

# An Adaptive And Low-Complexity Routing Protocol For Distributed Airborne Networks

Shivam Garg  
 San Diego State University  
 University of California, Irvine  
 sgarg@sdsu.edu

## Abstract

Airborne network (AN) is an essential component for the next generation air transportation system, which can self-configure to provide a seamless, low-cost and secure connectivity. The main characteristics of AN are high node density and fast mobility, which result in a dynamic topology with frequent link disruptions, high co-channel interference, and significant control and computational overheads. These challenges become more difficult to address for a decentralized, multi-hop AN, which does not have a supervisory node for centralized control.

In this paper, an adaptive, low-complexity, optimized link state routing (OLSR) protocol is presented to address the above-mentioned issues in a decentralized, multi-hop AN. In the proposed scheme, (i) a novel, multi-parametric route selection metric is introduced for optimal route selection, (ii) a proactive route switching mechanism is used, which prevents packet transmission over broken routes and reduces the computational overhead, and (iii) a dynamic control packet structure is used to minimize the control overhead. Simulation results show that our proposed scheme has a superior throughput performance and a much lower routing computational complexity as compared to the standard OLSR.

## 1 Introduction

The airborne network (AN) is an essential component for the next generation air transportation system, which can self-configure to provide a seamless, low-cost and secure connectivity in real-time to a multitude of devices with stringent and dynamic quality of service (QoS) requirements, whenever and wherever needed [1, 2]. Such ANs consist of multiple, cooperative, manned and unmanned aerial vehicles (UAVs), which use flight-to-flight communication to reduce latency in information sharing [2], and provide scalability [1]. Some existing applications of AN in civilian and military sectors are environmental sensing and disaster management, traffic and urban monitoring, patrolling, and relaying networks [1].

The main characteristics of AN are high node density and fast mobility, which result in a dynamic topology with high co-channel interference and frequent link disruptions [1]. This leads to packet transmission over a broken route, which increases congestion, and thereby, degrades the flow QoS. To address it, periodic topology change notifications are broadcasted in the network to select an alternative route, which increases the control and computational overheads,

and energy consumption, which can lead to the node failure [1]. This reduces channel utilization, flow reliability, and increases latency. These challenges become more difficult to address for a decentralized, multi-hop AN, which does not have a supervisory node for centralized control.

To address these issues, we propose an adaptive, low-complexity optimized link state routing (OLSR) protocol for a decentralized, multi-hop AN.

### 1.1 Related Work

Routing protocols are typically categorized into geographic, reactive and proactive protocols [1]. In geographic routing protocols (e.g., aeronautical routing protocol), location servers store the updated GPS locations of all nodes in a fixed geographical region, which a source node queries to obtain the recent location of its destination node. However, a separate routing scheme (other than geographic routing) is required for the location servers to maintain the updated location information. Furthermore, geographic routing protocols are not suited for a distributed network with no central or regional servers [1].

Reactive routing protocol (e.g., ad-hoc on-demand distance vector (AODV)) discovers routes on-demand for a source-destination pair. These schemes have a smaller signaling traffic but a large route discovery delay when the route to destination is not available [1]. Therefore, frequent link breaks in a highly mobile airborne network can significantly increase the route rediscovery delay.

In proactive routing protocol (e.g., OLSR [3]), control messages are periodically exchanged among the nodes in the network, which incurs a large signaling overhead in a dense network. However, each node keeps the updated information for all nodes in the network and can immediately find a route to a destination node when needed, without any route discovery delay. Hence, it is suitable for dynamic topology, where source-destination pairs change with time [1].

#### 1.1.1 Related work on stable and long-lasting route selection

The shortest-hop routes generally select the edge nodes. Consequently, the signal strength at the receiver node is minimized, which increases the packet loss ratio and reduces PDR (packet delivery ratio) of the flow. Therefore, the routing scheme proposed in [4] differentiates the links based on their RSSI (received signal strength indicator) values using Chebyshev inequality and prefers links with a lower variance in RSSI values for the route selection. Since interference

from neighbor nodes increases significantly in a dense network, which causes inaccurate computation of RSSI values, it is ill-suited for the robust route selection.

A mobility-aware route selection scheme is proposed in [5], where the link stability is determined by the variance in the distance values (computed using GPS) of a node pair. Node pairs with a smaller variance in distance values are expected to remain in the range of each other for a longer duration due to their similar mobility patterns. However, [5] fails to select links where the nodes come closer because of the high variance in their monotonically decreasing distance values.

An SVM (support vector machine)-based link failure prediction scheme is proposed in [6], which uses SNR (signal to noise ratio) to classify the link stability. It reroutes the traffic when a link fails at an intermediate node. However, the route rediscovery at intermediate node can result in a longer hop-count route, which increases the latency and route failure probability [7].

To reduce the control overhead, each node in [8] uses the local information of its 1-hop neighborhood to select the best next-hop node for traffic forwarding. However, schemes like [8] fail to find globally optimum routes. The scheme discussed in [9] uses the distribution of the past LLT (link lifetime- the duration for which two nodes remained connected) and the current link age (i.e., the duration since the node pair is connected till the current timestamp) to estimate the residual LLT (i.e., the duration after which the link between the node pair would expire from the current timestamp). However, [9] does not consider the effect of a trajectory change, which can result in an inaccurate LLT computation because the trajectory change can cause nodes to either come closer or move away [7].

Mathematical formulation is proposed in [10] to compute LLT for a node pair using their speed, directions of movement and current coordinates. However, it is limited to the ground vehicles and cannot be used for the airborne node, where the direction of movement continuously changes due to the smooth mobility on a curve.

*Note that none of the above mentioned schemes are suitable for a decentralized, dynamic, multi-hop AN.*

## 1.2 Contributions of Our Proposed Scheme

To address the above-mentioned challenges, an adaptive, low-complexity OLSR protocol for a decentralized, multi-hop AN is presented in this paper. The main contributions of the proposed scheme are as follows:

1. A novel, multi-parametric route selection metric is introduced, which uses hop count, route lifetime (RLT), and route certainty index (RCI) metrics for the route selection.

2. A proactive route switching mechanism is used to prevent packet transmission over broken routes.

3. In the standard OLSR [3], a node computes a route to all nodes in the network, whenever it receives a control message. In our scheme, a node computes routes only for the active source-destination pair(s) and uses them for RLT duration, which significantly reduces the routing computational overhead as compared to the standard OLSR.

4. An adaptive control packet structure is used to minimize the control overhead.

**Paper organization:** The details of the standard OLSR and our proposed adaptive OLSR protocols are described in Sections 2 and 3, respectively. The performance comparison with the standard OLSR is discussed in Section 4, followed by the conclusion and future work in Section 5.

## 2 Brief Overview of OLSR

The OLSR protocol [3] uses two types of control packets-Hello and TC (topology control). Each node includes its 1-hop neighbors in its Hello packet, which is broadcasted periodically after Hello interval (the default value is 2s). Hello packets are used to construct the 1 and 2-hop neighbor sets (i.e.,  $N^1(X)$  and  $N^2(X)$ , respectively) at a node  $X$ .

Each node then finds the smallest subset of its  $N^1(X)$  (which is called MPR (multi-point relay) set) to cover all the nodes in its  $N^2(X)$ , and includes it in its Hello packet to inform its  $N^1(X)$ . A node also maintains an MPR selector set to store those 1-hop neighbors which have included it in their MPR set. Note that only the MPR nodes forward the TC packets of its MPR selector node(s), whereas Hello packets are not forwarded in OLSR.

Each MPR node includes its MPR selector nodes in its TC packet, which is broadcasted periodically after TC interval (the default value is 5s). A node uses TC packets to construct its Topology Set, which stores information of the links between an MPR node and its MPR selectors. Note that the Topology Set of a node represents its current knowledge of the network. Each node in OLSR uses Dijkstra algorithm on the graph built using its Topology Set to find a shortest-hop route to the destination node.

## 3 Description of Our Proposed Adaptive OLSR Protocol

The network modeling and assumptions are discussed below, followed by the description of our proposed scheme.

### 3.1 Network Modeling and Assumptions

Design and evaluation of routing protocols for ANs require such mobility models which can produce realistic node movements [1]. Note that airborne nodes can not make sharp turns due to their high speeds. Therefore, we use smooth-turn mobility model [2] in which each node independently selects a center and radius based on its history, and rotates around the center in the clockwise (or counter-clockwise) direction for a randomly selected duration, which results in a smooth realistic trajectory [2]. Here, a very large radius results in the straight trajectory. We use the buffer boundary model to prevent a node from leaving the network area.

Airborne nodes can be categorized into fixed-wing (FW) and rotatory-wing (RW) [11]. For dynamic missions, FW airborne nodes are preferred since they can attain higher speeds with a longer flight time due to their better aerodynamic design, which gives them stability against harsh environment characteristics, such as air drag [11].

Therefore, we consider a swarm of low SWaP (size, weight and power) fixed-wing UAVs, which is a widely-used AN in the literature [2, 11]. Note that the words *node* and *UAV* are used interchangeably in this document from here onward. We assume that each UAV ( $i$ ) broadcasts its trajectory information (GPS location, movement state (i.e., clockwise, counter-clockwise or straight), center and radius) to

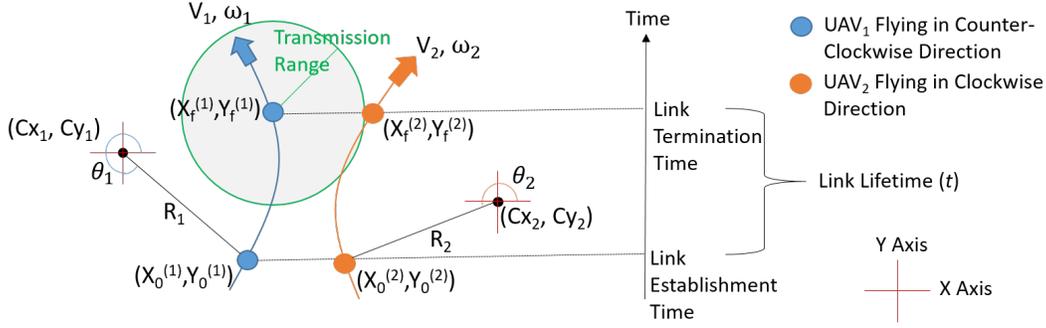


Figure 1. Link Lifetime computation when both UAVs fly in a curve.

$LLT = \{\min(t) \mid \text{Link Distance}(t) \geq \text{Transmission Range}\}$ , where

$$\text{Link Distance} = \left[ a^2 + b^2 + R_1^2 + R_2^2 - 2R_1R_2 \cos(\theta_1 + \omega_1 t - \theta_2 - \omega_2 t) + (\text{sign}_1)2R_1\sqrt{a^2 + b^2} \cos(\theta_1 + \omega_1 t - (\text{sign}_1\text{sign}_2)\alpha) - (\text{sign}_1)2R_2\sqrt{a^2 + b^2} \cos(\theta_2 + \omega_2 t - (\text{sign}_1\text{sign}_2)\alpha) \right]^{1/2}, \quad (1)$$

$$a = \text{abs}(Cx_1 - Cx_2), \text{sign}_1 = \frac{Cx_1 - Cx_2}{a}, b = \text{abs}(Cy_1 - Cy_2), \text{sign}_2 = \frac{Cy_1 - Cy_2}{b}, \text{ and } \alpha = \cos^{-1}\left(\frac{a}{\sqrt{a^2 + b^2}}\right)$$

its 1-hop neighbors<sup>1</sup>, (ii) follows a smooth trajectory, and therefore, cannot make sharp turns [2], (iii) has the collision avoidance capability, and (iv) is half-duplex.

### 3.2 Overview of Our Proposed Scheme

The formulation to compute LLT of a UAV pair and the dynamic packet structure are discussed in Section 3.2.1. Our novel, multi-parametric route-selection metric is described in Section 3.2.2. The routing computational complexity in our proposed scheme is compared with the standard OLSR in Section 3.2.3, followed by the discussion of the proactive route switching mechanism in Section 3.2.4.

#### 3.2.1 Formulation to compute link lifetime

Note that two UAVs establish a link at the *Link Establishment Time* when they exchange their Hello packets for the first time. The link between the UAV pair terminates when they move out of each other's transmission range. This time is called *Link Termination Time*. The LLT of a UAV pair is computed using the following steps:

**Step 1:** Find the coordinates of the future location (i.e., at time  $t$ ) for both UAVs.

**Step 2:** The link between a UAV pair breaks when the distance between them exceeds the node transmission range. It gives an equation with one unknown variable  $t$ .

**Step 3:** Compute root(s) of the above equation.

**Step 4:**  $LLT = \text{Link Termination Time} - \text{Link Establishment Time} = \text{root}$  which best approximates the equation in Step 2.

Based on the movement states for a UAV, the following three cases are possible for a UAV pair.

**Case A.** Both UAVs fly in a curve.

**Case B.** One UAV flies in a curve and other UAV moves in a

straight direction at any angle in range  $[0, 2\pi]$  w.r.t. X axis.

**Case C.** Both UAVs fly in a straight direction at random angles in range  $[0, 2\pi]$  w.r.t. X axis.

The pictorial representation for case A is shown in Fig. 1 and its LLT is computed using (1). For  $UAV_1$  and  $UAV_2$  pair, the current GPS locations are  $(X_0^{(1)}, Y_0^{(1)})$  and  $(X_0^{(2)}, Y_0^{(2)})$ , current trajectory centers are  $(Cx_1, Cy_1)$  and  $(Cx_2, Cy_2)$ , current radii are  $R_1$  and  $R_2$ , velocities are  $V_1$  and  $V_2$ , and the movement directions are  $Dir_1$  and  $Dir_2$ , respectively. Here,  $Dir$  is -1 for clockwise direction and +1 for counter-clockwise direction. The angular velocity for the UAV pair is computed as,  $\omega_1 = \frac{V_1 Dir_1}{R_1}$  and  $\omega_2 = \frac{V_2 Dir_2}{R_2}$ . The initial displacement at the link establishment time for each UAV is computed as,

$$\theta_1 = \tan^{-1}\left(\frac{Y_0^{(1)} - Cy_1}{X_0^{(1)} - Cx_1}\right) \text{ and } \theta_2 = \tan^{-1}\left(\frac{Y_0^{(2)} - Cy_2}{X_0^{(2)} - Cx_2}\right).$$

Similarly, the LLT value for the other two cases can be computed by using the four steps mentioned above. Note that the LLT value of a link is included in the control message in our scheme, which is broadcasted in the network.

• **Discussion on random UAV trajectory change:** During the lifetime of a link, any UAV of a given UAV pair can randomly experience a trajectory change. For example, both UAVs in Fig. 2(b) change their respective trajectory once before the link termination (see 1<sup>st</sup> and 2<sup>nd</sup> Link Trajectory Change in Fig. 2(b)). A link between two UAVs breaks after  $l^{\text{th}}$  trajectory change, where  $l \in [0, 1, 2, \dots]$ . Note that it is not possible to reliably predict the complete LLT, if the future trajectory changes are unknown to UAVs [7]. Therefore, the LLT value is reevaluated in our scheme when either of the UAVs changes its trajectory before the link break. Hence, a UAV pair calculates its LLT when the

<sup>1</sup>If a UAV does not know its trajectory, its center, radius and movement state can be computed by using its three consecutive GPS locations.

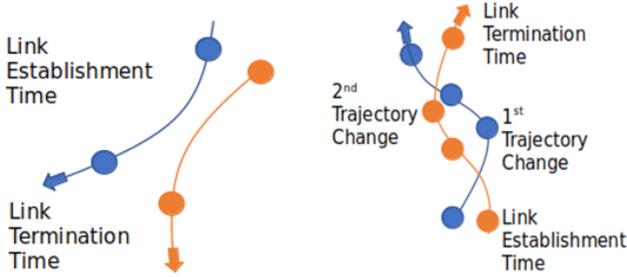


Figure 2. The link between a UAV pair can break (a) without any trajectory change (left image) or (b) after multiple link trajectory changes (right image).

link is established or UAV trajectory changes.

- **Dynamic control packet structure:** Introduction of the additional fields in the control packet used in our scheme increases the packet size and thereby the control overhead and packet error rate. Consequently, the control packets can collide, which results in obsolete route selection. To address this issue, a node includes its trajectory information, whenever it forms a new link or changes its current trajectory.

### 3.2.2 A novel route-selection metric

We use BFS (Breadth First Search) algorithm to compute candidate routes for a source-destination pair, and calculate the *route cost* for each candidate route  $R$  by using (2), where  $HC_R$ ,  $RLT_R$  and  $RCI_R$  are the hop count, route lifetime and route certainty index, respectively, for route  $R$ . In our scheme, route with the minimum *route cost* is selected, which corresponds to the route with a lower  $HC$  and higher  $RLT$  and  $RCI$ . Note that the  $RCI$  value represents the stability of a route.

$$\begin{aligned} \text{Route Cost } (R) &= \frac{HC_R}{RLT_R \sqrt{(1 + RCI_R)}}, \text{ where} \\ RLT_R &= \min_{\text{link } i \in R} (LLT_i), \\ RCI_R &= \frac{1}{HC_R} \sum_{\text{link } i \in R} \log_{10}(1 + LLT_i - RLT_R) \end{aligned} \quad (2)$$

### 3.2.3 Discussion on computational overhead

In the standard OLSR [3], a node recomputes routes to all nodes in the network, whenever it receives a new control message, which results in a huge computational overhead. In our scheme, a node computes route only for the active source-destination pair(s), where a data generating node is called active source, and the node for which data is generated is called active destination. Once a route  $R$  is selected, node uses it for  $RLT_R$  duration. However, if an updated LLT value is received for any link on route  $R$ , which is lower than the current  $RLT_R$ , node selects a new route using (2)<sup>2</sup>.

<sup>2</sup>Note that an RW node can hover at a particular position for indefinite time, which makes reliable LLT prediction impossible. Moreover, it can change its trajectory very frequently due to the high maneuverability capabilities, which results in the frequent LLT recomputations, and thereby, route reevaluations. As a result, the importance of the route lifetime and its stability in the route selection metric diminishes. Therefore, only the hop

### 3.2.4 Proactive route switching mechanism

In the standard OLSR [3], a node discards a link from its routing table only when it does not receive an update for that link for a predefined large interval. Therefore, nodes in the standard OLSR fail to quickly identify a link break, which leads to the packet transmission over a broken route. In our scheme, each node uses a route  $R$  for  $RLT_R - TTL$  duration, where TTL represents the packet time-to-live value. Then it selects a new route. This mechanism improves the PDR.

## 4 Performance Comparison with Standard OLSR Protocol

The simulation setup is discussed in Section 4.1, followed by the performance comparison of our scheme with the standard OLSR for two evaluation metrics in Section 4.2.

### 4.1 Simulation Setup

Simulations are run in NS-3 version 3.29. The network topology consists of 30 FW-UAVs moving under the smooth-turn mobility model [2] at 20 m/s. Packet size is 1 kB, queue size is 1000 packets, channel rate is 11 Mbps, and the MAC protocol used is CSMA/CA (carrier-sense multiple access with collision avoidance) in which the RTS (request to send)-CTS (clear to send) exchange is enabled.

We consider a video streaming application with a 10 s packet TTL value. Transmission ranges used are 1 km and 1.5 km, which results in two different co-channel interference levels. We consider the line-of-sight communication and Friis propagation loss model. Data is generated for 75 s. Each experiment is repeated 10 times. A source-destination pair is randomly selected in each run.

The following two metrics are used in the simulation to compare the performance of our proposed scheme with the standard OLSR for different data rates.

(i) **PDR** for a flow is the ratio of total packets received by the destination node over total packets generated at the source. Since PDR represents a normalized throughput, the flow throughput can be computed as PDR x Data Rate.

(ii) **Average number of the routing table computations.**

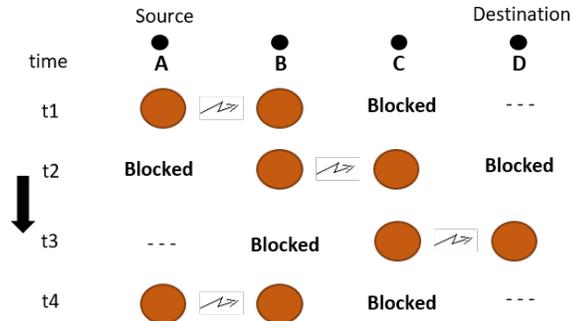
Note that the maximum achievable flow throughput for an omnidirectional communication in a multi-hop network cannot exceed 33% of its channel capacity (see Fig. 3). In fact, after considering the control overhead, the maximum achievable link throughput for a flow on a multi-hop route would be less than 33% of the channel capacity. We observed a very low PDR value for 1 km transmission range at data rates higher than 2 Mbps due to the heavy network congestion. Therefore, the performance is shown only for 1, 1.5 and 2 Mbps data rates.

### 4.2 Simulation Results

The average number of link breaks experienced by the routing schemes are 73 and 117, respectively, for 1 km and 1.5 km transmission range. The PDR obtained by both schemes is shown in Table 1 for three different data rates (1 Mbps, 1.5 Mbps and 2 Mbps) at two different transmission ranges (1 km and 1.5 km).

Recall that the standard OLSR fails to quickly adapt to the link breaks, whereas our scheme selects a stable, longer-

count should be considered in (2) for the route selection for RW nodes.



**Figure 3.** The channel access mechanism for a 3-hop topology, where the source node A transmits packets to destination node D over the route A-B-C-D in over a single frequency band. Here, node A forwards its packet to node B in t1 time slot. In t2 and t3 time slots, node A should not communicate to prevent collisions at node B. Therefore, node A can access the channel once in every three time slots.

lasting route and proactively switches to a new route. Therefore, our scheme has a superior PDR performance, which is higher by 15% to 38% as compared to the standard OLSR scheme. Note that the congestion increases with the data rate, which results in the packet drop due to the buffer overflow and TTL expiry. Therefore, the PDR values of both schemes reduce as the data rate increases.

The increase in the transmission range reduces the route length (in terms of hops). As a result, fewer packets are dropped due to TTL expiry, which improves the flow PDR. Therefore, the PDR values of both schemes increase when the transmission range increases from 1 km to 1.5 km.

**Table 1.** Comparison of PDR for Both Schemes

Data Rate →	1 Mbps		1.5 Mbps		2 Mbps	
Scheme ↓	1 km	1.5 km	1 km	1.5 km	1 km	1.5 km
<b>OLSR</b>	47%	66%	45%	61%	39%	58%
<b>Our Scheme</b>	79%	100%	71%	99%	54%	93%

In addition, the comparison of computational overheads in the standard OLSR and our scheme are shown in terms of the average number of the routing table computations in Table 2. Since our scheme computes routes only for the active source-destination pairs and selects more stable and longer-lasting routes, its average routing table computations is significantly lower than that of the standard OLSR for both transmission ranges.

As mentioned earlier, an increase in the transmission range reduces the hop count, which reduces the number of control messages forwarded. Therefore, the total routing table computations is lower at 1.5 km transmission range as compared to 1 km transmission range for both schemes.

## 5 Conclusion and Future Work

In this paper, a novel, adaptive, proactive routing protocol is presented for a dynamic, multi-hop airborne network. Its main features are: (i) a novel, multi-parametric route selection metric is proposed to select a route with a

**Table 2.** Comparison of Average Routing Table Computations for Both Schemes

Transmission Range	OLSR	Our Scheme
<b>1 km</b>	35,949	956
<b>1.5 km</b>	22,369	397

lower hop count and higher route lifetime and stability, (ii) a proactive route switching mechanism is used to prevent the packet transmission over a broken route, (iii) low computational overhead, and (iv) a dynamic control packet structure is used to minimize the control overhead. The simulation results show that our scheme has a superior throughput performance and a much lower routing computational complexity as compared to the standard OLSR.

## 5.1 Ongoing and Future Work

The importance of a routing metric depends on several factors, including the network topology, node density and number of flows, and it varies with time. Therefore, we are in the process of dynamically adjusting the importance of each metric at each node using Bayesian optimization and reinforcement learning, in order to support the traffic flows of different data rates, latency, and priority.

We will then upgrade the above adaptive OLSR protocol to facilitate the parallel packet transmissions over multiple, interference-free paths, which will further improve the flow throughput and reduce the end-to-end delay.

## 6 Acknowledgments

The author would like to thank his advisors Prof. Sunil Kumar and Prof. Alexander Ihler for their constant support and supervision.

## 7 References

- [1] M. Y. Arafat and S. Moh. Routing protocols for unmanned aerial vehicle networks: A survey. *IEEE Access*, 7:99694–99720, 2019.
- [2] J. Xie et al. A survey and analysis of mobility models for airborne networks. *IEEE Commun. Surv. Tutor.*, 16(3):1221–1238, 2013.
- [3] T. Clausen et al. Optimized link state routing protocol (olsr). *RFC 3626*, 2003.
- [4] C. Pu. Link-quality and traffic-load aware routing for uav ad hoc networks. In *Int. Conf. Collaboration Internet Comput.*, pages 71–79, 2018.
- [5] Y. Zheng et al. A mobility and load aware olsr routing protocol for uav mobile ad-hoc networks. *Int. Conf. Commun. Technol.*, pages 1–7, 2014.
- [6] K. Bao et al. Intelligent software-defined mesh networks with link-failure adaptive traffic balancing. *IEEE Trans. Cogn. Commun. Netw.*, 4(2):266–276, 2018.
- [7] M. Gerharz et al. Link stability in mobile wireless ad hoc networks. In *IEEE Conf. Local Comput. Netw.*, pages 30–39, 2002.
- [8] G. Oddi et al. A proactive link-failure resilient routing protocol for manets based on reinforcement learning. In *Mediterranean Conf. Control Automat. (MED)*, pages 1259–1264. IEEE, 2012.
- [9] Z. Guo et al. Multi-objective olsr for proactive routing in manet with delay, energy, and link lifetime predictions. *Appl. Math. Model.*, 35(3):1413–1426, 2011.
- [10] Z. Li and Z. J. Haas. On residual path lifetime in mobile networks. *IEEE Commun. Lett.*, 20(3):582–585, 2016.
- [11] A. G. Perez and M. D. Cano. Flying ad hoc networks: A new domain for network communications. *Sensors*, 18(10):3571, 2018.