

Delay Tolerant Packet Forwarding Algorithm Based on Location Estimation for Micro Aerial Vehicle Networks

Shiji Li^{1,2*}, Guyu Hu¹, Youwei Ding³ and Yun Zhou⁴

¹ Command and Control Engineering College, Army Engineering University of PLA
Nanjing, 210007 - China
[e-mail: lshjiii@sina.com]

² Unit 94619 of PLA
Lu'an, 237132 - China

³ School of Artificial Intelligence and Information Technology, Nanjing University of Chinese Medicine
Nanjing, 210016 - China

⁴ Unit 78111 of PLA
Chengdu, 610011 - China

*Corresponding author: Shiji Li

*Received July 15, 2019; revised September 29, 2019; accepted December 18, 2019;
published March 31, 2020*

Abstract

In search and rescue mission, micro aerial vehicles (MAVs) are typically used to capture image and video from an aerial perspective and transfer the data to the ground station. Because of the power limitation, a cluster of MAVs are required for a large search area, hence an ad-hoc wireless network must be maintained to transfer data more conveniently and fast. However, the unstable link and the intermittent connectivity between the MAVs caused by MAVs' movement may challenge the packet forwarding. This paper proposes a delay tolerant packet forwarding algorithm based on location estimation for MAV networks, called DTN_{est} algorithm. In the algorithm, ferrying MAVs are used to transmit data between MAVs and the ground station, and the locations of both searching MAVs and ferrying MAVs are estimated to compute the distances between the MAVs and destination. The MAV that is closest to the destination is selected greedy to forward packet. If a MAV cannot find the next hop MAV using the greedy strategy, the packets will be stored and re-forwarded once again in the next time slot. The experiment results show that the proposed DTN_{est} algorithm outperforms the typical DTN_{geo} algorithm in terms of packet delivery ratio and average routing hops.

Keywords: Delay tolerant network, ferrying MAV, location estimation, micro aerial vehicle network, packet forwarding algorithm, store-carry-forward

1. Introduction

Micro aerial vehicles (MAVs) which weigh only a few kilograms, have been widely used in many applications, such as aerial photography, agricultural monitoring, disaster relief, military and so on. MAVs are usually equipped with computing and wireless communication devices, sensors and cameras to collect image and video and transmit them to the ground station [1][2]. With the characteristics of low cost, portability, and no casualty risk, MAVs have become a hotspot in both academic and industrial fields.

A cluster of MAVs is usually required to complete large and complex tasks coordinately, since the energy, flight distance and communication range of each MAV are limited. The cluster of MAVs is typically organized as an ad-hoc multi-hop network to transmit data from MAVs to the ground station [3]. However, packet forwarding is a challenge in practical MAV networks because of unstable links and intermittent connectivity of the network caused by rapid movements of MAVs [4].

Since global positioning system (GPS) and inertial measurement unit provide the location information of MAVs [5][6], Mozaffari et al. proposed a geographic routing based method to forward packets to the nodes that are spatially closer to the destination [7]. In practical applications, the motion and the unstable link of MAVs may change the topology of the network frequently, but these geographic routing methods are not adequate for MAV networks that face intermittent connectivity [8].

To cope with intermittent connectivity, delay tolerant networking (DTN) is a feasible approach. Using the communication opportunity formed by nodes flying into each other's transmission range to transmit messages hop by hop, Raffelsberger et al. integrated the "store-carry-forward" with the adaptive routing method [9]. However, traditional DTN concepts often use flooding or limited flooding methods based on stochastic knowledge about moving nodes, which do not fit well in practical MAV networks because node trajectories are usually deterministic [10].

With the increasing demand for data communication in the challenging MAV networks environment, it is urgent to develop new packet forwarding technologies. A single packet forwarding strategy cannot fit well with MAV network. Fortunately, it is found that properly combining different strategies can improve the performance of packet forwarding [11]. Inspired by this hybrid strategy, this paper combines the geographic location based routing method and the "store-carry-forward" concept in DTN. Furthermore, ferrying MAVs are introduced to transmit message between MAVs.

This paper proposes a packet forwarding algorithm called DTN_{est} for MAV networks based on the location and velocity of MAVs. The main contributions of this paper are as follows:

(1) The MAV network is modeled as a DTN network, and uses the "store-carry-forward" strategy for packet forwarding. To solve the problem of intermittent connectivity, ferrying MAVs are introduced to transmit messages between disconnected subnets. MAVs's flight paths are usually planned in advanced. Therefore, this paper introduces a linear extrapolation method to estimate the position of MAVs to obtain a more accurate network topology.

(2) Two new routing mechanisms are designed. First, this paper introduces a location estimation mechanism, which can greatly reduce the packet forwarding failure caused by outdated location and velocity information. Second, motion driven packet forwarding mechanism are developed to distinguish the ferrying MAV from other MAVs, which can

alleviate the ping-pong problem caused by ferrying MAVs' periodic movement and decrease the routing hops required for packet transmission.

(3) The packet forwarding algorithm DTN_{est} is designed and implemented. Location estimation mechanism and motion driven packet forwarding mechanism are used to determine the optimal next hop. Simulation results show that DTN_{est} is superior to the existing algorithm in terms of packet delivery ratio and average routing hops.

The rest of this paper is organized as follows. Section 2 reviews the related work. The network model is formulated in Section 3, some new packet forwarding mechanisms and a packet forwarding algorithm DTN_{est} are developed in Section 4. Section 5 evaluates the proposed algorithm, and Section 6 concludes the paper and points out the future research.

2. Related Work

The existing routing algorithms for MAV ad-hoc networks can be mainly divided into three categories.

(1) Network topology-based routing algorithms. The optimized link state routing (OLSR) protocol was applied to the network consisting of two micro airplanes and a ground station [12]. The results show that the OLSR cannot deal with the rapidly changing topology in a MAV network properly and quickly enough. In [13], Asadpour *et al.* demonstrated that the B.A.T.M.A.N routing protocol has long route convergence time because of topology changes. The main reason is that the mobile ad-hoc routing protocols such as AODV [14], OLSR [15], TORA [16], DSR [17] need stable link to converge, while the network topology of unmanned aerial vehicles changes quickly, and links are established and broken frequently.

(2) DTN routing algorithms. DTN [18] is an applicable method for intermittent connectivity. Flooding (or limited flooding) method is usually adopted to deal with the long-time disconnection of networks in pure DTN routing algorithm. In Epidemic routing algorithm [19] messages were copied to all nodes that in the communication range to improve the packet delivery ratio, but unlimited copies increase the overhead and decline the transmission efficiency of unmanned aerial vehicles. Although the Spray and Wait algorithm was proposed to limit the number of copies [20], the methods with both unlimited and limited copies will incur a large overhead and a long delay when UAV transmits large data [21].

(3) Geographical location-based routing algorithms. Since the global positioning system such as GPS can provide the location information of the UAV, it is a feasible method for UAV to transmit messages to nodes that are spatially closer to the destination node. A geographical routing protocol was developed in [22], and the UAV link lifetime was estimated using mobility direction and velocity of UAVs. However, pure geographic routing can not achieve packets forwarding well in UAV networks due to the intermittent connectivity.

As the network topology of UAVs may change frequently, and network connectivity cannot be guaranteed, a single routing strategy cannot achieve data forwarding well. It is found that properly combining different strategies can improve the effectiveness of routing methods. The GeoDTN+Nav algorithm combined the geographic location-based routing and DTN algorithm to utilize the position information and the storage-carry-forward concept [23]. A location-aided delay tolerant routing (LADTR) UAV network protocol for application in post-disaster operations was proposed in [24], which adopted single-copy data forwarding to avoid replication of each message. These approaches may not work because of the nature of UAV mobility in free space.

MAVs' motion usually results in long time disconnection in MAV networks. In order to solve the data transmission failure caused by network interruption, ferrying MAVs were introduced in [25] to transfer data between MAVs and ground station by moving back and forth. A motion-driven packet forwarding algorithm DTN_{geo} was proposed. If an end-to-end shortest path to the destination exists, the MAV forwards the message to the neighbor in the shortest path. Otherwise, the MAV forwards the message to the neighbor closer to the destination. If there is no neighbor node closer to the destination, the message will be carried. The DTN_{geo} algorithm only considers the location of the MAV at current moment, and forwards messages to the nodes closer to destination, but probably the source itself may be closer to the destination in the future, and the message will be transmitted back to the source. This kind of message transmission back and forth is called the "ping-pong" effect.

A MAV can sense its own location using GPS, so the authors assumed that each MAV is aware of the real time geographic position of all MAVs [24][25]. In practical application, this premise is not realistic. MAVs obtain their own location, but usually broadcast periodically, and then other MAVs can receive the location updated packet. Therefore, the location of other MAVs is usually out of date and inaccurate, which will result in greatly reduced packet delivery ratio.

To solve the "ping-pong" problem and packet loss problem mentioned in above paragraphs, this paper proposes a delay tolerant packet forwarding algorithm based on location estimation (DTN_{est}) for MAV networks, which yields excellent results in terms of packet delivery ratio and average routing hops.

3. MAV Network Model

The deployment of MAV network is generally sparse. Due to the limited transmission distance of high-throughput links, the MAV network is prone to be interrupted. To maintain connectivity, two typical solutions are adopted, using more searching MAVs to maintain the link quality between each MAV and the ground station, and using ferrying MAVs to transmit the data generated by the searching MAVs to the ground station. In the former solution, lots of energy of each MAV is wasted on maintaining the link quality, and the allocation of MAVs will be highly complex. While in the ferrying MAVs solution, data will be transmitted to the ground station through multi-hops path which consists of both searching MAVs and ferrying MAVs, this paper adopts the ferrying solution because the total MAVs are less and path planning of searching MAVs will not be modified.

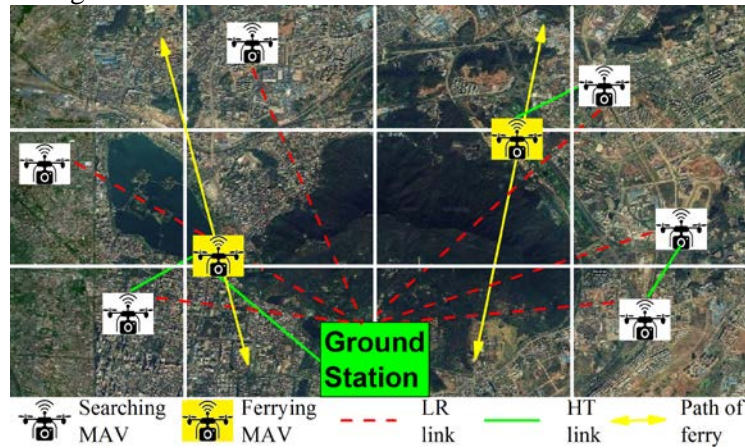


Fig. 1. A search and rescue scenario

In a search and rescue scenario as shown in **Fig. 1**, multiple MAVs are used to search for particular objects and monitor the given area with geo-tagged camera. Assuming that the MAV cluster is deployed at a certain altitude (such as 100m) in the air, they can move freely on the plane (i.e., there is no obstacle on the plane), each MAV (and ground station) knows its position and velocity information, and the ground station remains stationary and is typically deployed at the boundary of the search area. The area will be divided into many subareas, each of which is searched by an individual MAV. Ferrying MAVs are used to transmit the data generated by each searching MAV to the ground station via high throughput links. All the MAVs and the ground station are aware of their real-time position.

In **Fig. 1**, six searching MAVs transfer images to the ground station using high-throughput links (HT links) with limited ranges. Further, long-range links (LR links) are leveraged for control information. Two ferrying MAVs establish HT connectivity by flying back and forth between the searching MAVs and the ground station.

Suppose the search area is an $X \times Y$ rectangle, where X and Y are the length and width of the search areas respectively. If the actual search area is not a rectangle, the Minimum Bounding Rectangle (MBR) is used as the practical search area. Let D_{\max} be the maximum transmission range. Assume that the ground station G_0 is deployed at the midpoint of the horizontal side. There are M ferrying MAVs, denoted as F_i , $i \in \{1, 2, \dots, M\}$; N searching MAVs, denoted as S_j , $j \in \{1, 2, \dots, N\}$. Each MAV is equipped with a GPS+inertial measurement unit to obtain its real-time position and velocity information and broadcast the information periodically. Each MAV saves the position and velocity information of all MAVs in the topological table. The main content of the topological table items is $\{ID_i, P_i, V_i, T_i, K_i\}$. Where ID_i represents the identity of the MAV, $P_i = (px_i, py_i)$ represents the location of the MAV, $V_i = (vx_i, vy_i)$ represents the velocity of the MAV, and T_i is the information update time, $K_i \in \{G, F, S\}$ indicates the type of MAV (G is the ground station, F is the ferrying MAV, and S is the searching MAV). Suppose that all MAVs are at the same altitude, so this paper neglects the height of MAVs and just assumes all MAVs to be flying in a 2-dimensional plane.

In search and rescue scenarios, task assignment and route planning of the MAVs are usually performed in advance to save energy and improve the efficiency. Each searching MAV is responsible for a sub-area and uses a certain strategy to search (such as carpet search, etc.). Ferrying MAVs transfer data between the searching MAVs and the ground station, and the ferrying path is planned in advance too. Therefore, the MAV position can be estimated from the motion information.

4. Delay Tolerant Packet Forwarding Algorithms Based on Location Estimation: DTN_{est}

In practical applications, a MAV can sense its own location and broadcast periodically, and then it can know the location of other MAVs. Hence, the location of other MAVs is usually out of date and inaccurate, which will result in greatly reduced packet delivery ratio.

4.1 Location Estimation Mechanism

Each MAV broadcasts its location and velocity periodically. Once receiving the broadcasted information from the neighbors, a MAV updates its topological table. The MAVs use location information and greedy strategy to select the next hop. Then the next hop will be selected greedily using the topological table. However, the location received is out of date, and the MAV has moved to other location at current, which may lead to routing failure and ping-pong problems.

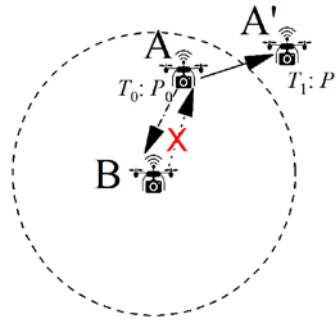


Fig. 2. Packet loss caused by MAV leaving communication range after location update

As shown in **Fig. 2**, MAV *A* sends its location P_0 to MAV *B* at time T_0 . Afterwards, MAV *B* can send data to MAV *A* at time T_1 ($T_1 > T_0$), because MAV *A* is the neighbor of *B* according to the topological table maintained at MAV *B*. However, MAV *A* has been out the communication range of *B* at time T_1 , and then the packet sent from *A* to *B* will be lost.

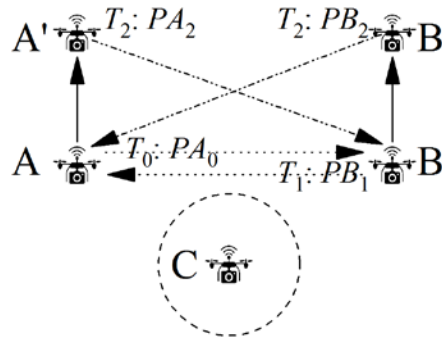


Fig. 3. Ping-pong event caused by outdated location information

As shown in **Fig. 3**, MAVs *A* and *B* are out of the communication range of destination MAV *C*. At time T_0 , MAV *A* sends its location PA_0 to MAV *B*. At time T_1 , MAV *B* sends its location PB_1 to MAV *A*. When sending data to destination *C* at Time T_2 , MAV *A* will select *B* as next hop because *B* is closer to *C* than *A* according to the topological table on MAV *A*. Similarly, MAV *A* will be selected as next hop for MAV *B*. The data will be repeatedly transmitted between *A* and *B*, which is called ping-pong effect. However, the locations of MAVs *A* and *B* at Time T_2 have been changed to PA_2 and PB_2 . Suppose PB_2 is closer to destination *C*, then *A* will send data to *B*, but *B* will not send data to *A*. Hence, the ping-pong problem is solved. Note that this paper does not discuss about whether

MAV C is mobile or static, as MAV C is keeping out of the communication range of MAVs A and B.

The main reason for the above problems is inaccurate location information of each MAV. One could shorten the updated interval to obtain more accurate location, but too small interval will lead too many updated packets in the network and increases the transmission time. Location estimation is an alternative solution to obtain approximately accurate location of each MAV.

Suppose the location of each MAV is in Euclidean space. Given the location (px_u, py_u) and velocity (vx_u, vy_u) of a MAV at update time T_u . The current time is denoted as T_c . Using linear extrapolation, the estimated location of the MAV at current with estimation time interval T_e ($T_e = T_c - T_u$) can be calculated:

$$(px_e, py_e) = (px_u + vx_u \times T_e, py_u + vy_u \times T_e) \quad (1)$$

Suppose the real position of the MAV at current time is (px_r, py_r) , the prediction error of MAV position is calculated as follows:

$$P_{error} = \sqrt{(px_r - px_e)^2 + (py_r - py_e)^2} \quad (2)$$

Because MAVs (such as quadrotor MAVs) usually have powerful flight capabilities, they can quickly change their flight direction and speed in a very short time. To simplify the analysis, we ignore the time T_{change} required for the MAV to change the flight direction and speed (i.e. the MAV can change the direction and speed instantaneously). Let the maximum flight speed of the MAV be V_{max} , then the possible position of the MAV at the current time is on the circle with (px_u, py_u) as the center and $V_{max} \times T_e$ as the radius. Then we have:

$$(px_r - px_u)^2 + (py_r - py_u)^2 = (V_{max} \times T_e)^2 \quad (3)$$

When the real flight direction is exactly opposite to the velocity direction in the topological table (at an angle of 180°), the position prediction error is maximum. The maximum position prediction error is:

$$\max(P_{error}) < (\sqrt{vx_u^2 + vy_u^2} + V_{max}) \times T_e \quad (4)$$

Note that $\max(P_{error})$ is an unreachable upper limit due to the existence of T_{change} . In the worst case, the position prediction error will increase with the estimation time gets longer.

It is easy to know that if the real flight velocity is same with the velocity in the topological table, then the minimum position prediction error is:

$$\min(P_{error}) = 0 \quad (5)$$

4.2 Motion Driven Packet Forwarding Mechanism

MAVs in this paper are divided into two types, searching MAVs and ferrying MAVs. Each searching MAV is only responsible for performing tasks within a certain sub-area, and the topology between the searching MAVs corresponds to the topology of the sub-areas, so the topology remains stable. While the ferrying MAVs are used to transmit data between MAVs and a ground station, which will cause great changes in the network topology. The path of the ferrying MAVs is planned in advance and usually moves back and forth between the source and the destination, hence one can predict the location of each ferrying MAV at every time.

As shown in Fig. 4, the ferrying MAV is out of the communication range of destination MAV C and is moving away from the destination. At time T_0 , the ferrying MAV was passing P_0 . When sending data to destination C at time T_0 , MAV A will select the ferrying MAV as next hop because the ferrying MAV is closer to C than A . Nevertheless, the ferrying MAV was out of the communication range of destination MAV C , so it carried the packet. Afterwards, the ferrying MAV moves to a new position P_1 at time T_1 . Then the ferrying MAV sends packets back to MAV A , causing the ping-pong problem. Note that there may be a large number of packets transfer during the ferrying MAV moving.

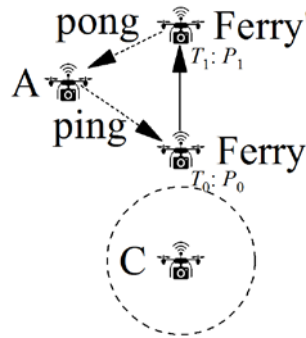


Fig. 4. The ping-pong problem caused by ferrying MAVs

The solution to the problem is that MAV A carries the packet by itself if necessary. At time T_0 , MAV A found that its optimal neighbor node was a ferrying MAV and the ferrying MAV was not in the communication range of the destination. Predicting the motion of the ferrying MAV and finding that it is moving away from the destination, MAV A carries the packet rather than send to the ferrying MAV. This can alleviate the ping-pong problem and reduce the routing hops.

4.3 The DTN_{est} Algorithm Based on Location Estimation

This paper develops a location-aware packet forwarding algorithm DTN_{est} with DTN support. Each MAV maintains its location and velocity and a topological table, which will be broadcasted periodically. The topological table on a MAV contains information such as MAV ID, location, velocity, update time, and MAV type. After receiving the topological table from other MAVs, a MAV will update its own topological table. Then each MAV can obtain all MAVs' information, and usually the information is out of date.

The algorithm works well in both connected and intermittently connected networks. Data is carried by MAVs and forwarded in a greedy manner based on DTN mechanisms. When source MAV A forwards data to destination D , the neighbor of A closest to the destination is found. The closest neighbor is selected as the next hop if it is closer to the destination than MAV A (i.e. the optimal neighbor), otherwise, the data will be carried by A , and the MAV keeps trying to find the next hop.

Since the periodically updated topological table will cause inaccurate position, packet loss and ping-pong problems, this paper employs linear extrapolation model to estimate the current position of MAVs.

Ferrying MAVs' movement also causes ping-pong problem. As the movement of ferrying MAVs is planned in advance, one can predict the location of ferrying MAVs when forwarding

the packets in the network. If a ferrying MAV is the optimal neighbor of the source MAV to the destination, it will be selected as the next hop of the source MAV except if it is flying away from the destination and cannot find its next hop.

Algorithm 1 completely describes the processing process after receiving a packet in a MAV. Let the maximum communication range of MAVs be D_{\max} . The MAV generates or receives a packet, which contains data, destination ID, destination location and other information. Each MAV is aware of its own identity and location, and maintains a topological table to save other MAVs' information. Note that all MAV positions in the algorithm are estimated by Equation (1).

Algorithm 1. DTN_{est} Algorithm.

Input: transmitted image data (*packet*), destination node's ID (*dstID*), local node's ID (*localID*), destination node's position (*dstPos*), local node's position (*localPos*), topological table (*topoTable*= $\{ID_i, P_i, V_i, T_i, K_i\}, i \in \{1, 2, \dots, N\}$)

Output: the optimal neighbour node which the data forwards to (*nextHop*)

```

1: The local MAV receives packet from other MAV or creates a packet by itself
2: if dstID == localID then //the packet is sent to the current node
3:   receive(packet)
4:   return
5: end if
6: for  $i \in \{1, 2, \dots, N\}$  do
7:   if  $ID_i == dstID$  then
8:     estPos = Estimation ( $P_i, V_i, T_i$ ) //estimate MAV's position using Equation (1)
9:     if CalculateDistance (estPos, localPos) <  $D_{\max}$  then
10:      nextHop =  $ID_i$ 
11:      return
12:    end if
13:  end if
14: end for
15:  $i = \text{FindBestNeighbour}(\text{localID}, \text{localPos}, \text{dstID}, \text{dstPos})$  //see Algorithm 2
16: if  $i \neq \text{NULL}$  then
17:    $\{bestID, Pos, Vel, Time, Kind\} = \{ID_i, P_i, V_i, T_i, K_i\}$ 
18:   if  $kind == F$  then //Ferry MAV
19:     estPos = Estimation (Pos, Vel, Time)
20:     if CalculateDistance (estPos, dstPos) >  $D_{\max}$  and CalculateDistance (estPos, dstPos) >
CalculateDistance (Pos, dstPos) then
21:       if  $\text{FindBestNeighbour}(bestID, Pos, dstID, dstPos) == \text{NULL}$  then
22:         nextHop = NULL
23:         return
24:       end if
25:     end if
26:     nextHop = bestID
27:   return
28: end if
29: else
30:   nextHop = NULL
31:   return
32: end if

```

When receiving a packet, the MAV checks whether it is the destination of the packet. If it is the destination, the forwarding is completed (as shown in steps 1-5). Else, the MAV checks whether the destination is its own neighbor node. If the destination is the neighbor, the packet is forwarded to the destination, and the algorithm ends (as shown in steps 6-14). Otherwise, the MAV that is the optimal neighbor i is found by *FindBestNeighbour* function (step 15). The next hop is determined according to the results of the function *FindBestNeighbour* that is detailed in Algorithm 2.

(1) If i is not **NULL**, the algorithm checks MAV i . If MAV i is a ferrying MAV, not in the communication range of destination and is moving away from the destination, then algorithm 2 is called again to find the optimal neighbor of MAV i (steps 16-21).

(2) If the optimal neighbor does not exist, the packet is carried by the MAV and the algorithm ends. Otherwise, the packet is forwarded to i and the algorithm ends (steps 21-28).

(3) If i is **NULL**, the packet is carried by the MAV and the algorithm ends (steps 29-32).

Algorithm 2 describes the details of *FindBestNeighbour* function, which finds out the nearest optimal neighbor of the *refer* node based on the topological table. If the nearest neighbor is closer to the destination than the *refer* node, the index of the neighbor will be returned; otherwise, **NULL** will be returned.

Algorithm 2. FindBestNeighbour.

Input: destination node's ID ($dstID$), refer node's ID ($referID$), destination node's position ($dstPos$), refer node's position ($referPos$), topological table ($topoTable = \{ID_i, P_i, V_i, T_i, K_i\}, i \in \{1, 2, \dots, N\}$)

Output: the index of the best neighbour which the data forwards to

```

1:  $bestDistance = CalculateDistance(DstPos, referPos)$ 
2:  $index = \mathbf{NULL}$ 
3: for  $i \in \{1, 2, \dots, N\}$  do
4:    $estPos = Estimation(P_i, V_i, T_i)$  //estimate MAV's position using Equation (1)
5:   if  $CalculateDistance(referPos, estPos) < D_{max}$  and  $CalculateDistance(DstPos, estPos) < bestDistance$  then //  $ID_i$  is in the communication range of refer node and  $ID_i$  is closer to destination node
6:      $bestDistance = CalculateDistance(DstPos, estPos)$ 
7:      $index = i$ 
8:   end if
9: end for
10: return  $index$ 

```

The brief idea of Algorithm 2 is as follows. Initializes the current optimal distance as the distance between the *refer* node and destination, and the optimal neighbor as **NULL**. The topological table is traversed and the distance between the *refer* node's neighbor and the destination is calculated using the estimated position. If the distance is less than the current optimal distance, the optimal neighbor and the current optimal distance are updated.

5. Performance Evaluation

The simulation experiment is carried out under the conditions set in [25], using the state-of-the-art network simulator NS-3. The simulation scenarios are inspired by search and rescue missions. This paper uses a shared memory to emulate the LR link (see Fig. 1) for sharing MAVs' position and velocity information. MAVs connect to each other via ad-hoc WiFi 802.11n at 5GHz (HT link). The WiFi communication range is set to 200 meters. Each MAV generates 5 messages per second (56 Kbit/s) addressed to the ground station. The

MAVs' typical cruise speed is 5 m/s and each test is executed for about 8 minutes. A summary of parameters used in the simulation is given in **Table 1**.

The area in simulation is $800\text{m} \times 800\text{m}$. **Fig. 5** visualizes the placement of all MAVs and their trajectories. The scenarios are constructed by one ground station, four ferrying MAVs and nine searching MAVs. All MAVs are deployed at the same altitude of 100m. The ground station stands at the middle of the horizontal edge. The searching MAVs are placed out of communication range of the ground station. Each searching MAV can fly into the communication range of at least one ferrying MAV during the simulation time. Each searching MAV is assigned a $200\text{m} \times 200\text{m}$ region to search and collect data. The searching MAVs move along the zigzag trajectory to cover a region. The four ferrying MAVs fly back and forth along the dash line to transfer data to the ground station.

Table 1. Simulation parameters

Name	Value
Number of ground station	1
Number of MAVs	6-16
Experiment time	8 minutes
MAVs' cruise speed	5 m/s
Mobility model	Waypoint Model
Max communication range	200m
Packet size	1400 bytes
Message creation rate	5 packets/s
Simulation area	$800\text{m} \times 800\text{m}$

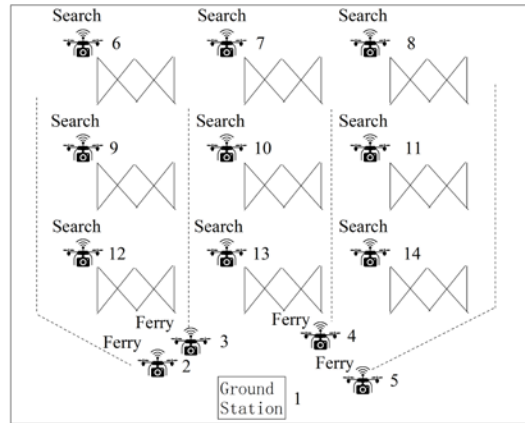


Fig. 5. Simulation scenario with one ground station, four ferrying MAVs and nine searching MAVs.

This paper studies the performance of the algorithm under different number of MAV networks, from 6 to 14 nodes. The serial numbers beside the nodes in **Fig. 5** represent different networks. For example, in a network with 9 nodes, only nodes with serial number 1-9 are arranged, while nodes with serial number 10-14 are not arranged. In other words, each network with n nodes consists of the ground station and the MAVs $\{MAV_2, \dots, MAV_n\}$.

The following metrics are used to evaluate the performance of the packet forwarding algorithms:

(1) *Delivery ratio*: The delivery ratio is defined as the fraction of messages that have been successfully delivered to the destination out of the messages that have been generated. This metric is a measure of the reliability of the packet forwarding algorithm.

(2) *Average delay*: The delay is calculated by the time when the packet arrives at the destination successfully minus the time when the packet was generated. The delay includes transmission time and carry time. The average delay is the mean delay of all packets that arrive at their destination successfully.

(3) *Average hop count*: The hop count is the number of hops a message passes until it reaches the destination. This metric shows the efficiency of a packet forwarding algorithm. The average hop count is the mean hops of all packets that arrive at their destination successfully.

The performance of DTN_{est} is evaluated in comparison with DTN_{geo} algorithm. The DTN_{geo} algorithm is a classical packet forwarding approach that represents the-state-of-the-art in DTN [25]. The main idea of DTN_{geo} algorithm is as follows: if the source MAV can establish an end-to-end shortest path to the destination, the message will be forwarded along the shortest path. Otherwise, the message is forwarded to the neighbor closer to the destination. If there is no neighbor closer to the destination, the message will be carried.

5.1 Performance of DTN_{est} and DTN_{geo} under different number of nodes

To investigate effects of different MAV densities and placements, this paper compares the performance of the two algorithms with varying number of nodes n (from 6 to 14). Each MAV generates 5 messages per second, and the update time interval is 1 second.

(1) *Delivery ratio*. Fig. 6 shows the delivery ratio of DTN_{est} and DTN_{geo} . Interestingly, in scenarios with 9 nodes or more, DTN_{est} outperforms DTN_{geo} and achieves a higher delivery ratio. In particular, when the number of network nodes exceeds 11, DTN_{est} 's delivery ratio is about 10% higher than DTN_{geo} . This is mainly because DTN_{est} uses estimated location, which greatly improves the accuracy of packet forwarding. Moreover, as shown by the figure, the delivery ratio generally increases for a better connected network, with larger number of nodes.

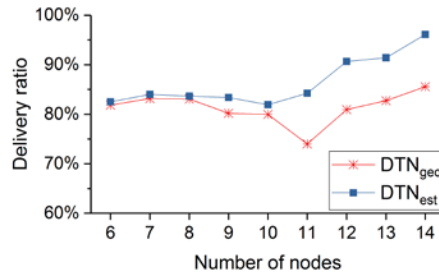


Fig. 6. Delivery ratio under different number of nodes

(2) *Average delay*. As visualized in Fig. 7, with the increased of the number of nodes, the average delay tends to decrease, which is due to the improvement of network connectivity and the decrease of connection interruption time. It is worth noting that DTN_{est} achieves an average delay result similar to DTN_{geo} , but with higher delivery ratio. In DTN_{est} , the location estimation is used, and the forwarding decisions of some packets are changed, such as selected other node as next hop or carried packet. However, the greedy forwarding strategy and the store-carry-forward strategy are not changed, so the transmission delay is hardly affected.

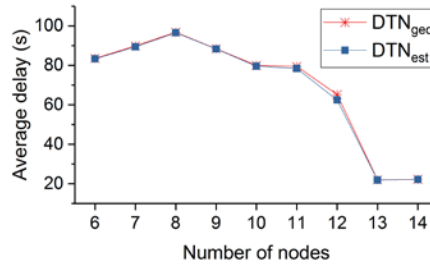


Fig. 7. Average delay under different number of nodes

(3) *Average hop count*. The measured average hop count is depicted in Fig. 8. In principle, the average hop count increases with increasing number of nodes, simply because many packets traverse more hops to reach the ground station from further distances. Due to the improved network connectivity, packets can be transmitted to the ground station much faster through searching MAVs, rather than relying solely on ferrying MAV to ferry. As expected, DTN_{est} achieves superior results, especially when the number of nodes exceed 7, the average hop count of DTN_{est} is about 8% lower than that of DTN_{geo}. The reason is that DTN_{est} uses location estimation, which reduces the occurrence of ping-pong problem and thus reduces the number of hops.

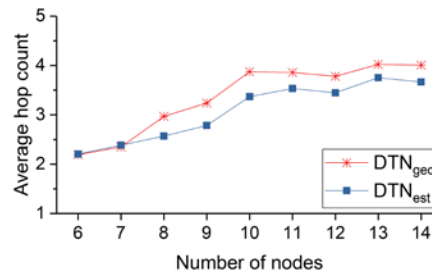


Fig. 8. Average hop count under different number of nodes

5.2 Performance of DTN_{est} and DTN_{geo} under different estimation time

As this paper employ a linear mobility model, the location estimation error would be increased with increasing the estimation time. The estimation time is determined by the update time interval. To investigate effects of different estimate time, this paper compare the performance of the two algorithms with the update time interval changing from 1 second to 10 seconds. Each MAV generates 5 messages per second, and the number of network nodes is 14.

(1) *Delivery ratio*. Fig. 9 confirms that the proposed DTN_{est} approaches higher delivery ratio compared with DTN_{geo} thanks to location estimation. DTN_{est} keeps a high delivery ratio (more than 90%) in a short update time interval (less than 4s), which is due to the small error of location estimation with a short estimated time. As the update time interval becomes longer, the error of location estimation becomes larger and both algorithms' delivery ratio decreases.

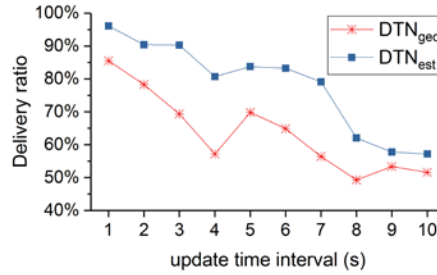


Fig. 9. Delivery ratio under different update time interval

(2) *Average delay.* **Fig. 10** plots the average delay of the two algorithms. As the update time interval gets longer, the location estimation error gets larger and then causes fluctuation of delay. However, the DTN_{est} is basically approximate to DTN_{geo} in average delay. The main reason is that the use of location estimation does not change the greedy forwarding strategy and the store-carry-forward strategy, which has no effect on the transmission delay.

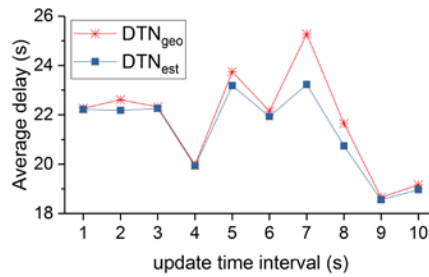


Fig. 10. Average delay under different update time interval

(3) *Average hop count.* **Fig. 11** compares the average hop count of the two algorithms. It is shown that the average hop count of both algorithms fluctuates, but DTN_{est} maintains a lower average hop count. When the update time interval is short (less than 4 seconds), the average hop count of DTN_{est} is stable, because the estimation information error with short estimation time is small. As the estimation time becomes longer, the error of the location estimation also becomes larger, and the average hop count experiences a slight fluctuation, but still maintains an excellent level.

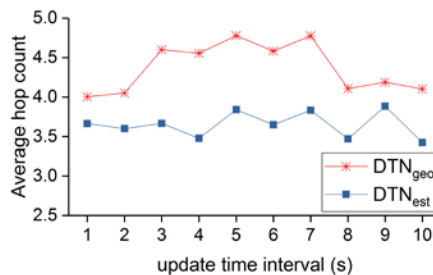


Fig. 11. Average hop count under different update time interval

5.3 Performance of DTN_{est} and DTN_{geo} under different motion of ferrying MAVs

In this subsection, four different motion of ferrying MAVs are designed to verify the performance of the two algorithms, as shown in **Fig. 12**. The ferrying MAVs move back and forth along the dotted line. The initial position and initial direction of motion are given in **Fig.**

12. Note that the ferrying MAVs motion used in other subsections is M1. In this subsection, each MAV generates 5 messages per second, the update time interval is 1 second, and the number of network nodes is 14.

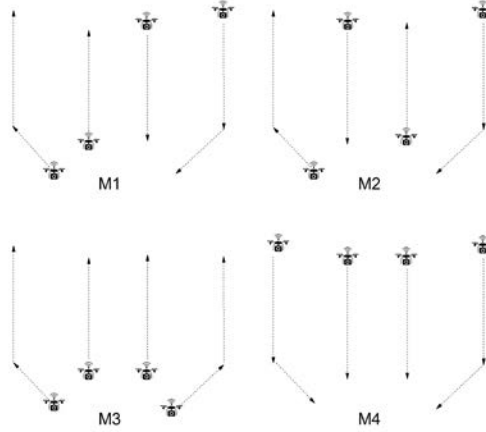


Fig. 12. Different motion of ferrying MAVs

(1) *Delivery ratio*. Fig. 13 evaluates the delivery ratio of the two algorithms with different ferrying MAVs motion. There are only four ferrying MAVs that can communicate directly with the ground station in the MAV network. The ferrying paths of the ferrying MAVs are extremely long, and the ferrying MAVs spend most of the time outside the communication range of the ground station. Therefore, during the ferrying MAVs' period, a large number of packets will be carried. When the ferrying MAVs fly into the ground station's communication range, the large number of carried packets can be delivered. The longer the total contact time between all ferrying MAVs and the ground station is, the higher the delivery ratio will be. Different motion will affect the total contact time between all ferrying MAVs and the ground station, thus affecting the delivery ratio. Still, DTN_{est} achieves a higher delivery ratio than DTN_{geo}, because location estimation reduces the packet loss.

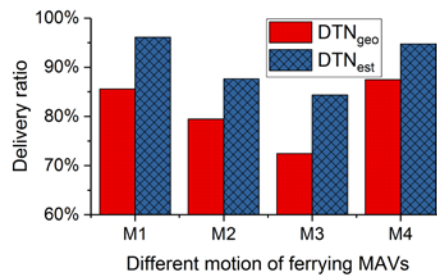


Fig. 13. Delivery ratio under different motion of ferrying MAVs

(2) *Average delay*. From Fig. 14, it can be observed that different ferrying MAVs' motion result in different average delays. This is because no searching MAVs are within the communication range of the ground station, and ferry data only by ferrying MAVs. Then different ferrying MAVs' motion will lead to different transmission delays. Nevertheless, the two algorithms are almost the same in average delay. The main reason is that the transmission delay is determined by the greedy forwarding strategy and the store-carry-forward strategy, and both algorithms use these two strategies for forwarding.

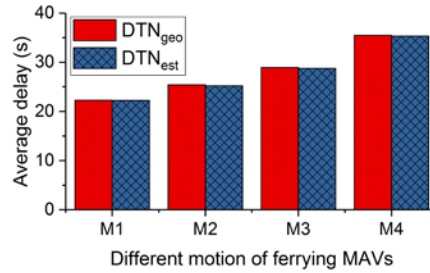


Fig. 14. Average delay under different motion of ferrying MAVs

(3) *Average hop count.* **Fig. 15** investigates the average hop count of the two algorithms. It reveals that different ferrying MAVs' motion lead to different average hop counts. Since the network topologies are different under different ferrying MAVs' motion, the packet forwarding paths are usually different and then routing hops vary. However, DTN_{est} has a lower average hop count than DTN_{geo}. This is mainly because DTN_{est} uses location estimation, which reduces the occurrence of ping-pong problem and thus reduces the number of routing hops.

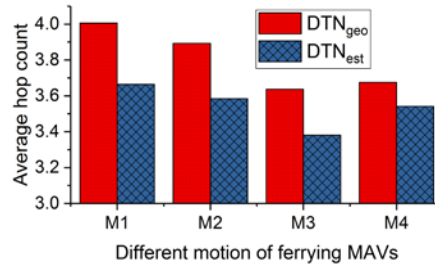


Fig. 15. Average hop count under different motion of ferrying MAVs

5.4 Performance of DTN_{est} and DTN_{geo} under different loads

This study investigated the two algorithms under different loads. Three cases are compared: every MAV generates 5 messages per second (base scenario, 56 Kbit/s per MAV), 10 messages per second (112 Kbit/s per MAV), and 20 messages per second (224 Kbit/s per MAV) respectively. The update time interval is 1 second. Note that the number of network nodes varies from 6 to 14.

(1) *Delivery ratio.* **Fig. 16** summarizes the delivery ratio of DTN_{est} and DTN_{geo}. DTN_{est} generally maintains a higher delivery ratio than DTN_{geo} because the location estimation strategy reduces the occurrence of packet loss. It can be observed that the delivery ratio of the two algorithms both decreases significantly with increasing load, from almost 91 percent (5 messages per second) to almost 77 percent (20 messages per second) in the worst performing scenario (12 nodes) of DTN_{est}, and from almost 80 percent (5 messages per second) to almost 56 percent (20 messages per second) in the worst performing scenario (10 nodes) of DTN_{geo}. As the load increases, the queue length will increase, and the wireless channel contention will intensify. Especially, there may be bottleneck nodes in the network (most of the data packets are forwarded by the node). Therefore, the packet loss problem can be severe, causing the delivery ratio to decrease.

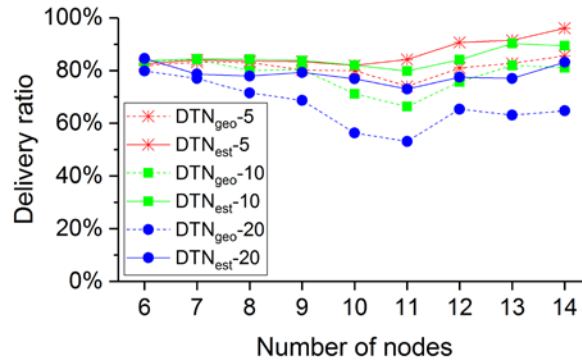


Fig. 16. Delivery ratio under different loads

It can be further observed that there is an interesting change of delivery ratio when the number of network nodes varies. When the number of nodes is less than 11, the delivery ratio decreases with the increase of the number of nodes, which is due to poor network connectivity and serious bottleneck problem. In contrast, when the number of nodes exceeds 11, the delivery ratio increases with the increase of nodes. This is because the network connectivity is improved, the number of alternative paths for packet delivery increases, and the bottleneck problem is alleviated.

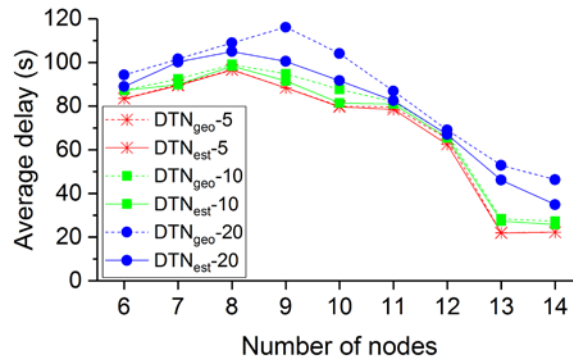


Fig. 17. Average delay under different loads

(2) *Average delay*. It can be concluded from the analysis of Fig. 17 that when the load increases, the average delay increases. That is because the queue length will increase as the load increases. When the load is low (5 messages per second), the average delay of DTN_{est} is similar to that of DTN_{geo}. It is worth noting that when the load increases (10 or 20 messages per second), the average delay of DTN_{est} is less than that of DTN_{geo}. The reason is that the location estimation mechanism increases the delivery ratio in DTN_{est}, additionally, more packets with shorter delay are successfully transferred to the destination in the heavy load scenario.

(3) *Average hop count*. One can see from Fig. 18 that when the load increases, the average hop count decreases. The reason is that packets with less hop counts are easier to deliver successfully. The average hop count of DTN_{est} is generally lower than DTN_{geo}. This is because the location estimation reduces the occurrence of the ping-pong problem and reduces the number of hops.

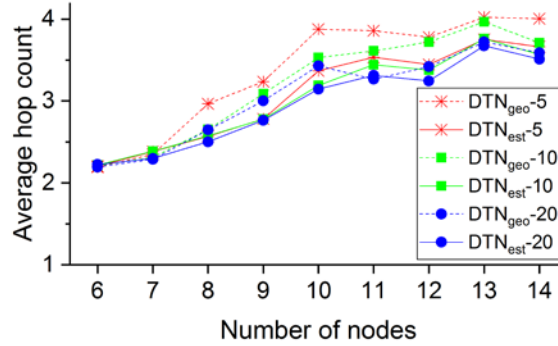


Fig. 18. Average hop count under different loads

5.5 Performance of DTN_{est} and DTN_{geo} under different schemes

In this section, we evaluate the effectiveness of the two algorithms, DTN_{est} and DTN_{geo}, under different relay schemes. To transmit message between disconnected subnets, this paper adopts ferrying MAVs that move back and forth in the MAV network (see Fig. 5). In addition, fixed relay MAVs that stay at a single position is also a common scheme. Fig. 19 visualizes an example of fixed relay MAVs schemes. Relays 2 and 3 maintain communication with the ground station as a gateway between the searching MAVs and the ground station. Relays 4 and 5 provide communication support between searching MAVs.

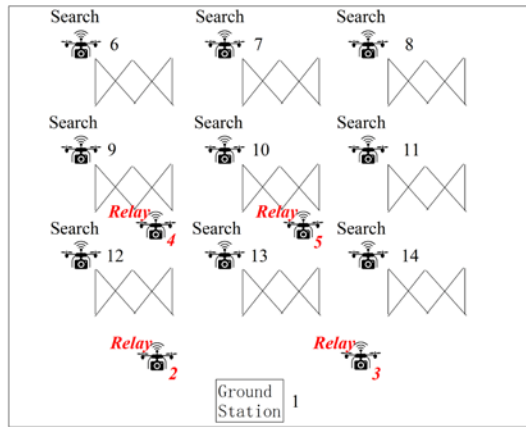


Fig. 19. The deployment diagram of fixed relay network scheme

To investigate effects of different schemes, this paper compares the performance of the two algorithms with varying number of nodes n (from 6 to 14). Each MAV generates 5 messages per second, and the update time interval is 1 second.

- (1) *Delivery ratio*. As shown in Fig. 20, when the number of nodes is less than 11 in fixed relay MAVs scheme, all the searching MAVs cannot communicate with the relays 2 and 3, and then lose contact with the ground station. The ground station only receives the messages from relays 2 and 3, so the delivery ratio is meaningless. However, the messages can be delivered to the ground station due to the movement of ferrying MAVs and delay tolerance mechanism. This show that for sparse networks, the ferrying MAVs scheme can achieve better message delivery ratio than the fixed relay MAVs scheme under the premise of intermittent connectivity.

When the number of nodes exceeds 12, the delivery ratio of the fixed relay MAVs scheme increases rapidly, even outperform the ferrying MAVs scheme. This is mainly due to the improved network connectivity and the fact that the fixed relay MAVs scheme does not have periodic motion problems like ferrying MAVs scheme (see **Section 5.3**). In the network with the fixed relay MAVs scheme, the delivery ratio of DTN_{est} is still higher than DTN_{geo} , indicating the effectiveness of DTN_{est} .

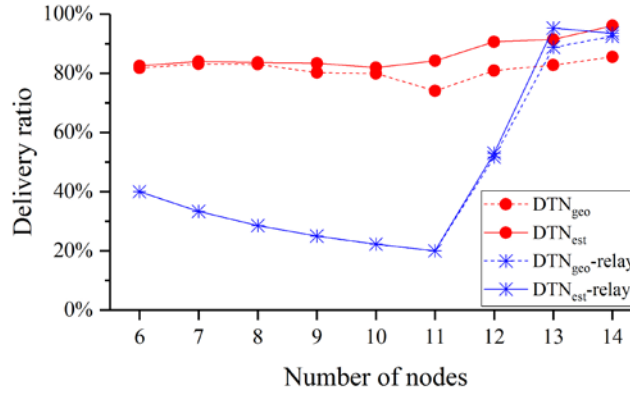


Fig. 20. Delivery ratio under different schemes

(2) *Average delay.* It can be concluded from the analysis of **Fig. 21** that when the number of nodes is less than 11 in relay MAVs scheme, the average delay is meaningless because only relays 2 and 3 communicate with the ground station. In contrast, using the ferrying MAVs scheme, although the average delay is large (more than 80s), the messages can be successfully delivered to the ground station.

It can be further observed when the number of nodes exceeds 12, the average delay of the fixed relay MAVs scheme is maintained at a good level (less than 10s). While using ferrying MAVs scheme, the average delay decreases rapidly. However, the average delay of the ferrying MAVs scheme is worse (more than 20s) than that of fixed relay MAVs scheme due to periodic motion. Furthermore, it can be seen that the average delay of DTN_{est} is almost the same as DTN_{geo} in the fixed relay MAVs scheme.

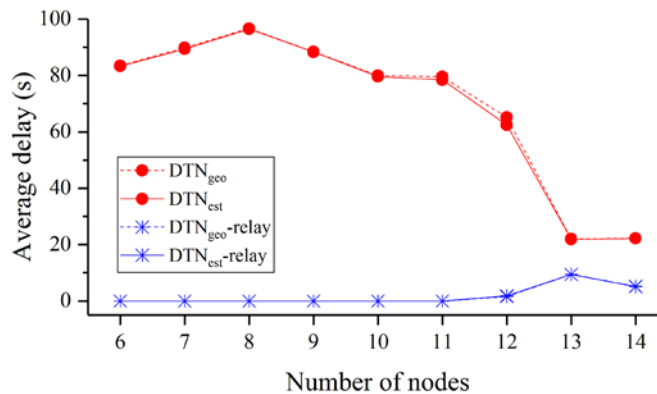


Fig. 21. Average delay under different schemes

(3) *Average hop count.* One can see from **Fig. 22** that in the fixed relay MAVs scheme, when the number of nodes is less than 11, since only relays 2 and 3 communicate with the

ground station, the average hop count is always 1, which is meaningless. While using the ferrying MAVs scheme, messages can be successfully transmitted to the ground station with 2-4 hops of forwarding.

When the number of nodes is greater than 12, the average hop count of the fixed relay MAVs scheme remains less than 3, while it is more than 3.5 in the ferrying MAVs scheme. This is due to the ping-pong problem caused by ferrying MAVs, resulting in an increase in the number of hops in message delivery. The average hop count of DTN_{est} is almost the same as DTN_{geo} in the fixed relay MAVs scheme. While in the ferrying MAVs scheme, the average hop count of DTN_{est} is less than DTN_{geo} because it reduces the occurrence of the ping-pong problem.

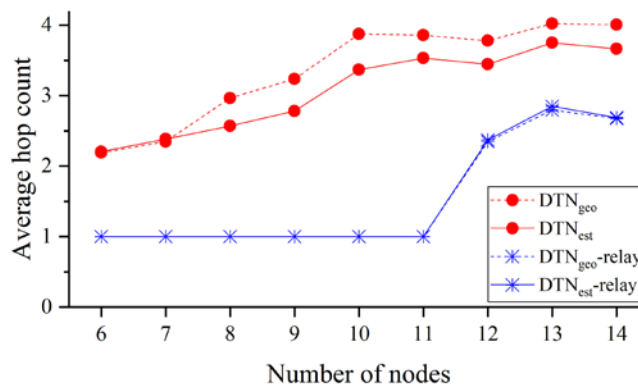


Fig. 22. Average hop count under different schemes

The fixed relay MAVs scheme cannot deal well with the interruption of network in a sparse network, which will result in the failure of message delivery. However, the ferrying MAVs scheme works well with intermittent network. Even in poor connectivity condition, the ferrying MAVs scheme can obtain a good delivery ratio with a large average delay.

In the case of good network connectivity, the delivery ratio of fixed relay MAVs scheme is no worse than that of ferrying MAVs scheme. Nevertheless, the average delay and average hop count of ferrying MAVs scheme are worse than that of fixed relay MAVs scheme. So in the case of a large sufficient number of nodes, the fixed relay MAVs scheme can be considered to improve network performance. Of course, the motion mode of the ferrying MAVs scheme can also be further studied to improve performance of message delivery.

In general, DTN_{est} achieves higher delivery ratio than DTN_{geo} in both fixed relay MAVs scheme and ferrying MAVs scheme. Besides, the average hop count of DTN_{est} is generally superior to DTN_{geo} , and the average delay of DTN_{est} is almost remaining the same as DTN_{geo} .

6. Conclusion

Using the motion information of MAVs, this paper has proposed the DTN_{est} algorithm, which is a location-aware packet forwarding approach with DTN support. In DTN_{est} , by employing location estimation mechanism and motion driven forwarding mechanism, the packet loss has been reduced and the ping-pong problem has been alleviated. The experimental results show that the proposed DTN_{est} algorithm outperforms the typical DTN_{geo} algorithm that represents the state-of-the-art in MAV networks in terms of packet delivery ratio and average routing hops.

This paper used the linear extrapolation method to estimate MAVs' position, which can obtain small location estimation error when the estimation time is short. However, the location estimation error is large when the estimation time is long and can cause the delivery ratio to decrease. Moreover, the average delay is large due to the periodic motion of ferrying MAVs. We will improve the accuracy of the location estimation methods and plan the ferrying MAVs' path reasonably in the future.

References

- [1] M. Asadpour, B. Van den Bergh, D. Giustiniano, K. Hummel, S. Pollin, and B. Plattner, "Micro aerial vehicle networks: An experimental analysis of challenges and opportunities," *IEEE Commun. Mag.*, vol. 52, no. 7, pp. 141–149, Jul. 2014. [Article \(CrossRef Link\)](#).
- [2] T. Andre et al., "Application-driven design of aerial communication networks," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 129–137, May 2014. [Article \(CrossRef Link\)](#).
- [3] Y. Zhou, N. Cheng, N. Lu, and X. S. Shen, "Multi-UAV-aided networks: Aerial-ground cooperative vehicular networking architecture," *IEEE Veh. Technol. Mag.*, vol. 10, no. 4, pp. 36–44, Dec. 2015. [Article \(CrossRef Link\)](#).
- [4] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1123–1152, Nov. 2016. [Article \(CrossRef Link\)](#).
- [5] H.-S. Ahn and C.-H. Won, "DGPS/IMU integration-based geolocation system: Airborne experimental test results," *Aerosp. Sci. Technol.*, vol. 13, no. 6, pp. 316–324, 2009. [Article \(CrossRef Link\)](#).
- [6] A. K.-S. Wong, T. K. Woo, A. T.-L. Lee, X. Xiao, V. Luk, and K. W. Cheng, "An AGPS-based elderly tracking system," in *Proc. of 1st Int. Conf. Ubiquitous Future Netw.*, pp. 100–105, Jun. 2009. [Article \(CrossRef Link\)](#).
- [7] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Unmanned aerial vehicle with underlaid device-to-device communications: Performance and tradeoffs," *IEEE Trans. Wireless Commun.*, vol. 15, no. 6, pp. 3949–3963, Jun. 2016. [Article \(CrossRef Link\)](#).
- [8] R. Shirani, M. St-Hilaire, T. Kunz, Y. Zhou, J. Li, and L. Lamont, "The performance of greedy geographic forwarding in unmanned aeronautical ad-hoc networks," in *Proc. of 9th IEEE Annu. Commun. Netw. Services Res. Conf. (CNSR)*, pp. 161–166, May 2011. [Article \(CrossRef Link\)](#).
- [9] C. Raffelsberger and H. Hellwagner, "A hybrid MANET-DTN routing scheme for emergency response scenarios," in *Proc. of IEEE Int. Conf. Pervasive Comput. Commun. Workshops*, pp. 505–510, Mar. 2013. [Article \(CrossRef Link\)](#).
- [10] H. Guo, X. Wang, H. Cheng, and M. Huang, "A location aided controlled spraying routing algorithm for delay tolerant networks," *Ad Hoc Netw.*, vol. 66, pp. 16–25, Nov. 2017. [Article \(CrossRef Link\)](#).
- [11] M. Harounabadi and A. Mitschele-Thiel, "Applying message forwarding and replication to multi-UAV message ferry networks," *Mobile Netw. Appl.*, vol. 23, pp. 1337–1346, Mar. 2018. [Article \(CrossRef Link\)](#).
- [12] A. Jimenez-Pacheco, D. Bouhired, Y. Gasser, J.-C. Zufferey, D. Floreano, and B. Rimoldi, "Implementation of a wireless mesh network of ultra light MAVs with dynamic routing," in *Proc. of IEEE Globecom Workshop*, pp. 1591–1596, Dec. 2012. [Article \(CrossRef Link\)](#).
- [13] M. Asadpour, S. Egli, K. A. Hummel, and D. Giustiniano, "Routing in a fleet of micro aerial vehicles: First experimental insights," in *Proc. of 3rd ACM MobiHoc Workshop Airborne Netw. Commun.*, pp. 9–10, 2014. [Article \(CrossRef Link\)](#).
- [14] F. Taqwa, A. Abdulkareem Ali, "Compressed fuzzy logic based multi-criteria AODV routing in VANET environment," *International Journal of Electrical and Computer Engineering (IJECE)*, 9(1), 397–401, Feb. 2019. [Article \(CrossRef Link\)](#).
- [15] M. Usha and B. Ramakrishnan, "An Enhanced MPR OLSR Protocol for Efficient Node Selection Process in Cognitive Radio Based VANET," *Wireless Personal Communications*, 106, 763–787,

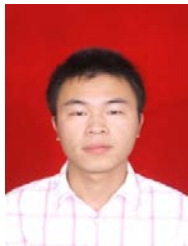
2019. [Article \(CrossRef Link\)](#).
- [16] P. Kuppasamy, K. Thirunavukkarasu and B. Kalaavathi, "A study and comparison of OLSR, AODV and TORA routing protocols in ad hoc networks," in *Proc. of 2011 3rd International Conference on Electronics Computer Technology, Kanyakumari*, pp. 143-147, 2011. [Article \(CrossRef Link\)](#).
 - [17] V. Sharma, B. Alam, and M.N. Doja, "An Improvement in DSR Routing Protocol of MANETs Using ANFIS," *Applications of Artificial Intelligence Techniques in Engineering*. Springer, Singapore, 569-576, 2018. [Article \(CrossRef Link\)](#).
 - [18] F. Kevin and S. Farrell, "DTN: an architectural retrospective," *IEEE Journal on Selected Areas in Communications*, 26(5), 828-836, 2008. [Article \(CrossRef Link\)](#).
 - [19] A. Vahdat and D. Becker, "Epidemic routing for partially-connected ad hoc networks," Dept. Comput. Sci., Duke Univ., Durham, NC, USA, Tech. Rep. CS-200006, 2000. [Article \(CrossRef Link\)](#).
 - [20] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *Proc. of ACM SIGCOMM Workshop Delay-Tolerant Netw. (WDTN)*, pp. 252–259, 2005. [Article \(CrossRef Link\)](#).
 - [21] A. Balasubramanian, B. N. Levine, and A. Venkataramani, "Replication routing in DTNs: A resource allocation approach," *IEEE/ACM Trans. Netw.*, vol. 18, no. 2, pp. 596–609, Apr. 2010. [Article \(CrossRef Link\)](#).
 - [22] M. Harounabadi, A. Puschmann, O. Artemenko, and A. Mitschele-Thiel, "TAG: Trajectory aware geographical routing in cognitive radio ad hoc networks with UAV nodes," *Ad Hoc Networks*. Cham, Switzerland: Springer, pp. 111–122, 2015. [Article \(CrossRef Link\)](#).
 - [23] P.-C. Cheng, K. C. Lee, M. Gerla, and J. Härri, "GeoDTN+Nav: Geographic DTN routing with navigator prediction for urban vehicular environments," *Mobile Netw. Appl.*, vol. 15, no. 1, pp. 61–82, 2010. [Article \(CrossRef Link\)](#).
 - [24] M. Y. Arafat and S. Moh, "Location-Aided Delay Tolerant Routing Protocol in UAV Networks for Post-Disaster Operation," *IEEE Access*, vol. 6, pp.59891-59906, 2018. [Article \(CrossRef Link\)](#).
 - [25] M. Asadpour, K. A. Hummel, D. Giustiniano, and S. Draskovic, "Route or carry: Motion-driven packet forwarding in micro aerial vehicle networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 3, pp. 843–856, Mar. 2017. [Article \(CrossRef Link\)](#).



Shiji Li received the B.S. and M.S. degrees from the Nanjing University of Aeronautics and Astronautics, China, in 2009 and 2014, respectively. He is currently pursuing the Ph.D. degree in computer science and technology with Command & Control College, Army Engineering University, China. His research interests include ad hoc networks, unmanned aerial networks, and autonomous cooperative control for multiple unmanned aerial vehicles.



Guyu Hu received the B.S. degree in Radio Communication from Zhejiang University in 1983 (Hangzhou, China), M.S. degree in Computer Application Technology from Nanjing Institute of Communications in 1989 (Nanjing, China), and Ph.D. degree in Communications and Information Systems from Nanjing Institute of Communications in 1992. Since 1990, he devotes to the research on network management. From 1997, he has been a full professor in the PLA University of Science and Technology, China. From 1998, his research interests include intelligent of network management, mainly on failure-finding from data with pattern recognition, machine learning and neural networks.



Youwei Ding is currently a lecturer in College of Information Technology, Nanjing University of Chinese Medicine. His research interests include energy efficient data management and big data mining. He received his Ph.D. degree in computer science from Nanjing University of Aeronautics and Astronautics in 2016.



Yun Zhou is currently a senior engineer. His research interests include network management and network security. He participated in a number of computer network construction projects, computer network management system development projects, and business center network security system construction.