JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 36, 851-864 (2020)
DOI: 10.6688/JISE.202007_36(4).0010

A Novel Mobility-Aware Gradient Forwarding Algorithm for Unmanned Aerial Vehicle Ad Hoc Networks

JUN LI[1,2] AND MING CHEN[3,1]
[1]*College of Command Information Systems*
Army Engineering University of PLA
Nanjing, 210000 P.R. China
[2]*School of Computer and Information*
Anhui Polytechnic University
Wuhu, 241000 P.R. China
[3]*College of Computer Science and Technology*
Nanjing University of Aeronautics and Astronautics
Nanjing, 210000 P.R. China
E-mail: edmondlee@ahpu.edu.cn; mingchennj@163.com

Currently, unmanned aerial vehicle ad hoc networks (UAANETs) have played important roles in both the military and civilian fields. Due to the fast moving speed of UAV nodes, the topology frequently changes, and it is difficult to maintain end-to-end connectivity. When deterministic routing protocols are applied in UAANETs, they cannot achieve ideal performance. In this paper, we propose a new mobility-aware gradient forwarding algorithm (MGFA), which aims to reduce the impairments caused by the highly dynamic topology. Our proposed algorithm introduces mobility information into the computation of the routing cost, which guides the relaying nodes to make forwarding decisions. The simulation results indicate that the MGFA can indeed outperform other deterministic routing protocols in terms of the packet delivery ratio, routing overhead, throughput, and average end-to-end delay.

Keywords: gradient forwarding, mobility-aware, highly dynamic topology, deterministic routing, UAV, UAANETs

1. INTRODUCTION

During the past few decades, unmanned aerial vehicles (UAVs) have gained increasingly more applications in both military and civilian fields due to their easy development, flexibility, hovering ability, and low maintenance costs. Because multiple UAVs can carry out missions more efficiently and reliably than a single UAV, they are extensively used for surveillance and reconnaissance [1], border patrol [2], remote sensing [3], search and rescue [4], good delivery [5], and disaster communication [6].To execute complex tasks efficiently and economically, multiple UAVs must communicate and cooperate via a wireless ad hoc network, which is named the unmanned aerial vehicle ad hoc networks (UAANET) [7, 8]. Compared with mobile ad hoc networks (MANETs) and vehicular ad hoc networks (VANETs), UAANETs have some unique characteristics,including highly dynamic topology, fast movement speed, and unstable radio-link quality [9]. These unique characteristics bring additional challenges to the design of routing protocols for UAANETs.

The fast moving speed and high mobility of UAV nodes leads to frequently varying distances between all UAV nodes in the UAANET, which produces a continually changing

Received October 3, 2019; revised November 12 & December 27, 2019; accepted January 28, 2020.
Communicated by Xiaohong Jiang.

network topology, frequent breakage of the end-to-end path, and unstable radio-link quality. Due to the highly dynamic topology, deterministic routing protocols can suffer from significant control overhead, degradation of the packet delivery ratio, and increased end-to-end delay. In this paper, we develop a mobility-aware gradient forwarding algorithm (MGFA) for UAANETs, which is an adaptation of the existing gradient routing (GRAd) routing protocol [10]. In the GRAd routing protocol, the sending node broadcasts messages to its neighboring nodes. Then, the receivers make forwarding decisions based on the gradient of the routing cost. To address the highly dynamic topology of UAANETs, the MGFA introduces mobility information into the computation of the routing cost. Simulation results illustrate that compared to other deterministic routing protocols, the MGFA can perform very well in UAANETs.

The main contributions of this paper can be summarized as follows:

1. We propose a new mobility-aware gradient forwarding algorithm for UAANETs, which is named the MGFA. In the MGFA, the sending node does not select a fixed next-hop node, but rather broadcasts messages to its neighboring nodes. One of the neighboring nodes forwards the message, which has the lowest routing cost to the destination node. The proposed algorithm is based on two major schemes: gradient forwarding and a new routing cost computation based on the mobility information of nodes.
2. A new routing cost computation scheme is proposed to guide the receivers to make the forwarding decisions. And it is based on the position and velocity of the nodes. The proposed scheme adds position information and velocity information in the header of the message. When an intermediate node receives the message, it can compute the routing cost to the source node based on position information and velocity information of itself and the source node. Then intermediate nodes make the forwarding decisions based on the computed routing cost.
3. We implement the proposed algorithm using the NS-3 network simulator. We evaluate the performance of our algorithm in terms of the packet delivery ratio, throughput, average end-to-end delay, and routing overhead and compare its results with those of other deterministic routing protocols.

The rest of the paper is structured as follows: In Section 2, we discuss related studies on the routing protocols for UAANETs. The design of the mobility-aware gradient routing algorithm is formally described in Section 3. Section 4 reports the simulation and analysis of the performance of the proposed routing algorithm. The last section summarizes this study.

2. RELATED WORKS

For decades, the design of routing protocols for UAANETs has received increasing attention from both academic research and industry. Attempts have been made to provide reliable and sufficient packet delivery for UAANETs. Most of these routing protocols were adapted from traditional routing protocols for MANETs and VANETs. These routing protocols are deterministic protocols characterized by preselecting a set of nodes to forward messages, followed by carrying messages through these selected nodes to the destination [9, 11, 12].

The first category includes proactive routing protocols, such as Optimized Link State Routing Protocol(OLSR) [13], Destination-Sequenced Distance Vector(DSDV) [14], and Better Approach To Mobile Ad hoc Networking (B.A.T.M.A.N) [15]. The OLSR is often adapted for UAANETs, and it is based on the periodical exchange of Hello and topology control (TC) messages to maintain the topology information of the whole network. It selects the multippoint relay (MPR) nodes to obtain the link-state information and forward messages. In recent years, some extensions of the OLSR ,e,g, the ML-OLSR [16], the P-OLSR [17], the D-OLSR [18] and the OLSR_PMD [19], have been proposed for UAANETs. The P-OSLR uses UAV nodes' GPS signals to predict the wireless link quality in order to avoid packet loss [17]. The ML-OLSR utilizes the position information and speed of the neighboring nodes to select the MPR nodes [16]. The D-OLSR routing protocol tends to select the fastest UAV node as an MPR node [17]. The OLSR_PMD chooses stable neighboring nodes as the MPR nodes based on mobility prediction [19]. An extension of the DSDV is illustrated in [20], which is a table-driven routing protocol based on a modified Bellman-Ford algorithm.The B.A.T.M.A.N has been used with UAANETs [21]. It proactively maintains the information about all the UAV nodes based on the exchange of the OriGinator Message (OGM). In this kind of modified proactive routing protocol for UAANETs, the UAV node periodically collects topology information to update its routing table and to establish the end-to-end path between every pair of UAVs in the network. In UAANETs, the high mobility of the UAV nodes causes frequent end-to-end connection failures and highly dynamic topology. These modified proactive routing protocols must update their routing table with topology changes, which introduces significant control overhead. They also respond slowly to changes in the network topology, which causes additional delay.

The second category comprises reactive routing protocols, such as DSR [22] and AODV [23]. They work exactly on demand, and they discover routes between the source node and the destination node when one node tries to communicate with another. In the DSR routing protocol, every message contains the complete source-to-destination addresses for forwarding messages, which causes performance degradation, especially when the topology is highly dynamic [24]. Different from DSR routing protocols, in AODV, each message contains the address of the destination node. Then, the intermediate node forwards messages using the routing table, which is better suited to a dynamic environment. However, under highly dynamic topology, the end-to-end path usually suffers from breakages. These reactive routing protocols frequently have to use routing recovery to find a new route between the source and destination, which results in significantly high delay [25].

In an environment with the frequently changing topology and varying wireless-link quality, the above-mentioned deterministic routing protocols can suffer from frequent routing recovery, high control overhead, high congestion, slow convergence time, and high end-to-end delay. Hence, these routing protocols cannot provide reliable and efficient message delivery under highly dynamic topology. Poor first proposed the GRAd routing protocol, which has excellent immunity to the changing topology in MANETs [10]. In the GRAd routing protocol, the sending node does not determine the next-hop node. Receivers make their routing decisions based on the routing cost, which is computed using the number of hops. The sending node broadcasts a message containing the routing cost, and the receivers decide whether to relay the message or not, which is based on whether its routing cost is lower than that of the received message. As this process repeats, the message is relayed from the source to the destination, such as a stone rolling downhill. Because UAV nodes have a faster movement speed and the network topology of UAANETs changes more frequently compared to MANETs, the GRAd routing protocol suffers degraded performance due to outdated routing costs when it is applied in UAANETs. The MGFA attempts to utilize the mobility information of nodes to compute the routing cost. The routing cost is evaluated using the current location of the source node and the previous location recorded when the source node sends the messages. The cost information of the source node and the destination node is efficiently updated via flooding.

3. THE DESIGN OF THE MOBILITY-AWARE GRADIENT FORWARDING ALGORITHM

The main idea of the mobility-aware gradient routing protocol is to introduce the mobility information of nodes into the routing cost computation. In the gradient forwarding algorithm, each node has a routing cost value for relaying messages to the destination, which is computed based on the number of hops to the destination node. When a node sends one message to the destination node, it will add cost information to the message and broadcast to all neighboring nodes within the communication range. All the receivers simultaneously receive the message, but only those nodes can forward the message, which have a lower routing cost than the message. When other receivers overhear the same message delivered by other nodes with the same cost or smaller cost, they will cancel forwarding. The sender will repeatedly broadcast the message until it overhears the transmission of its neighboring nodes. Whe the GRAd routing protocol is applied in UAANETs without any adaptation, due to the high mobility of UAV nodes, the routing cost information quickly becomes outdated.

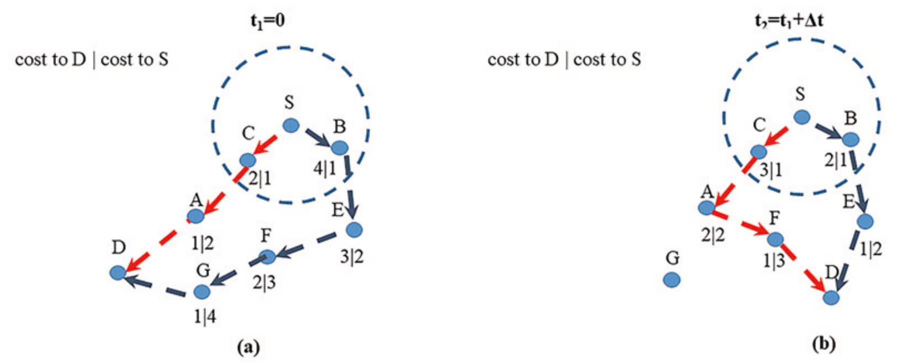

Fig. 1. Impact of the highly dynamic topology.

An example is shown in Fig. 1 (a), where we assume all nodes in the network have initialized their cost toward the source node S or the destination node D at time t_1. When node S sends messages to node D, the messages should be relayed by node C first, and not by node B, because node C has a small cost. In the GRAd routing protocol, the cost is calculated based on the number of hops to the destination. As illustrated in Fig. 1 (b), because UAANETs have faster-moving nodes and a highly dynamic topology compared to MANETs, after a short time Δt, node D has arrived at a new position, which is close to node E and node F. The routing cost becomes outdated in all nodes, which means that the available information is not sufficiently accurate to make forwarding decisions. The network performance significantly degrades when receivers make forwarding decisions based on outdated routing costs.

To address the highly dynamic topology in UAANETs, we modify the cost value calculation method by adding the mobility information of the nodes. In the MGFA, one node sends a message to its destination node, and it adds its current position information and velocity information to the header of the message. When an intermediate node receives the message from an original node (source node or destination node), it extracts the position information and velocity information from the received message. Then, the receiver computes both the current location of the original node and the relative distance between itself and the original node. If the current distance between receivers and the original node is larger than the distance when the message is sent, it shows that the original node tends to move away from the receiver and a higher routing cost should be set. If the current distance is smaller than the distance between the receiver and the original node when the message is sent, it indicates that the receiver tends to move toward the original node. Then, a smaller routing cost should be set. In this way, the routing cost can better reflect the dynamic topology of UAANETs than the conventional GRAd routing protocol. It reduces the impact of the fast movement speed of UAV nodes.

An example of the MGFA routing algorithm is shown in Fig. 2. All messages are relayed from the source node S to the destination node D. When node S broadcasts messages , its neighboring nodes simultaneously receive messages, including node B, node A, and node C. Because the moving direction of node C is toward node D, node C has a smaller cost value than the other receivers, such as node B and node A. The routing cost of node E should be lower than that of node F and node G when node C broadcasts messages to its neighboring nodes, including node E, node F, and node G. This is because node E will be closer to node D than other nodes in the future. Therefore, all messages from node S to node D will be transmitted by node C and node E.

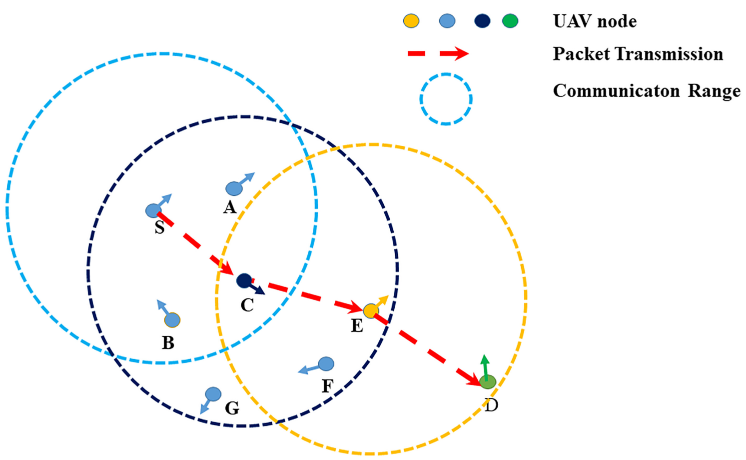

Fig. 2. An example of the MGFA routing algorithm.

3.1 MGFA Message Structure

The structure of the message header of the MGFA routing protocol is illustrated in Fig. 3.

- Type: This field takes two values; MSG_REQUEST starts the route-request process and MSG_DATA transmits application data.
- Seq_Number: The value of this field depends on the original node. When the original node sends a new message, the field is incremented by one. In practice, we

<table>
<tr><td>Type</td><td>Seq Number</td><td>ORIGIN_ ID</td><td>TARG_ID</td><td>COST</td><td>TTL</td></tr>
<tr><td colspan="2">(X_{ORIGN}, Y_{ORIGIN}, Z_{ORIGIN})</td><td colspan="2">(V_x, V_y, V_z)</td><td colspan="2">Timestamp</td></tr>
</table>

Fig. 3. Structure of the message header in the MGFA.

Algorithm 1 Algorithm of the route request process.

Input:

Output:

```
Construct an MSG_REQUEST message, set the Type field as MSG_REQUEST, set
an initial value of the Seq_Number field, set the COST field as zero and fill in other
information into the corresponding fields of the MSG REQUEST message;
repeat
    Broadcast the MSG_REQUEST message and start Timer;
    Wait for Timer to expire;
    if Timer expired then
        Seq_Number field is increment by one;
    end if
until Receive the MSG_REQUEST message from the destination node
Return;
```

combine the ORIGIN_ID and Seq_Number to distinguish a new message from the messages already received.

- ORIGIN_ID: Identifier of the source node.
- TARG_ID: Identifier of the destination nodes.
- COST:When the original node sends an MSG_REQUEST message, the value of this field is set to 0. When the MSG_REQUEST message is received, this field is updated by the receiving node. When the original node sends an MSG_DATA message, the value of this field is obtained from its routing cost table.
- TTL: This field indicates the lifetime of the message. When a message is relayed by one hop, TTL decrements by one. When it reaches zero, the message is dropped. The value of this field is set to the max number of hops to any node in the network.
- (X_{ORIGN}, Y_{ORIGIN}, Z_{ORIGIN}): This field records the position of the original node when it sends a message. The relaying node uses this field and its position information to calculate the previous distance between itself and the original node.
- (V_x, V_y, V_z): The original node adds its velocity information in the message when it sends a message. When the relaying node receives the message, it can use this information to predict the current position of the source node. The MGFA uses this information to compute the current distance between the source node and the relaying node, which will be introduced to update the cost.
- Timestamp: This field records the moment when the original node sends one message. When an intermediate node receives one message, it extracts the value of this field to compute the elapsed time between the current of time and the moment when the original node sends the message.

As Algorithm 1 shows a MSG_REQUEST type (MMR) message is sent by the source node when it wants to transmit data to the destination node. At this time, the source node does not have any cost information about the destination node. The source node starts a timer after it finishes sending the MMR message. If it does not receive a reply from the destination before the timer expires, the source node resends another MMR message to its neighboring nodes.

Algorithm 2 Algorithm of processing received message.

```
Input:   received message M;
Output:
 1: if M.TARG_ID == id of receiving node then
 2:   if M.TYPE == MSG_DATA then
 3:     Receive M;
 4:     Return;
 5:   end if
 6:   if M.TYPE == MSG_REQUEST then
 7:     Extract information from M;
 8:     Invoke function update_cost_table() with extracted information as arguments;
 9:     if M.ORIGIN_ID does not exist in cost_table then
10:       Send MSG_REQUEST message to the souce node;
11:     end if
12:     Return;
13:   end if
14: end if
15: if M.TARG_ID ≠ id of receiving node then
16:   if M.TYPE == MSG_DATA then
17:     Extract information from M;
18:     Retrieve cost value CV from cost_table according to M.TARG_ID;
19:     if CV <M.COST then
20:       repeat
21:         Try to broadcast M to other nodes;
22:       until Overhear other nodes that broadcast M with smaller cost OR success-
          fully broadcast M;
23:     end if
24:     Invoke function update_cost_table() with extracted information as arguments;
25:     Return;
26:   end if
27:   if M.TYPE == MSG_REQUEST then
28:     Extract information from M;
29:     Invoke function update_cost_table() with extracted information as arguments;
30:     Retrieve cost value CV from cost_table and update M.COST field with CV;
31:     Broadcast M to other nodes;
32:   end if
33: end if
```

As lines 27 through 32 in Algorithm 2 show, when an intermediate node receives an MMR message, the intermediate node extracts the relevant information, and it then updates the value of the COST field in the cost table, by the function update_cost_table. Then, it updates the COST field of the MMR message using the routing cost, and it broadcasts the MMR message to other nodes.

As lines 16 through 26 in Algorithm 2 show, when an intermediate node receives an MMD message, if the cost in the cost table is smaller than that of the message, it will relay the message. Meanwhile, the intermediate node opportunistically updates its routing cost to the source node when it receives MMD messages from other nodes.

Lines 6 through 13 in Algorithm 2 and Algorithm 3 illustrate the process of dealing with received MMR messages at the destination. If the TARG_ID does not exist in the cost table, the destination node will insert a new cost entry that contains the cost to the source node and return a new MMR message to inform the source node to end the request process. Then, it keeps updating the cost to the source node when it receives a new message containing a smaller cost.

3.2 Cost Table and Computing Cost

The structure of the cost table entry is illustrated in Fig. 4 .

ID	Seq_Number	COST	LIFE

Fig. 4. Structure of the message header in the MGFA.

- ID: Identifier of a UAV node.
- Seq_Number: This field distinguishes if the incoming message is the newest message.
- COST: This field records the latest cost for sending a message to the ID.
- LIFE: This field indicates the expiration time of a cost-table entry. If a cost-table entry is updated, then the field is reset. If the current time ever exceeds LIFE, the cost-table entry is deleted.

Referring to Eq. (1), the cost in the MGFA consists of two components: cost_{hop} and $cost_{position}$. The cost_{hop} is computed based on the number of hops to deliver messages to the original node. When the relaying node receives an MMR message from the original node, it extracts the cost from the received message as $cost_{pre}$, and then it adds one to $cost_{pre}$ to compute cost_{hop}, as illustrated by Eq. (2).

$$cost = cost_{hop} + cost_{position} \tag{1}$$

$$cost_{hop} = cost_{pre} + 1 \tag{2}$$

At the time that the message is received, the relaying node extracts the position (x_o, y_o, z_o) of the original node from the received message. The relaying node calculates the distance between itself and the original node at the moment when the original node sends the message. To do this, it uses its position (x_r,y_r,z_r,) and the original node's position information:

$$dist_{pre} = \sqrt{(x_r - x_o)^2 + (y_r - y_o)^2 + (y_r - y_o)^2}. \tag{3}$$

We use the velocity of the relaying node (v_{xr}, v_{yr}, v_{zr}) and the velocity of the original node(v_{xo}, v_{yo}, v_{zo}) to compute the current distance between the relaying node and the original node when the relaying node receives the message. The calculation is illustrated by Eqs. (4)-(7). Time Δt is the elapsed time between the current time and the moment when the original node sends the message.

$$dist_{cur-x} = x_r - x_o + (v_{xr} - v_{xo}) \times \Delta t \tag{4}$$

Algorithm 3 function update_cost_table.

Input:
cost_table
M.Seq_Number
M.ORIGIN_ID
M.COST
M.Timestamp
position information M.((X_{ORIGN}, Y_{ORIGIN}, Z_{ORIGIN})
velocity information M.(V_x, V_y, V_z)
system current time NOW

Output:
Δt=NOW- M.Timestamp;
if M.ORIGIN_ID does not exist in cost_table **then**
    Compute the cost CV based on Eqs. (1)-(8)
    Insert new cost entry in cost_table where ID = M.ORIGIN_ID and COST = CV;
    Return;
end if
if M.ORIGIN_ID exists in cost_table AND cost_table.COST <M.COST **then**
    Return;
end if
if M.ORIGIN_ID exists in cost_table AND cost_table.COST >M.COST **then**
    Compute the cost CV based on Eqs. (1)-(8)
    Update the cost entry in cost_table with CV when ID=M.ORIGIN_ID;
    Return;
end if

$$dist_{cur-y} = y_r - y_o + (v_{yr} - v_{yo}) \times \Delta t \tag{5}$$

$$dist_{cur-z} = z_r - z_o + (v_{yz} - v_{yz}) \times \Delta t \tag{6}$$

$$dist_{cur} = \sqrt{dist_{cur-x}^2 + dist_{cur-y}^2 + dist_{cur-z}^2} \tag{7}$$

The calculation of $cost_{position}$ is done using Eq. (8). If the current distance between the relaying node and the original node is greater than the distance when the original node sends the message, $cost_{position}$ is set to 1; otherwise, it is set to 0.

$$cost_{position} = \begin{cases} 0, & \frac{dist_{cur}}{dist_{pre}} \leq 1 \\ 1, & \frac{dist_{cur}}{dist_{pre}} \geq 1 \end{cases} \tag{8}$$

As Algorithm 3 shows, in the MGFA routing algorithm, each node keeps a cost record (cost_table), which records the cost to any node in a UAANET. Every intermediate node makes the forwarding decisions based on the value of the COST field in the cost table. When a node receives a message, it checks the ORIGIN_ID field of the message. If the source node is not in its cost table, it adds a new cost table entry and then calculates the new cost to the source node. If the source node is in the cost table, it checks whether the arriving message has a lower cost. If the cost of the message is small, it extracts the position information and velocity information of the source node from the arriving message. It computes the new cost, and then it updates the cost table.

4. SIMULATION AND PERFORMANCE

The simulation of the MGFA was performed by using the NS-3 network simulator. The performance of the MGFA was evaluated with respect to the average end-to-end delay, packet delivery ratio, throughput, and routing overhead, and the results were compared to those of the P-OLSR and AODV routing protocols.

In the simulation, the UAANET included 30 UAV nodes, and its mission area was 1000 m*1200 m. The mobility of the UAV nodes was random waypoint mobility. The movement speed of the UAV node was between 5 m/s and 25 m/s. We used the CBR protocol to generate the simulation data from the source nodes. The parameters of the simulation are listed in Table 1.

Table 1. Simulation parameters.

Parameter	Value
Network size	1000 m*1200 m
Number of UAVs	30
Node mobility model	Random Waypoint Mobility
Node transmission radius	80 m
UAV movement speed	5-25 m/s
Antenna model	Omni
Channel capacity	10 Mbps
Routing protocol	MGFA,AODV,P-OLSR
Traffic type	CBR
Simulation time	200 s

To evaluate the performance of the MGFA routing algorithm, four key performance metrics were utilized.

- Packet delivery ratio: The ratio of the number of packets successfully received by the target and the number of packets sent by the source node.
- Average end-to-end delay: The time consumed by the messages transmitted along the route from a source node to a target node.
- Throughput: The number of messages transmitted by the network per time slot.
- Routing overhead: The ratio of the number of control packets and the number of packets transmitted by the source node.

The impact of node mobility on the packet delivery ratio is illustrated in Fig. 5. As the movement speed of the UAV nodes increases, compared to the MGFA, the AODV and P-OLSR suffer from significant message losses. As the movement speed of the UAV nodes increases, the topology becomes more dynamic and the quality of the wireless link becomes more unstable. Compared to the AODV and P-OLSR, the MGFA has the advantage of using gradient forwarding and the mobility information of the UAV nodes to achieve the best performance in terms of the packet delivery ratio. This is because the AODV and P-OLSR fail to rediscover an alternative route promptly in the face of the highly dynamic topology. In contrast, the MGFA uses multiple neighboring nodes to relay messages and lets receivers make the forwarding decisions. When one neighboring node moves out of the communication range, other neighboring nodes can transmit messages without rediscovering a new route.

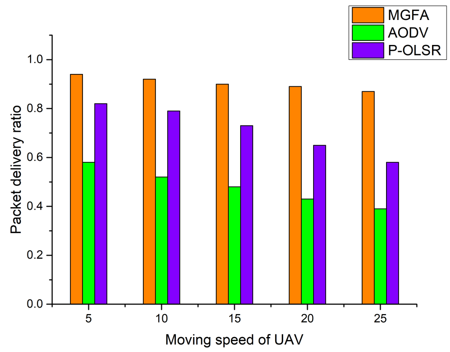

Fig. 5. Comparison of the packet delivery ratio.

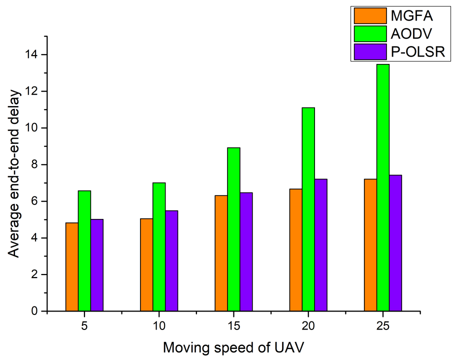

Fig. 6. Comparison of the average end-to-end delay.

Fig. 6 shows the average end-to-end delay of the three routing algorithms. As the movement speed of the UAV nodes increases, the AODV experiences increased average end-to-end delay because it spends more time rediscovering the end-to-end route between the source node and the destination node. Because the MGFA relays messages based on the routing cost, which is computed by the combination of the number of hops and mobility information of nodes, the MGFA delivers messages from source nodes to destination nodes with less retransmission. Therefore, the average end-to-end delay of the MGFA increases slightly as the movement speed of the UAV nodes increases. The average end-to-end delay of the P-OLSR is similar to that of the MGFA. However, the P-OLSR shows limited performance in terms of the packet delivery ratio, throughput, and routing overhead.

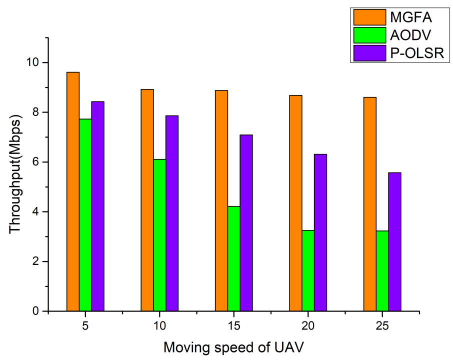

Fig. 7. Comparison of the throughput.

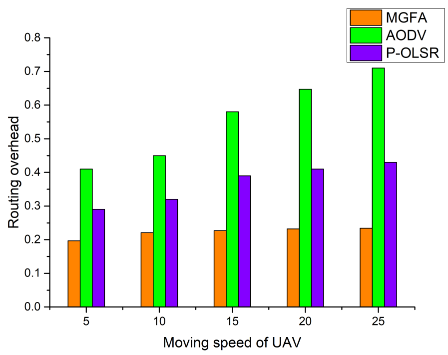

Fig. 8. Comparison of the routing overhead.

Fig. 7 shows how the network throughput of the three routing algorithms varies with the movement speed of the UAV nodes. For the AODV and P-OLSR, the network throughput decreases as the movement speed increases. The AODV and P-OLSR suffer from more packet loss and frequent end-to-end routing rediscovery. In the MGFA, the sending node broadcasts messages to its neighboring nodes, and receivers make forwarding decisions based on whether they are close to the position of the destination node without routing discovery or updating the global topology information. The MGFA avoids frequent route recovery and more packet loss under the highly dynamic topology. Therefore, it can alleviate the decreased network throughput.

Fig. 8 shows the routing overhead for the three routing algorithms. It indicates that the AODV generates a significantly larger number of control messages for frequent route discovery and route recovery. The P-OLSR also has a larger number of control messages to collect topology information to update the routing table, when dealing with the highly dynamic topology. In the MGFA, control messages are added to the route discovery process, and the maintenance of the route is not based on control messages. The MGFA uses gradient forwarding based on the routing cost, which is opportunistically updated, to guard against frequent end-to-end connection failures. As a result, the routing overhead of the MGFA is the lowest among the three routing protocols.

5. CONCLUSIONS

This study has proposed a novel mobility-aware gradient forwarding algorithm for unmanned aerial vehicle ad hoc networks. In our routing algorithm, receivers make forwarding decisions based on the routing cost, which is different from previous deterministic routing protocols. Additionally, we have also introduced mobility information into the computation of the routing cost to reduce the impairments caused by the highly dynamic topology. As shown by the simulation results, our proposed routing algorithm outperforms other deterministic routing protocols in terms of the packet delivery ratio, routing overhead, and average end-to-end delay.

ACKNOWLEDGMENT

This research was partly supported by the National Natural Science Foundation of China under Grant No. 61772271, the Natural Science Foundation of Anhui under Grant No. 1908085MF183, and the Natural Science research project in the Anhui Provincial Department of Education (No. KJ2017A104, NO. KJ2018A0116, TSKJ2017B29).

REFERENCES

1. T. Kopfstedt, M. Mukai, M. Fujita, and C. Ament, “Control of formations of uavs for surveillance and reconnaissance missions,” *IFAC Proceedings Volumes*, Vol. 41, 2008, pp. 5161-5166.
2. Z. Sun, P. Wang, M. C. Vuran, M. A. Al-Rodhaan, A. M. Al-Dhelaan, and I. F. Akyildiz, “Bordersense: Border patrol through advanced wireless sensor networks,” *Ad Hoc Networks*, Vol. 9, 2011, pp. 468-477.
3. I. Jawhar, N. Mohamed, J. Al-Jaroodi, D. P. Agrawal, and S. Zhang, “Communication and networking of uav-based systems: Classification and associated architectures,” *Journal of Network and Computer Applications*, Vol. 84, 2017, pp. 93-108.
4. P. Vincent and I. Rubin, “A framework and analysis for cooperative search using uav swarms,” in *Proceedings of ACM Symposium on Applied Computing*, 2004, pp. 79-86.
5. A. Goodchild and J. Toy, “Delivery by drone: An evaluation of unmanned aerial vehicle technology in reducing co2 emissions in the delivery service industry,” *Transportation Research Part D: Transport and Environment*, Vol. 61, 2018, pp. 58-67.
6. S. Chowdhury, A. Emelogu, M. Marufuzzaman, S. G. Nurre, and L. Bian, “Drones for disaster response and relief operations: A continuous approximation model,” *International Journal of Production Economics*, Vol. 188, 2017, pp. 167-184.

7. T. Andre, K. A. Hummel, A. P. Schoellig, E. Yanmaz, M. Asadpour, C. Bettstetter, P. Grippa, H. Hellwagner, S. Sand, and S. Zhang, "Application-driven design of aerial communication networks," *IEEE Communications Magazine*, Vol. 52, 2014, pp. 129-137.
8. S. Hayat, E. Yanmaz, and R. Muzaffar, "Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint," *IEEE Communications Surveys and Tutorials*, Vol. 18, 2016, pp. 2624-2661.
9. L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Communications Surveys and Tutorials*, Vol. 18, 2016, pp. 1123-1152.
10. R. Poor, Z. Smith, M. Paris, A. Wheeler, and R. Kelsey, "Ad hoc wireless network using gradient routing," Patent 10/457,205, 2004.
11. O. K. Sahingoz, "Networking models in flying ad-hoc networks (fanets): Concepts and challenges," *Journal of Intelligent & Robotic Systems*, Vol. 74, 2014, pp. 513-527.
12. J. Jiang and G. Han, "Routing protocols for unmanned aerial vehicles," *IEEE Communications Magazine*, Vol. 56, 2018, pp. 58-63.
13. T. H. Clausen and P. Jacquet, "Optimized link state routing protocol (OLSR)," *RFC*, Vol. 3626, 2003, pp. 1-75.
14. C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *Proceedings of ACM SIGCOMM Conference on Communications Architectures, Protocols and Applications*, 1994, pp. 234-244.
15. B. Sliwa, D. Behnke, C. Ide, and C. Wietfeld, "B.a.t.mobile: Leveraging mobility control knowledge for efficient routing in mobile robotic networks," in *Proceedings of IEEE Globecom Workshops*, 2016, pp. 1-6.
16. Y. Zheng, Y. Wang, Z. Li, L. Dong, Y. Jiang, and H. Zhang, "A mobility and load aware olsr routing protocol for uav mobile ad-hoc networks," in *Proceedings of International Conference on Information and Communications Technologies*, 2014, pp. 1-7.
17. S. Rosati, K. Kruzelecki, G. Heitz, D. Floreano, and B. Rimoldi, "Dynamic routing for flying ad hoc networks," *IEEE Transactions on Vehicular Technology*, Vol. 65, 2016, pp. 1690-1700.
18. A. I. Alshbatat and L. Dong, "Cross layer design for mobile ad-hoc unmanned aerial vehicle communication networks," in *Proceedings of IEEE International Conference on Networking*, 2010, pp. 331-336.
19. M. Song, J. Liu, and S. Yang, "A mobility prediction and delay prediction routing protocol for UAV networks," in *Proceedings of the 10th International Conference on Wireless Communications and Signal Processing*, 2018, pp. 1-6.
20. A. Garcia-Santiago, J. Castañeda-Camacho, J. Guerrero-Castellanos, and G. M. Aguilar, "Evaluation of AODV and DSDV routing protocols for a FANET: further results towards robotic vehicle networks," in *Proceedings of the 9th IEEE Latin American Symposium on Circuits & Systems*, 2018, pp. 1-4.
21. I. A. Kaysina, D. S. Vasiliev, A. Abilov, D. S. Meitis, and A. E. Kaysin, "Performance evaluation testbed for emerging relaying and coding algorithms in flying ad hoc networks," in *Proceedings of Moscow Workshop on Electronic and Networking Technologies*, 2018, pp. 1-5.
22. D. B. Johnson, Y. Hu, and D. A. Maltz, "The dynamic source routing protocol (DSR) for mobile ad hoc networks for ipv4," *RFC*, Vol. 4728, 2007, pp. 1-107.

23. C. E. Perkins, E. M. Belding-Royer, and S. R. Das, “Ad hoc on-demand distance vector (AODV) routing,” *RFC*, Vol. 3561, 2003, pp. 1-37.
24. M. Khan, I. Khan, A. Safi, and I. Quershi, “Dynamic routing in flying ad-hoc networks using topology-based routing protocols,” *Drones*, Vol. 2, 2018, pp. 27-42.
25. K. Singh and A. K. Verma, “Experimental analysis of aodv, dsdv and olsr routing protocol for flying adhoc networks (fanets),” in *Proceedings of IEEE International Conference on Electrical, Computer and Communication Technologies*, 2015, pp. 1-4.

Jun Li received M.S. degree in Computer Science from Anhui Polytechnic University, Wuhu, China. He is currently pursuing Ph.D. at Army Engineering University of PLA, Nanjing, China. He is also working as a Lecturer at School of Computer and Information, Anhui Polytechnic University. His research interests include UAV ad hoc network, software-defined wireless networking, and vehicular ad hoc network.

Ming Chen received Ph.D. degree in Information System from the Institute of Communication Engineering in 1991, Nanjing, China. He is currently a Professor and Ph.D. supervisor in College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China. He is also a Ph.D. supervisor at the Army Engineering University of PLA. His research interests include network architecture, unmanned aerial vehicle networks, network measurement, and future networks.