial coordinates (x''_j, y''_j, z''_j) into cylindrical coordinates using Equation 18.

$$\begin{aligned} \rho_j &= \sqrt{x''^2_j + y''^2_j} \\ \phi_j &= \arctan\left(\frac{y''_j}{x''_j}\right) \\ z_j &= z''_j \end{aligned} \tag{18}$$

- **Step 4:** If the cylindrical coordinates of U_j, i.e. (ρ_j, ϕ_j, z_j), satisfy the inequality presented in Equation 19, U_j is inside the local filtering zone of its previous-hop node (U_i). Hence, it rebroadcasts the RREQ packet. Otherwise, the RREQ packet is deleted.

$$0 \le \rho_j \le R, 0 \le z_j \le R, 0 \le \phi_j \le 2\pi \tag{19}$$

In the example presented in Figure 12, only two UAVs, namely U_5 and U_6, are allowed to rebroadcast the RREQ packet. Now, these nodes update some fields of this RREQ packet. To generally express how the RREQ packet is updated, assume that the previous-hop UAV and the next-hop one are U_i and U_j, respectively. U_j updates the fields of the RREQ packet below.

- **Number of hops (HC_r):** For updating this field of the RREQ packet, U_j must add one unit to HC_r. This is stated in Equation 20.

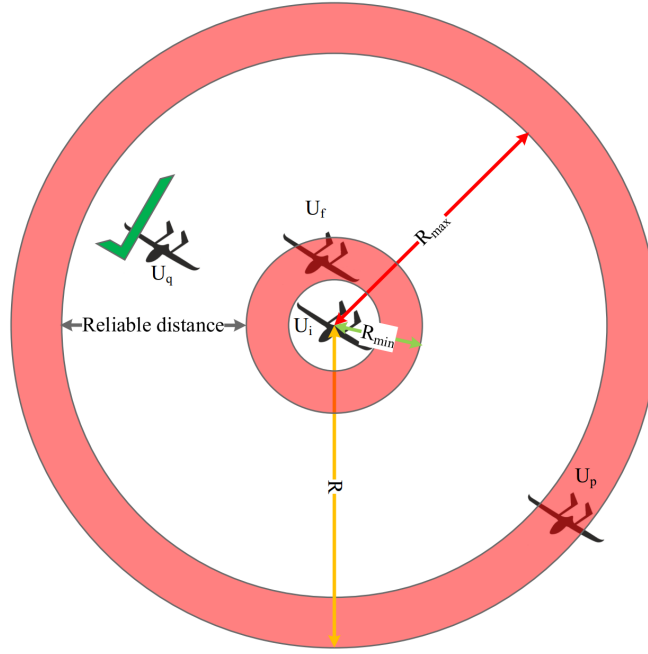$$HC_r = HC_r + 1 \tag{20}$$

Figure 13. Concept of reliable distance in the communication range of U_i.

- **Route energy (E_r):** U_j obtains its current energy capacity and normalizes it according to Equation 21.

$$E_j^{norm} = \frac{E_j}{E_{max}} \tag{21}$$

where E_j indicates the residual energy of U_j and E_{max} denotes the initial energy of UAVs. Now, U_j compares E_j^{norm} with the value inserted in the field E_r. If $E_j^{norm} < E_r$, then U_j updates E_r in the RREQ packet and inserts E_j^{norm} into this field.

- **Reliable distance (Ψ_r):** To update this field in the RREQ packet, U_j calculates the distance to U_i (i.e. its previous-hop node) based on Equation 22.

$$D_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \tag{22}$$

Here, (x_i, y_i, z_i) and (x_j, y_j, z_j) show the spatial coordinates of U_i and U_j, respectively.

Now, U_j obtains its reliable distance (i.e. Ψ_{ij}) based on D_{ij}. Figure 13 clarifies this concept. In this figure, R is the communication radius of UAVs. According to this figure, the three UAVs, namely U_f, U_p, and U_q are in the communication range of U_i. U_f and U_i have a very short distance from each other i.e. $D_{if} \leq R_{min}$. In this case, if U_f is chosen as the next-hop node, there are a high number of hops in the communication route. This increases latency in the routing path. On the other hand, U_p is very close to the communication boundary of U_i, i.e. $R_{max} < D_{ip} \leq R$. If U_p is chosen as the next-hop node, this path has low PDR. This is because U_p may leave the communication range of U_i, and the created path is extremely unstable. However, the distance between U_q and U_i is reliable, i.e. $R_{min} \leq D_{iq} \leq R_{max}$. If U_q is chosen as the next-hop node, the communication path is stable, and data packets arrive at the destination successfully. R_{min} and R_{max} are the lower and upper borders of Ψ_{ij}, and their values are empirically selected. In[35], the authors stated that if $R_{max} = 0.9R$, the data delivery ratio reaches 100%. In addition, the lower border is empirically considered $R_{min} = 0.1R$. According to the points mentioned above, Ψ_{ij} is calculated through Equation 23.

$$\Psi_{ij} = \begin{cases} \frac{D_{ij}}{R_{min}}, & 0 < D_{ij} < R_{min} \\ 1, & R_{min} \leq D_{ij} \leq R_{max} \\ 1 - \left(\frac{D_{ij} - R_{max}}{R - R_{max}} \right), & R_{max} < D_{ij} \leq R \end{cases} \tag{23}$$

Now, U_j compares Ψ_{ij} to the value inserted into the field Ψ_r. If $\Psi_{ij} < \Psi_r$, then U_j updates Ψ_r and inserts Ψ_{ij} into this field.

- **Movement similarity (ϑ_r):** To update this field in the RREQ packet, the similarity between the two speed vectors of U_j and U_i is calculated through Equation 24.

$$cs_{\overrightarrow{V_{ij}}} = \frac{V_i^x V_j^x + V_i^y V_j^y + V_i^z V_j^z}{\sqrt{(V_i^x)^2 + (V_i^y)^2 + (V_i^z)^2} \times \sqrt{\left(V_j^x\right)^2 + \left(V_j^y\right)^2 + \left(V_j^z\right)^2}} \tag{24}$$

where $\left(V_i^x, V_i^y, V_i^z\right)$ and $\left(V_j^x, V_j^y, V_j^z\right)$ show two speed vectors of U_i and U_j, respectively. They are calculated through Equations 25 and 26, respectively.

$$\begin{aligned} V_i^x &= V_i \sin \varphi_i \cos \theta_i \\ V_i^y &= V_i \sin \varphi_i \sin \theta_i \\ V_i^z &= V_i \cos \varphi_i \end{aligned} \tag{25}$$

$$\begin{aligned} V_j^x &= V_j \sin \varphi_j \cos \theta_j \\ V_j^y &= V_j \sin \varphi_j \sin \theta_j \\ V_j^z &= V_j \cos \varphi_j \end{aligned} \tag{26}$$

where $(V_i, \theta_i, \varphi_i)$ and $(V_j, \theta_j, \varphi_j)$ indicate the speed information of U_i and U_j in the neighbor table, respectively. Note that $cs_{\overrightarrow{V_{ij}}}$ is limited to $[-1, 1]$, but in LFEAR, this parameter is normalized through Equation 27 to be limited to $[0, 1]$.

$$CS_{\overrightarrow{V_{ij}}} = \frac{cs_{\overrightarrow{V_{ij}}} + 1}{2} \tag{27}$$

Now, U_j compares $CS_{\overrightarrow{V_{ij}}}$ to the value inserted into the field ϑ_r. If $CS_{\overrightarrow{V_{ij}}} < \vartheta_r$, then U_j updates ϑ_r and records $CS_{\overrightarrow{V_{ij}}}$ in it.

After updating the RREQ packet, U_5 broadcasts the RREQ packet for its neighbors, namely U_9, U_{10}, and U_{11}. This is shown in Figure 14 (a). In this figure, U_{11} is outside the filtering area and removes the RREQ packet. U_6 transmits the RREQ packet to its neighboring UAVs, namely U_8 and U_9. This process is shown in Figure 14 (b). In this figure, U_8 is outside the local filtering and removes the RREQ packet, but U_9 rebroadcasts this packet. In the next step, U_9 and U_{10} execute the RREQ rebroadcast process, which is displayed in Figure 15 (a) and Figure 15 (b), respectively. At the last step, U_{13} and U_{14} execute the RREQ rebroadcast process, which is depicted in Figure 16 (a) and Figure 16 (b), respectively. Finally, the RREQ packet reaches U_D, and the path discovery process ends. The pseudo-code of this process is stated in Algorithm 3.

4.2.4 Selection of the best route

After ending the route discovery procedure, the two paths are constructed between U_S and U_D. As shown in Figure 17, these two paths are $PATH1 : U_S - U_5 - U_{10} - U_{13} - U_D$ and $PATH2 : U_S - U_6 - U_9 - U_{14} - U_D$.

Now, U_D begins the route selection process. In the current phase, U_D should choose the best route from the constructed paths. Therefore, U_D extracts information about each route P_r, namely, HC_r, E_r, Ψ_r, and ϑ_r from the relevant RREQ packet and calculates the score S_r based on Equation 28.

$$S_r = \ell_1 \left(1 - \frac{HC_r}{n-1}\right) + \ell_2 E_r + \ell_3 \Psi_r + \ell_4 \vartheta_r \tag{28}$$

where ℓ_1, ℓ_2, ℓ_3, and ℓ_4 are weight coefficients in $[0, 1]$ so that $\ell_1 + \ell_2 + \ell_3 + \ell_4 = 1$. In LFEAR, these coefficients have the same value i.e. $\ell_1 = \ell_2 = \ell_3 = \ell_4 = \frac{1}{4}$. Also, n is the number of UAVs in the network.

Then, U_D selects P_r with the highest route and sends the RREP packet to U_S through it. In Figure 18, U_D chooses $PATH1 : U_S - U_5 - U_{10} - U_{13} - U_D$. Then, U_S keeps this path in its neighbor table to exploit it for communicating with U_D. The pseudo-code of this process is stated in Algorithm 4.

5 Simulation and result evaluation

In this section, the simulation process of LFEAR is described using the NS2 simulator. In the simulation environment, the length, width, and height of the network are $X = 10000\,m$, $Y = 10000\,m$, and $Z = 10000\,m$, respectively. In this environment, the number of flying nodes varies from 5 to 30, and the speed of these nodes is limited to $10 - 110\,m/s$. The movement pattern
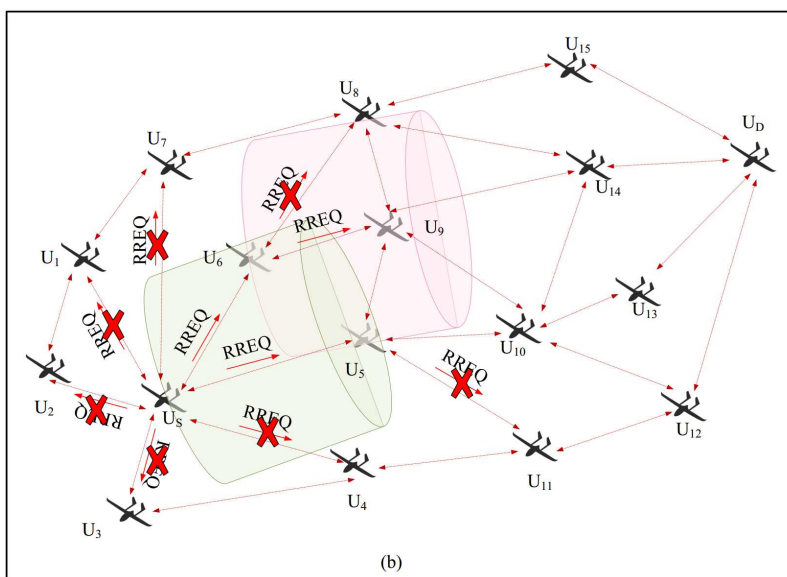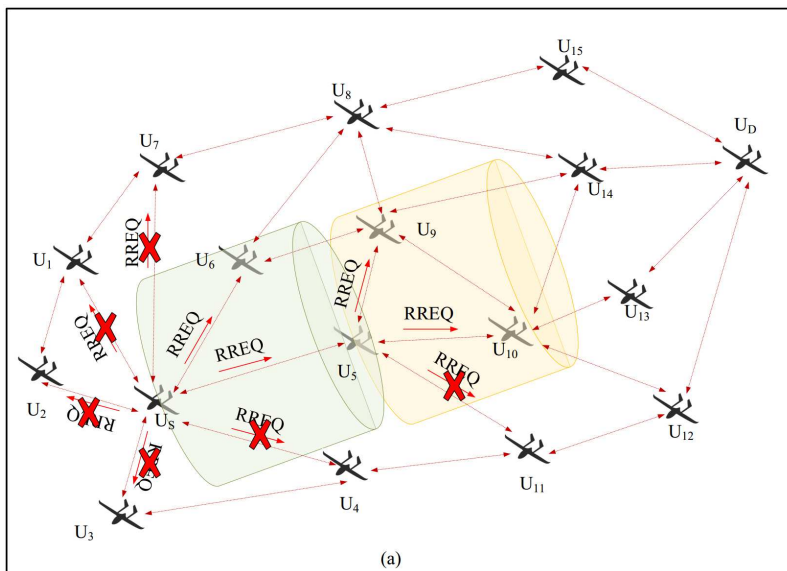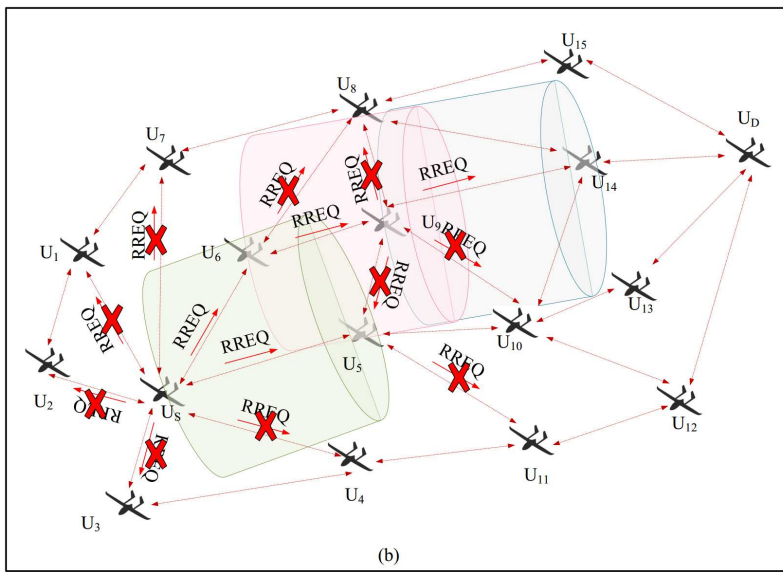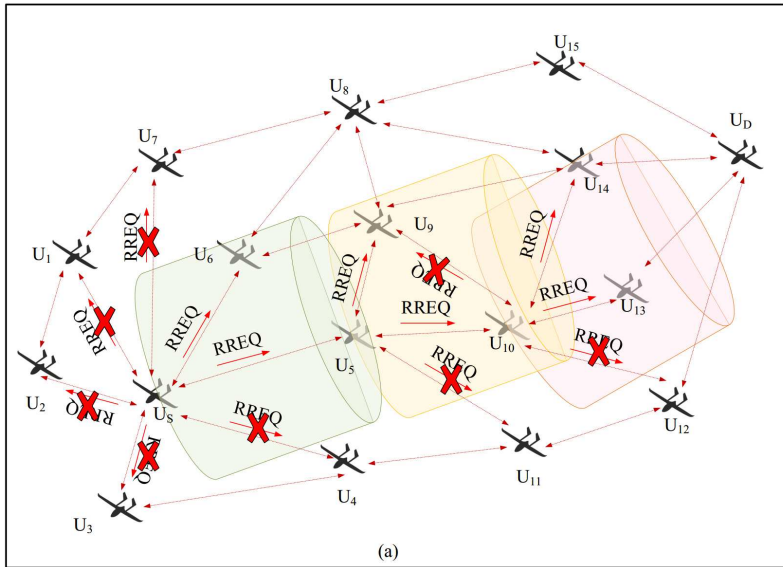
Figure 14. Route discovery process in the second step.

Figure 15. Route discovery process in the third step.
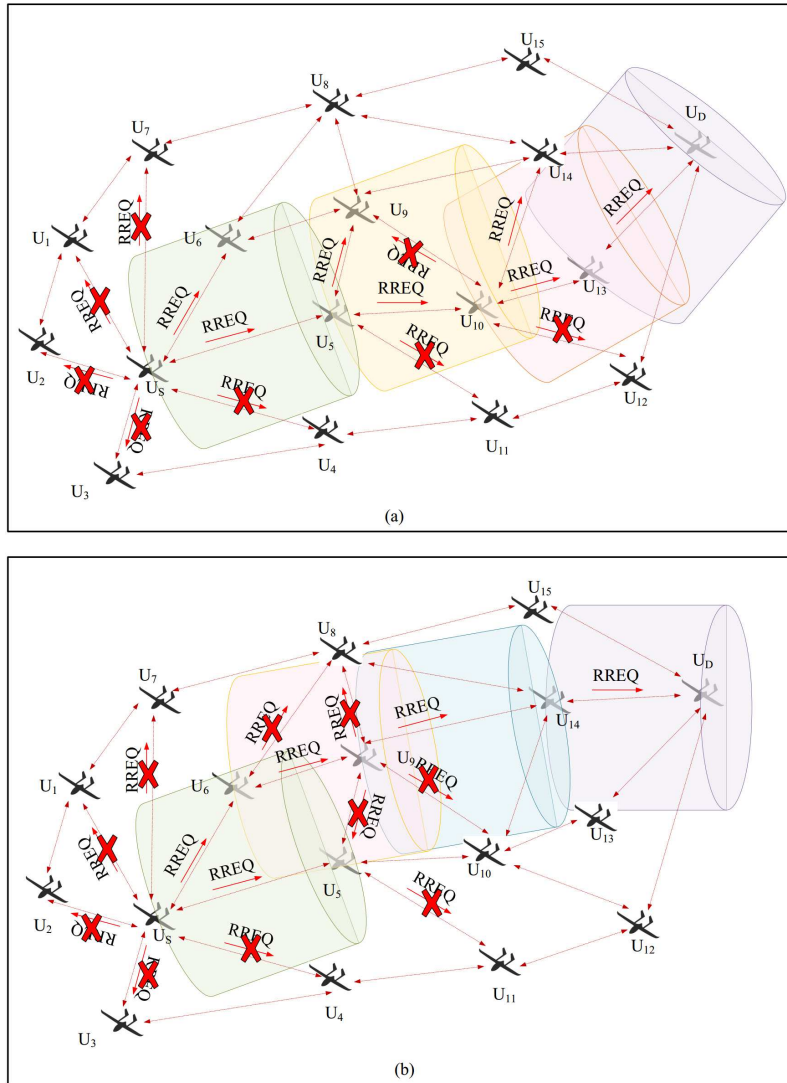
Figure 16. Route discovery process in the fourth step.

Algorithm 3 Route finding

Input: U_S: Source UAV
$\quad\quad$ U_D: Destination UAV
Output: Creating a route from U_S to U_D
$\quad$ **Begin**
1: **if** U_S and U_D attempt to communicate with each other **and** U_S cannot find valid route to U_D **then**
2: $\quad$ **U$_S$:** Make a route request (RREQ) packet in accordance with Figure 6;
3: $\quad$ **U$_S$:** Disseminate RREQ around its communication area;
4: $\quad$ **U$_S$:** Nominate itself (i.e. current hop) as U_i;
5: $\quad$ **while** U_D receives RREQ **do**
6: $\quad\quad$ **if** U_j belongs to NT_i **and** U_j receives RREQ from U_i **then**
7: $\quad\quad\quad$ **U$_j$:** Calculate its transferred spatial coordinates using Equation 15;
8: $\quad\quad\quad$ **U$_j$:** Rotate its transferred spatial coordinates according to Equations 16 and 17;
9: $\quad\quad\quad$ **U$_j$:** Convert its rotated spatial coordinate to cylindrical coordinates using Equation 18;
10: $\quad\quad\quad$ **if** the cylindrical coordinates of U_j meets Equation 19 **then**
11: $\quad\quad\quad\quad$ **U$_j$:** Refresh HC_r based on Equation 20;
12: $\quad\quad\quad\quad$ **U$_j$:** Calculate its energy level (E_j^{norm}) based on Equation 21;
13: $\quad\quad\quad\quad$ **if** $E_j^{norm} < E_r$ **then**
14: $\quad\quad\quad\quad\quad$ **U$_j$:** Refresh E_r based on E_j^{norm} in RREQ packet;
15: $\quad\quad\quad\quad$ **end if**
16: $\quad\quad\quad\quad$ **U$_j$:** Calculate the Euclidean distance between U_i and U_j based on Equation 22;
17: $\quad\quad\quad\quad$ **U$_j$:** Obtain the reliable distance between U_i and U_j (Ψ_{ij}) using Equation 23;
18: $\quad\quad\quad\quad$ **if** $\Psi_{ij} < \Psi_r$ **then**
19: $\quad\quad\quad\quad\quad$ **U$_j$:** Replace Ψ_r with Ψ_{ij} in RREQ packet;
20: $\quad\quad\quad\quad$ **end if**
21: $\quad\quad\quad\quad$ **U$_j$:** Calculate the velocity vector of U_i i.e. (V_i^x, V_i^y, V_i^z) using Equation 25;
22: $\quad\quad\quad\quad$ **U$_j$:** Compute the velocity vector of U_j i.e. $\left(V_j^x, V_j^y, V_j^z\right)$ using Equation 26;
23: $\quad\quad\quad\quad$ **U$_j$:** Calculate the similarity between (V_i^x, V_i^y, V_i^z) and $\left(V_j^x, V_j^y, V_j^z\right)$ using Equation 24;
24: $\quad\quad\quad\quad$ **U$_j$:** Normalize the velocity similarity $cs_{\overrightarrow{V_{ij}}}$ based on Equation 27;
25: $\quad\quad\quad\quad$ **if** $CS_{\overrightarrow{V_{ij}}} < \vartheta_r$ **then**
26: $\quad\quad\quad\quad\quad$ **U$_j$:** Update ϑ_r with regard to $CS_{\overrightarrow{V_{ij}}}$ in RREQ packet;
27: $\quad\quad\quad\quad$ **end if**
28: $\quad\quad\quad\quad$ **U$_j$:** Disseminate the updated RREQ packet around its communication area;
29: $\quad\quad\quad\quad$ **U$_j$:** Nominate itself (i.e. current hop) as U_i;
30: $\quad\quad\quad$ **else**
31: $\quad\quad\quad\quad$ **U$_j$:** Discard RREQ packet;
32: $\quad\quad\quad$ **end if**
33: $\quad\quad$ **end if**
34: $\quad$ **end while**
35: **end if**
$\quad$ **End**

Algorithm 4 Selection the best route

Input: P_r: Discovered routes in Algorithm 3 so that $r = 1, ..., f$
Output: Selection the best route from U_S to U_D
$\quad$ **Begin**
1: **for** $r = 1$ to f **do**
2: $\quad$ **U$_D$:** Calculate S_r for P_r according Equation 28;
3: **end for**
4: **U$_D$:** Order the discovered routes P_r based on S_r;
5: **U$_D$:** Select P_r with the maximum S_r as the final route between U_S and U_D;
6: **U$_D$:** Make a route reply packet (RREP) in accordance with Figure 7;
7: **U$_D$:** Send back RREP to U_S via the final route;
8: **U$_S$:** Save the final route in its routing table;
9: **U$_S$:** Communicate with U_D and transfer data through the final route;
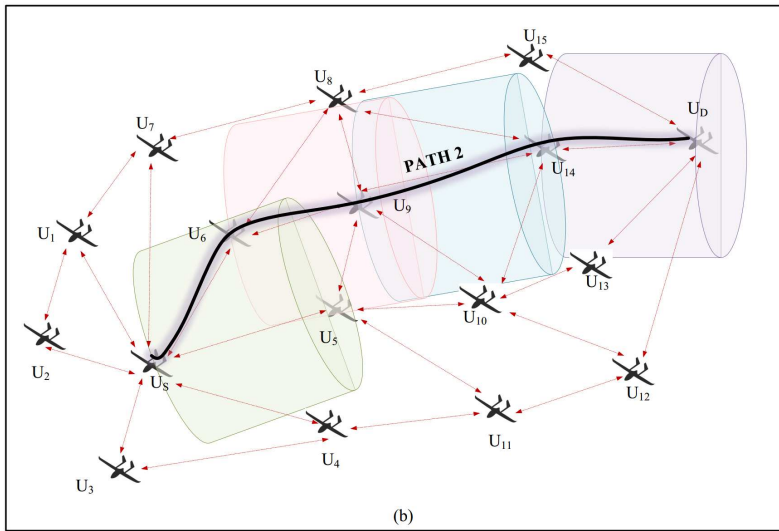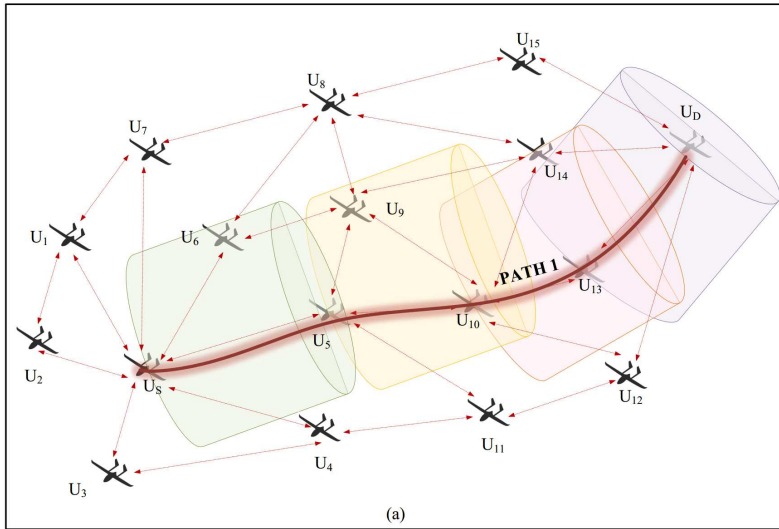$\quad$ **End**

Figure 17. Construction of two routes between U_S and U_D.
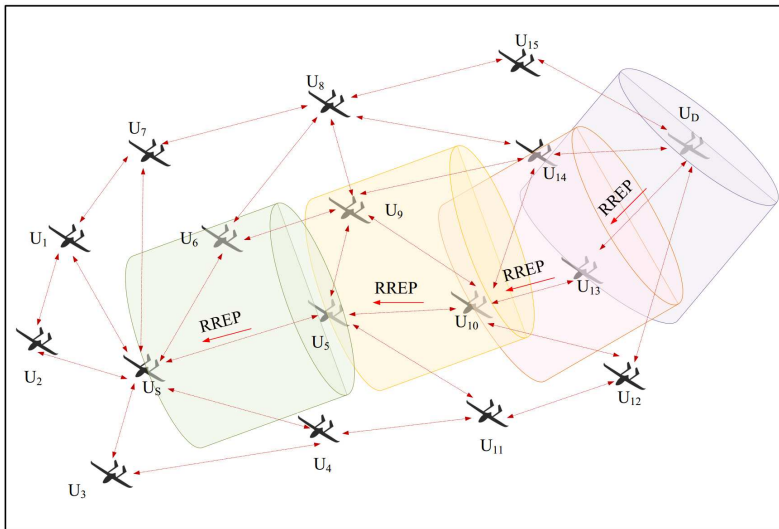
Figure 18. Selection of the best route.

Table 2. Simulation settings.

Parameter	Value
Simulation software	NS2
Simulation environment	$10000 \times 10000 \times 10000\,m^3$
Number of UAVs	5-30 nodes
Maximum energy of UAVs	100 J
Speed of UAVs	$10 - 110\,m/s$
Communication of UAVs	250 m
Mobility model	Random waypoint (RWP)
Transmission power	10 dBm
Evaluation criteria	Energy consumption, data delivery rate, network lifespan, routing overhead, delay
Compared schemes	LFEAR, EARVRT, LoCaL, and O-LAR
Antenna	Omni-Antenna
Runtime	150 s
Mac standard	IEEE 802.11g
Traffic model	Constant bit rate (CBR)
Packet size	512 bytes
Traffic rate	2 Mbps

of these nodes in the network environment is defined based on the random waypoint mobility model. Each flying node has a 250-meter communication radius, and its initial energy equals 100 joules. The total simulation time is 150 seconds. The traffic model is defined according to the constant bit rate model (CBR). In this model, data packets are 512 bytes, and the bit rate is 2 Mbps. Table 2 introduces the most important simulation parameters. Note that two simulation scenarios are considered to accurately evaluate network performance in different methods. The first scenario assumes that network density varies from 5 to 30 nodes, and these flying nodes have a speed of $60\,m/s$. The second scenario assumes that the speed of flying nodes varies from 10 to 110 m/s, while the network density is adjusted to 10 nodes. Then, network performance is evaluated in terms of five criteria, namely energy consumption, data delivery rate, network lifespan, routing overhead, and delay. These simulation results compare and evaluate the four routing methods, including LFEAR, EARVRT[39], LoCaL[40], and O-LAR[41].

5.1 Energy consumption

Figure 19 compares several routing approaches in terms of energy consumption in the first simulation scenario. In this figure, energy consumption in LFEAR is 1.33%, 11.59%, and 24.45% less than that in EARVRT, LoCaL, and O-LAR, respectively. Figure 20 also shows the energy consumption of different methods in the second simulation scenario. According to this figure, LFEAR reduces energy consumption by 5.55%, 6.98%, and 16.79% compared to EARVRT, LoCaL, and O-LAR, respectively. There are several reasons, which are addressed below. In LFEAR, the number of relay nodes in the route discovery process is managed using local filtering. Hence, the broadcast range of the RREQ packet is limited in the network and the broadcasting storm issue in FANET does not occur. This can improve energy consumption in the network. Furthermore, in the routing process, the RREQ packet stores the route energy parameter, which shows the lowest energy level of UAVs in this route. This field determines the energy crisis point in the routing path. In the route selection process, the destination node seeks to find the route with the best energy. This increases the stability of the selected path and improves energy consumption in the network. In addition, this reduces the number of low-energy UAVs participating in the route construction process. As a result, the energy consumption of UAVs is balanced in the network. This enhances network lifespan. On the other hand, LFEAR considers the three parameters, namely the route energy, reliable distance, and movement similarity, in the path selection process. The purpose of these three parameters is to increase the stability of the constructed paths to balance the energy consumption of UAVs and improve network lifespan. Among the different routing approaches, EARVRT also works well in terms of energy consumption because it is an energy-aware routing approach and restricts the broadcasting range of RREQs through a virtual tunnel. Additionally, LoCaL utilizes a filtering technique to limit RREQs. However, it does not guarantee path stability. As a result, it is weaker than LFEAR in terms of energy consumption. O-LAR has the worst energy consumption because the filtering technique used in this scheme does not work well.

5.2 Routing overhead

Figure 21 compares the routing schemes in terms of routing overhead in the first simulation scenario. According to this figure, the routing overhead in LFEAR is 16.51% more than that in EARVRT. However, it is 2.33% and 23.03% lower than that in LoCaL and O-LAR, respectively. Figure 22 evaluates the routing approaches in the second simulation scenario. As shown in this figure, the routing overhead in LFEAR is about 23% more than that in EARVRT. However, LFEAR decreases routing overhead by 1.6% and 26.79% in comparison with LoCaL and O-LAR, respectively. Thus, the performance of the proposed method is weaker than EARVRT because LFEAR uses beacon messages to obtain local network topology, but EARVRT does not require these messages. However, LFEAR has less routing overhead than LoCaL and O-LAR. This is because it controls the broadcasting range of RREQs using local filtering. On the other hand, LFEAR takes into consideration three parameters,
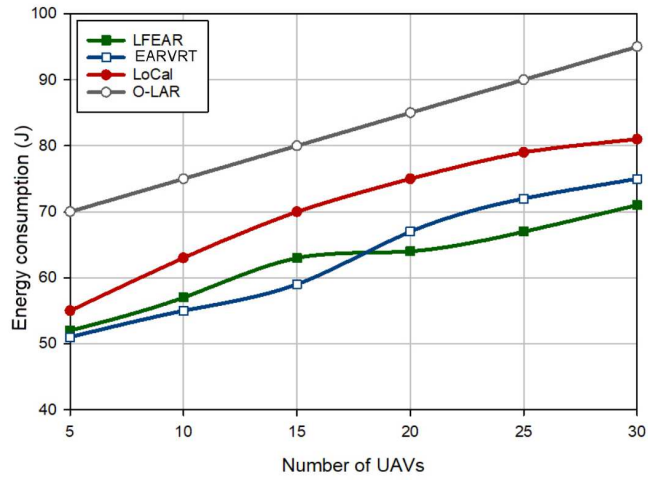
Figure 19. Energy consumption in the first simulation scenario.

Figure 20. Energy consumption in the second simulation scenario.
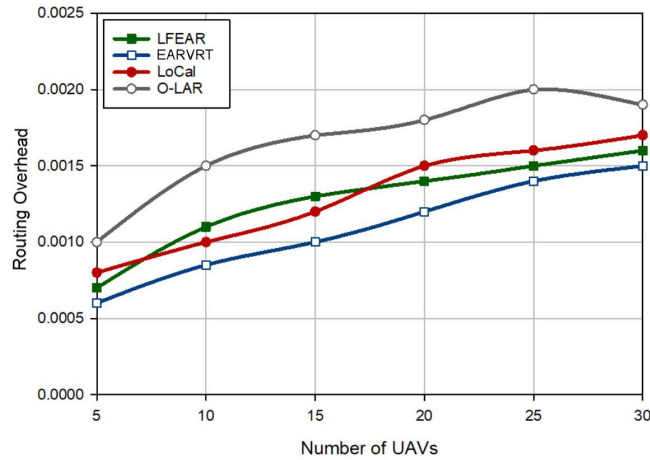
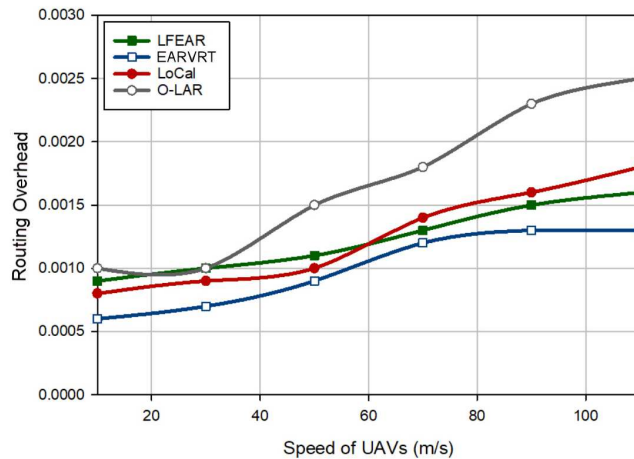**Figure 21.** Routing overhead in the first simulation scenario.

Figure 22. Routing overhead in the second simulation scenario.

including route energy, reliable distance, and movement similarity in the path selection process. These parameters help LFEAR find more stable and energy-efficient paths in FANET. This reduces the number of failed routes in the network. Hence, LFEAR lowers the need to find new paths and rebuild failed routes. As a result, routing overhead is better than LoCaL and O-LAR.

5.3 Packet delivery rate

Figure 23 evaluates the routing approaches in terms of packet delivery rate in the first simulation scenario. According to this evaluation, LFEAR increases PDR by 1.77%, 5.14%, and 13.04% compared to EARVRT, LoCaL, and O-LAR, respectively. Figure 24 also tests the routing approaches in terms of PDR in the second simulation scenario. This experiment shows that the performance of the proposed scheme is weaker than EARVRT at high speeds, and it has less PDR (approximately 2.29%) than EARVRT. This is because LFEAR is dependent on beacon messages to access local network topology. At high speeds, the network topology experiences more changes. Therefore, this increases errors in the routing process. As a result, the constructed paths are volatile and may be cut off. This lowers PDR. However, EARVRT does not need information about local network topology. Thus, topological changes in the network have less effect on the performance of this method. However, in the second simulation scenario (Figure 24), LFEAR has a higher PDR than LoCaL and O-LAR. This is because LFEAR uses a criterion called reliable distance in the RREQ packet. This criterion attempts to choose two consecutive UAVs in a routing path in such a way that minimizes the likelihood of leaving the communication range of each other. This means that these UAVs are not close to the communication boundary of each other. As a result, the communication path is valid for a longer time. This improves PDR in the network. In addition, another criterion, namely movement similarity, is inserted into the RREQ packet. This criterion indicates the speed similarity of intermediate UAVs. The purpose of this field is to select the intermediate nodes with similar speeds in the routing path to form stable paths in FANET. In this case, data packets reach the destination successfully.
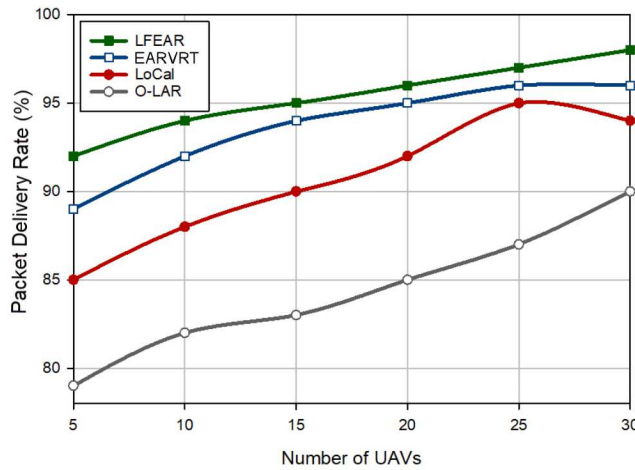
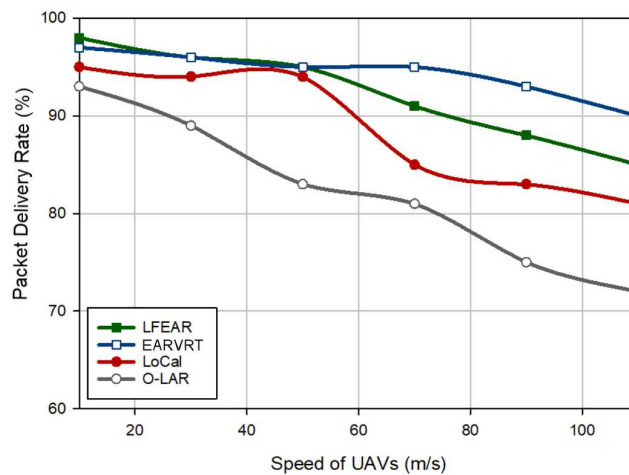**Figure 23.** Packet delivery rate in the first simulation scenario.

Figure 24. Packet delivery rate in the second simulation scenario.

Considering the energy criterion in the routing process helps LFEAR to build stable routes and increase PDR in the network.

5.4 Network lifetime

Figure 25 compares the routing approaches in terms of network lifetime (when the first node dies (FND) in the network) in the first simulation scenario. According to this figure, LFEAR increases network lifespan by 6.74%, 3.15%, and 10.85% compared to EARVRT, LoCaL, and O-LAR, respectively. In addition, Figure 26 tests different routing schemes in terms of network lifespan in the second simulation scenario. The experiment shows that the network lifetime in LFEAR is about 5.67%, 4.64%, and 22.63% more than that in EARVRT, LoCaL, and O-LAR, respectively. The most important reason is that LFEAR can optimize energy consumption in the network so that UAVs consume energy uniformly. Other reasons for this are mentioned in Section 5.1. On the other hand, LFEAR uses a new criterion called reliable distance to choose optimal paths. This criterion causes the two consecutive UAVs in a routing path to be not close to each other. This minimizes the likelihood of leaving the communication range of each other and increases the stability of the constructed paths. As a result, UAVs consume less energy to discover new routes or rebuild failed paths. This improves network lifetime. LFEAR also uses another criterion called movement similarity in the route selection process. This criterion causes intermediate nodes with almost similar speeds to be chosen in the routing path, and consequently, stable paths are constructed in FANET. This stability has a good effect on improving network lifetime because communication paths are valid for a longer time, and LFEAR has little need to find new routes.
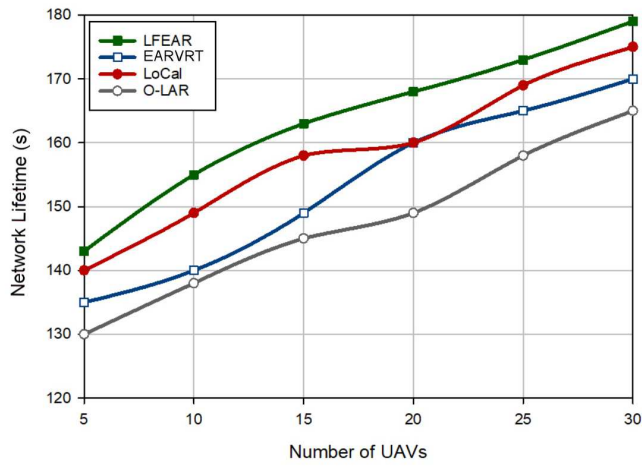
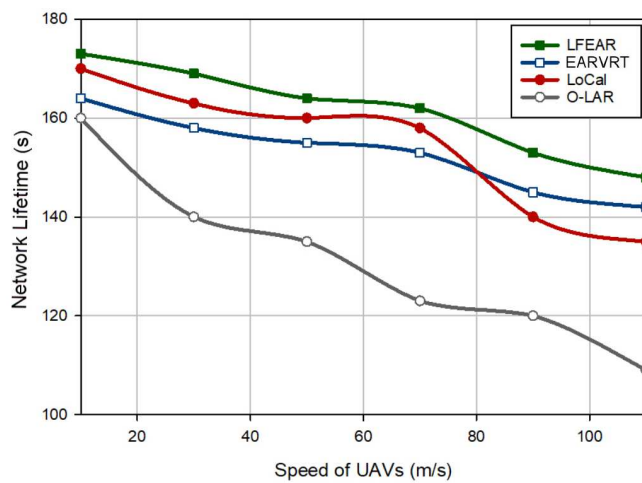**Figure 25.** Network lifetime in the first simulation scenario.

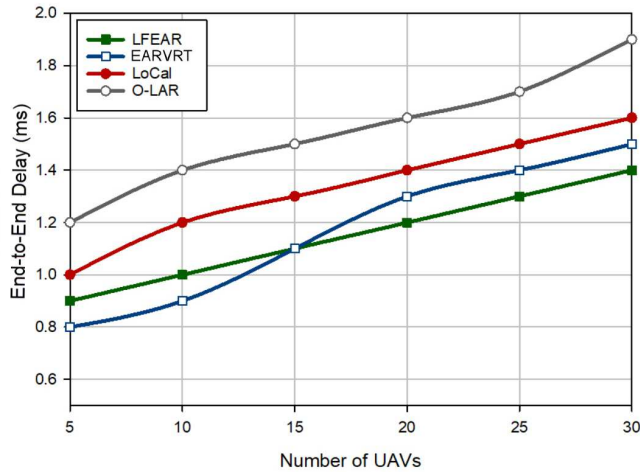**Figure 26.** Network lifetime in the second simulation scenario.

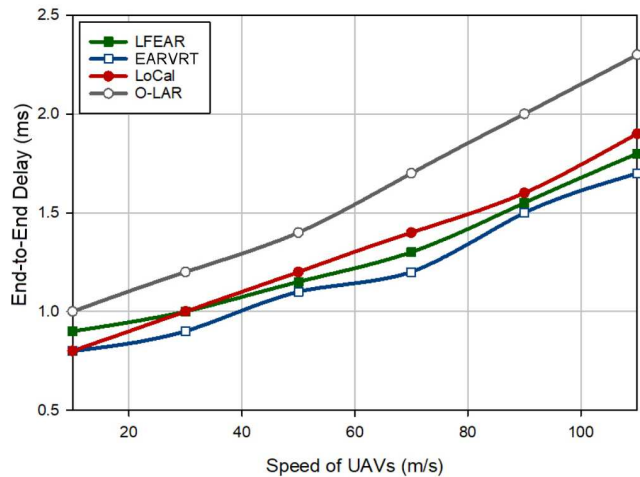**Figure 27.** Delay in the first simulation scenario.

Figure 28. Delay in the second simulation scenario.

5.5 Delay

Figure 27 evaluates the routine approaches in terms of delay in the first simulation scenario. This figure shows that LFEAR reduces latency by 1.71%, 13.53%, and 25.81% compared to EARVRT, LoCaL, and O-LAR, respectively. In addition, Figure 28 compares the routing schemes in terms of delay in the second simulation scenario. According to this figure, the delay in LFEAR is about 6.67% more than that in EARVRT. However, it is about 3.03% and 20% better than that in LoCaL and O-LAR, respectively. The proposed scheme has weaker performance than EARVRT. This is because LFEAR has a neighboring table construction phase, which does not exist in EARVRT. This phase leads to accurate decisions in the route discovery process, but it increases delay. However, the better performance of the proposed scheme compared to LoCaL and O-LAR is due to local filtering, which controls the broadcasting range of RREQs in the network. This accelerates the path discovery process in the network. On the other hand, the route selection process considers the three parameters, including energy, reliable distance, and movement similarity, and consequently, it has a good ability to find stable routes for FANET. This reduces the need for rebuilding the broken paths and lowers delay in the network.

6 Conclusion

In this paper, a local filtering-based energy-aware routing scheme (LFEAR) was suggested for FANETs. LFEAR seeks to improve the adaptability of AODV to FANET and solve the broadcasting storm issue in AODV. Overall, LFEAR consists of two phases: neighbor table construction and route discovery. To solve the broadcasting storm issue, LFEAR uses local filtering to limit the broadcasting range of RREQs. Additionally, LFEAR improves the template of RREQ by adding three other

fields, namely route energy, reliable distance, and movement similarity to create stable and energy-efficient routes between UAVs. After discovering different paths, the route selection process begins, and a score is calculated for each path. This score combines the number of hops, route energy, reliable distance, and movement similarity. Finally, the route with the highest score performs the data transmission process. Finally, the simulation process of LFEAR is performed using the NS2 simulator, and two simulation scenarios are defined based on the network density and the speed of UAVs. In the first scenario, LFEAR improves energy consumption, PDR, network lifetime, and delay by 1.33%, 1.77%, 6.74%, and 1.71%, respectively. However, routing overhead in LFEAR is approximately 16.51% more than that in EARVRT. In the second scenario, LFEAR optimizes energy consumption and network lifetime by 5.55% and 5.67% respectively, but its performance in terms of routing overhead, PDR, and delay is 23%, 2.29%, and 67.67% weaker than EARVRT. In future research directions, we attempt to increase the adaptability of the proposed routing to the dynamic environment of FANET and improve its performance at high speeds. In this regard, future research attempts to decrease routing overhead in the network by adjusting an adaptive beacon period. To design this period, swarm intelligence techniques and machine learning (ML) can be used.

References

1. Lee, S.W., Ali, S., Yousefpoor, M.S., Yousefpoor, E., Lalbakhsh, P., Javaheri, D., Rahmani, A.M. and Hosseinzadeh, M., 2021. An energy-aware and predictive fuzzy logic-based routing scheme in flying ad hoc networks (fanets). IEEE Access, 9, pp.129977-130005. doi: 10.1109/ACCESS.2021.3111444.

2. Rahmani, A.M., Ali, S., Yousefpoor, E., Yousefpoor, M.S., Javaheri, D., Lalbakhsh, P., Ahmed, O.H., Hosseinzadeh, M. and Lee, S.W., 2022. OLSR+: A new routing method based on fuzzy logic in flying ad-hoc networks (FANETs). Vehicular Communications, 36, p.100489. doi: 10.1016/j.vehcom.2022.100489.

3. Shokrollahi, S. and Dehghan, M., 2023. TGRV: A trust-based geographic routing protocol for VANETs. Ad Hoc Networks, 140, p.103062. doi: 10.1016/j.adhoc.2022.103062.

4. Pasandideh, F., da Costa, J.P.J., Kunst, R., Islam, N., Hardjawana, W. and Pignaton de Freitas, E., 2022. A review of flying ad hoc networks: Key characteristics, applications, and wireless technologies. Remote Sensing, 14(18), p.4459. doi: 10.3390/rs14184459.

5. Mansoor, N., Hossain, M.I., Rozario, A., Zareei, M. and Arreola, A.R., 2023. A Fresh Look at Routing Protocols in Unmanned Aerial Vehicular Networks: A Survey. IEEE Access. doi: 10.1109/ACCESS.2023.3290871.

6. da Costa, L.A.L., Kunst, R. and de Freitas, E.P., 2021. Q-FANET: Improved Q-learning based routing protocol for FANETs. Computer Networks, 198, p.108379. doi: 10.1016/j.comnet.2021.108379.

7. Hosseinzadeh, M., Ali, S., Ionescu-Feleaga, L., Ionescu, B.S., Yousefpoor, M.S., Yousefpoor, E., Ahmed, O.H., Rahmani, A.M. and Mehmood, A., 2023. A novel Q-learning-based routing scheme using an intelligent filtering algorithm for flying ad hoc networks (FANETs). Journal of King Saud University-Computer and Information Sciences, 35(10), p.101817. doi: 10.1016/j.jksuci.2023.101817.

8. Mohsan, S.A.H., Othman, N.Q.H., Li, Y., Alsharif, M.H. and Khan, M.A., 2023. Unmanned aerial vehicles (UAVs): Practical aspects, applications, open challenges, security issues, and future trends. Intelligent Service Robotics, 16(1), pp.109-137. doi: 10.1007/s11370-022-00452-4.

9. Mukherjee, A., Panja, A.K., Dey, N. and Crespo, R.G., 2023. An intelligent edge enabled 6G-flying ad-hoc network ecosystem for precision agriculture. Expert Systems, 40(4), p.e13090. doi: 10.1111/exsy.13090.

10. Ghosh, A., Mistry, C. and Biswas, M., 2022. An Extensive Analysis of Flying Ad-Hoc Network Applications and Routing Protocols in Agriculture. In Artificial Intelligence Applications in Agriculture and Food Quality Improvement (pp. 129-147). IGI Global. doi: 10.4018/978-1-6684-5141-0.ch008.

11. Kumar, P. and Verma, S., 2022. Implementation of modified OLSR protocol in AANETs for UDP and TCP environment. Journal of King Saud University-Computer and Information Sciences, 34(4), pp.1305-1311. doi: 10.1016/j.jksuci.2019.07.009.

12. Lansky, J., Rahmani, A.M., Zandavi, S.M., Chung, V., Yousefpoor, E., Yousefpoor, M.S., Khan, F. and Hosseinzadeh, M., 2022. A Q-learning-based routing scheme for smart air quality monitoring system using flying ad hoc networks. Scientific Reports, 12(1), p.20184. doi: 10.1038/s41598-022-20353-x.

13. Lansky, J., Rahmani, A.M., Malik, M.H., Yousefpoor, E., Yousefpoor, M.S., Khan, M.U. and Hosseinzadeh, M., 2023. An energy-aware routing method using firefly algorithm for flying ad hoc networks. Scientific Reports, 13(1), p.1323. doi: 10.1038/s41598-023-27567-7.

14. Hosseinzadeh, M., Yoo, J., Ali, S., Lansky, J., Mildeova, S., Yousefpoor, M.S., Ahmed, O.H., Rahmani, A.M. and Tightiz, L., 2023. A cluster-based trusted routing method using fire hawk optimizer (FHO) in wireless sensor networks (WSNs). Scientific Reports, 13(1), p.13046. doi: 10.1038/s41598-023-40273-8.

15. Lansky, J., Ali, S., Rahmani, A.M., Yousefpoor, M.S., Yousefpoor, E., Khan, F. and Hosseinzadeh, M., 2022. Reinforcement learning-based routing protocols in flying ad hoc networks (FANET): A review. Mathematics, 10(16), p.3017. doi: 10.3390/math10163017.

16. Lansky, J., Rahmani, A.M. and Hosseinzadeh, M., 2022. Reinforcement Learning-Based Routing Protocols in Vehicular Ad Hoc Networks for Intelligent Transport System (ITS): A Survey. Mathematics, 10(24), p.4673. doi: 10.3390/math10244673.

17. Perkins, C.E. and Bhagwat, P., 1994. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. ACM SIGCOMM computer communication review, 24(4), pp.234-244. doi: 10.1145/190809.190336.

18. Clausen, T. and Jacquet, P. eds., 2003. RFC3626: Optimized link state routing protocol (OLSR). doi: 10.17487/RFC3626.

19. Johnson, D., Hu, Y.C. and Maltz, D., 2007. The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4 (No. rfc4728). doi: 10.17487/RFC4728.

20. Perkins, C., Belding-Royer, E. and Das, S., 2003. RFC3561: Ad hoc on-demand distance vector (AODV) routing. doi: 10.17487/RFC3561.

21. Park, V.D. and Corson, M.S., 1998, June. A performance comparison of the temporally-ordered routing algorithm and ideal link-state routing. In Proceedings Third IEEE Symposium on Computers and Communications. ISCC'98.(Cat. No. 98EX166) (pp. 592-598). IEEE. doi: 10.1109/ISCC.1998.702600.

22. Karp, B. and Kung, H.T., 2000, August. GPSR: Greedy perimeter stateless routing for wireless networks. In Proceedings of the 6th annual international conference on Mobile computing and networking (pp. 243-254). doi: 10.1145/345910.345953.

23. Alam, M.M. and Moh, S., 2022. Joint topology control and routing in a UAV swarm for crowd surveillance. Journal of Network and Computer Applications, 204, p.103427. doi: 10.1016/j.jnca.2022.103427.

24. Khedr, A.M., Salim, A., PV, P.R. and Osamy, W., 2023. MWCRSF: Mobility-based weighted cluster routing scheme for FANETs. Vehicular Communications, 41, p.100603. doi: 10.1016/j.vehcom.2023.100603.

25. Hosseinzadeh, M., Yoo, J., Ali, S., Lansky, J., Mildeova, S., Yousefpoor, M.S., Ahmed, O.H., Rahmani, A.M. and Tightiz, L., 2023. A fuzzy logic-based secure hierarchical routing scheme using firefly algorithm in Internet of Things for healthcare. Scientific Reports, 13(1), p.11058. doi: 10.1038/s41598-023-38203-9.

26. Zheng, B., Zhuo, K., Zhang, H. and Wu, H.X., 2022. A novel airborne greedy geographic routing protocol for flying Ad hoc networks. Wireless Networks, pp.1-15. doi: 10.1007/s11276-022-03030-9.

27. Derhab, A., Cheikhrouhou, O., Allouch, A., Koubaa, A., Qureshi, B., Ferrag, M.A., Maglaras, L. and Khan, F.A., 2023. Internet of drones security: Taxonomies, open issues, and future directions. Vehicular Communications, 39, p.100552. doi: 10.1016/j.vehcom.2022.100552.

28. Jin, H., Jin, X., Zhou, Y., Guo, P., Ren, J., Yao, J. and Zhang, S., 2023. A survey of energy efficient methods for UAV communication. Vehicular Communications, p.100594. doi: 10.1016/j.vehcom.2023.100594.

29. Shahbazi, M., Simsek, M. and Kantarci, B., 2023. AI-enabled cluster head selection through modified density based clustering in Aeronautical Ad Hoc Networks. Ad Hoc Networks, 148, p.103209. doi: 10.1016/j.adhoc.2023.103209.

30. Ameur, A.I., Lakas, A., Yagoubi, M.B. and Oubbati, O.S., 2022. Peer-to-peer overlay techniques for vehicular ad hoc networks: Survey and challenges. Vehicular Communications, 34, p.100455. doi: 10.1016/j.vehcom.2022.100455.

31. Laghari, A.A., Jumani, A.K., Laghari, R.A. and Nawaz, H., 2023. Unmanned aerial vehicles: A review. Cognitive Robotics, 3. doi: 10.1016/j.cogr.2022.12.004.

32. Hosseinzadeh, M., Mohammed, A.H., Rahmani, A.M., A. Alenizi, F., Zandavi, S.M., Yousefpoor, E., Ahmed, O.H., Hussain Malik, M. and Tightiz, L., 2023. A secure routing approach based on league championship algorithm for wireless body sensor networks in healthcare. Plos one, 18(10), p.e0290119. doi: 10.1371/journal.pone.0290119.

33. Oubbati, O.S., Lakas, A., Zhou, F., Güneş, M. and Yagoubi, M.B., 2017. A survey on position-based routing protocols for Flying Ad hoc Networks (FANETs). Vehicular Communications, 10, pp.29-56. doi: 10.1016/j.vehcom.2017.10.003.

34. Lakew, D.S., Sa'ad, U., Dao, N.N., Na, W. and Cho, S., 2020. Routing in flying ad hoc networks: A comprehensive survey. IEEE Communications Surveys & Tutorials, 22(2), pp.1071-1120. doi: 10.1109/COMST.2020.2982452.

35. Kumar, S., Raw, R.S., Bansal, A. and Singh, P., 2023. UF-GPSR: Modified geographical routing protocol for flying ad-hoc networks. Transactions on Emerging Telecommunications Technologies, 34(8), p.e4813. doi: 10.1002/ett.4813.

36. De Lucia, L., Palazzi, C.E. and Vegni, A.M., 2023. ENSING: Energy saving based data transmission in internet of Drones for 3D connectivity in 6G networks. Ad Hoc Networks, p.103211. doi: 10.1016/j.adhoc.2023.103211.

37. Hosseinzadeh, M., Tanveer, J., Ionescu-Feleaga, L., Ionescu, B.S., Yousefpoor, M.S., Yousefpoor, E., Ahmed, O.H., Rahmani, A.M. and Mehmood, A., 2023. A greedy perimeter stateless routing method based on a position prediction mechanism for flying ad hoc networks. Journal of King Saud University-Computer and Information Sciences, 35(8), p.101712. doi: 10.1016/j.jksuci.2023.101712.

38. Hosseinzadeh, M., Mohammed, A.H., Alenizi, F.A., Malik, M.H., Yousefpoor, E., Yousefpoor, M.S., Ahmed, O.H., Rahmani, A.M. and Tightiz, L., 2023. A novel fuzzy trust-based secure routing scheme in flying ad hoc networks. Vehicular Communications, 44, p.100665. doi: 10.1016/j.vehcom.2023.100665.

39. Hosseinzadeh, M., Ali, S., Mohammed, A.H., Lansky, J., Mildeova, S., Yousefpoor, M.S., Yousefpoor, E., Ahmed, O.H., Rahmani, A.M. and Mehmood, A., 2024. An energy-aware routing scheme based on a virtual relay tunnel in flying ad hoc networks. Alexandria Engineering Journal. doi: 10.1016/j.aej.2024.02.006.

40. Kumar, S., Raw, R.S. and Bansal, A., 2023. LoCaL: Link-optimized cone-assisted location routing in flying ad hoc networks. International Journal of Communication Systems, 36(2), p.e5375. doi: 10.1002/dac.5375.

41. Kumar, S., Raw, R.S., Bansal, A., Mohammed, M.A., Khuwuthyakorn, P. and Thinnukool, O., 2021. 3D location oriented routing in flying ad-hoc networks for information dissemination. IEEE Access, 9, pp.137083-137098. doi: 10.1109/AC-CESS.2021.3115000.

42. Wang, F., Chen, Z., Zhang, J., Zhou, C. and Yue, W., 2019, July. Greedy forwarding and limited flooding based routing protocol for UAV flying ad-hoc networks. In 2019 IEEE 9th International conference on electronics information and emergency communication (ICEIEC) (pp. 1-4). IEEE. doi: 10.1109/ICEIEC.2019.8784505.

43. Heinzelman, W.R., Chandrakasan, A. and Balakrishnan, H., 2000, January. Energy-efficient communication protocol for wireless microsensor networks. In Proceedings of the 33rd annual Hawaii international conference on system sciences (pp. 10-pp). IEEE. doi: 10.1109/HICSS.2000.926982.

Funding

Acknowledgements

Author contributions statement

M.H., M.S.Y., and H.M.: Initial conceptualization. F.M.H., J.L.: Experimental setup. J.L., M.H.: Field testing J.L., H.M., M.H., F.M.H., and M.S.Y.: Wrote the main manuscript text. M.S.Y., and M.H.: Prepared figures. M.S.Y., M.H., J.L., H.M.: Wrote the analysis section. H.M. and J.L.: Reviewed the final manuscript.

The data availability statement

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.