Vehicular Communications 52 (2025) 100879

Contents lists available at ScienceDirect

Vehicular Communications

journal homepage: www.elsevier.com/locate/vehcom

A novel cylindrical filtering-based greedy perimeter stateless routing scheme in flying ad hoc networks

Amir Masoud Rahmani [a,1], Amir Haider [b,1], Khursheed Aurangzeb [c], May Altulyan [d], Entesar Gemeay [e], Mohammad Sadegh Yousefpoor [f], Efat Yousefpoor [f], Parisa Khoshvaght [g,h], Mehdi Hosseinzadeh [i,j,*]

[a] Future Technology Research Center, National Yunlin University of Science and Technology, Yunlin, Taiwan
[b] Department of Artificial Intelligence and Robotics, Sejong University, Seoul 05006, Republic of Korea
[c] Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, P. O. Box 51178, Riyadh, 11543, Saudi Arabia
[d] Department of Computer Engineering, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia
[e] Department of Computer Engineering, Computer and Information Technology College, Taif University, Taif, Saudi Arabia
[f] Center of Research and Strategic Studies, Lebanese French University, Kurdistan Region, Iraq
[g] Institute of Research and Development, Duy Tan University, Da Nang, Viet Nam
[h] Center of Excellence in Genomics and Precision Dentistry, Department of Physiology, Faculty of Dentistry, Chulalongkorn University, Bangkok 10330, Thailand
[i] School of Computer Science, Duy Tan University, Da Nang, Viet Nam
[j] Jadara University Research Center, Jadara University, Irbid, Jordan

ARTICLE INFO

Keywords:
Artificial intelligence (AI)
Cylindrical coordinate system
Flying ad hoc networks (FANETs)
Routing
Unmanned aerial vehicles (UAVs)
Quality of service (QoS)

ABSTRACT

Flying ad hoc networks (FANETs) are a new example of ad hoc networks, which arrange unmanned aerial vehicles (UAVs) in an ad hoc form. The features of these networks, such as the movement of UAVs in a 3D space, high speed of UAVs, dynamic topology, limited resources, and low density, have created vital challenges for communication reliability, especially when designing routing methods in FANETs. In this paper, a novel cylindrical filtering-based greedy perimeter stateless routing scheme (CF-GPSR) is suggested in FANETs. In CF-GPSR, cylindrical filtering reduces the size of the initial candidate set to accelerate the selection of the next-hop node. In this phase, the formulation of the cylindrical filtering construction process is expressed in the cylindrical coordinate system because the filtered area is a cylinder enclosed within the communication range of flying nodes. The cylindrical filtering construction process includes three steps, namely transferring coordinate axes, rotating coordinate axes, and cylinder construction. When selecting the next-hop node, CF-GPSR first uses this cylindrical filtering to limit the candidate set of each flying node. Then, CF-GPSR decides on the best next-hop UAV based on a merit function, which includes four criteria, namely velocity factor, ideal distance, residual energy, and movement angle, and selects a candidate node with the highest merit value as the next-hop UAV. Finally, the simulation process is performed using the NS 3.23 simulator, and four simulation scenarios are defined based on the number of UAVs, the communication area of nodes, network connections, and the size of packets to evaluate CF-GPSR. In the simulation process, CF-GPSR is compared with the three GPSR-based routing schemes, namely UF-GPSR, GPSR-PPU, and GPSR in terms of delay, data delivery ratio, data loss ratio, and throughput. In the first scenario, namely the change in the number of flying nodes, CF-GPSR improves delay, PDR, PLR, and throughput by 17.34%, 4.83%, 16%, and 7.05%, respectively. Also, in the second scenario, namely the change in communication range, the proposed method optimizes delay, PDR, PLR, and throughput by 4.91%, 5.71%, 6.12%, and 8.45%, respectively. In the third scenario, namely the change in the number of connections, CF-GPSR improves EED, PDR, PLR, and throughput by 18.41%, 9.09%, 9.52%, and 7.03%, respectively. In the fourth simulation scenario, namely the change in the packet size, CF-GPSR improves delay, PDR, PLR, and throughput by 14.81%, 19.39%, 7.19%, and 0.39%, respectively.

* Corresponding author at: School of Computer Science, Duy Tan University, Da Nang, Viet Nam.
E-mail address: mehdihosseinzadeh@duytan.edu.vn (M. Hosseinzadeh).

[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.vehcom.2025.100879
Received 19 May 2024; Received in revised form 6 November 2024; Accepted 12 January 2025
Available online 20 January 2025

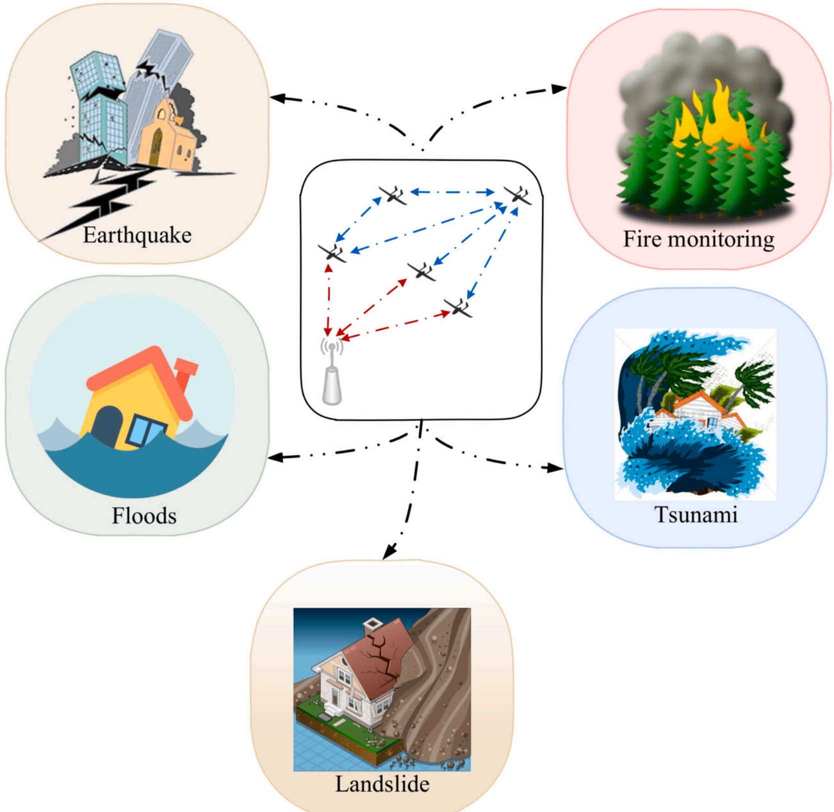

Fig. 1. FANET applications in natural disasters.

1. Introduction

Flying ad hoc network (FANET) is a modern wireless network, which has attracted a lot of interest from industrial and research communities. Due to the development of novel communication technologies such as micro-embedded systems, sensors, Wi-Fi, control technologies, artificial intelligence (AI), and positioning systems, the use of drones (also called unmanned aerial vehicles (UAVs)) is significantly increased in different areas [1,2]. FANET is made up of numerous drones, which fly in the air in a coordinated form and work together to carry out a specific mission. These networks have a dynamic topology due to the high movement of drones. In addition to military applications, FANETs are used in many civil and commercial activities, such as in natural disasters [3,4]. They can evaluate extensive and unpredictable physical or financial damage in damaged areas. Different natural disasters are forest fires, earthquakes, floods, tsunamis, and landslides. They cause economic, financial, environmental, and physical damage [5,6]. Hence, it is essential to prevent the increase in financial and physical damage and protect the environment during natural disasters using new technologies such as FANET because drones can be used to find suitable and safe locations to make rescue camps. Also, they can estimate damage, perform rescue operations, and evaluate natural disasters [7,8]. Therefore, researchers found that FANET is one of the most powerful tools when occurring any natural disaster. During any catastrophe, transportation and communication infrastructures are limited or almost impossible. As a result, only multi-UAV systems such as FANET can be used in this condition since UAVs can fly up to 30 meters on the top of the ground surface. Therefore, they are a useful and powerful tool to monitor damaged areas and rescue people from danger [9,10]. Fig. 1 depicts different FANET applications in natural disasters.

These networks have a variety of features such as unstable communication, resource restrictions, high mobility of UAVs, deployment in 3D space, rapid topology changes, and self-organization ability. As a result, it is essential to design an effective routing strategy to maintain network performance and ameliorate the quality of service (QoS) [11,12]. Routing protocols available in mobile ad hoc networks (MANETs) or vehicular ad hoc networks (VANETs) need some improvements to get a suitable performance in FANETs. A routing protocol plays a significant role in enhancing the reliability and stability of communication in the data transfer process. In this network, each UAV acts as a host and router simultaneously [13,14]. Note that routing paths include multiple hops, and consequently, the improvement of routing algorithms can significantly increase network performance. However, the high changes in network topology caused by the rapid movement of UAVs, as well as their random entry and exit from the network environment, can increase the number of broken links and reduce QoS requirements [15,16]. Thus, routing protocols must be compatible with the specific features of FANETs. The most important routing schemes in FANET include topology-based (proactive, reactive, and hybrid) and position-based routing schemes [17,18].

In recent years, many research studies have been provided in the field of routing protocols in FANETs. These studies show that position-based routing protocols are better than the topology-based ones. However, position-based routing protocols are dependent on the location information of UAVs in FANETs [19,20]. To achieve this information, flying nodes must be equipped with positioning systems such as GPS. Therefore, the performance of these routing protocols is dependent on the correctness of the position information obtained from the positioning system [21,22]. If the positioning system fails to determine the accurate position of UAVs for any reason, such as weather conditions or the deployment of FANETs in indoor environments, the aforementioned routing protocols experience poor performance. Of course, the many advantages of these protocols, such as the ability to build stable paths, low number of hops, low delay, and low routing overhead cannot be neglected. Therefore, these routing protocols are an appropriate option for FANETs [23,24].

Greedy perimeter stateless routing protocol (GPSR) [25] is the most popular and well-known position-based protocol, which uses two forwarding techniques, including greedy and perimeter to transfer data packets to the destination. In GPSR, the Euclidean distance between UAVs is a criterion to select the next-hop node. To calculate this distance, each flying node needs information about the position of its neighboring UAVs. To obtain this information, flying nodes regularly and periodically exchange their position through hello messages with each other and store the information collected from adjacent UAVs in a table called the neighbor table. According to the greedy forwarding technique, each flying node considers a candidate set containing its neighboring nodes to designate the best next-hop UAV [26,27]. Then, it chooses the neighboring node closest to the destination from its candidate set to transmit data packets to the destination. If the search process in the candidate set is not successful in finding the closest flying node to the destination, this flying node faces a local optimization issue and stops the greedy forwarding technique. Then, it uses the second forwarding technique, namely perimeter mode, and selects the next node according to the right-hand rule [28,29]. When the local optimum issue is resolved, the forwarding technique switched from the perimeter mode to the greedy mode. The procedure is repeated until the data packet gets to the destination. Note that GPSR is successful in reducing delay and routing overhead. However, it is necessary to improve the greedy forwarding technique in this method [30,31]. As mentioned above, this technique focuses only on the distance criterion to select the next-hop UAV, but it is not efficient in FANETs. Criteria such as the movement information of UAVs, residual energy, and movement angle can increase the quality of this technique. On the other hand, in GPSR, the candidate set contains all neighboring nodes in the neighbor table, but many of these UAVs do not deserve to be candidate nodes, hence, they must be removed from the candidate set. This filtering operation accelerates the next-hop selection process and reduces delay in the routing process [32,33].

According to the points mentioned above, this paper presents a novel cylindrical filtering-based greedy perimeter stateless routing scheme (CF-GPSR) in FANETs. In CF-GPSR, when a flying node wants to forward its data packets to another node and does not have a route, it creates a multi-hop route using two forwarding techniques, including greedy and perimeter. In CF-GPSR, the greedy technique is modified using cylindrical filtering and merit function to select the best next-hop UAV. However, the perimeter forwarding strategy in CF-GPSR resembles it in GPSR. In GPSR, all neighboring nodes belong to the candidate set. As a result, the source node must search all flying nodes in the candidate set to find the flying node closest to the destination. Now, if the network density is too high, the size of this candidate will be enlarged. As a result, it takes a long time to find the best next-hop UAV. Hence, CF-GPSR tries to solve this challenge by reducing the size of the initial candidate set using cylindrical filtering. On the other hand, in GPSR, the distance criterion alone is intended to choose the best next-hop UAV. However, this criterion alone is not efficient for the 3D environment of FANET with extremely moving nodes and limited energy resources. Therefore, CF-GPSR decides on the next-hop UAV based on a merit function including four factors, namely velocity factor, ideal distance, residual energy, and movement angle. In general, CF-GPSR first filters the existing nodes in the candidate set, then evaluates these nodes based on the merit function, and finally selects the flying node with the most merit value as the next-hop UAV. Note that if CF-GPSR is involved with the local optimum, it uses the perimeter forwarding strategy used in GPSR to solve this issue. In the following, the most important innovations in this paper are stated:

- In CF-GPSR, each flying node uses the regular exchange of hello packets in its communication range to build its neighbor table and decide on the next-hop node. This section explains the format of hello messages, the template of the neighbor table, updating process of this table.
- In CF-GPSR, each flying node generates a multi-hop route using two forwarding techniques, namely greedy and perimeter. The proposed method improves the greedy forwarding technique using cylindrical filtering and merit function so that cylindrical filtering is responsible for restricting the candidate set of each flying node, and the merit function is used as a criterion to select the next-hop UAV.
- In CF-GPSR, the size of the initial candidate set is reduced through cylindrical filtering to obtain the filtered candidate set. This accelerates the next-hop selection process and reduces delay in the routing process. The cylindrical filtering construction process consists of three steps, namely transferring coordinate axes, rotating coordinate axes, and cylinder construction.
- In CF-GPSR, each flying node decides on the best next-hop UAV based on a merit function, which includes four criteria, namely velocity factor, ideal distance, residual energy, and movement angle.

The structure of this paper is as follows: Section 2 introduces the most recent routing methods in FANETs. Section 3 includes the network and energy models used in CF-GPSR. Section 4 describes the different steps of CF-GPSR. In Section 5, the simulation results are presented, and Section 6 states the conclusion of this paper.

2. Related works

In [34], a utility function-based GPSR called UF-GPSR is suggested for FANETs. It optimizes the greedy transmission scheme by applying critical scales, such as movement direction, residual energy, velocity, distance, and risk of links. Furthermore, UF-GPSR calculates a utility function based on these scales to improve the routing process when determining the next flying node. In UF-GPSR, the authors consider three points. 1) Among the neighboring UAVs of a flying node, the candidate node closest to the destination is not always the best next-hop. 2) In scattered and dynamic environments, routing loops are likely, and routing schemes are susceptible to the local optimum problem. 3) When designating the subsequent node, access to the location information of flying nodes in the network environment is not enough. The implementation of UF-GPSR is carried out in NS3, and the results indicate its powerful and useful performance in comparison to other GPSR-based routing methods.

In [35], a mobility-aware clustering-based routing technique called MWCRSF is designed in FANETs. MWCRSF employs the sparrow search algorithm (SSA) because it has a high ability to search efficiently in the network environment. The SSA-based-clustering algorithm creates primary clusters to separate flying nodes in different clusters. Then, a weighted CH selection algorithm is presented. It merges the average distance to its neighbors, relative velocity, correlation degree, residual energy, and distance to the ground control station (GCS). In addition, MWCRSF includes a cluster-updating algorithm to refresh clusters at all times. To get a reliable and immediate data transmission process between flying nodes and GCS, the next-hop selection process considers various elements such as distance between flying nodes and GCS, link lifespan, and residual energy. In the following, MWCRSF employs a multi-criteria decision-making system based on skyline operators to decrease the dimensions of the search space. This accelerates the CH selection and routing procedures. The simulation results show the powerful performance of MWCRSF.

In [36], an adaptive and multi-path GPSR called AM-GPSR is offered for FANETs. Two new techniques, adaptive hello broadcasting and multi-path greedy routing, are introduced in this routing scheme. Firstly, each drone utilizes a dynamic hello interval based on its movement information and the difference between two approximated and real locations. Additionally, greedy route management filters the set of potential forwarders by eliminating border drones. Then, the priority of these candidate drones is calculated based on the destination arrival time and buffer volume, and the drones, which obtain more priority can play the role of forwarders in the forwarding path. AM-GPSR uses a

Table 1
Juxtaposition of related works.

Technique	Benefits	Weaknesses
UF-GPSR [34]	Enhancing data delivery ratio and throughput, decreasing latency and routing overhead, obtaining high energy efficiently	Low scalability, not adjusting a dynamic time interval for broadcasting hello messages in the network
MWCRSF [35]	Having good network scalability, increasing throughput and data delivery ratio, optimizing energy consumption, decreasing latency, accelerating the clusters construction process, forming stable clusters	High computational complexity, not adjusting a dynamic time interval for broadcasting hello messages in the network
AM-GPSR [36]	Adjusting a dynamic time interval for broadcasting hello messages in the network, improving data reliability, improving latency, delivery success rate, and throughput in the data transfer procedure, filtering the set of candidate nodes	High routing overhead, low scalability
GPSR+ [37]	Improving energy consumption and extending network longevity, enhancing the reliability and stability of the created paths, adapting to frequent topology changes in FANET, high fault tolerance, preventing routing loops	Increasing latency, high computational complexity, low scalability
OLSR+ [38]	Increasing data delivery ratio and throughput, optimizing energy consumption, high adaptability to FANET	High routing overhead, low scalability
FTSR [39]	High accuracy in identifying distrusted flying nodes, increasing data delivery ratio, improving reliability in the data transmission process, enhancing quality of service (QoS)	Low scalability, high latency in the routing procedure
GPSR [25]	Decreasing latency and routing overhead, improving throughput and efficiency	Low reliability in the data transmission process, dealing with routing loops, low adaptability to the FANET environment
GPSR-PPU [40]	Increasing reliability and energy efficiency in the routing process, stabilizing the formed routes in the network, low delay, high data delivery ratio	Low scalability, low adaptability to FANET, not adjusting a dynamic time interval for broadcasting hello messages in the network

multi-path routing technique to improve data reliability. Lastly, simulation results show that AM-GPSR has a successful performance compared to other routing schemes.

In [37], a location prediction strategy-based GPSR named GPSR+ is offered in FANETs. It uses a location prediction technique to estimate the future location of flying nodes in the network. In GPSR+, this technique is applied to modify the time interval related to the periodic broadcasting of hello messages. This modification improves the suitability of GPSR+ for FANET. In GPSR+, a candidate set is chosen based on a spherical removal method. Then, GPSR+ applies this candidate set to improve the selection process of the next-hop node and accelerate the routing process. To achieve this goal, the most stable UAV is singled out from the candidate set. This stabilizes the created routes between UAVs. Evaluation results show the successful performance of GPSR+ in comparison with other methods. However, the latency in GPSR+ has increased slightly.

In [38], a fuzzy logic-based optimized link state routing approach called OLSR+ is presented for FANETs. This scheme has four phases. Firstly, OLSR+ gets information about local network topology using hello messages. This phase defines an exponential function containing movement direction, link quality, relative velocity, and distance to give a new formulation of link lifespan. The next phase defines a fuzzy system, which includes three criteria, namely residual energy, neighbor degree, and link lifespan to choose multi-point relays (MPRs). The third phase modifies the template of topology control message and inserts two fields, namely the energy and lifespan of the relevant route into this message. The last phase determines how to obtain the routing tables of flying nodes. Evaluation results show the superiority of OLSR+ in comparison with G-OLSR and OLSR.

In [39], a fuzzy logic-based and trust-aware secure routing method (FTSR) is offered for FANETs. FTSR evaluates the trust of flying nodes based on two techniques, namely local trust and route trust. Firstly, each flying node applies the distributed local trust technique to find trusted and distrusted adjacent flying nodes locally. Therefore, the trusted flying nodes cooperate with each other to find secure routes in the network, and the distrusted flying nodes are isolated in the network. The local trust technique reduces the likelihood of fake routes in the network. However, this is still very likely because the local technique may not detect all distrusted flying nodes. Hence, the second trust technique in FTSR, namely route trust, is responsible for finding other distrusted flying nodes. This technique uses a fuzzy system to check the trust status of each route globally and find securest routes. The evaluation results clearly indicate the powerful and useful performance of FTSR.

In [25], the GPSR algorithm is introduced in wireless networks. It dispatches data packets according to the position of nodes. In addition, GPSR designs a greedy forwarding pattern based on nodes' position to designate the next-hop node toward the destination. If a data packet arrives at an area that fails to apply the greedy pattern, it recovers the routing process around that area. In GPSR, each node keeps local network topology in its memory, hence it works better than the shortest path algorithm. Additionally, it can manage changes in local network topology when finding new routes. Evaluation results show that GPSR works well in high-density networks.

In [40], an improved GPSR algorithm called GPSR-PPU is suggested in FANETs. GPSR-PPU considers the location prediction process and uncertainty to strengthen the routing process in FANETs. In GPSR-PPU, the location information of flying nodes may include uncertainties due to measurement errors or external factors. Hence, flying nodes, which are associated with high uncertainties, get a lower priority to transmit data packets in the network. Furthermore, GPSR-PPU modifies the next-hop selection process in dynamic environments with high topology changes such as FANET. It predicts the future location of flying nodes based on their current location and velocity. This prediction helps the routing process to make accurate decisions and form stable routes between source and destination. GPSR-PPU considers a mechanism to avoid routing loops. This mechanism uses a table, which stores the ID of the flying nodes during the routing process. Hence, these flying nodes cannot be met again in the future. In addition, GPSR-PPU employs a route maintenance process that periodically updates routing information.

A brief overview of the benefits and drawbacks of each approach discussed in this section is provided in Table 1.

3. System model

In this section, assumptions about the proposed system are expressed in two sections, namely the network model and the energy model.

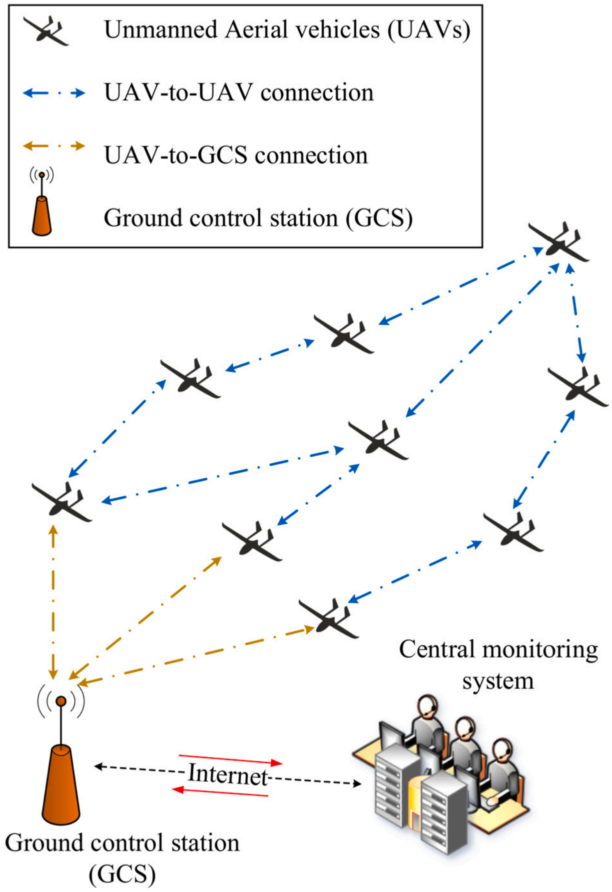

Fig. 2. Network model in CF-GPSR.

3.1. Network model

The network model in CF-GPSR can be shown in the form of a network graph $G = (U, L)$ so that U and L indicate two sets containing vertices and edges, respectively. In the set U, each member represents a UAV or a flying node in the network so that $U = \{u_i | 1 \le i \le N\}$, where u_i indicates i-th flying node, and N is the total number of flying nodes in the network. Each flying node like u_i is known using a special identifier (ID_i) to distinguish it from other UAVs. In addition, each member of the set L indicates a direct link between two neighboring nodes. If the distance between two nodes such as u_i and u_j is smaller than their communication radius (i.e. r), then u_i and u_j are neighbors, and the link between them, L_{ij}, belongs to the set L i.e. $L = \left\{ L_{ij} | \underset{i \ne j}{\forall}\ u_i, u_j \in U \text{ and } d_{ij} \le r \right\}$ so that d_{ij} indicates the distance from u_i to u_j and is derived from Equation (1).

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

here (x_i, y_i, z_i) and (x_j, y_j, z_j) show the location coordinates of u_i and u_j, respectively. This information is obtained through the positioning system installed on these flying nodes.

If u_i and u_j have a direct link in the set L, the connection between these nodes is directly established through the mentioned link. Otherwise, u_i and u_j need to form a multi-hop route to communicate with each other. In this case, flying nodes use UAV-to-UAV communication in FANET [43]. Fig. 2 shows this network model. Note that the ground control station (GCS) is a powerful node with an unlimited energy resource in the network. GCS is directly connected to the central monitoring system and is responsible for transmitting commands from the central system to flying nodes and sending data collected from drones to the central system. GCS also communicates with a small number of flying nodes at any moment through the GCS-to-UAV connection, while other flying nodes are indirectly connected to GCS via multi-hop routes [43].

3.2. Energy model

In CF-GPSR, the energy model proposed by Heinzelman et al. [41] is used to calculate the energy consumption of flying nodes in the data transfer process. It consists of two energy models, namely free space (fs) and multi-path fading (mp). In these two energy models, there is a direct relationship between the energy consumed by the transmitter flying node (such as u_i) and its distance to the receiver flying node (such as u_j) i.e. d_{ij} (calculated by Equation (1)). If d_{ij} is large, u_i consumes high energy in the data transfer process and vice versa. To transmit k bits from u_i to u_j, the energy consumed by u_i and u_j is calculated through Equations (2) and (3), respectively.

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

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

so that E_{elec} is the energy needed for the electronic board of u_i and u_j to transmit one bit. $\varepsilon_{fs}(d_{ij})^2$ and $\varepsilon_{mp}(d_{ij})^4$ are the energy needed to amplify signals in two models, namely fs and mp, respectively. Also, d_t is the distance threshold, which is equal to $d_t = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}}$. If $d_{ij} \le d_t$, the energy consumed in u_i is calculated based on the fs model. Otherwise, it is obtained from the mp model.

On the other hand, when flying or hovering, the energy consumption of a flying node such as u_i is obtained with regard to the method presented in [42]. According to this method, the hovering power of u_i is computed through Equation (4).

$$P_H = \sqrt{\frac{(m_u g)^3}{2\pi r_\omega n_\omega \rho_a}} \tag{4}$$

So that m_u indicates the mass of u_i, g shows the gravitational acceleration, r_ω is the wing radius, n_ω denotes the number of wings, and ρ_a indicates the density of the air.

Therefore, u_i uses Equation (5) to calculate the energy needed for hovering in the air.

$$E_H = P_H T_H \tag{5}$$

so that T_H is a time interval for hovering in the air.

Also, the flight power of u_i is computed through Equation (6).

$$P_F = (P_{\max} - P_H) \frac{V_u(t)}{V_{\max}} \tag{6}$$

so that $V_{\max}$ denotes the most velocity of u_i, $V_u(t)$ indicates the current velocity of u_i, $P_{\max}$ shows the flight power of u_i at the velocity $V_{\max}$.

As a result, u_i uses Equation (7) to calculate the energy needed for flying in the air.

$$E_F = \int_0^{T_F} P_F \, dt \tag{7}$$

where T_F is a time interval for flying in the air.

4. Proposed method

In this section, a cylindrical filtering-based greedy perimeter stateless routing scheme (CF-GPSR) is offered for FANETs. In CF-GPSR, when a flying node, such as u_s, does not find a path to transmit its data packets to the desired flying node (u_d), it must create a multi-hop route to

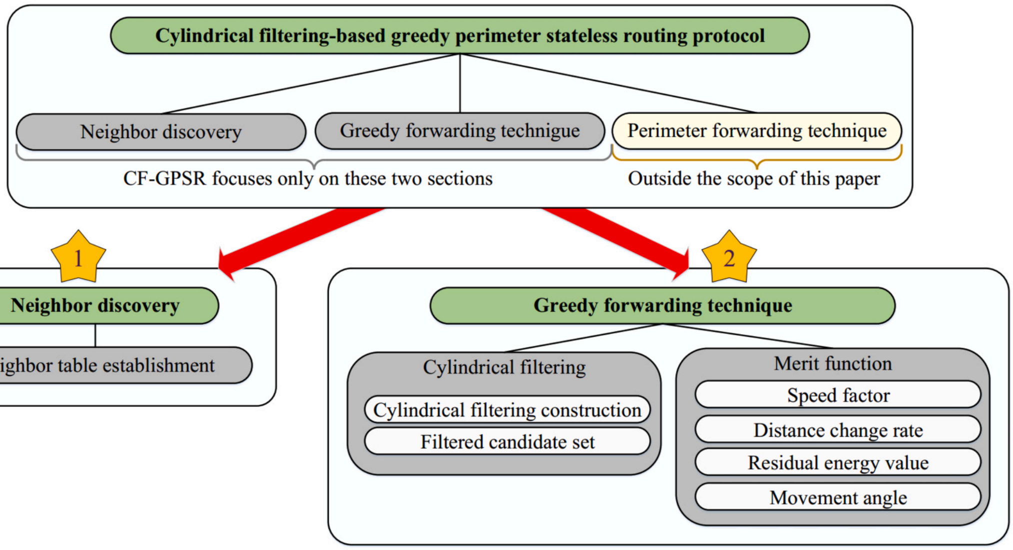

Fig. 3. The schematic design of the proposed scheme.

u_d using two greedy and perimeter forwarding techniques. In CF-GPSR, the greedy technique is modified by adding cylindrical filtering and utilizing the merit function to select the best next-hop UAV. However, the perimeter forwarding technique in CF-GPSR is fully similar to that in GPSR. Therefore, CF-GPSR includes two parts:

- Neighboring discovery
- Greedy forwarding strategy

In the following, each section is detailed. The schematic design of CF-GPSR is shown in Fig. 3. Additionally, the most important symbols used in CF-GPSR are stated in Table 2.

4.1. Neighbor discovery

Before starting the routing process, each flying node, such as u_i, requires information about local network topology and its adjacent flying nodes so that it can decide on the next-hop node in the routing process. To address this need and achieve the desired information, u_i uses the regular exchange of hello packets in its communication radius i.e. r. According to the format presented in Table 3, a hello message contains information such as identifier (ID_i), residual energy (E_i), position (x_i, y_i, z_i), and velocity (V_i, Θ_i, ϕ_i). Note that V_i, Θ_i, and ϕ_i denote the velocity size, its angle with the xy-plane, and its angle with the positive z-axis, respectively.

After completing a period of hello messages, u_i gets a number of hello messages from its adjacent flying nodes. To build local network topology, u_i processes hello messages of flying nodes to store their information in a neighbor table (NT_i). See Table 4. In addition, u_i is responsible for updating NT_i in any period. Accordingly, u_i attaches new entries to store the information of new flying nodes in NT_i. Furthermore, two modes are intended for existing entries related to the old neighbors in NT_i: 1) Remove the entries of invalid neighbors who have left the communication range of u_i, and 2) Refresh the information of neighbors who are still within the communication range of u_i. To determine the time interval to remove an entry from NT_i, a validity time (VT_j) is included in this table. VT_j and the hello period are directly related to each other. If u_i receives a hello packet from its old neighbor, VT_j will be reset, and this entry will be valid during the next time. Otherwise, if VT_j is finished, the relevant entry will be removed. This expresses the importance of the hello period. In dynamic environments such as FANET, the hello dissemination period cannot be long because the movement of flying nodes changes the local network topology, and UAVs cannot properly adapt their neighbor table to these changes. On the other hand, a short hello time imposes a lot of routing overhead in the network. In the best case, the hello packet time should be adjusted based on the velocity of flying nodes. If flying nodes fly rapidly at the network, CF-GPSR considers a short hello broadcast time, otherwise this period can be longer. This creates a balance between routing overhead and adaptation to the FANET environment. Algorithm 1 shows the pseudo-code related to this process.

Algorithm 1 Neighbor discovery.

Input: $U = \{u_i | 1 \leq i \leq N\}$: A set of flying nodes
r: Communication range of flying nodes
(x_i, y_i, z_i): Position of u_i
(V_i, Θ_i, ϕ_i): Speed information of u_i
T_{NET}: A timer that counters the simulation time.
Output: NT_i: Neighbor table related to u_i
Begin
$\mathbf{u_i}$**:** Adjust its hello period with regard to the speed of flying nodes;
repeat
  if $(T_{NET}$ mod $hello\,period) = 0$ **then**
    for $i = 1$ to N **do**
      $\mathbf{u_i}$**:** Obtain its information like ID_i, E_i, (x_i, y_i, z_i), and (V_i, Θ_i, ϕ_i);
      $\mathbf{u_i}$**:** Insert the mentioned information into a hello packet;
      $\mathbf{u_i}$**:** Disseminate the hello packet for its surrounding flying nodes;
    end for
  end if
  if u_i gets a hello packet from its surrounding node u_j **and** NT_i includes ID_j **then**
    $\mathbf{u_i}$**:** Renew the information of u_j, like (x_j, y_j, z_j), (V_j, Θ_j, ϕ_j), and E_j in NT_i;
    $\mathbf{u_i}$**:** Restart VT_j in the entry corresponding to u_j in NT_i;
  else if u_i gets a hello packet from its surrounding node u_j **and** NT_i does not include ID_j **then**
    $\mathbf{u_i}$**:** Put a new entry in NT_i;
    $\mathbf{u_i}$**:** Insert ID_j, (x_j, y_j, z_j), (V_j, Θ_j, ϕ_j), and E_j in NT_i;
    $\mathbf{u_i}$**:** Restart VT_j in this new entry with regard to hello period;
  end if
  for $j = 1$ to N_i **do**
    if NT_i includes ID_j **and** VT_j is ended **then**
      $\mathbf{u_i}$**:** Eliminate the entry related to u_j from NT_i;
    end if
  end for
until $T_{NET} \leq Simulation\ Time$
End

Table 2
The most important symbols used in CF-GPSR.

Symbol	Description
$G=(U,L)$	Network graph
U	A set of flying nodes in the network
L	A set of direct links in the network
u_i	Flying node i
N	Number of flying nodes
ID_i	Identifier of u_i
r	Communication radius of flying nodes
L_{ij}	A communication link between u_i and u_j
d_{ij}	Euclidean distance between u_i and u_j
(x_i, y_i, z_i)	Position of u_i
fs	Free space propagation model
mp	Multi-path fading propagation model
E_{elec}	Energy consumed by electrical boards of transceivers
ε_{fs}	Energy needed to amplifiers in the free space model
ε_{mp}	Energy needed to amplifiers in the multi-path fading model
d_t	Threshold distance in the energy model
P_H	Hovering power of flying nodes
m_u	Mass of flying nodes
g	Gravity acceleration
r_ω	Wing radius
n_ω	Number of wings
ρ_a	Air density
E_H	Hovering energy consumption of flying nodes
T_H	Hovering time
P_F	Flight power of flying nodes
$P_{\max}$	Flight power of flying nodes whose speed is maximum
$V_u(t)$	Current speed of flying nodes
$V_{\max}$	Maximum speed of flying nodes
E_F	Flight energy consumption of flying nodes
T_F	Flying time
u_s	Source flying node
u_d	Destination flying node
S	Position of u_s
D	Position of u_d
E_i	Residual energy of u_i
(V_i, Θ_i, ϕ_i)	Speed information of u_i
NT_i	Neighbor table related to u_i
N_i	Number of members of NT_i
VT_j	Validity time of the entry related to u_j in NT_i
ICS_i	Initial candidate set of u_i
FCS_i	Filtered candidate set of u_i
ρ	Length of the projection of the point P onto the xy-plane
φ	Angle between ρ and the positive x-axis
z	Regular z-coordinate
θ	Rotation angle
$R_z(\theta)$	Rotation matrix
VF_{ij}	Velocity factor of u_j with regard to u_i
$IDis_{ij}$	Ideal distance between u_i and u_j
λ_j	Movement angle of u_j with regard to u_i and u_d
F_{merit}	Merit function

Table 3
Hello message format.

Hello broadcast period	Identifier (ID_i)	Hello identifier
Position (x_i, y_i, z_i)	Velocity (V_i, Θ_i, ϕ_i)	Residual energy (E_i)

Table 4
Template of NT_i.

Identifier	Position	Velocity	Residual energy	Validity time
ID_j	(x_j, y_j, z_j)	(V_j, Θ_j, ϕ_j)	E_j	VT_j

4.2. Greedy forwarding strategy

In CF-GPSR, when the flying node u_s does not find a path to transfer its data packets to u_d, it creates a multi-hop path to u_d using two greedy and perimeter forwarding techniques. The proposed scheme modifies the greedy forwarding technique using cylindrical filtering and merit function. However, the perimeter forwarding strategy in CF-GPSR resembles it in GPSR. An example is presented in Fig. 4 to clearly explain the greedy forwarding strategy in CF-GPSR. According to this figure, u_s is in the center of the circular area with the radius r, and all flying nodes in this area, namely u_1, u_2, u_3, u_4, u_5, u_6, u_7, and u_8, are the neighbors of u_s. They can be connected to u_s through a direct link. In GPSR, all these UAVs belong to a candidate set. As a result, u_s must search all flying nodes in the candidate set to find the flying node closest to u_d. Now, if the network density is too high, the size of this candidate will be enlarged. As a result, it takes a long time to find the best next-hop UAV. Hence, CF-GPSR tries to solve this challenge by reducing the size of the initial candidate set using cylindrical filtering. On the other hand, in GPSR, the distance criterion alone is intended to choose the best next-hop UAV. However, this criterion alone is not efficient for the 3D environment of FANET with extremely moving nodes and limited energy resources. Therefore, CF-GPSR decides on the next-hop UAV based on a merit function including four factors, namely velocity factor, ideal distance, residual energy, and movement angle. In general, CF-GPSR first filters the existing nodes in the candidate set, then evaluates these nodes based on the merit function, and finally selects the flying node with the most merit value as the next-hop UAV. Note that if CF-GPSR is involved with the local optimum, it uses the perimeter forwarding strategy used in GPSR to solve this issue. Pay regard to that the perimeter forwarding strategy is outside the field of this paper and is not addressed here.

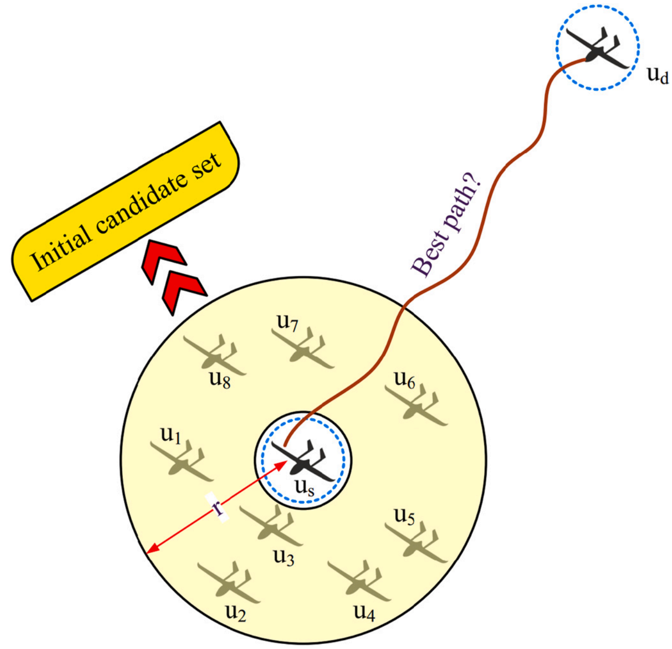

Fig. 4. Communication range of u_s.

4.2.1. Cylindrical filtering

According to Fig. 4, u_s is located in the center of the circular area with the radius r, and its neighbors include u_1, u_2, u_3, u_4, u_5, u_6, u_7, and u_8. Based on Section 4.1, these neighbors have been recorded in the neighbor table of u_s. Therefore, u_s have access to their information such as position, velocity, and remaining energy. According to the information in the neighbor table, u_s reduces the size of its initial candidate set (ICS_s) using cylindrical filtering to obtain a filtered candidate set (FCS_s) in Fig. 5. This section includes two steps:

- Cylindrical filtering construction
- Calculating the filtered candidate set

Cylindrical filtering construction. The purpose of this step is to calculate the cylindrical filtered area enclosed within the communication range of u_s in Fig. 5. Fig. 6 shows the simplified shape of cylindrical filter-

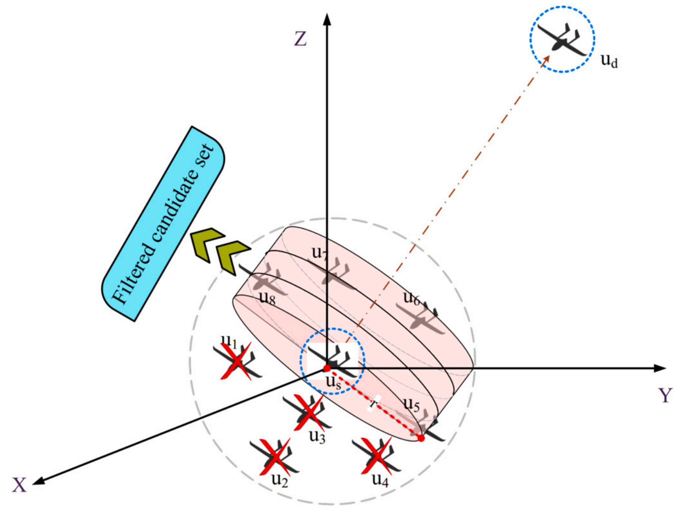

Fig. 5. Filtered candidate set (FCS_s).

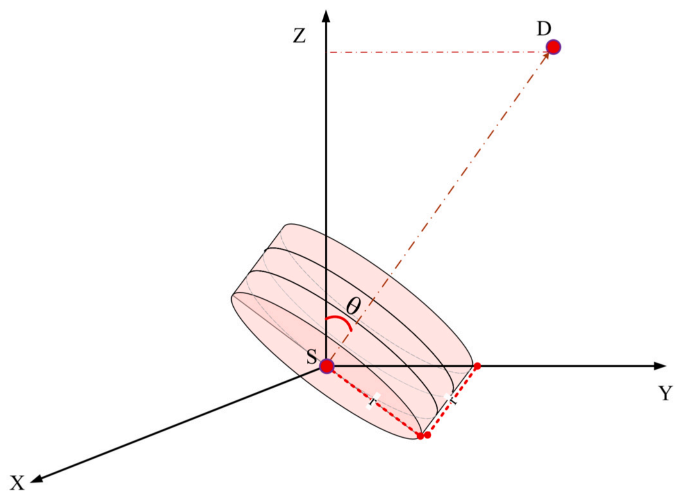

Fig. 6. Simplified shape of the filtered area.

ing. Furthermore, Algorithm 2 expresses the pseudo-code related to the cylindrical filtering construction process. Note that in this process, the positions of u_s and u_d are displayed as $S = \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix}$ and $D = \begin{bmatrix} x_d \\ y_d \\ z_d \end{bmatrix}$, respectively.

The equations of cylindrical filtered area must be calculated in the cylindrical coordinate system because this area in Fig. 6 is cylindrical. Note that the cylindrical coordinates in the three-dimensional space are a combination of polar coordinates on the xy-plane and the z-axis in the Cartesian coordinates. According to Fig. 7, the cylindrical coordinates of each point such as $P = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$ are defined as $P = \begin{bmatrix} \rho \\ \varphi \\ z \end{bmatrix}$ so that ρ indicates the distance from the origin to the projection of P on the xy-plane. φ denotes the angle between the positive x-axis and the line segment from the origin to the projection of P. The third cylindrical coordinate is the z coordinate, which shows the distance from P to the xy-plane. Equations (8) and (9) specify the relationship between cylindrical coordinates and Cartesian coordinates.

$$\begin{bmatrix} \rho \\ \varphi \\ z \end{bmatrix} = \begin{bmatrix} \sqrt{x^2 + y^2} \\ \arctan\left(\frac{y}{x}\right) \\ z \end{bmatrix} \tag{8}$$

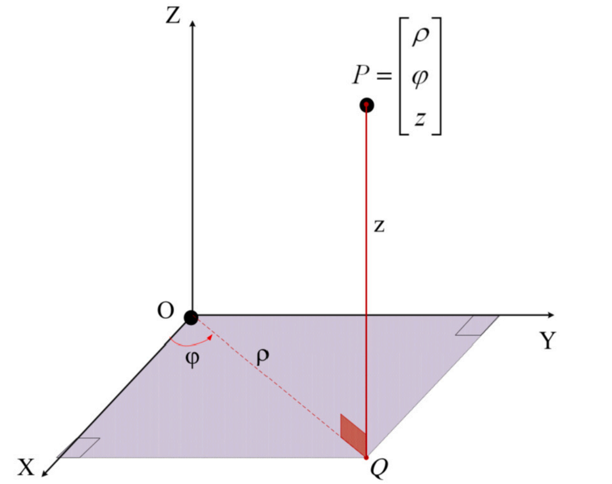

Fig. 7. Cylindrical coordinates of the point P in the 3D space.

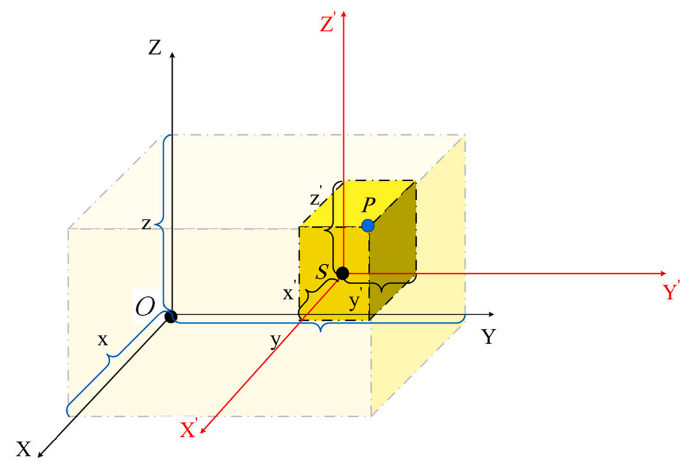

Fig. 8. Transferring coordinate axes to the point S.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \rho\cos\varphi \\ \rho\sin\varphi \\ z \end{bmatrix} \tag{9}$$

To calculate the cylindrical filtered area presented in Fig. 5, three steps are carried out:

- **First step) Transferring coordinates axes:** As shown in Fig. 6, the origin of the coordinate axes is the point S. Hence, the first step in the cylindrical filtering construction process is to transfer the coordinate axes from $O = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$ to $S = \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix}$. In this case, the transferred coordinate axes i.e. X', Y', and Z' are parallel to the initial coordinate axes i.e. X, Y, and Z. This is stated in Fig. 8. In this figure, if the Cartesian coordinates of a point like P are equal to $P = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$, its coordinates in the transferred coordinate system are $P = \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}$. In this case, the relationship between these two coordinate systems is obtained through the transformation function provided in Equation (10).

$$\begin{aligned} X' &= X - x_s \\ Y' &= Y - y_s \\ Z' &= Z - z_s \end{aligned} \tag{10}$$

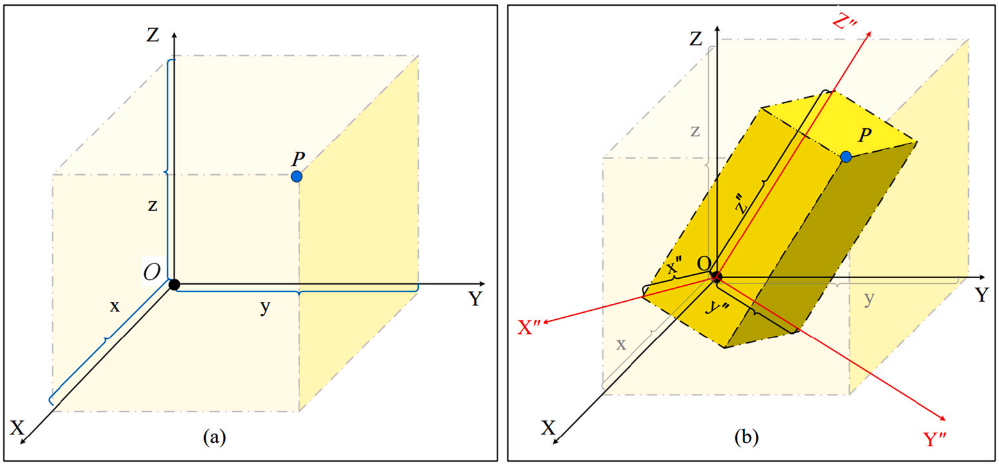

Fig. 9. Rotating the coordinate axes around the origin.

here (x_s, y_s, z_s) indicates the spatial coordinates of S.

- **Second step) Rotating coordinate axes:** In Fig. 6, if the coordinate axes are rotated around the origin with regard to the angle θ, the central axis of the cylindrical filtered area matches the z-axis, and the formulation of this cylindrical filtering is very simple. To calculate the rotation angle θ, a vector is first drawn from S to D. Now, the angle between the vector $\overrightarrow{SD}$ and the z-axis is calculated based on Equation (11).

$$\theta = \arccos\left(\frac{z_d - z_s}{\sqrt{(x_d-x_s)^2+(y_d-y_s)^2+(z_d-z_s)^2}}\right), \quad 0 \le \theta \le 2\pi \tag{11}$$

In this case, the rotated coordinate axes are X'', Y'', and Z'', and the initial axes are X, Y, and Z. If the initial coordinates of a point like P are $P = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$ (see Fig. 9(a)), its rotated coordinates are equal to $P = \begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix}$ (see Fig. 9(b)). Note that CF-GPSR uses the cylindrical coordinate system. As a result, the rotation matrix in this coordinate system is $R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$. In this case, the relationship between two initial and rotated coordinate systems is obtained from the transformation function presented in Equations (12) and (13).

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{12}$$

Then,

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = \begin{bmatrix} x\cos\theta - y\sin\theta \\ x\sin\theta + y\cos\theta \\ z \end{bmatrix} \tag{13}$$

- **Third step) Constructing cylindrical filtering:** After performing steps one and two, Equation (14) expresses the inner area of the cylinder in Fig. 6 whose radius and height are equal to r, in the cylindrical coordinate system.

$$0 \le \rho \le r, 0 \le z \le r, 0 \le \varphi \le 2\pi \tag{14}$$

Calculating the filtered candidate set. In this phase, u_s should evaluate its neighboring nodes in NT_s based on the following steps to determine whether a neighboring flying node, such as u_j with the position (x_j, y_j, z_j), is inside the cylindrical filtering or not. Algorithm 3 presents a pseudo-code for calculating the filtered candidate set.

Algorithm 2 Cylindrical filtering construction.

Input: u_s: Source flying node
u_d: Destination flying node
(x_s, y_s, z_s): Position of u_s
(x_d, y_d, z_d): Position of u_d
Output: Cylindrical filtering formulation
Begin
1: $\mathbf{u_s}$: Consider the position of u_s as $S = \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix}$;
2: $\mathbf{u_s}$: Consider the position of u_d as $D = \begin{bmatrix} x_d \\ y_d \\ z_d \end{bmatrix}$;
3: $\mathbf{u_s}$: Transfer x-axis, y-axis, and z-axis from the origin O to the point S based on Equation (10);
4: $\mathbf{u_s}$: Calculate the rotation angle θ based on Equation (11);
5: $\mathbf{u_s}$: Rotate the transferred x-axis, y-axis, and z-axis based on Equations (12) and (13);
6: $\mathbf{u_s}$: Obtain the cylindrical filtering formulation based on Equation (14);
End

- **Step 1:** u_s transfers $u_j = \begin{bmatrix} x_j \\ y_j \\ z_j \end{bmatrix}$ to a coordinate system with the origin S based on Algorithm 2. As a result, the new position of this flying node is calculated through Equation (15) and using the transformation function in Equation (10).

$$u_j = \begin{bmatrix} x'_j \\ y'_j \\ z'_j \end{bmatrix} = \begin{bmatrix} x_j - x_s \\ y_j - y_s \\ z_j - z_s \end{bmatrix} \tag{15}$$

- **Step 2:** u_s rotates $u_j = \begin{bmatrix} x'_j \\ y'_j \\ z'_j \end{bmatrix}$ around the origin S based on the rotation angle θ. Therefore, the rotated coordinates of u_j are calculated using Equation (16) according to Algorithm 2 and based on the rotation matrix in Equation (12).

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

- **Step 3:** u_s converts $u_j = \begin{bmatrix} x''_j \\ y''_j \\ z''_j \end{bmatrix}$ into cylindrical coordinates using Equations (8) and (9) to obtain Equation (17).

$$u_j = \begin{bmatrix} \rho_j \\ \varphi_j \\ z_j \end{bmatrix} = \begin{bmatrix} \sqrt{{x''_j}^2 + {y''_j}^2} \\ \arctan\left(\frac{y''_j}{x''_j}\right) \\ z''_j \end{bmatrix} \tag{17}$$

- **Step 4:** If $u_j = \begin{bmatrix} \rho_j \\ \varphi_j \\ z_j \end{bmatrix}$ meets Equation (18), u_j is a member of the filtered candidate set FCS_s.

$$0 \le \rho_j \le r, 0 \le z_j \le r, 0 \le \varphi_j \le 2\pi \tag{18}$$

Otherwise, u_j is removed from the candidate set and does not belong to FCS_s.

Algorithm 3 Filtered candidate set.

Input: u_s: Source flying node
NT_s: Neighbor table related to u_s
u_d: Destination flying node
(x_s, y_s, z_s): Position of u_s
(x_d, y_d, z_d): Position of u_d
Output: FCS_s: Filtered candidate set related to u_s

```
    Begin
1:  for j = 1 to N_s do
2:    if u_j belongs to NT_s then
3:      u_s: Transfer u_j to a new coordinate system with the origin S = [x_s; y_s; z_s]
        based on Equation (15);
4:      u_s: Calculate the rotation angle θ based on Equation (11);
5:      u_s: Rotate the transferred coordinates (u_j = [x'_j; y'_j; z'_j]) around the angle θ
        based on Equation (16);
6:      u_s: Convert u_j = [x''_j; y''_j; z''_j] into the cylindrical coordinates (u_j = [ρ_j; φ_j; z_j])
        based on Equation (17);
7:      if u_j = [ρ_j; φ_j; z_j] meets Equation (18) then
8:        u_s: Insert u_j into FCS_s;
9:      end if
10:   end if
11: end for
    End
```

4.2.2. Merit function

In CF-GPSR, each flying node, such as u_i, decides on the best next-hop UAV based on a merit function, which consists of four criteria, namely velocity factor, ideal distance, residual energy, and movement angle. In this case, u_i first uses cylindrical filtering to limit its candidate set. Then, it calculates a merit value for each neighboring node such as u_j in FCS_i, and chooses a candidate node with the highest merit value as its next-hop UAV. Algorithm 4 shows the pseudo-code related to this process. To calculate the merit function, four criteria are introduced below:

- **Velocity factor ($\mathbf{VF_{ij}}$):** This criterion indicates the relative velocity of u_j with regard to u_i. The importance of this criterion in the merit function is that flying nodes in a routing path have a relatively similar velocity. As a result, they may stay in the neighboring range of each other for a longer time interval and build a more stable path from u_s to u_d. This issue has a positive effect on reducing the number of broken communication links and lost packets, and consequently lowering data retransmission. In the merit function, the desirable state occurs when VF_{ij} approaches zero, and the velocity of UAVs in the routing path is fully similar. VF_{ij} is obtained using Equation (19).

$$\left|\overrightarrow{VF_{ij}}\right| = \sqrt{\left(V_{ij}^x\right)^2 + \left(V_{ij}^y\right)^2 + \left(V_{ij}^z\right)^2} \tag{19}$$

so that,

$$V_{ij}^x = V_i \sin\phi_i \cos\Theta_i - V_j \sin\phi_j \cos\Theta_j \tag{20}$$

$$V_{ij}^y = V_i \sin\phi_i \sin\Theta_i - V_j \sin\phi_j \sin\Theta_j \tag{21}$$

$$V_{ij}^z = V_i \cos\phi_i - V_j \cos\phi_j \tag{22}$$

where (V_i, Θ_i, ϕ_i) and (V_j, Θ_j, ϕ_j) are the velocity vectors of u_i and u_j, which are obtained from NT_i.

Now, a normalization process is implemented on $\left|\overrightarrow{VF_{ij}}\right|$ through Equation (23) to limit its value to $[0, 1]$. Note that the merit function consists of four criteria with different units. Therefore, if the normalization process is not carried out on these criteria, their value will have a different effect on the merit function.

$$\left|\overrightarrow{VF_{ij}^{norm}}\right| = \frac{\left|\overrightarrow{VF_{ij}}\right|}{\max\limits_{\forall u_j \in FCS_i}\left\{\left|\overrightarrow{VF_{ij}}\right|\right\}} \tag{23}$$

Here, $\max\limits_{\forall u_j \in FCS_i}\left\{\left|\overrightarrow{VF_{ij}}\right|\right\}$ represents the maximum relative velocity of flying nodes in FCS_i with regard to u_i.

- **Ideal Distance ($\mathbf{IDis_{ij}}$):** According to the greedy forwarding strategy in GPSR, u_i searches its candidate set to find the closest node to u_d (such as u_j). Then, it considers u_j as its next-hop UAV. In this case, u_j is very close to the boundary of the communication range of u_i. This strategy is not especially suitable for highly dynamic networks such as FANET because these UAVs quickly get out of the communication range of each other. As a result, routes include a high number of broken links, which increase lost data packets. This causes the instability of the paths from u_s to u_d. The authors stated in [34] that if $d_{ij} \le 0.9r$, the data delivery rate is approximately 100%. However, if $0.9r < d_{ij} \le r$, data packets may be extremely lost. Here, d_{ij} is the distance between u_i and u_j calculated through Equation (1). r denotes the communication radius of UAVs in the network. However, if $d_{ij} \le 0.1r$, u_j is still not a good option for playing the role of next-hop node because the proximity of u_i and u_j increases the number of hops and delay in the data transfer process. Hence, CF-GPSR considers the ideal distance i.e. $0.1 \le d_{ij} \le 0.9r$. According to the merit function, if d_{ij} is close to the ideal distance, u_j can play the role of the next-hop node properly. Fig. 10 illustrates the concept of ideal distance in the communication range of u_i. The normalization process of $IDis_{ij}$ is presented in Equation (24) to restrict its value to $[0, 1]$.

$$IDis_{ij}^{norm} = \begin{cases} \frac{d_{ij}}{0.1r}, & 0 < d_{ij} < 0.1r \\ 1, & 0.1r \le d_{ij} \le 0.9r \\ 1 - \left(\frac{d_{ij} - 0.9r}{r - 0.9r}\right), & 0.9r < d_{ij} \le r \end{cases} \tag{24}$$

- **Residual energy ($\mathbf{E_j}$):** In the next-hop selection process, u_i uses this important criterion to evaluate the merit value of flying nodes in FCS_i because a neighboring flying node with low energy dies quickly if it plays the role of the next-hop UAV in the routing path. As a result, the data transfer process must be repeated again. This creates an imbalanced energy consumption in FANET and reduces network lifespan. In contrast, if u_j has high energy levels and participates in the routing process, this route is valid for a long time and the data transfer process is successful. Note that E_j is extracted from

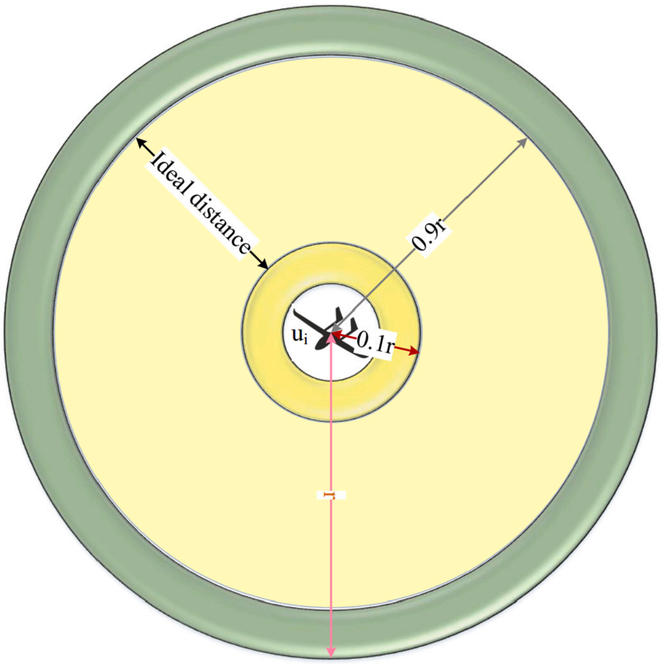

Fig. 10. Ideal distance between u_i and flying nodes in FCS_i.

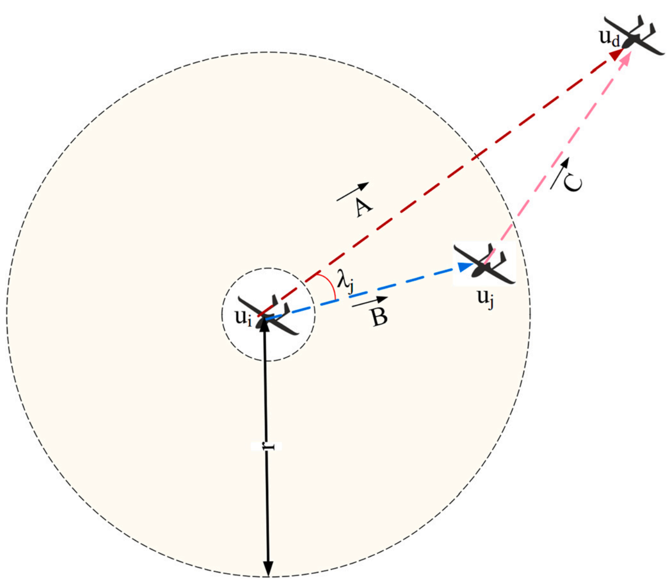

Fig. 11. Movement angle.

NT_i, and a normalization process is done on E_j based on Equation (25) to adjust its value in $[0, 1]$.

$$E_j^{norm} = \frac{E_j}{\max\limits_{\forall u_j \in FCS_i} \{E_j\}} \tag{25}$$

where $\max\limits_{\forall u_j \in FCS_i} \{E_j\}$ indicates the highest levels of UAVs in FCS_i.

- **Movement angle (λ_j):** The purpose of this criterion is that the movement angle of the next-hop UAV such as u_j is close to that of the previous-hop node i.e. u_i, and the two UAVs move in the same direction. This reduces the number of hops and delay in the data transfer process. To achieve this goal, in accordance with Fig. 11, a triangle is drawn between the three flying nodes, namely u_i, u_j, and u_d. Each vertex in this triangle corresponds to one of these three UAVs in FANET. Now, the angle between u_i and u_j (i.e. λ_j) is calculated using Equation (26).

$$\lambda_j = \arccos\left(\frac{a_1 b_1 + a_2 b_2 + a_3 b_3}{|A|\,|B|}\right) \tag{26}$$

So that,

$$A = \underbrace{\left(x_i - x_d\right)}_{a_1} i + \underbrace{\left(y_i - y_d\right)}_{a_2} j + \underbrace{\left(z_i - z_d\right)}_{a_3} k \tag{27}$$

And,

$$B = \underbrace{\left(x_i - x_j\right)}_{b_1} i + \underbrace{\left(y_i - y_j\right)}_{b_2} j + \underbrace{\left(z_i - z_j\right)}_{b_3} k \tag{28}$$

Hence,

$$|A| = \sqrt{\left(x_i - x_d\right)^2 + \left(y_i - y_d\right)^2 + \left(z_i - z_d\right)^2} \tag{29}$$

And,

$$|B| = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 + \left(z_i - z_j\right)^2} \tag{30}$$

As a result, λ_j is normalized using Equation (31).

$$\lambda_j^{norm} = \frac{\lambda_j}{\pi}, \quad 0 < \lambda_j < \pi \tag{31}$$

According to Equation (31), if λ_j^{norm} approaches zero, u_i and u_j move in a similar direction.

Finally, the merit function is calculated based on Equation (32).

$$F_{merit} = \ell_1 \left(1 - \left|\overrightarrow{VF}_{ij}^{norm}\right|\right) + \ell_2 IDis_{ij}^{norm} + \ell_3 E_j^{norm} + \ell_4 \left(1 - \lambda_j^{norm}\right) \tag{32}$$

where ℓ_1, ℓ_2, ℓ_3, and ℓ_4 are the weight coefficients in $[0, 1]$. When choosing their value, the sum of these weights must be one i.e. $\sum_{i=1}^{4} \ell_i = 1$. In CF-GPSR, these coefficients have the same value, i.e. $\ell_1 = \ell_2 = \ell_3 = \ell_4 = \frac{1}{4}$ so that the four mentioned criteria, namely velocity factor, ideal distance, residual energy, and movement angle, have the same effect on the merit function.

5. Simulation and evaluation of results

Here, the simulation process is performed using the NS 3.23 simulator to compare CF-GPSR with three other routing schemes, namely UF-GPSR [34], GPSR-PPU [40], and GPSR [25]. In the simulation environment, 20 to 200 flying nodes are randomly distributed in a network with the size $1000 \times 1000 \times 1000\ m^3$ [34]. The initial energy of these flying nodes is $100\ J$, and their communication radius changes from 200 to 300 meters. The random waypoint mobility model is used to simulate the movement of flying nodes in the network, and the velocity of flying nodes changes from 5 to 10 m/s [34]. In addition, the simulation time is 500 seconds. Other simulation parameters are presented in Table 5. To evaluate the performance of CF-GPSR under different conditions, four simulation scenarios are defined based on change in the number of flying nodes, change in the communication range of UAVs, change in the number of connections, and change in the size of data packets. These four simulation scenarios are detailed in Table 6. In the simulation process, CF-GPSR is compared with UF-GPSR, GPSR-PPU, and GPSR in terms of delay, data delivery ratio, data loss ratio, and throughput. To increase the accuracy of the evaluation process, each test is repeated 20 times, the average results are presented, and the confidence interval is equal to 95%. The four evaluation criteria are introduced below:

Table 5
Simulation parameters [34].

Parameter	Value
Simulator	NS 3.23
Compared protocols	CF-GPSR, UF-GPSR, GPSR-PPU, and GPSR
Evaluation criteria	End-to-end delay, PDR, PLR, and throughput
Network size	$1000 \times 1000 \times 1000\ m^3$
Number of flying nodes	20-200
Initial energy of flying nodes	100 J
Velocity of flying nodes	$5 - 10\ m/s$
Communication range of flying nodes	$200 - 300\ m$
Mobility model	Random waypoint
Traffic model	Constant bit rate (CBR)
Data rate	54 Mbps
Packet size	512, 1024, 1500, and 2048 bytes
Mac layer standard	IEEE 802.11
Antenna	Omni Antenna
CBR connections	5, 10, 15, and 20
Simulation time	500 s
Iterations of the simulation process	20
Transmission power	20 dBm

Algorithm 4 Merit function.

Input: u_i: Flying node i
NT_i: Neighbor table related to u_i
FCS_i: Filtered candidate set related to u_i
r: Communication range of flying nodes
(x_i, y_i, z_i): Position of u_i
(V_i, Θ_i, ϕ_i): Speed information of u_i
Output: Merit value related to neighbors of u_i
Begin
for $j = 1$ to $|FCS_i|$ **do**
  if u_j belongs to FCS_i **then**
    $\mathbf{u_i}$: Calculate $\left|\overrightarrow{VF_{ij}}\right|$ based on Equation (19);
    $\mathbf{u_s}$: Normalize $\left|\overrightarrow{VF_{ij}}\right|$ in accordance with Equation (23);
    $\mathbf{u_i}$: Obtain d_{ij} from Equation (1);
    $\mathbf{u_s}$: Normalize d_{ij} and obtain $IDis_{ij}$ using Equation (24);
    $\mathbf{u_i}$: Extract E_j from NT_i;
    $\mathbf{u_s}$: Normalize E_j in accordance with Equation (25);
    $\mathbf{u_i}$: Get λ_j from Equation (26);
    $\mathbf{u_s}$: Normalize λ_j based on Equation (31);
    $\mathbf{u_s}$: Obtain the merit value of u_j based on Equation (32);
  end if
end for
for $j = 1$ to $|FCS_i|$ **do**
  $\mathbf{u_s}$: Find a flying node with the maximum merit value (such as u_j) from FCS_i;
  $\mathbf{u_s}$: Select u_j as its next-hop node;
  $\mathbf{u_s}$: Send its data packets to u_j;
end for
End

- **End-to-end delay (EED):** Equation (33) is used to calculate this evaluation criterion. This criterion represents a time interval from when the source flying node transmits a data packet until it arrives at the desired destination.

$$EED = \frac{\sum_{PK_i \in P = \{PK_1, \ldots, PK_p\}} \left(t_D\left(PK_i\right) - t_S\left(PK_i\right)\right)}{\sum_{i=1}^{p} PK_i} \tag{33}$$

so PK_i is i-th packet delivered to the destination UAV. P is the packet set and p is the total number of these data packets. In addition, $t_D\left(PK_i\right)$ and $t_S\left(PK_i\right)$ are two delivery and transmission times of PK_i.

- **Packet delivery rate (PDR):** Equation (34) is used to calculate PDR, which denotes the percentage of packets that have been successfully delivered to the destination node.

$$PDR = \frac{\sum_{d=1}^{n_d} PK_d}{\sum_{s=1}^{n_s} PK_s} \times 100 \tag{34}$$

where PK_d and n_d are d-th packet derived to the destination node and the total number of these delivered data packets, respectively. Additionally, PK_s and n_s indicate s-th packet sent from the source node and the total number of these packets, respectively.

- **Packet loss rate (PLR):** Equation (35) is used to calculate PLR. This criterion represents the percentage of packets that have not reached the destination node.

$$PLR = \frac{\sum_{l=1}^{n_l} PK_l}{\sum_{s=1}^{n_s} PK_s} \times 100 \tag{35}$$

where PK_l and n_l indicate l-th lost packet and the number of lost packets, respectively. Also, PK_s and n_s are s-th packet sent from the source node and the total number of these packets, respectively.

- **Throughput:** Equation (36) is used to calculate throughput. This criterion shows the ratio of the packets delivered to the destination to the time needed to transfer these packets.

$$Throughput = \frac{\sum_{d=1}^{n_d} PK_d}{t_{transmission}} \tag{36}$$

so that PK_d and n_d are d-th packet derived to the destination node and the total number of these delivered data packets, respectively. Also, $t_{transmission}$ indicates the time needed to send these packets.

5.1. End-to-end delay (EED)

Here, the different routing methods are evaluated in terms of the first criterion, namely EED, according to the four simulation scenarios mentioned in Table 6. Fig. 12 is related to the first scenario that evaluates the effect of the number of flying nodes on EED in different schemes. According to this figure, CF-GPSR has the lowest EED compared to other methods. In CF-GPSR, EED is approximately 17.34%, 24.85%, and 32.33% lower than that in UF-GPSR, GPSR-PPU, and GPSR, respectively. The main reason for this is the use of cylindrical filtering that reduces the size of the initial candidate set and accelerates the next-hop selection process. On the other hand, CF-GPSR uses a merit function, including velocity factor, ideal distance, residual energy, and movement

Table 6
Simulation scenarios.

Simulation scenario	Parameter	Value
Scenario 1	Number of flying node	Number of flying nodes changes from 20 to 200.
	Communication radius of flying nodes	Communication radius of flying nodes is fixed. It is set on 250 meters.
	CBR connections	Number of CBR connections is fixed. It is set on 15.
	Packet size	The size of packets is fixed. It is set on 2048 bytes.
Scenario 2	Number of flying node	Number of flying nodes is fixed. It is set on 100 node.
	Communication radius of flying nodes	Communication radius of flying nodes changes from 200 to 300 meters.
	CBR connections	Number of CBR connections is fixed. It is set on 15.
	Packet size	The size of packets is fixed. It is set on 2048 bytes.
Scenario 3	Number of flying node	Number of flying nodes is fixed. It is set on 100 node.
	Communication radius of flying nodes	Communication radius of flying nodes is fixed. It is set on 250 meters.
	CBR connections	Number of CBR connections change from 5 to 20.
	Packet size	The size of packets is fixed. It is set on 2048 bytes.
Scenario 4	Number of flying node	Number of flying nodes is fixed. It is set on 100 node.
	Communication radius of flying nodes	Communication radius of flying nodes is fixed. It is set on 250 meters.
	CBR connections	Number of CBR connections is fixed. It is set on 15.
	Packet size	The size of packets changes from 512 to 2048 bytes.

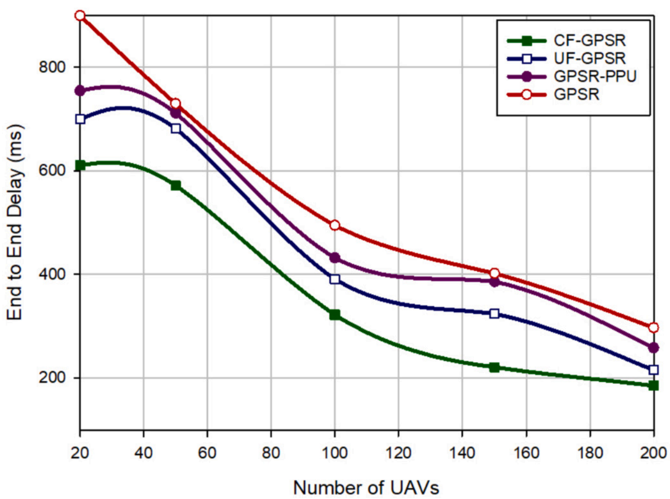

Fig. 12. End-to-end delay based on the number of flying nodes.

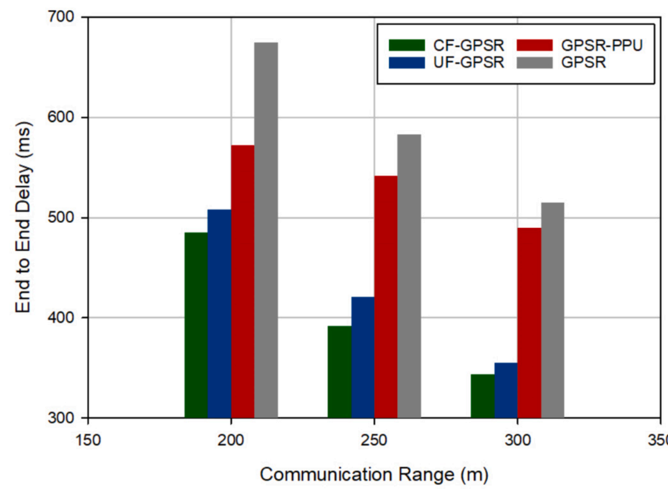

Fig. 13. End-to-end delay based on the communication range.

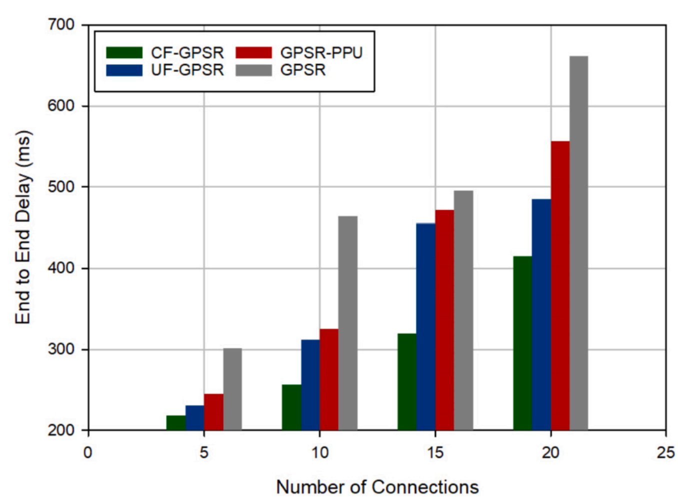

Fig. 14. End-to-end delay based on the number of connections.

angle to decide on the best next-hop UAV. The main purpose of this merit function is to form stable paths between source-destination pairs. The route stability severely reduces the number of broken links on this path. Hence, EED is reduced in the data transfer process. Fig. 12 shows a reverse relationship between EED and the number of flying nodes, so when the number of nodes is increasing in the network environment, EED is decreasing in all routing methods and vice versa. This is a rational outcome because when the network density is high, the flying nodes are closer to each other. As a result, they stay in the communication range of each other at a longer time. This increases the stability of the formed paths and decreases EED in the data transfer process. Fig. 13 is related to the second simulation scenario that evaluates the effect of communication range on delay in different schemes. According to this figure, CF-GPSR has the lowest EED and decreases this evaluation criterion by 4.91%, 23.88%, and 31.13% in comparison with UF-GPSR, GPSR-PPU, and GPSR, respectively. This is because CF-GPSR and UF-GPSR take into account the distance, movement direction, and velocity of UAVs in the next-hop selection process, so they have good EED. However, GPSR and GPSR-PPU have the weakest EED because these techniques have only relied on the distance criterion in the greedy forwarding technique. However, the position prediction mechanism helps GPSR-PPU to make better decisions when choosing the next-hop UAV, and its performance is better than GPSR. Fig. 13 shows the reverse relationship between communication range and EED. Obviously, the increase in communication range has a positive effect on improving route stability and reduces delay in the routing process. Fig. 14 is related to the third simulation scenario that shows the effect of connections on EED in different methods. According to this figure, CF-GPSR has the best performance in terms of delay compared to other methods. EED in CF-GPSR is approximately 18.41%, 24.33%, and 37.08% lower than UF-GPSR, GPSR-PPU,

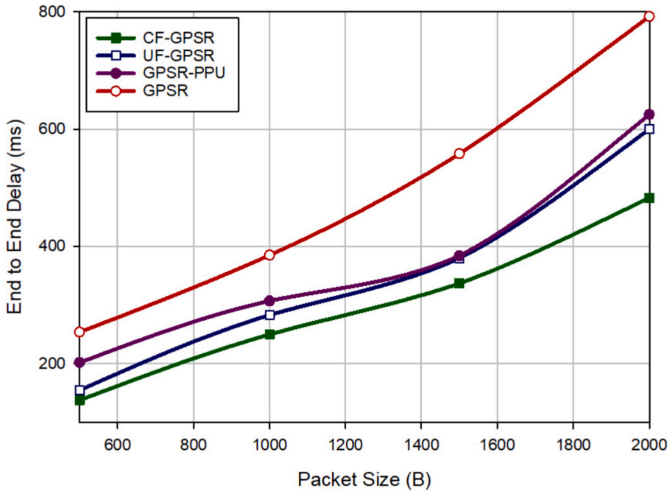

Fig. 15. End-to-end delay based on the size of packet.

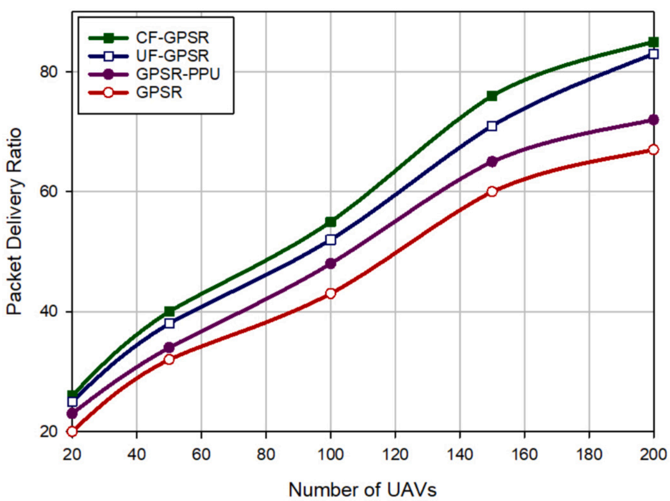

Fig. 16. Packet delivery rate based on the number of flying nodes.

and GPSR, respectively. This figure shows a direct relationship between EED and CBR connections. When the number of connections is increasing in the network, the data transfer process experiences more delay in all GPSR-based routing methods. Obviously, when the network includes more connections, traffic and congestion are high in FANET. As a result, UAVs need more time to deliver their data packets. Finally, Fig. 15 is related to the fourth scenario and indicates the effect of packet size on EED. According to this figure, delay in CF-GPSR is about 14.81%, 20.42%, and 27.39% lower than that in UF-GPSR, GPSR-PPU, and GPSR, respectively. Obviously, it takes more time to deliver large data packets to the destination because they cause more traffic in FANET. In all four simulation scenarios, the superiority of the proposed method to reduce delay is due to its ability to find stable routes between source-destination pairs. This has greatly reduced the number of broken communication routes. This decreases the need for data retransmission in the network.

5.2. Packet delivery rate (PDR)

Here, different routing methods are evaluated in terms of the second evaluation criterion, namely PDR, according to the four simulation scenarios introduced in Table 6. Fig. 16 is related to the first simulation scenario that shows the effect of the number of flying nodes on PDR in different schemes. According to this figure, CF-GPSR has the highest PDR and improves it by 4.83%, 16.53%, and 27.03% in comparison with UF-GPSR, GPSR-PPU, and GPSR, respectively. In Fig. 16, when the number of nodes is 20, PDR in CF-GPSR, UF-GPSR, GPSR-PPU, and GPSR are approximately 26%, 25%, 23%, and 20%, respectively. This shows that in low-density networks, communication links between UAVs are very weak and are quickly cut off. As a result, a large number of data packets are lost. However, when the number of UAVs is 200, PDR in CF-GPSR, UF-GPSR, GPSR-PPU, and GPSR is equal to 85%, 83%, 72%, and 67%, respectively. This shows that the performance of these routing schemes is desirable in high-density networks because the distance between flying nodes is reduced in this case, and the more stable links are made between UAVs. As a result, the routes are valid for a longer time interval. This leads to high PDR. The better performance of CF-GPSR is because the residual energy of flying nodes is an important parameter in the merit function, and when a flying node with a good energy level plays the role of the next-hop UAV, this formed route is valid for a long time. Consequently, the data transfer process is successful. Fig. 17 is related to the second simulation scenario, which shows the effect of communication range on PDR in different schemes. In this figure, CF-GPSR has the highest PDR and increases the evaluation criterion by 5.71%, 13.27%, and 26.13% compared to UF-GPSR, GPSR-PPU, and GPSR, respectively. Among these routing methods, GPSR has the lowest PDR because every flying node searches its candidate set to find the nearest node to the destination. Therefore, the next-hop node is very close to the boundary of the communication range of the previous node. This strategy is not suitable for highly dynamic networks such as FANETs because these UAVs quickly get out of the communication range of each other. As a result, the number of broken links is high, and data packets are lost. However, UF-GPSR employs a new concept called ideal distance. Based on this new concept, nodes close to the communication boundary are not a good option for playing the role of the next-hop node because it increases lost packets. Also, a flying node should not choose nodes, which are very near to itself as the next-hop UAV because the proximity of the two consecutive nodes raises the number of hops in the routing path and increases delay in the data transfer process. In CF-GPSR, it is desirable that the distance between the two consecutive UAVs in the routing path is close to the ideal distance. This concept improves PDR in CF-GPSR. Fig. 18 is related to the third simulation scenario that evaluates the effect of the number of connections on PDR in different methods. This figure shows the superiority of the proposed method compared to other GPSR-based routing schemes. As shown in this figure, when the number of connections is 5, PDR is very low in all four routing methods so that PDR in CF-GPSR, UF-GPSR, GPSR-PPU, and GPSR is equal to 21%, 18%, 14%, and 9%, respectively. However, when the number of connections is 20, PDR in CF-GPSR, UF-GPSR, GPSR-PPU, and GPSR is 63%, 57%, 39%, and 34%, respectively. In fact, when the network includes more connections, it experiences fewer communication interruptions, and data packets successfully reach the destination UAV. Fig. 19 is re-

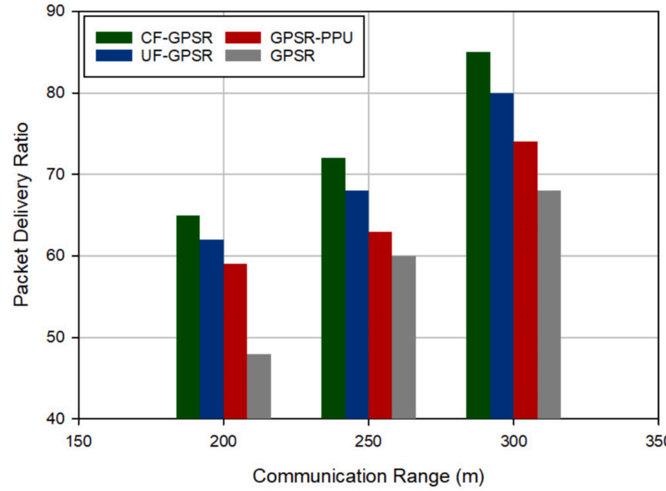

Fig. 17. Packet delivery rate based on communication range.

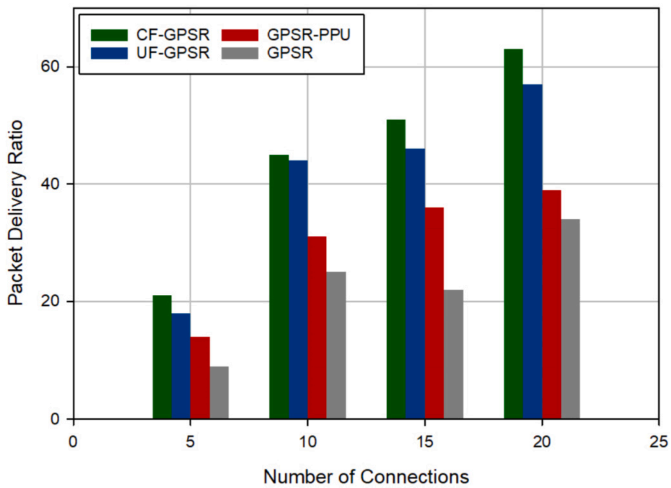

Fig. 18. Packet delivery rate based on connections.

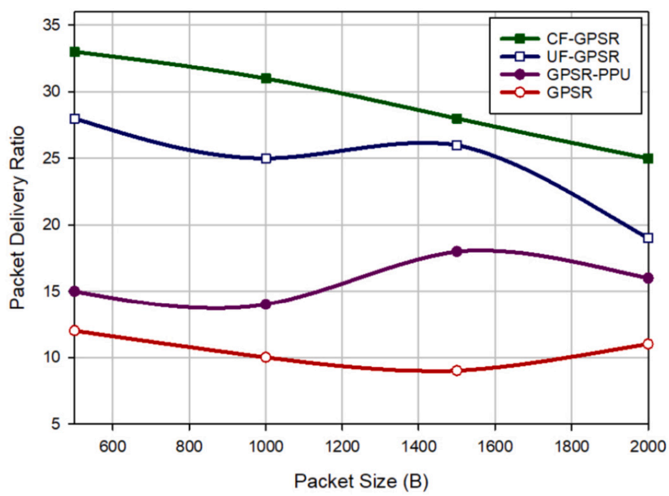

Fig. 19. Packet Delivery rate based on packet size.

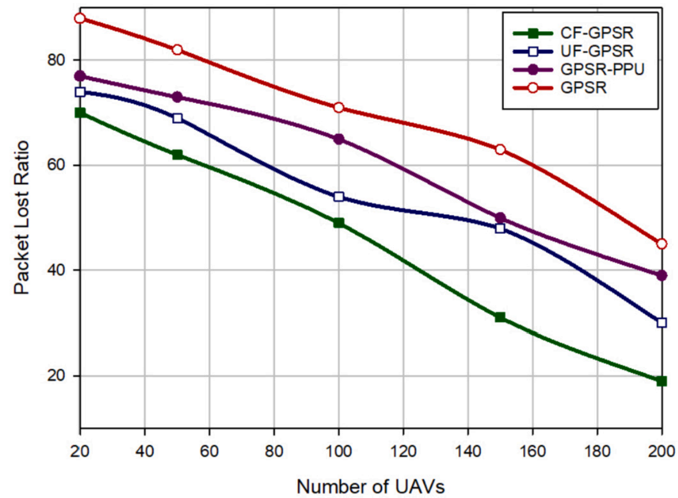

Fig. 20. Packet loss rate based on the number of nodes.

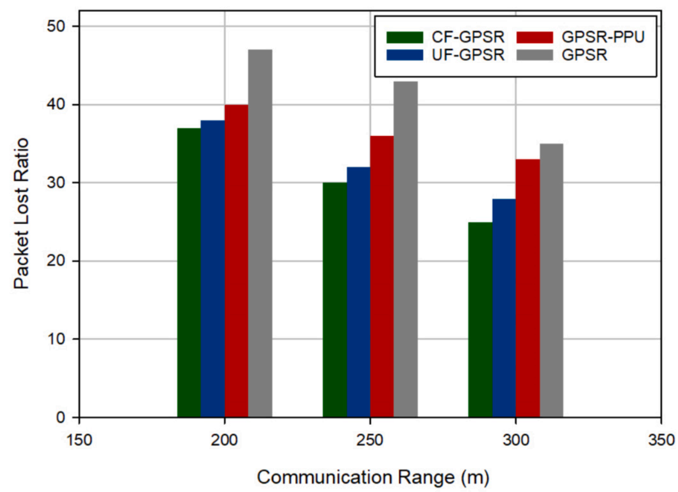

Fig. 21. Packet loss rate based on communication range.

lated to the fourth simulation scenario that shows the effect of packet size on PDR in different schemes. According to this figure, CF-GPSR has the best PDR than other routing approaches. As shown in Fig. 19, when packet size changes from 512 bytes to 2048 bytes, PDR in CF-GPSR decreases from 33% to 25%. In this case, PDR in UF-GPSR changes between 28% and 19%, that in GPSR-PPU is between 15% and 16%, and that in GPSR changes between 12% and 11%. Obviously, when the size of the data packets is large, PDR decreases in all the routing methods because the network has more traffic, and some data segments may be lost. As a result, there is a need for re-transferring data packets. However, if the size of data packets is smaller, the network traffic is low, and the data packets reach the destination successfully.

5.3. Packet loss rate (PLR)

Here, different routing schemes are evaluated in terms of the third evaluation criterion, namely PLR, in the four simulation scenarios stated in Table 6. Fig. 20 is related to the first simulation scenario that shows the effect of the number of flying nodes on PLR in different approaches. As shown in this figure, CF-GPSR has the lowest PLR compared to other routing schemes and reduces this criterion by 16%, 24.01%, and 23.81% compared to UF-GPSR, GPSR-PPU, and GPSR, respectively. The most important reason for this issue is that CF-GPSR can effectively reduce the search space in the candidate set. As a result, the selection process of the best next-hop UAV is very fast. Also, the flying nodes in the routing path are close to each other in terms of velocity and movement direction. Additionally, they keep a suitable distance from each other. CF-GPSR tries to choose these UAVs from the flying nodes with sufficient energy. As a result, the formed paths are stable. This stability improves PDR and reduces PLR. Fig. 21 is related to the second simulation scenario that shows the effect of communication range on PLR. According to this figure, CF-GPSR improves PLR approximately by 6.12%, 15.58%, and 26.40% in comparison with UF-GPSR, GPSR-PPU, and GPSR, respectively. Fig. 21 expresses a reverse relationship between communication range and PLR. When flying nodes have a large communication range, communication links between these nodes are valid for a longer time. As a result, the formed routes are more stable, and data packets reach the destination successfully. Fig. 22 is related to the third scenario that shows the effect of connections on PLR in different routing schemes. According to this figure, CF-GPSR reduces PLR by 9.52%, 21.13%, and 33.86% in comparison with CF-GPSR, GPSR-PPU, and GPSR, respectively. Obviously, when the network includes high connections, the formed paths are more stable, and PLR is low. Finally, Fig. 23 is related to the fourth scenario and shows the effect of packet size on PLR. In CF-GPSR, PLR is approximately 7.19%, 10.86%, and 16.62% lower than that in UF-GPSR, GPSR-PPU, and GPSR, respectively. The better performance of CF-GPSR and UF-GPSR in this scenario is due to their high adaptability to FANET because these two schemes take into account the movement of flying nodes in the routing process.

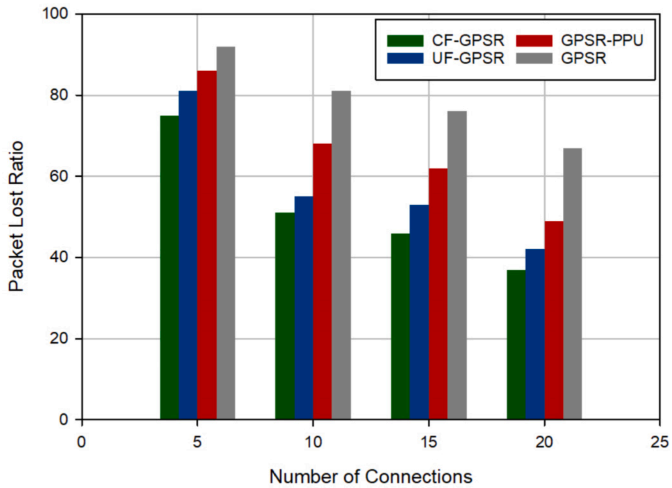

Fig. 22. Packet loss rate based on connection.

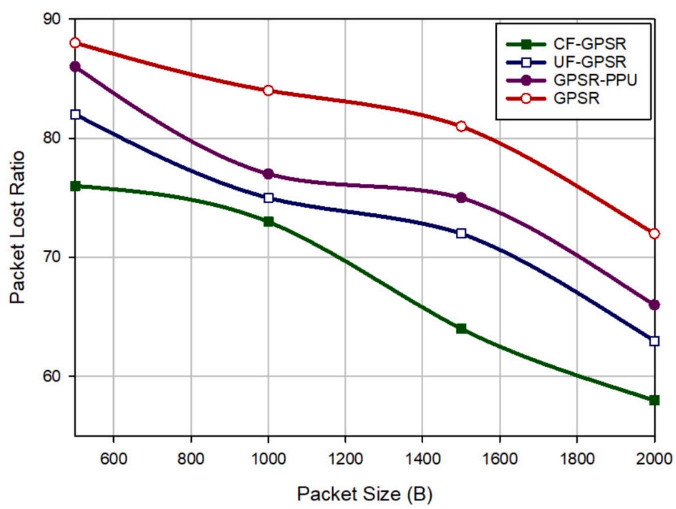

Fig. 23. Packet loss rate based on packet size.

While GPSR-PPU uses a position prediction strategy and chooses the best next-hop UAV based on the predicted distance. This strategy helps GPSR-PPU to improve the stability of routes, but this method focuses only on the distance criterion in the greedy forwarding process. As a result, its performance is weaker than CF-GPSR and UF-GPSR. GPSR has the weakest performance in FANET because it does not pay attention to the movement of flying nodes in the 3D space of FANET.

5.4. Throughput

Here, different routing schemes are examined in terms of the fourth evaluation criterion, namely throughput, in the four simulation scenarios mentioned in Table 6. Fig. 24 is related to the first simulation scenario that shows the effect of the number of flying nodes on throughput in different routing approaches. Based on this figure, CF-GPSR has the best throughput and increases the evaluation scale by 7.05%, 10.36%, and 19.11% in comparison with UF-GPSR, GPSR-PPU, and GPSR, respectively. The reasons for this issue were precisely stated in Section 5.2. Obviously, when the network includes a high density of flying nodes, throughput in all routing schemes is high because the distance between UAVs is low in this case, and communication links between these nodes are more stable. As a result, a high number of packets are successfully delivered to the destination. On the other hand, CF-GPSR performs quickly the routing process due to the use of cylindrical filtering. Hence, the throughput in CF-GPSR is higher than that in other routing schemes.

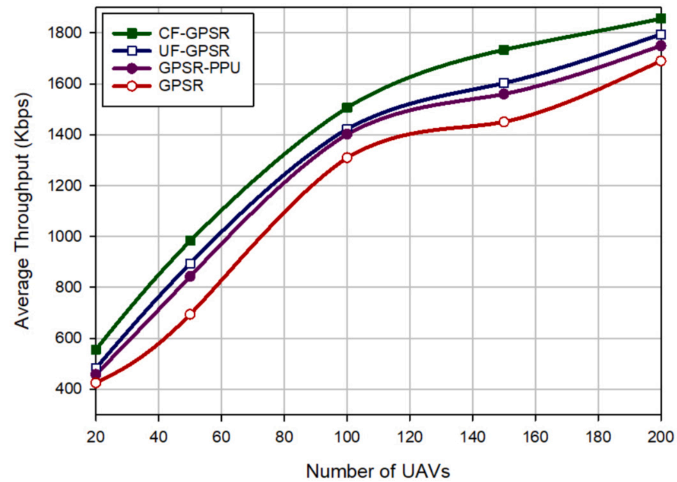

Fig. 24. Throughput based on the number of flying nodes.

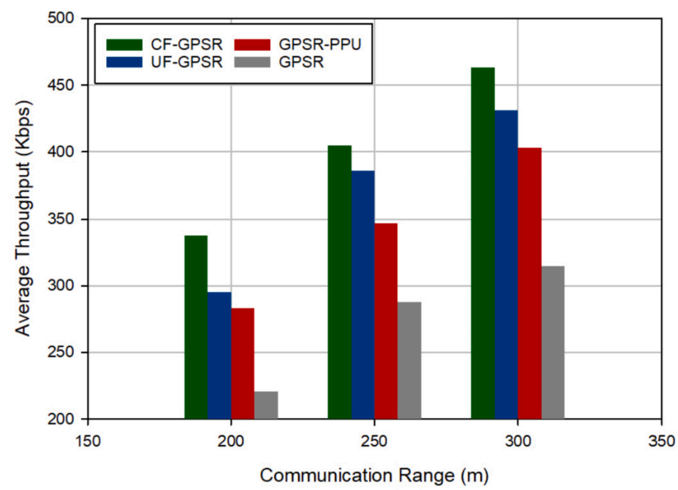

Fig. 25. Throughput based on communication range.

Fig. 25 is related to the second simulation scenario, which shows the effect of communication range on throughput. It is clear that the number of hops in the path is reduced when the communication range of flying nodes is larger. Therefore, data packets need a shorter time to reach the destination, this increases throughput. Fig. 26 is related to the third scenario, which shows the effect of connections on throughput. As the number of connections changes from 5 to 15, throughput in all routing methods is increasing. When the number of connections is 15, throughput in CF-GPSR, UF-GPSR, GPSR-PPU, and GPSR is 1530, 1506, 1211, and 1082 kbps, respectively. However, when the number of connections changes from 15 to 20, throughput in CF-GPSR, UF-GPSR, and GPSR-PPU decreases in the network due to high traffic in the network. Fig. 27 is related to the fourth simulation scenario and shows the effect of packet size on throughput. As shown in this figure, CF-GPSR has the best performance in terms of throughput and improves this evaluation criterion by 0.39%, 5.43%, and 13.49% compared to UF-GPSR, GPSR-PPU, and GPSR. The powerful performance of the proposed method in terms of throughput is due to its ability to reduce PLR, which was examined in Section 5.3.

6. Conclusion

In this paper, a novel cylindrical filtering-based greedy perimeter stateless routing scheme (CF-GPSR) was presented for FANET. CF-GPSR consists of two main steps: neighbor discovery and greedy forwarding strategy. In the greedy forwarding strategy, each flying node uses

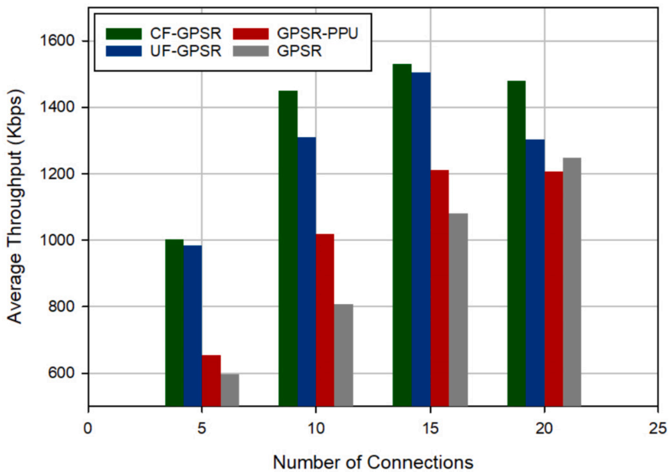

Fig. 26. Throughput based on connections.

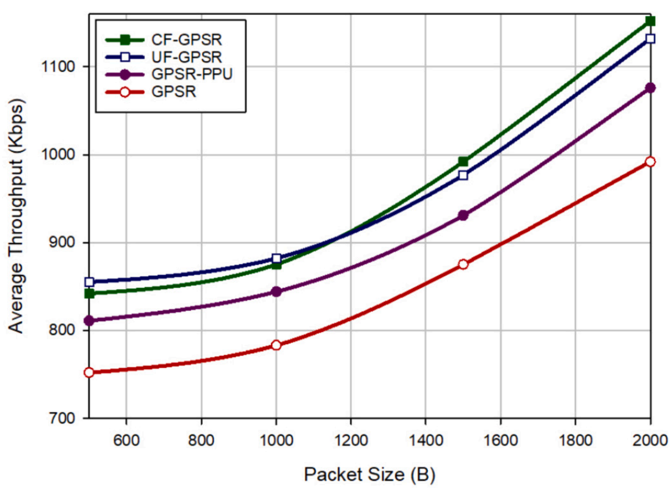

Fig. 27. Throughput based on the size of data packets.

a cylindrical filtering-based technique to reduce the size of its initial candidate set. Then, this flying node decides on the best next-hop UAV based on a merit function, which includes four criteria, namely velocity factor, ideal distance, residual energy, and movement angle, and selects a candidate node with the highest merit value as the next-hop UAV. In the simulation process, four simulation scenarios are defined based on the change in network density, the change in the communication range of nodes, the change in the number of network connections, and the change in the size of packets. Then, CF-GPSR was compared with UF-GPSR, GPSR-PPU, and GPSR in terms of delay, PDR, PLR, and throughput. These evaluations showed that CF-GPSR improves delay, PDR, PLR, and throughput by 17.34%, 4.83%, 16%, and 7.05%, respectively, in the first scenario, namely the change in the number of flying nodes. Also, the proposed method optimizes delay, PDR, PLR, and throughput by 4.91%, 5.71%, 6.12%, and 8.45%, respectively, in the second scenario, namely the change in communication range. In the third scenario, namely the change in the number of connections, CF-GPSR improves EED, PDR, PLR, and throughput by 18.41%, 9.09%, 9.52%, and 7.03%, respectively. In the fourth simulation scenario, namely the change in the packet size, CF-GPSR improves delay, PDR, PLR, and throughput by 14.81%, 19.39%, 7.19%, and 0.39%, respectively. In future research directions, we try to increase the adaptability of CF-GPSR to the dynamic environment. This goal is achieved by setting up a dynamic hello broadcast period. In addition, we will determine the weight coefficients in the merit function through extensive simulations under various scenarios. We also improve its performance by combining the proposed routing method and the machine learning (ML) techniques so that an intelligent filtering is created based on ML strategies, for example Q-learning, deep reinforcement learning (DRL), and optimization algorithms, such as fire hawk optimizer (FHO), crow search algorithm (CSA), grey wolf optimizer (GWO), and sparrow search algorithm (SSA) to decide on the next forwarder intelligently.

CRediT authorship contribution statement

Amir Masoud Rahmani: Methodology, Writing – review & editing. **Amir Haider:** Data curation, Writing – review & editing. **Khursheed Aurangzeb:** Validation, Writing – review & editing. **May Altulyan:** Resources, Writing – review & editing. **Entesar Gemeay:** Resources, Visualization, Writing – review & editing. **Mohammad Sadegh Yousefpoor:** Writing – original draft, Conceptualization, Validation. **Efat Yousefpoor:** Writing – original draft, Methodology, Validation. **Parisa Khoshvaght:** Visualization, Writing – review & editing. **Mehdi Hosseinzadeh:** Project administration, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors would like to acknowledge Deanship of Graduate Studies and Scientific Research, Taif University for funding this work.

Data availability

No data was used for the research described in the article.

References

[1] A. Hemmati, M. Zarei, A. Souri, UAV-based Internet of vehicles: a systematic literature review, Intell. Syst. Appl. (2023) 200226, https://doi.org/10.1016/j.iswa.2023.200226.
[2] M.M. Alam, S. Moh, Joint topology control and routing in a UAV swarm for crowd surveillance, J. Netw. Comput. Appl. 204 (2022) 103427, https://doi.org/10.1016/j.jnca.2022.103427.
[3] P. Kumar, S. Verma, Implementation of modified OLSR protocol in AANETs for UDP and TCP environment, J. King Saud Univ, Comput. Inf. Sci. 34 (4) (2022) 1305–1311, https://doi.org/10.1016/j.jksuci.2019.07.009.
[4] A.I. Abubakar, M.S. Mollel, O. Onireti, M. Ozturk, I. Ahmad, S.M. Asad, Y. Sambo, A. Zoha, S. Hussain, M.A. Imran, Coverage and throughput analysis of an energy efficient UAV base station positioning scheme, Comput. Netw. (2023) 109854, https://doi.org/10.1016/j.comnet.2023.109854.
[5] A. Kujawski, T. Dudek, Analysis and visualization of data obtained from camera mounted on unmanned aerial vehicle used in areas of urban transport, Sustain. Cities Soc. 72 (2021) 103004, https://doi.org/10.1016/j.scs.2021.103004.
[6] M. Aloqaily, O. Bouachir, I. Al Ridhawi, A. Tzes, An adaptive UAV positioning model for sustainable smart transportation, Sustain. Cities Soc. 78 (2022) 103617, https://doi.org/10.1016/j.scs.2021.103617.
[7] M.F. Khan, K.L.A. Yau, M.H. Ling, M.A. Imran, Y.W. Chong, An intelligent cluster-based routing scheme in 5G flying ad hoc networks, Appl. Sci. 12 (7) (2022) 3665, https://doi.org/10.3390/app12073665.
[8] Y. Cheriguene, F.Z. Bousbaa, C.A. Kerrache, S. Djellikh, N. Lagraa, M. Lahby, A. Lakas, COCOMA: a resource-optimized cooperative UAVs communication protocol for surveillance and monitoring applications, Wirel. Netw. (2022) 1–17, https://doi.org/10.1007/s11276-022-03031-8.
[9] R. Dhall, S. Dhongdi, Review of protocol stack development of flying ad-hoc networks for disaster monitoring applications, Arch. Comput. Methods Eng. 30 (1) (2023) 37–68, https://doi.org/10.1007/s11831-022-09791-y.
[10] J. Wang, C. Jiang, Introduction of flying ad hoc networks, in: Flying Ad Hoc Networks: Cooperative Networking and Resource Allocation, Singapore: Springer Singapore, 2021, pp. 1–10.
[11] K. Messaoudi, O.S. Oubbati, A. Rachedi, A. Lakas, T. Bendouma, N. Chaib, A survey of UAV-based data collection: challenges, solutions and future perspectives, J. Netw. Comput. Appl. (2023) 103670, https://doi.org/10.1016/j.jnca.2023.103670.

[12] F. Syed, S.K. Gupta, S. Hamood Alsamhi, M. Rashid, X. Liu, A survey on recent optimal techniques for securing unmanned aerial vehicles applications, Trans. Emerg. Telecommun. Technol. 32 (7) (2021) e4133, https://doi.org/10.1002/ett.4133.

[13] H. Jin, X. Jin, Y. Zhou, P. Guo, J. Ren, J. Yao, S. Zhang, A survey of energy efficient methods for UAV communication, Veh. Commun. (2023) 100594, https://doi.org/10.1016/j.vehcom.2023.100594.

[14] O.S. Oubbati, A. Lakas, F. Zhou, M. Güneş, M.B. Yagoubi, A survey on position-based routing protocols for Flying Ad hoc Networks (FANETs), Veh. Commun. 10 (2017) 29–56, https://doi.org/10.1016/j.vehcom.2017.10.003.

[15] A. Chriki, H. Touati, H. Snoussi, F. Kamoun, FANET: communication, mobility models and security issues, Comput. Netw. 163 (2019) 106877, https://doi.org/10.1016/j.comnet.2019.106877.

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

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

[18] J. Lansky, A.M. Rahmani, M.H. Malik, E. Yousefpoor, M.S. Yousefpoor, M.U. Khan, M. Hosseinzadeh, An energy-aware routing method using firefly algorithm for flying ad hoc networks, Sci. Rep. 13 (1) (2023) 1323, https://doi.org/10.1038/s41598-023-27567-7.

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

[20] J. Lansky, A.M. Rahmani, M. Hosseinzadeh, Reinforcement learning-based routing protocols in vehicular ad hoc networks for Intelligent Transport System (ITS): a survey, Mathematics 10 (24) (2022) 4673, https://doi.org/10.3390/math10244673.

[21] M.S. Yousefpoor, E. Yousefpoor, H. Barati, A. Barati, A. Movaghar, M. Hosseinzadeh, Secure data aggregation methods and countermeasures against various attacks in wireless sensor networks: a comprehensive review, J. Netw. Comput. Appl. 190 (2021) 103118, https://doi.org/10.1016/j.jnca.2021.103118.

[22] A.M. Rahmani, R.A. Naqvi, E. Yousefpoor, M.S. Yousefpoor, O.H. Ahmed, M. Hosseinzadeh, K. Siddique, A Q-learning and fuzzy logic-based hierarchical routing scheme in the intelligent transportation system for smart cities, Mathematics 10 (22) (2022) 4192, https://doi.org/10.3390/math10224192.

[23] M. Hosseinzadeh, S. Ali, L. Ionescu-Feleaga, B.S. Ionescu, M.S. Yousefpoor, E. Yousefpoor, O.H. Ahmed, A.M. Rahmani, A. Mehmood, A novel Q-learning-based routing scheme using an intelligent filtering algorithm for flying ad hoc networks (FANETs), J. King Saud Univ, Comput. Inf. Sci. (2023) 101817, https://doi.org/10.1016/j.jksuci.2023.101817.

[24] M. Gharib, F. Afghah, E.S. Bentley, LB-OPAR: load balanced optimized predictive and adaptive routing for cooperative UAV networks, Ad Hoc Netw. 132 (2022) 102878, https://doi.org/10.1016/j.adhoc.2022.102878.

[25] B. Karp, H.T. Kung, GPSR: greedy perimeter stateless routing for wireless networks, in: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, 2000, August, pp. 243–254.

[26] M. Ahman, A. Rechache, A. Kaci, O. Annad, A. Nacef, S. Ait-Chellouche, Toward an efficient geographical routing protocol for Internet of vehicles, in: Proceedings of the 2023 6th International Conference on Geoinformatics and Data Analysis, 2023, April, pp. 53–59.

[27] Y. Cui, Q. Zhang, Z. Feng, Z. Wei, C. Shi, H. Yang, Topology-aware resilient routing protocol for FANETs: an adaptive Q-learning approach, IEEE Internet Things J. 9 (19) (2022) 18632–18649, https://doi.org/10.1109/JIOT.2022.3162849.

[28] S. Shokrollahi, M. Dehghan, TGRV: a trust-based geographic routing protocol for VANETs, Ad Hoc Netw. 140 (2023) 103062, https://doi.org/10.1016/j.adhoc.2022.103062.

[29] J. Cui, L. Ma, R. Wang, M. Liu, Research and optimization of GPSR routing protocol for vehicular ad-hoc network, China Commun. 19 (10) (2022) 194–206, https://doi.org/10.23919/JCC.2022.00.031.

[30] A. Bengag, A. Bengag, M. Elboukhari, The GPSR Routing Protocol in VANETs: Improvements and Analysis, in: International Conference on Electronic Engineering and Renewable Energy Systems, Springer Nature Singapore, Singapore, 2022, May, pp. 21–28.

[31] B. Zheng, K. Zhuo, H. Zhang, H.X. Wu, A novel airborne greedy geographic routing protocol for flying ad hoc networks, Wirel. Netw. (2022) 1–15, https://doi.org/10.1007/s11276-022-03030-9.

[32] M. Mazouzi, K. Mershad, O. Cheikhrouhou, M. Hamdi, Agent-based reactive geographic routing protocol for Internet of vehicles, IEEE Access (2023), https://doi.org/10.1109/ACCESS.2023.3299230.

[33] G.A. Kakamoukas, P.G. Sarigiannidis, A.A. Economides, FANETs in agriculture-A routing protocol survey, Internet of Things 18 (2022) 100183, https://doi.org/10.1016/j.iot.2020.100183.

[34] S. Kumar, R.S. Raw, A. Bansal, P. Singh, UF-GPSR: modified geographical routing protocol for flying ad-hoc networks, Trans. Emerg. Telecommun. Technol. 34 (8) (2023) e4813, https://doi.org/10.1002/ett.4813.

[35] A.M. Khedr, A. Salim, P.R. PV, W. Osamy, MWCRSF: mobility-based weighted cluster routing scheme for FANETs, Veh. Commun. 41 (2023) 100603, https://doi.org/10.1016/j.vehcom.2023.100603.

[36] A.M. Rahmani, D. Hussain, R.J. Ismail, F. Alanazi, S. Belhaj, M.S. Yousefpoor, E. Yousefpoor, A. Darwesh, M. Hosseinzadeh, An adaptive and multi-path greedy perimeter stateless routing protocol in flying ad hoc networks, Veh. Commun. 50 (2024) 100838, https://doi.org/10.1016/j.vehcom.2024.100838.

[37] M. Hosseinzadeh, J. Tanveer, L. Ionescu-Feleaga, B.S. Ionescu, M.S. Yousefpoor, E. Yousefpoor, O.H. Ahmed, A.M. Rahmani, A. Mehmood, A greedy perimeter stateless routing method based on a position prediction mechanism for flying ad hoc networks, J. King Saud Univ, Comput. Inf. Sci. 35 (8) (2023) 101712, https://doi.org/10.1016/j.jksuci.2023.101712.

[38] A.M. Rahmani, S. Ali, E. Yousefpoor, M.S. Yousefpoor, D. Javaheri, P. Lalbakhsh, O.H. Ahmed, M. Hosseinzadeh, S.W. Lee, OLSR+: a new routing method based on fuzzy logic in flying ad-hoc networks (FANETs), Veh. Commun. 36 (2022) 100489, https://doi.org/10.1016/j.vehcom.2022.100489.

[39] M. Hosseinzadeh, A.H. Mohammed, F.A. Alenizi, M.H. Malik, E. Yousefpoor, M.S. Yousefpoor, O.H. Ahmed, A.M. Rahmani, L. Tightiz, A novel fuzzy trust-based secure routing scheme in flying ad hoc networks, Veh. Commun. 44 (2023) 100665, https://doi.org/10.1016/j.vehcom.2023.100665.

[40] A. Rodrigues, A.B. Reis, S. Sargento, GPSR-PPU: greedy perimeter stateless routing with position prediction and uncertainty for FANETs, in: 2020 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops, IEEE, 2020, March, pp. 1–6.

[41] W.R. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, in: Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, IEEE, 2000, January, p. 10.

[42] H. Ghazzai, M.B. Ghorbel, A. Kadri, M.J. Hossain, Energy efficient 3D positioning of micro unmanned aerial vehicles for underlay cognitive radio systems, in: 2017 IEEE International Conference on Communications (ICC), IEEE, 2017, May, pp. 1–6.

[43] C. Liu, Z. Zhang, On connectivity of flying ad hoc networks in the presence of ground terminal, China Commun. 18 (9) (2021) 1–10, https://doi.org/10.23919/JCC.2021.09.001.