Stochastic Packet Forwarding Algorithm in Flying Ad Hoc Networks

Cong Pu

***Abstract*—The drones, also officially referred to as unmanned airborne vehicles (UAVs), have captured the attention of hobbyists, researchers, and investors, and are becoming increasingly popular for various commercial, industrial, and public-safety applications. As an essential ingredient of Internet-of-Drones, Flying Ad Hoc Networks (FANETs) largely consisting of various drones are expeditiously proliferating and playing an important role in realizing the goal of coordinating the access of drones to controlled airspace and providing navigation services. However, packet forwarding in FANETs is challenged by the unique characteristics of FANETs, such as unstable wireless medium and intermittent connectivity caused by high mobility of drones. In this paper, we propose a stochastic packet forwarding algorithm, also called *SPA*, to provide efficient and reliable data transmission in FANETs. The basic idea of the *SPA* is to make a stochastic forwarding drone selection based on the combination of multiple real-time network metrics. By objectively allocating the weight to multiple real-time network metrics based on the entropy weight theory, the *SPA* computes the forwarding availability of each forwarding candidate drone. Then, the forwarding probability of each forwarding candidate drone is calculated, and the forwarding drone is stochastically chosen from all forwarding candidate drones based on the calculated forwarding probability. In experimental performance evaluation, we select link throughput and link expiration time as real-time network metrics, and evaluate the proposed stochastic packet forwarding algorithm through extensive simulation experiments using OMNeT++ and compare its performance with a prior motion-driven packet forwarding algorithm. Simulation results show that the *SPA* can improve the number of delivered packets as well as the average throughput, indicating a viable approach in FANETs.**

***Index Terms*—Unmanned Airborne Vehicles, Drones, Stochastic Packet Forwarding Algorithm, Flying Ad Hoc Networks, Internet-of-Drones**

I. Introduction

Thanks to the versatility, flexibility, easy installation, and relatively small operating expenses, the usage of unmanned airborne vehicles (UAVs), a.k.a. drones, has witnessed an extraordinary increase in a variety of civilian and military application areas, including aerial surveying and mapping, aerial surveillance and security, aerial inspection of infrastructure, and aerial delivery. In the United States, the drone logistics and transportation market was estimated to be valued at $11.20 billion in 2022 and is projected to reach $29.06 billion by 2027, at a combined annual growth rate of 21% [1]. Military business remains a major source of revenue for the drone industry, as demonstrated by the requested approximately $9.39 billion for unmanned systems and associated technologies in the Fiscal Year 2019 budget by the Department of Defense. As the proliferation of drone-based civilian and military applications, the development of Internet-of-Drones (IoD) and its applications are rapidly booming and developing, where a myriad of multi-sized and heterogeneous drones seamlessly interact with each other through zone service providers to realize the goal of coordinating the access of drones to controlled airspace and providing navigation services [2]. By leveraging edge and fog computing, and wireless communications and networking technologies, we envision that drones in the realm of IoD will find many new uses in enhancing our life further.

Endowed with the capabilities of sensing, computing, and wireless communicating, drones are qualified to collect, transmit, and deliver data. In addition, a fleet of drones may set up a Flying Ad Hoc Network (FANET) to establish connectivity in larger areas, and efficiently and collaboratively route data packets in order to achieve the goal of sharing information and knowledge and coordinating decisions. However, due to the unique characteristics of FANETs, such as unstable wireless link quality and intermittent connectivity caused by high mobility of drones, packet forwarding is challenged, and existing approaches in non-aerial multi-hop communication environments cannot be directly applied. Moreover, many routing protocols and communication algorithms specifically designed for Mobile Ad Hoc Networks (MANETs) and/or Vehicular Ad Hoc Networks (VANETs), such as traditional source, distance vector, or link state routing protocols, are not sensitive enough to adapt to the dynamics of FANETs in a timely manner due to a long convergence time and high routing overhead [3].

In this paper, we propose a stochastic packet forwarding algorithm, also called *SPA*, to provide efficient and reliable data transmission and communication in FANETs. The basic idea of the *SPA* is to make a stochastic forwarding drone selection based on the combination of multiple real-time network metrics in FANETs. The main contributions of this paper can be summarized as follows:

- In the *SPA*, we first objectively allocate the weight to multiple real-time network metrics based on the entropy weight theory and compute the forwarding availability of each forwarding candidate drone. And then, we calculate the forwarding probability of each forwarding candidate drone and stochastically choose the forwarding drone from all forwarding candidate drones based on the calculated forwarding probability.
- In the design of the *SPA*, multiple real-time network met-

Cong Pu (Email: puc@marshall.edu) is with the Weisberg Division of Computer Science, Marshall University, Huntington, WV 25755, USA.

rics are taken into consideration, thus, the *SPA* can detect and respond to link and connectivity changes nimbly and accurately. The *SPA* also provides good extensibility and flexibility, and additional network metrics, such as residual energy of forwarding drone and hop count of destination, can be easily added into the *SPA*.
- We select link throughput and link expiration time as real-time network metrics in experimental performance evaluation, and evaluate the *SPA* through extensive simulation experiments using OMNeT++ [4]. We also revisit a prior motion-driven packet forwarding algorithm [5], and implement and modify it to work in FANETs for performance comparison.

We develop a customized discrete event driven simulation framework by using OMNeT++ and evaluate its performance through extensive simulation experiments in terms of average link lifetime, number of delivered packets, and average throughput. The simulation results show that the proposed stochastic packet forwarding algorithm can not only improve the number of delivered packets, but also increase the average throughput, indicating a viable approach in FANETs.

The rest of the paper is organized as follows. Existing literature are discussed in Section II. A system model and the proposed stochastic packet forwarding algorithm are presented in Section III. In Section IV, simulation results are provided and analyzed. Finally, Section V concludes the paper with future research direction.

II. Related Work

In this section, we present and analyze a variety of routing protocols and communication algorithms in FANETs and similar environments.

In [5], a motion-driven packet forwarding algorithm is designed and implemented based on the link characterization in Micro Aerial Vehicle Networks. The algorithm unites location-aware end-to-end routing and delay-tolerant forwarding, extended by two heuristics that make use of anticipated future locations as well as estimated link capacity and connection time. In the end-to-end routing, the drone analyzes the network topology to find the shortest path to the destination of the message by implementing Dijkstra's algorithm. If such a path exists, the drone forwards the message to the neighbor that is a part of the shortest path. In case no end-to-end path is found, the drone forwards the message to the neighbor with the smallest virtual link weight or keeps the message in case the drone's own weight is equal or less than the weight of its neighbors. A smaller weight of a virtual link basically expresses a closer physical proximity to the destination.

In [6], a geolocation-based routing protocol is proposed for multihop network communication in FANETs, where the geolocation information of adjacent nodes is utilized to find the routing path to the destination node. Results from simulation experiments and field test have indicated that the proposed geolocation-based routing protocol can achieve low communication overheard and robustness to drastically changing network topology in FANETs. In [7], an adaptive hybrid communication protocol, which consists of a position-prediction-based directional MAC protocol (PPMAC) and a self-learning routing protocol based on reinforcement learning (RLSRP), is proposed to provide a comprehensive and high-performance communication scheme in FANETs. In the PPMAC, the exact position of each drone is predicted with directional antennas to overcome the directional deafness problem. In the RLSRP, all drones exchange their status information regularly with the FANET to update its stored data and implement the decision making on the routing path that has the shortest delivery delay.

In [8], a Software-Defined UAV Networking (SD-UAVNet) architecture is proposed to address the challenges of high dynamics, unstable aerial wireless links, and UAV collision probabilities in UAV network. In the SD-UAVNet architecture, based on the collected global UAV relevant context information, the centralized SDN UAV controller can optimize the UAVs' movements, determine the relay nodes deployment, select proper routing paths, and prevent UAVs from collisions, finally achieving the goal of satisfactory video quality. An aerial network management protocol built on top of a software defined networking (SDN) architecture is proposed to provide an efficient and robust end-to-end data relaying in [9], where each drone becomes SDN switch that performs under directives sent by a centralized controller. The [10] provides an overview of the state of the art of routing protocols for UAVs. The investigated routing protocols are compared in terms of multi-path capability, load balancing, loop-free ability, route update method, dynamic robustness, energy efficiency, and route metric. In [11], a comprehensive survey of position-based routing protocols for FANETs is provided along with a comparative study and the discussion of the advantages and weaknesses of each protocol.

In summary, most prior routing protocols and communication algorithms rely on end-to-end routing and delay-tolerant forwarding that introduce high communication delay as well as require a certain degree of link stability. Most predication based approaches only focus on certain network metrics, and can not be easily extended with additional network metrics. However, little attention has been paid to a stochastic packet forwarding algorithm endowed with good extensibility and flexibility in FANETs.

III. The Proposed Stochastic Packet Forwarding Algorithm

In this section, we first introduce the system model, and then propose a stochastic packet forwarding algorithm, also called *SPA*, to provide efficient and reliable data transmission and communication in FANETs.

A. System Model

In this paper, we consider a set of drones (later nodes) to form a FANET, where each node is uniquely identified by its node ID [12], and is able to obtain its current geographical position and mobility information through equipped Global Positioning System, Inertial Measurement Units, and digital map. Each node is also furnished with rechargeable batteries

TABLE I: Forwarding Candidate Table

Node ID	Link Throughput	Link Expiration Time	...	$Metrics^M$	Expiration Period
n_1	X_{11}	X_{12}	...	X_{1M}	t^1_{exp}
n_2	X_{21}	X_{22}	...	X_{2M}	t^2_{exp}
n_3	X_{31}	X_{32}	...	X_{3M}	t^3_{exp}
...	...	...	...	...	...
n_N	X_{N1}	X_{N2}	...	X_{NM}	t^N_{exp}

which can be recharged from environmental energy resources, such as wireless charging [13] or solar energy [14], thus, energy consumption is not a major problem and not considered in this paper. In most drone-based services and applications, drones like small quad-copters do not fly at high altitudes, therefore, we assume that drones have a low and constant altitude during the flight.

B. Stochastic Packet Forwarding Algorithm

The basic idea of the stochastic packet forwarding algorithm (*SPA*) is to make a stochastic forwarding node selection based on the combination of multiple real-time network metrics in FANETs. To be specific, the *SPA* first objectively allocates the weight to multiple real-time network metrics based on the entropy weight theory, and computes the forwarding availability of each forwarding candidate node. And then, the forwarding probability of each forwarding candidate node is calculated, and the forwarding node is stochastically chosen from all forwarding candidate nodes based on the calculated forwarding probability.

First, each node periodically broadcasts *HELLO* messages piggybacked with its own node ID, position coordinates, moving speed, and moving direction to neighbor nodes. *HELLO* messages are transmitted in the broadcast manner so that they are received or overheard by all one-hop neighbor nodes that are located within the communication range, but not relayed or broadcasted further. Through periodically exchanged *HELLO* messages, each node is aware of its one-hop neighbor nodes that will be considered as forwarding candidate nodes when it has data packets to send. In addition, these periodically exchanged *HELLO* messages permit each node to learn the network condition associated with each neighbor node, and then build its forwarding candidate table (*FT*). In the *FT*, each entry contains neighbor node ID (n_{id}), multiple real-time network metrics such as link throughput, link expiration time, etc., and candidate expiration time period (t_{exp}), respectively. Here, the candidate expiration time period t_{exp} is a system parameter, indicating that the node will remove the entry of neighbor node from the *FT* if it does not receive the *HELLO* message from this neighbor node before the t_{exp} expires. When the node receives the *HELLO* message from a neighbor node that is already in the *FT* before the t_{exp} expires, it updates the information of real-time network metrics associated with this neighbor node, and resets the t_{exp}. If the node receives a *HELLO* message from a new neighbor node, a new entry is added to the *FT*. Here, the format of the forwarding candidate table *FT* is shown in Table. I, where *M* number of network metrics are considered for *N* number of forwarding candidate (or neighbor) nodes.

$$\mathbb{R}^{N \times M} = \begin{pmatrix} X^*_{11} & X^*_{12} & X^*_{13} & \cdots & X^*_{1M} \\ X^*_{21} & X^*_{22} & X^*_{23} & \cdots & X^*_{2M} \\ X^*_{31} & X^*_{32} & X^*_{33} & \cdots & X^*_{3M} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ X^*_{N1} & X^*_{N2} & X^*_{N3} & \cdots & X^*_{NM} \end{pmatrix}$$

Fig. 1: The normalized matrix $\mathbb{R}^{N\times M}$.

Second, when a packet sender has data packets to send, it retrieves the information of the forwarding candidate table *FT* and removes the column of node ID n_{id} and candidate expiration time period t_{exp}. Then, the packet sender normalizes the value of each network metrics in the *FT* according to Eq. 1,

$$X^*_{ij} = \frac{max\{X_j\} - X_{ij}}{max\{X_j\} - min\{X_j\}} \times \alpha_j + (1 - \alpha_j), \quad i = 1, 2, \ldots, N, \quad j = 1, 2, \ldots, M, \tag{1}$$

and generates a normalized matrix of the value of network metrics $\mathbb{R}^{N\times M}$, as shown in Fig. 1. Here, α_j is the efficiency coefficient used to control the value range of the j^{th} network metrics, and $\sum_{j=1}^{M} \alpha_j$ = 1.0. The rational behind the design of α_j is to adjust the effect of the j^{th} network metrics for subjective preference. After that, the packet sender calculates the entropy of the j^{th} network metrics according to Eq. 2,

$$Ent_j = -\frac{1}{\ln N} \sum_{i=1}^{N} \Big(F_{ij} \cdot \ln F_{ij} \Big), \quad j = 1, 2, \ldots, M, \tag{2}$$

where F_{ij} is the proportion of the j^{th} network metrics associated with the forwarding candidate node n_i, and is represented as

$$F_{ij} = \frac{X^*_{ij}}{\sum_{k=1}^{N} X^*_{kj}}, \quad j = 1, 2, \ldots, M. \tag{3}$$

Based on the concept of entropy [15], the entropy weight of the j^{th} network metrics Υ_j can be defined as

$$\Upsilon_j = \frac{1 - Ent_j}{\sum_{k=1}^{M}(1 - Ent_k)}, \quad j = 1, 2, \ldots, M. \tag{4}$$

Thus, the forwarding availability of the forwarding candidate node n_i, denoted by Ava^{fwd}_i, can be calculated according to the normalized value of network metrics and the entropy weight of network metrics, which is represented as

$$Ava^{fwd}_i = \sum_{j=1}^{M} (\Upsilon_j \cdot X^*_{ij}), \quad i = 1, 2, \ldots, N. \tag{5}$$

Finally, the forwarding probability of the forwarding candidate node n_i, denoted by Pro^{fwd}_i, can be obtained from

$$Pro^{fwd}_i = 1 - \frac{Ava^{fwd}_i}{\sum_{k=1}^{N} Ava^{fwd}_k}, \quad i = 1, 2, \ldots, N. \tag{6}$$

Third, based on the calculated forwarding probability of each forwarding candidate node, the packet sender stochastically chooses the forwarding node, and then sends the data packets. In detail, the packet sender generates a random number (e.g., rand[0,1]) and compares it with the forwarding

Notations:
• *FT*, n_{id}, t_{exp}, $\mathbb{R}^{N \times M}$, *Ent*, Υ, Ava^{fwd}, and Pro^{fwd}: Defined before.
• $FT_i[j].metrics$: The information of real-time network metrics associated with node n_j stored at node n_i.
• $pkt[n_{id}, type]$: A packet containing a node ID (n_{id}) and packet type (*type*). Here, *type* can be either *Data* or *HELLO*.

◇ When a node n_i receives a *HELLO* packet $pkt[n_j, HELLO]$ from node n_j:
  if $n_j \in FT_i[n_{id}]$ /* n_j is the one-hop neighbor node of n_i */
    Update $FT_i[j].metrics$;
    Reset $FT_i[j].t_{exp}$;
  else /* n_j is the new neighbor node of n_i */
    Add a new entry of n_j into FT_i;
◇ When $FT_i[j].t_{exp}$ expires at node n_i:
  Remove the entry $FT_i[j]$;
◇ When a packet sender n_i has data packets to send:
  Retrieve FT_i; Remove the column of n_{id} and t_{exp} in FT_i;
  Normalize the value of metrics in FT_i according to Eq. 1;
  Generate $\mathbb{R}^{N \times M}$ as shown in Fig. 1;
  Calculate *Ent* of each metrics according to Eq. 2;
  Calculate Υ of each metrics according to Eq. 4;
  Calculate Ava^{fwd} of each candidate node according to Eq. 5;
  Calculate Pro^{fwd} of each candidate node according to Eq. 6;
  for $n_k \in FT_i[n_{id}]$ /* Stochastically select forwarding node */
    r = rand[0,1];
    if $Pro_k^{fwd} < r$
      continue;
    Select n_k as forwarding node;
    break;
  Send $pkt[i, Data]$ to n_k;

Fig. 2: The pseudo code of the proposed *SPA* algorithm.

probability of one forwarding candidate node. If the forwarding probability of this forwarding candidate node is larger than the randomly generated number, the packet sender selects it as the forwarding node, and then sends the data packets. Otherwise, the packet sender will generate a new random number and compare it with the forwarding probability of another forwarding candidate node, at which the same aforementioned operations will be applied to select the forwarding node. This selection process will continue until the packet sender successfully chooses the forwarding node whose forwarding probability is larger than the randomly generated number. In order to improve fault tolerance and network resiliency, more than one forwarding node can be chosen to send data packets, but we limit the number of forwarding node to one in this paper. Major operations of the *SPA* are summarized in Fig. 2.

For example, as shown in Fig. 3, suppose that the packet sender n_s has three forwarding candidate nodes (or one-hop neighbor nodes), n_a, n_b, and n_c, to select and send data packets. We consider link throughput and link expiration time as real-time network metrics to calculate the forwarding probability of each forwarding candidate node. Here, the modeling of link throughput and link expiration time is introduced in Subsection III-C. Thus, n_s first generates the normalized matrix $\mathbb{R}^{3\times2}$ according to Eq. 1.

$$\mathbb{R}^{3\times2} = \begin{bmatrix} 0.83 & 0.75 \\ 0.5 & 0.5 \\ 1.0 & 1.0 \end{bmatrix}$$

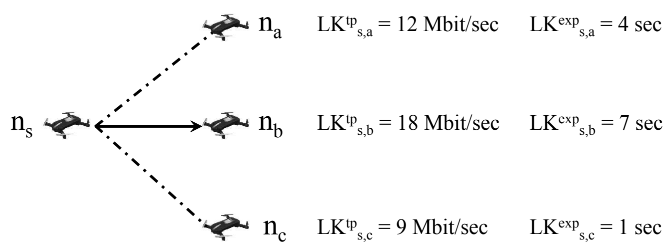


Fig. 3: A snapshot of the network, where the packet sender n_s has three forwarding candidate nodes, n_a, n_b, and n_c, to select and send data packets. Here, $LK_{s,i}^{tp}$ and $LK_{s,i}^{exp}$ is the link throughput and link expiration time between node n_s and node n_i, and solid line represents the forwarding of data packets.

After that, n_s calculates the entropy and the entropy weight of link throughput and link expiration time according to Eq. 2 and Eq. 4, respectively.

$$Ent_{LK^{tp}} = 0.80 \quad Ent_{LK^{exp}} = 0.77$$
$$\Upsilon_{LK^{tp}} = 0.47 \quad \Upsilon_{LK^{exp}} = 0.54$$

Thus, n_s calculates the forwarding availability and forwarding probability of each forwarding candidate node based on Eq. 5 and Eq. 6, respectively.

$$Ava_a^{fwd} = 0.79 \quad Ava_b^{fwd} = 0.50 \quad Ava_c^{fwd} = 1.0$$
$$Pro_a^{fwd} = 0.66 \quad Pro_b^{fwd} = 0.78 \quad Pro_c^{fwd} = 0.56$$

Finally, n_s generates a random number rand[0,1], compares it with the forwarding probability of n_a, n_b, and n_c, and selects the forwarding node. Note that the forwarding candidate node n_b provides the largest throughput (18 Mbit/sec) and longest link expiration time (7 sec) as shown in Fig. 3. If n_b was selected as forwarding node, more data packets could be forwarded from n_s before the link between n_s and n_b is broken. Thus, n_b has the largest forwarding probability ($Pro_b^{fwd} = 0.78$) to be selected as forwarding node, compared to that of n_a and n_c.

C. Modeling Link Expiration Time and Link Throughput

In this paper, since each node can obtain its geographical location and mobility information which are provided by Global Positioning System and Inertial Measurement Units, a new link expiration time (LET) prediction method is proposed to estimate the link expiration time between two nodes based on [16]. Suppose that two nodes n_i and n_j are within the communication range r of each other, and the two-dimensional position coordinates of n_i and n_j are denoted by (x_i, y_i) and (x_j, y_j), respectively. We also assume that v_i and v_j are the moving speeds, and θ_i and θ_j ($0 \leq \theta_i, \theta_j < 2\pi$) are the moving directions of n_i and n_j, respectively. Thus, the duration of time that n_i and n_j will stay connected, which is denoted by $LK_{i,j}^{exp}$, is represented by

$$LK_{i,j}^{exp} = \frac{-(a \cdot b + c \cdot d) + \sqrt{(a^2 + c^2) \cdot r^2 - (a \cdot d - b \cdot c)^2}}{a^2 + c^2}, \tag{7}$$

where

$$a = v_i \cdot \cos\theta_i - v_j \cdot \cos\theta_j, \quad (8)$$

$$b = x_i - x_j, \quad (9)$$

$$c = v_i \cdot \sin\theta_i - v_j \cdot \sin\theta_j, \quad (10)$$

$$d = y_i - y_j. \quad (11)$$

When $v_i = v_j$ and $\theta_i = \theta_j$, which indicates n_i and n_j moving at the same speed and direction, $LK_{i,j}^{exp}$ becomes ∞. Thus, the predicted connection time $LK_{i,j}^{exp}$ is the link expiration time between n_i and n_j.

In addition, the link throughput can be estimated as a function of the known geographical distance between the sending node and the receiving node. According to [5], [17], the derived empirical link throughput between node n_i and node n_j, denoted by $LK_{i,j}^{tp}$, is given as follows

$$LK_{i,j}^{tp} = 10^6 \cdot (-9.09 \cdot \log_2(dist(i,j)) + 72.58), \quad (12)$$

where $dist(i, j)$ is the spatial distance between n_i and n_j. Here, the unit of spatial distance $dist(i, j)$ and link throughput $LK_{i,j}^{tp}$ is meter and bit/second, respectively.

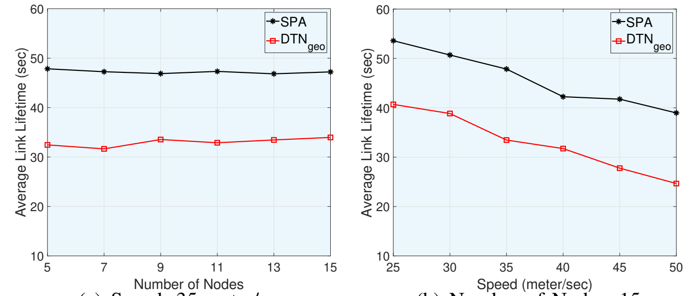
IV. Performance Evaluation

A. Simulation Testbed

We conduct extensive simulation experiments using OMNeT++ [4] for performance evaluation and analysis. 5 to 15 nodes are uniformly and initially distributed in a 1000×1000 (m^2) square network area. Nodes are equipped with IEEE 802.11p radio transceiver and communicate through two-way ground propagation channel. The communication range of each node is set to 300 meters. The random waypoint mobility model [18], [19] is deployed in the network, where each node travels toward a randomly selected destination in the network with a constant speed of 25 to 50 meter/sec and a zero pause time. The source node generates data traffic at the packet rate of 3.0 packet/sec and the size of each packet is 512 Bytes. The total simulation time is 500 seconds, and each simulation scenario is repeated 10 times with different randomly generated seeds to obtain steady state performance metrics. In this paper, we measure the performance in terms of average link lifetime, average number of delivered packets, and average throughput ratio by changing key simulation parameters, including number of nodes and moving speed. We also revisit a prior motion-driven packet forwarding algorithm [5], and implement and modify it to work in FANETs for performance comparison.

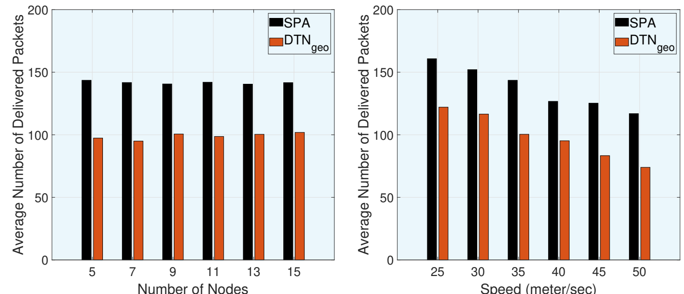
B. Simulation Results and Analysis

First, we observe the average link lifetime of selected links by varying the number of nodes and moving speed in Fig. 4. As shown in Subfig. 4(a), the overall average link lifetime of selected links of *SPA* and DTN_{geo} are not very sensitive to the changes of number of nodes in the network, where the speed is 35 meter/sec. However, the average link lifetime of *SPA* is still 15% larger than that of DTN_{geo}. This is because the *SPA* takes into account of link expiration time to select forwarding



(a) Speed: 35 meter/sec (b) Number of Nodes: 15

Fig. 4: The performance of average link lifetime of selected links against number of nodes and moving speed.

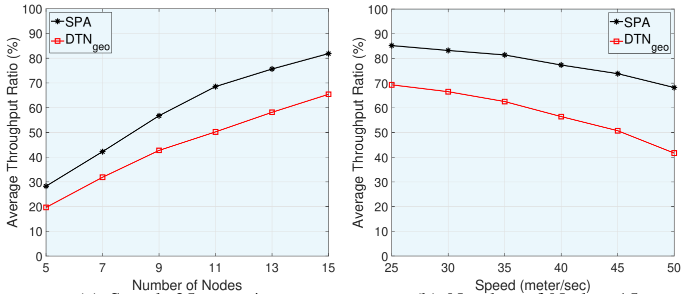


(a) Speed: 35 meter/sec (b) Number of Nodes: 15

Fig. 5: The performance of average number of delivered packets against number of nodes and moving speed.

node, and the candidate nodes with larger link expiration time have more chances to be selected. However, the DTN_{geo} analyzes the network topology to find the shortest path in terms of transmission delay to the destination of the message, and the forwarding nodes along the routing path may not provide the longest link connection time. In Subfig. 4(b), the overall average link lifetime of selected links of two schemes significantly decrease as the moving speed of nodes increases from 25 to 50 meter/sec. As the moving speed increases, the link between each two nodes becomes less stable, thus, the link expiration time decreases. The *SPA* still outperforms the DTN_{geo} because the forwarding candidate node with a larger link expiration time has larger probability to be selected to forward data packets in the *SPA*.

Second, in Fig. 5, the average number of delivered packets are observed with varying number of nodes and moving speed. As shown in Subfig. 5(a), more packets can be delivered by *SPA* than DTN_{geo}. In the *SPA*, when the forwarding candidate node has a larger link throughput as well as link expiration time, which means more data packets can be delivered before the link is broken, it has a larger probability to be selected as forwarding node. As a result, a larger number of data packets can be delivered to the next-hop node. In the DTN_{geo}, the source node only selects the candidate node who can provide the lower transmission delay to send data packets. However, the selected forwarding node may provide lower link connection time, and a smaller number of data packets can be delivered before the link is broken. In Subfig. 5(b), as the moving speed increases, the average number of delivered packets decreases linearly. Since the link is less stable and



(a) Speed: 35 meter/sec (b) Number of Nodes: 15

Fig. 6: The performance of average throughput ratio against number of nodes and moving speed.

becomes disconnected suddenly with the higher mobility, a smaller number of data packets can be delivered. However, the *SPA* still shows a better performance than the DTN_{geo}.

Third, we measure the average throughput ratio by changing the number of node and moving speed in Fig. 6. In Subfig. 6(a), it is shown that the average throughput ratio of *SPA* and DTN_{geo} significantly increases as the number of nodes in the network increases. When the node density increases, each node can have more forwarding candidate nodes to select, thus, more data packets can be forwarded and a larger throughput ratio is observed. Moreover, the *SPA* can achieve a higher throughput ratio than the DTN_{geo}, because the *SPA* selects more reliable and stable link in terms of link expiration time to forward data packets, and a larger number of data packets can be forwarded and a higher throughput ratio is observed. As the speed increases from 25 to 35 meter/sec, the average throughput ratio of both schemes decrease in Subfig. 6(b), where the number of node in the network is 15. This is because the node with higher mobility can move suddenly, and the link becomes less stable or even disconnected. As a result, a smaller number of data packets can be forwarded and the throughput ratio decreases accordingly. However, the observed throughput ratio of *SPA* is still larger than that of DTN_{geo}, because the forwarding candidate node with a more reliable and stable link will be selected with a big probability, which results in more data packets being delivered and a higher throughput ratio.

V. Conclusion and Future Work

In this paper, a stochastic packet forwarding algorithm is proposed to provide efficient and reliable data transmission and communication in FANETs, where each node objectively allocates the weight to multiple real-time network metrics based on the entropy weight theory, computes the forwarding probability of each forwarding candidate node, and then stochastically chooses the forwarding node based on the calculated forwarding probability. In experimental study, we select link throughput and link expiration time as real-time network metrics. We also develop a customized discrete event driven simulation framework by using OMNeT++ and evaluate its performance through extensive simulation experiments in terms of average link lifetime, average number of delivered packets, and average throughput ratio. The simulation results indicate that the proposed stochastic packet forwarding algorithm is a viable approach for efficient and reliable data transmission and communication in FANETs. As a future work, we plan to improve the proposed stochastic packet forwarding algorithm by considering additional real-time network metrics, such as traffic load and residual energy of forwarding candidate node, to achieve the goal of load balancing and extending the network lifetime. In terms of performance evaluation, since radio propagation and its channel dynamics cannot easily be captured by discrete event driven simulation model and framework, we plan to develop a small-scale testbed with small and safe quad-copters, and deploy a real outdoor environment to see the full potential of the proposed approach.

Acknowledgment

This research was supported by Startup grant in the Weisberg Division of Computer Science and 2019 Summer Research Award at Marshall University.

References

[1] *2019 UAV Report*, https://www.gpsworld.com/uav-report-growth-trends-opportunities-for-2019.
[2] A. Koubâa *et al.*, "Dronemap planner: A service-oriented cloud-based management system for the internet-of-drones," *Ad Hoc Networks*, vol. 86, pp. 46–62, 2019.
[3] M. Asadpour, S. Egli, K. Hummel, and D. Giustiniano, "Routing in a Fleet of Micro Aerial Vehicles: First Experimental Insights," in *Proc. ACM Airborne*, 2014, pp. 9–10.
[4] A. Varga, *OMNeT++*, 2014, http://www.omnetpp.org/.
[5] M. Asadpour *et al.*, "Route or Carry: Motion-Driven Packet Forwarding in Micro Aerial Vehicle Networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 3, pp. 843–856, 2017.
[6] S. Choi and Others, "Performance Evaluation of Scientific Research Program in Zhejiang Colleges Based on Uncertainty Analysis," in *Proc. IEEE ICUFN*, 2018, pp. 50–52.
[7] Z. Zheng *et al.*, "Adaptive Communication Protocols in Flying Ad Hoc Network," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 136–142, 2018.
[8] Z. Zhao *et al.*, "Software-defined unmanned aerial vehicles networking for video dissemination services," *Ad Hoc Networks*, vol. 83, pp. 68–77, 2019.
[9] G. Secinti, P. Darian, B. Canberk, and K. Chowdhury, "SDNs in the Sky: Robust End-to-End Connectivity for Aerial Vehicular Networks," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 16–21, 2018.
[10] J. Jiang and G. Han, "Routing Protocols for Unmanned Aerial Vehicles," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 58–63, 2018.
[11] O. Oubbati *et al.*, "A survey on position-based routing protocols for Flying Ad hoc Networks (FANETs)," *Elsevier Vehicular Communications*, vol. 10, pp. 29–56, 2017.
[12] C. Pu and S. Lim, "A Light-Weight Countermeasure to Forwarding Misbehavior in Wireless Sensor Networks: Design, Analysis, and Evaluation," *IEEE Systems Journal*, vol. 12, no. 1, pp. 834–842, 2018.
[13] T. Long, M. Ozger, O. Cetinkaya, and O. Akan, "Energy Neutral Internet of Drones," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 22–28, 2018.
[14] C. Pu, T. Gade, S. Lim, M. Min, and W. Wang, "Lightweight Forwarding Protocols in Energy Harvesting Wireless Sensor Networks," in *Proc. IEEE MILCOM*, 2014, pp. 1053–1059.
[15] C. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.
[16] W. Su *et al.*, "Mobility prediction and routing in ad hoc wireless networks," *Int. J. Network Mgmt*, vol. 11, no. 1, pp. 3–30, 2001.
[17] C. Pu, "Jamming-Resilient Multipath Routing Protocol for Flying Ad Hoc Networks," *IEEE Access*, vol. 6, pp. 68 472–68 486, 2018.
[18] H. Pucha, S. Das, and Y. Hu, "The Performance Impact of Traffic Patterns on Routing Protocols in Mobile Ad Hoc Networks," *Elsevier Computer Networks*, vol. 51, no. 12, pp. 3595–3616, 2007.
[19] C. Pu, "Link-Quality and Traffic-Load Aware Routing for UAV Ad Hoc Networks," in *Proc. IEEE CIC*, 2018, pp. 71–79.