

Journal Pre-proof

Optimized Unmanned Aerial Vehicles deployment for static and mobile targets monitoring

Fadi Al-Turjman, Hadi Zahmatkesh, Ibrhaim Al-Oqily, Reda Daboul



PII: S0140-3664(19)30695-4

DOI: <https://doi.org/10.1016/j.comcom.2019.10.001>

Reference: COMCOM 5968

To appear in: *Computer Communications*

Received date: 27 June 2019

Revised date: 6 September 2019

Accepted date: 1 October 2019

Please cite this article as: F. Al-Turjman, H. Zahmatkesh, I. Al-Oqily et al., Optimized Unmanned Aerial Vehicles deployment for static and mobile targets monitoring, *Computer Communications* (2019), doi: <https://doi.org/10.1016/j.comcom.2019.10.001>.

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2019 Published by Elsevier B.V.

Optimized Unmanned Aerial Vehicles Deployment for Static and Mobile Targets Monitoring

Fadi Al-Turjman¹, Hadi Zahmatkesh², Ibrhaim Al-Oqily³, and Reda Daboul¹

¹Department of Computer Engineering, Antalya Bilim University, Antalya, Turkey.

²Department of Computer Engineering, Middle East Technical University, Northern Cyprus Campus.

³The Hashemite University, P. O. Box 150459, Zarqa 13115, Jordan and Al-Yamamah University, P.O. Box 45180, Riyadh 11512, Saudi Arabia.

E-mails: fadi.alturjman@antalya.edu.tr, hadi.zahmatkesh@gmail.com, ialoqily@ieee.org, reda.dabol@antalya.edu.tr

Abstract— In the recent decade, drones or Unmanned Aerial Vehicles (UAVs) are getting increasing attention by both industry and academia. Due to the support of advanced technologies, they might be soon an integral part of any smart-cities related project. In this paper, we propose a cost-effective framework related to the optimal placement of drones in order to monitor a set of static and/or dynamic targets in the IoT era. The main objective of this study is to minimize the total number of drones required to monitor an environment while providing the maximum coverage, which in turn leads to significant reduction in cost. Our simulation results show that by increasing the battery capacity of the drones, the drones' visibility range would also increase and thus, the number of drones would be reduced. Moreover, when the targets are sparsely distributed across a large number of different regions, a further increase to the targets does not require an increase in the number of drones needed to monitor them.

Index Terms—Smart city, Unmanned Aerial Vehicle (UAV), Drone, Internet of Things (IoT).

1. Introduction

In recent decade, due to massive growth in the telecommunication sector, there is a high demand for providing high quality services. Moreover, the rapid growth in population and increase in the number of mobile connected devices have brought several challenges such as network coverage and capacity. One promising way to overcome such challenges is to utilize intelligent systems towards smart projects such as smart cities. Smart cities [1] are led by strategic administrations that support technology and innovation. The aim of smart cities is to maximize the efficient use of

valuable resources to foster sustainable growth. Unmanned aerial vehicle (UAV) is considered a crucial part of smart cities. The main objective of a smart city is to improve its resident's life by providing low-cost services and efficient infrastructure. UAVs are already being used to document accident scenes [2], support first responder activities, and monitor construction sites [3], but they are ready to become an integral part of a smart city's network as well. UAVs can be used to gather key intelligence data on movements of potential threats and to help in determining locations of threats and providing detailed topographic information in real time. They can also be utilized in providing an accurate representation of an area using images, which can help to rescue human and animals in case of a disaster [4]. In addition, it is important to develop model-based controllers in order to monitor the movement of the UAVs in real-time. In this regard, discrete event modelling of embedded systems can be used as a formal method to develop software for embedded systems in order to implement the system software as a model [44].

The concept of smart city is converting cities into digital societies, transforming the life of its citizens to an easy life in every facet, and intelligent transport system (ITS) becomes an indispensable component among all. ITS [5] is considered as the application of sensing, control, analysis, and communication technologies to ground transportation in order to enhance security and mobility as well as efficiency. It includes a wide range of applications that process and share information to ease congestion, enhance traffic management, minimize environmental impact, and increase the benefits of transportation to commercial users and the public in general. As our contributions, we propose a cost-effective framework to minimize the total number of drones needed to monitor an area while providing the maximum visionary coverage for the target. In other words, we optimize the number and location of drones to have full coverage of an area which in turn reduces the overall cost of the network.

In order to fully understand drones and the proposed framework in this study, we first discuss various applications of drones in smart cities. The concept of smart city is based on the integration of Information and Communication Technology (ICT) and its trends. Smart cities play a significant role in development of a sustainable environment and drones play a major role in it as well. Drones are widely utilized in various areas. Their applications can be classified into environmental- and industrial-based applications. Some typical ones of these applications are listed and described in the following subsections.

1.1 Environmental-Based Applications

Drones or UAVs are becoming increasingly popular for monitoring the environment. The technology has entered various fields such as surveillance and search and rescue operations. Drones can be equipped with sensors and cameras, making them ideal for monitoring environment. In this section, we briefly discuss some important applications of drones in monitoring environments.

Disaster Management: Disasters are affecting different regions of the world every year. They are unstoppable events that are either natural or man-made, such as wildfire, earthquake, terrorist attacks, and floods. One of the major challenges faced by the rescue team during an enormous disaster is to find survivors and victims as early as possible and to take them out of the disaster area to ensure that they are not stuck under the destroyed area. In this regard, drones can help to detect people in disaster areas [6]. They can be equipped with sensors and camera to identify the precise location of survivors as early as possible. The data can be sent to the rescue team for further investigation and action.

Vegetation Management: Drones provide an important innovation in vegetation monitoring. By using the right sensors and an appropriate camera attached to it, it is possible to map the health of

the crops by determining soil quality, humidity, and pollution in the area. The advantage of using drone-based system for vegetation monitoring is that unlike satellite images, drones can provide more information in relatively smaller areas. Moreover, the cost of using drones is much lower compared with manned flights, and therefore, it makes the technology more accessible.

Water Resource Management: Water management is one of the main issues in agriculture, in which new technologies such as drones can provide solutions. The use of drones in water management can help to provide solution on how to manage irrigation water and maximize its efficiency. For example, integration of UAV photogrammetry and image recognition technology can be used to solve the limitations of the existing measuring tools and techniques for water level measurements in the field [7].

1.2 Industrial-Based Applications

Drones or UAVs are playing a significant part in the industrial internet of things (IIoT) [8]. They can be valuable in industrial applications such as mining [9], oil and gas [10], and construction [11]. We briefly discuss some important aspects of these industrial applications in this section.

Mining Activities: Drones or UAVs can enhance security in applications related to mining activities with real-time information, such as latest surface surveys for enhanced blast patterns, quick and accurate pre- and post-blast information, and recognizing of misfire and wall damage. Moreover, drones can provide an effective approach to monitor stores and assist with area exploration as well as general management. In addition, miners can gain benefits from the use of drones in the design of roads and dumps, as they help them to find out more efficient approaches in terms of environmental impacts.

Oil and Gas: UAVs or drones have been deployed and used by several operators in oil and gas sector for various activities in difficult environments [10]. These activities include data collection,

inspection, and exploration. Using inspection drones in oil and gas sector has several advantages over traditional inspection methods. For example, it eliminates major dangers to personnel involved in traditional inspection activities in dangerous environments. Moreover, a significant reduction in cost is achieved due to ease of access in difficult environments.

Construction: Aerial craft can be used in almost every stage of the engineering process, from planning to final construction. Helicopters and airplanes are already being used in civil engineering for different purposes such as mapping from a plane and producing marketing films for tourist destinations. Utilizing drones can significantly reduce the expense and time traditionally involved in various stages of the engineering process, such as construction of roadways and forest road, and coastal erosion.

The contribution of this study is that the proposed cost-effective framework deals with a cost minimization problem related to the optimal placement of drones which in turn monitors a set of static or dynamic targets. The minimization problem aims to reduce the number of drones in the environment while providing the maximum coverage, given a constant value of battery capacity. Moreover, the proposed framework can be integrated with Artificial Intelligence (AI) and deep learning for the problem of drone detection and tracking challenges [39]. For example, AI and deep learning can be used together with drones equipped with sensors as a promising solution in intrusion detection systems [40][41]. In addition, the study in [42] shows that Machine Learning (ML) algorithms can be applied in deployment of drone BSs in wireless networks to analyze the traffic pattern and estimate the traffic demand in the target system. Similarly, the authors in [43] utilize deep learning approaches for on-demand drone deployment in emergency and temporary conditions since the position of drones is a crucial factor that affects the available capacity to the data flows which is being served.

The remainder of this article is organized as follows. Section 2 presents the existing studies related to the use of UAVs in the IoT era. Problem description and the proposed framework are outlined in Section 3. Section 4 discusses the performance metrics, results, and findings of the study. Finally, Section 5 concludes this paper. A list of abbreviations together with their brief definitions used throughout the paper is provided in Table 1 to help the readers in understanding the abbreviated terms.

Table 1. List of Abbreviations	
Abbreviated	Name
AED	Automated External Defibrillator
AI	Artificial Intelligence
CPMS	Canonical Particle Multi-path Swarm
CPS	Canonical Particle Swarm
DSP	Drone Scheduling Problem
ECA	Emission Control Area
FMPS	Fully Multi-path Particle Swarm
GPS	Global Positioning System
ICT	Information and Communication Technology
IIoT	Industrial Internet of Things
INS	Inertial Navigation System
IoT	Internet of Things
ITS	Intelligent Transport System
KF	Kalman Filter
LRBA	Lagrangian Relaxation-Based Approach
ML	Machine Learning
ODP	Optimal Drone Placement
OPA	Optimized Placement Approach
OS	Operating System
QoS	Quality of Service
TSP	Traveling Salesman Problem
UAV	Unmanned Aerial Vehicle
WBS	Wireless Base Station

2. Related Works

This section provides a review on existing studies related to use of drones and their deployment approaches in the IoT era. Recently, the use of UAVs or drones to control the emissions of sailing vessels has attracted too much attention due to its significant potential for performing regulations in Emission Control Areas (ECAs). In [12], the authors propose a drone scheduling problem (DSP) such that a group of planned tours is developed for drones to examine the vessels in ECAs. The dynamics of sailing vessels are modeled by utilizing a location function in real-time in a deterministic manner. They also propose a Lagrangian relaxation-based approach (LRBA), which is able to gain the best solution for the problem in large-scale cases. The results reveal that the proposed approach outperforms the commercial ones for the problem of up to 100 vessels. Drones have also been well utilized for military purposes [13-16]. For instance, the routing of a set of drones to destroy a determined group of targets that are prioritized differently is studied in [13]. The authors propose a two-phase approach that considers resolving a sub-problem related to target assignment for each drone in the initial phase. The second phase in this solution framework is to solve a travelling salesman problem (TSP) to obtain a routing plan. Similarly, the authors in [14] develop an integer-programming model for an environment where new targets may emerge dynamically. This model reassigned UAVs to the updated group of tasks regarding any changes in the battleground. In [15], the routing of UAVs is considered for military surveillance purposes, where UAVs gather information from targeted area using sensors. The proposed strategy chooses the sensors for each UAV by including payload capacity restrictions. Then, following these constraints, a group of UAVs is routed to develop a region-sharing approach by considering uncertainty on the data gained from observations. This strategy dynamically sends drones to gather information instead of focusing on a predetermined routing plan. The results of the study prove that

the proposed approach is effective in a contemporary battleground where communications between UAVs and ground base stations are frequently blocked.

Besides military purposes, several studies consider the problem of routing drones in logistic delivery operations [17-19]. In this regard, drones assist trucks to deliver items to the customers who are located geographically. However, drones are often limited to carry only one package that makes the routing decision of the drones easier and enhance the operational efficiency [12]. For instance, a joint scheduling problem for trucks and drones is studied in [27]. In this study, drones are used to deliver packages to customers close to the storage, and trucks are responsible to deliver parcels far. The results reveal that, with such a delivery system, customers receive their orders faster. Moreover, it reduces cost of distributions as well as environmental impacts. Another similar problem is studied in [18]. In this study, trucks are permitted to carry UAVs in specific routes so that they can fly from the trucks and deliver parcels to people who are far from the storage. In [19], the authors prove that the potential improvement in delivery efficiency of the cooperation between drones and trucks depend on the speed of drones and the square root of the ratio of the speed of trucks.

UAVs or drones are increasingly proposed for medical use cases as well. For example, the study in [20] develops a new optimization model to help in the deployment of a network of automated external defibrillator (AED)-enabled medical drones to minimize the time it takes to reach to a patient's side. The proposed approach can optimally locate drones by considering the problem of backup coverage location with complementary coverage. At the same time, it improves backup coverage with insignificant loss of initial coverage.

In several studies, drones have been used for the tasks related to trajectory planning and task allocation [21, 22]. For example, in [21], the authors propose an automated surveillance system to

track several mobile ground targets. The aim of this study is to reduce the total energy consumption and to find the exact location of the targets. The study in [22] proposes a system containing several operating drones and a control station. The drones receive control information from the control station and send their location information and the sensed parameters back to the control station. The results show the effectiveness of the proposed task allocation algorithm in terms of task completion.

Apart from the studies where drones are used for ideal trajectory planning problems, some studies have utilized drones to track different targets using various sensors. For example, in [23], an algorithm is proposed to track a mobile target in a cooperative manner using several drones equipped with cameras. The goal is to keep the mobile target in the position visible by cameras from various angles while achieving a low computational complexity. In addition, the authors in [24] investigated a similar problem by considering multiple criteria, such as the number of drones, the satisfaction of customers, and the total distance moved by the drones simultaneously. The objective is to detect the exact location of mobile targets using the sensors placed on the target. Although the mentioned studies so far reveal the use of drones and their deployment approaches in the IoT era, but none of them consider the optimal drone placement (ODP) problem and the issues related to the target coverage. However, these issues are extensively investigated in the following studies [25-27]. For example, the optimal placement of a group of drones is considered in [26], with the assumption that a large number of drones are available to cover a group of mobile targets. The main objective of this study is to reduce the total amount of energy consumption. A similar study is presented in [27], where mobile targets are monitored by a group of drones that have restricted energy resources. The aim of this study is to reduce the number of required drones to monitor a piece of plane where the targets are moving. The authors mathematically present the

problem under study by using mixed integer non-linear optimization models. In addition, heuristic procedures according to restricted mixed integer-programming formulation are defined for the problem. Finally, the behavior of the proposed model is assessed and a comparison is provided between the proposed model and the mixed integer-programming-based heuristic models in terms of efficiency and effectiveness. In [25], the authors propose a mathematical model to formulate the ODP problem. They provide an improved model that considers the energy of each drone, and design an ideal approach to solve the placement problem of static or mobile drones. Using two low-complexity centralized algorithms, samples of the mentioned problem with more than 50 targets and a large number of possible locations for the drones can be solved.

Although these studies tried to solve the ODP problem with the reduced amount of energy consumption, but they did not attempt to provide the maximum coverage for the drones while minimizing the total number of drones in monitoring the environment which is considered in this study.

In [32], the authors examine and simulate real time Inertial Navigation System (INS) and Global Positioning System (GPS) in UAV navigation using a two-level Kalman Filter (KF). The proposed approach is based on predicting the error in position of the INS and then removing it from its corresponding position besides the second level of applied KF for the entire integrated GPS/INS errors. The results obtained show that the KF-based module is able to decrease the INS position error and prevent its growth even in the long-term period. The study in [33] introduces an optimized data delivery framework called Canonical Particle Swarm (CPS) for multimedia delivery using drones in the 5G/IoT era. In the proposed framework, multi-swarm strategy is utilized to specify the optimal direction while carrying out a multi-path routing. The results show the performance improvement of the proposed method compared to other ordinary optimization

approaches such as Canonical Particle Multi-path Swarm (CPMS) and Fully Multi-path Particle Swarm (FMPS).

In [34], the authors present a study which aims to find the optimal locations for static drones in a given area in order to maximize the coverage. The algorithm solves the ODP problem for both clustered and uniformly targets. The results show the effectiveness of the proposed approach in solving the placement problems of drones. However, the approach does not consider the mobile targets in the area.

The authors in [35] use an algorithm based on gradient projection in order to find the optimal placement of a single drone in case it can be used as aerial Wireless Base Station (WBS) when cellular networks are out of service. They consider the uplink scenario as a constraint and find an optimal location for the drone in a way to maximize the sum of durations of the time of uplink transmissions. A similar study in [36] presents an algorithm to optimally locate drone BS in an area with various target densities. The authors aim to minimize the number of drones and their 3D placement in a way that all the users are served.

Another study on 3D deployment of drone BSs is presented in [37]. The authors propose a framework to maximize the number of covered users with various Quality of Service (QoS) requirements. They model the problem as a placement problem with multiple circles and propose an algorithm that uses an exhaustive search in a closed region over a 1D parameter. In addition, the authors propose the maximal weighted area algorithm to deal with the placement problem. Such use cases are important due to the growth of data traffic caused by multimedia applications [38].

3. Proposed Methodology

One of the major drawbacks of drones is their limited range [28], which is due to the capacity of the battery. Another challenging issue regarding the usage of drones is related to their high price. Therefore, there is a need to optimize the number and location of drones to have full coverage of an area. In this regard, we propose a method called optimized placement approach (OPA) to minimize the total number of drones required while providing maximum coverage. This in turn leads to reduction in cost.

3.1 Optimized Placement Approach

The relationship between the drone's height and the coverage area of the target can be formulated by $A = \pi(R^2 - h^2)$ where h is the height of the drone, and A and R are the drone's coverage area and the radius of the drone's wireless transmitter respectively. Clearly $A = \pi R^2$ when $h = 0$. In our model, we assume that there are N drones in the area in which they can fly to a maximum height of h_{max} and minimum height of h_{min} maintaining a particular coverage radius [25]. We assume that there is a location (x, y, z) that each drone can be placed in the area.

Please note that deploying drones to cover targets is not a simple problem. The deployment strategy should reduce the overall cost by minimizing the number of drones required to control a target. They should be placed in a way to cover all the targets while having minimum overlapped coverage between drones. Moreover, there should be high quality wireless communications between the drones and the ground targets which can be achieved by reducing the altitude of the drones. In addition, multi-hop connectivity between the drones and a BS can be provided using air-to-air communications. Therefore, connectivity, full coverage of the targets and the quality of the coverage are the main objectives and constraints of our problem.

Let D denotes a group of available drones and T represents the group of targets to be monitored. The objective function is to minimize the total number of required drones for monitoring the environment. Assume that each target is determined by the values (x, y, z) , where x , y , and z signify length, width, and height of the target, respectively. Therefore, given a drone d , it is located at a coordinate (x_d, y_d, z_d) with a target T to be monitored. It is possible to define the distance between d and T_i when $z = 0$ as follows [31]:

$$U_{t_i}^{x_d y_d} = \sqrt{(x_{t_i} - x_d)^2 + (y_{t_i} - y_d)^2} \quad (1)$$

Drones have a visibility of θ , which is signified by a disk on the plane with radius r^z depending on z_d . The drone visibility is also dependent on the angle of camera lens. Moreover, the position where each drone $d \in D$ should be located (x_d, y_d, z_d) , and the target $t_i \in T$ monitored by the drone should be decided. Therefore, the first decision variables are defined as follows:

$$\delta_{xyz}^d = \begin{cases} 1 & \text{if } d \text{ is located at coordinate } (x, y, z) \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$\gamma_{t_i}^d = \begin{cases} 1 & \text{if target } t_i \text{ is observed by drone } d \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

The goal is to carefully control and watch all the targets with minimum one UAV to minimize the required number of drones as well as total energy consumption which will be formulated in section 3.2. Furthermore, according to [25][27][30] the energy consumption of each drone can be formulated as follows:

$$E = (\beta + \alpha k)t + P_{max}(K/S) \quad (4)$$

where β is the minimum power for the drone required to stay in the air and α is a motor speed multiplier. α and β are both dependent on the weight of the drone, and the features of the motor it is using. P_{max} , S , and t are maximum power of the motor, speed, and the operating time,

respectively. The term αk indicates the relationship between the height and power, and P_{max} (K/S) is the power required to move up to height K with speed S . The objective function is to minimize the total number of required drones for monitoring the environment, which can be formulated as follows:

$$\text{Min } f(\delta) \text{ s.t. } \sum_{(x,y,z)} \delta_{xyz}^d \leq 1 \forall d \in D \quad (5)$$

$$\gamma_{t_i}^d \leq \sum_{(x,y,z)} \delta_{xyz}^d \left(\frac{r^{z_d}}{D_{t_i}^{dy}} \right) \quad \forall d \in D, t_i \in T \quad (6)$$

3.2 Equations of the Drone's Location

The flying zone for the drone is represented by $Z = [Z_{min}, Z_{max}]$. This is the area that is parallel to the plane containing the targets. Detection of the target above Z_{max} is not possible, and the drones are not permitted to fly above this threshold in the region. In addition, drones cannot fly below Z_{min} . The flying zone is presented by a rectangle of length X_{max} and width Y_{max} such that

$$\sum_{d \in D} \gamma_{t_i}^d \geq 1 \quad \forall t_i \in T \quad (7)$$

When the drones fly, they need to observe the target for a specified amount of time. Additionally, the target can move in the region, particularly a time window $[\tau_{min}^{t_i}, \tau_{max}^{t_i}]$ is associated with each target $t_i \in T$, meaning that at the beginning the target t_i is placed at the point of coordinate, and it has been detected in the time range specified by the time window. If the target is moving to catch mobility in the system, a sequence coordinate C_i is associated with each target. According to [26], it is assumed that:

$$|C_i| = \left[\frac{\tau_{max}^{t_i} - \tau_{min}^{t_i}}{\Delta \tau} \right] \quad (8)$$

where $\Delta\tau$ represents the time interval such that a new location of the target t_i is obtained.

Considering all the constraints and objectives discussed earlier, the equations to minimize the number of required drones and the total energy consumption can be formulated as follows:

$$f(\delta) = N * \sum_{(x,y,z)} \sum_{d \in D} \delta_{xyz}^d t \quad (9)$$

$$f(\delta) = A - N * \sum_{d \in D} \delta_{xyz}^d * A_i \quad (10)$$

Where A is the entire area to be monitored and N and A_i are the number of drones and the area monitored by the i^{th} drone respectively.

$$E = N * (\beta \sum_{(x,y,z)} \sum_{d \in D} \delta_{xyz}^d t + a \sum_{(x,y,z)} \sum_{d \in D} Z \delta_{xyz}^d + \frac{P_{max}}{S} \sum_{(x,y,z)} \sum_{d \in D} Z \delta_{xyz}^d) \quad (11)$$

4. Performance Evaluation and Results

In this section, to assess our proposed model, we discuss the performance metric and parameters as well as the results obtained by simulation.

4.1 Simulation Setup

Equations (1)–(10) in Section 3 are aimed to minimize the number of drones. The simulation has been implemented in Octave programming language, namely GNU 4.4.1 [29] which is a high-level scientific programming language primarily intended for numerical computations. Octave is an important open-source and free software tool used in robotics which is equivalent to Matlab.

Our simulation script was executed on a device with Windows 8 Operating System (OS). The device has the following specifications: Intel(R) Atom(TM) CPU Z2760 @ 1.80 GHz, 1,800 MHz, 2 Core(s), 4 Logical Processor(s). The usage of RAM was low, and the computation time was from 4 to 6 s. There are a couple of assumptions that were made during the simulation phase. First, the battery capacity is not subjected to optimization, that is, the optimum value for battery capacity

corresponding to the minimum number of drones are determined through trial and error. This is because various applications have different requirements which may affect the battery capacity. Therefore, through the trial and error, we determine the optimum value for the battery capacity of the drones. Second, each drone covers an area that is a square of 1 km^2 . The reason for choosing such a large coverage area for the drones is that we are targeting a large-scale scenario such as street coverage in smart cities. Finally, for simplicity purposes, the communication range between drones is considered as circular disks.

4.2 Performance Metrics and Parameters

To assess the proposed framework, the following performance metric is considered in the script.

Target Coverage: This is the coverage area of drones while flying over a target. The metric is evaluated while varying the following parameters:

Energy (E): It represents the initial capacity of the drone's battery.

Visibility angle (θ): It is the opening of the drone visibility range.

Horizontal Energy Consumption (γ): This is the energy consumed due to the horizontal movement of the drone.

Vertical Energy Consumption (α): It represents the energy consumed due to the vertical movement of the drone.

Number of Targets (nt): It represents the number of targets to be monitored by the drones.

Please note that there are also some important parameters such as delay which can affect the critical applications of the UAVs. For example, the average experienced delay in the system may change if the height of the drone changes. In this regards, if the height of the drone increases, the delay may increase too. This is due to the increase of coverage in the space that permits less hops between the target and the final destination. Another critical communication metric is the network

throughput which can be defined as the amount of useful works such that connected drones can carry out per the unit of time in terms of bytes that have been delivered successfully to the base station. If the number of drones increases, the network throughput increases too. This is clear when there are more drones available, there will be more routes available towards the base station. This in turn causes improvement in the data delivery. However, by increasing the number of drones, the total energy consumption of the network increases as well. Therefore, it is important to find the optimal number of required drones for monitoring an area in order to have an energy-efficient system.

4.3 Results and Discussions

In this section, we discuss the results obtained from the simulation tool implemented to evaluate the performance of the proposed framework. During the simulation, we considered two phases: static and dynamic targets.

4.3.1 Static Targets

Static targets have fixed positions and do not change their locations. We considered two situations regarding static targets. First, given a fixed number of targets ($nt = 100$), the results for the optimal solution (e.g. minimum number of drones) are obtained as follows. Figure 1 shows a scenario where the drones, which are flying away either horizontally or vertically or both, have a narrower visibility range than the ones close to the $(0, 0)$ coordinates. The number of drones is initially 23 drones. If we decrease the energy consumption of flying horizontally to $\gamma = 2$ with less battery capacity as in Figure 2, it can be seen that drones flying to the right and to the top right of Figure 2 are covering more targets due to flying further. Doing the same steps of decreasing energy cost

of moving horizontally and the battery capacity makes the drones fly further and higher. Therefore, a bigger visibility range is achieved as shown in Figure 3.

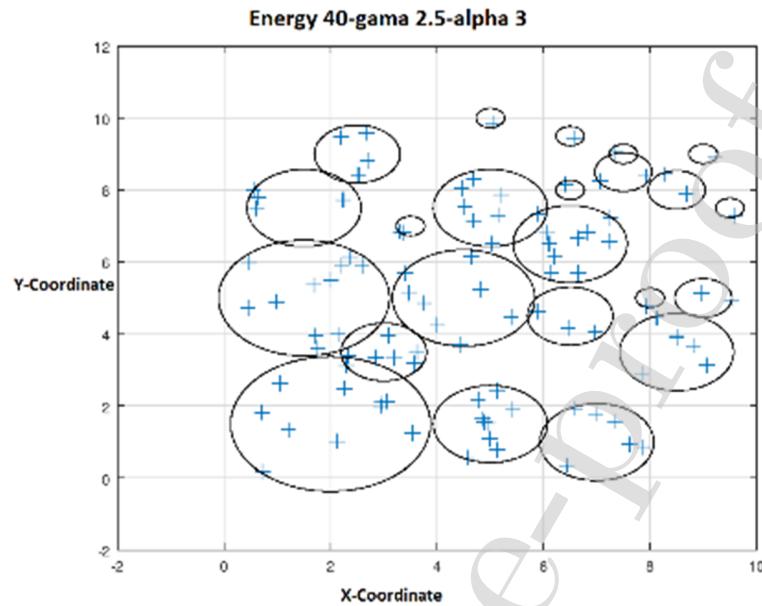


Figure 1 - Energy = 40, gamma = 2.5, alpha = 3.

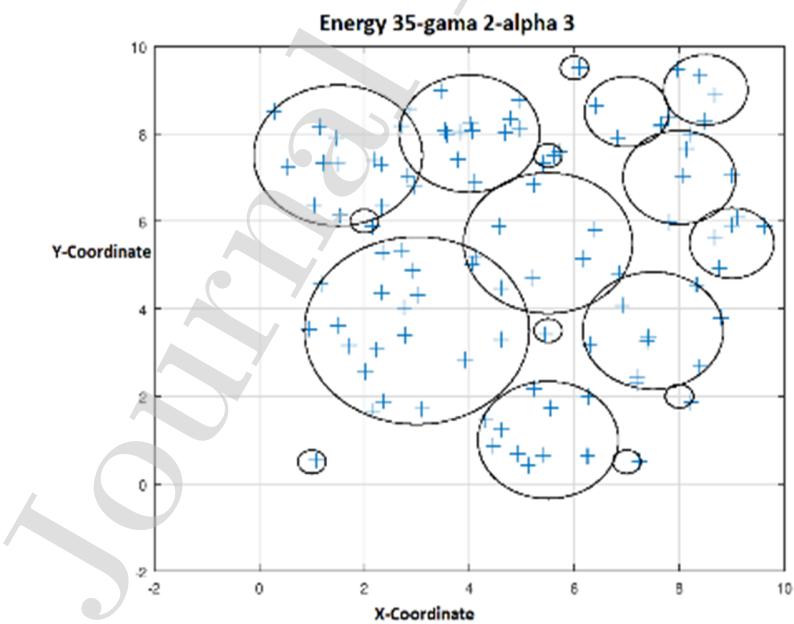


Figure 2 - Energy = 35, gamma = 2, alpha = 3.

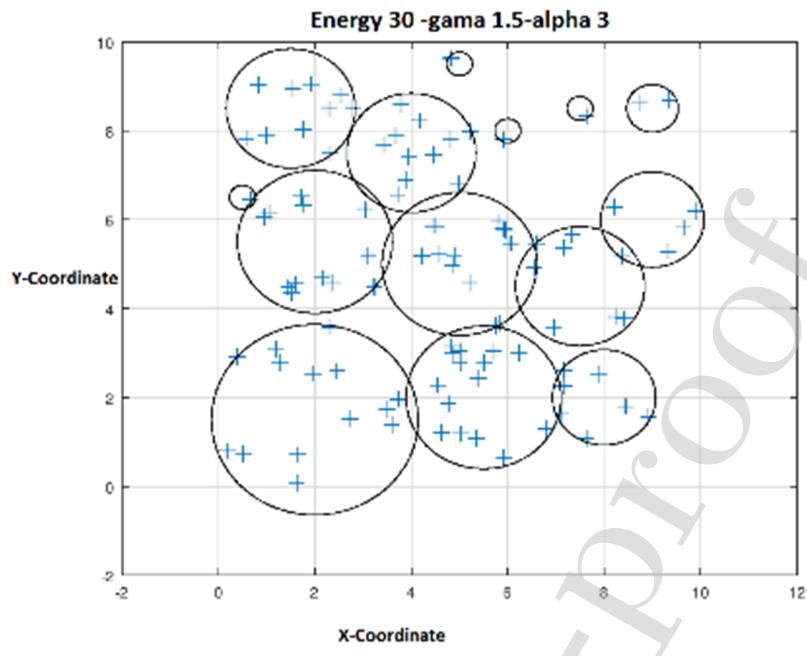


Figure 3 - Energy = 30, gamma = 1.5, alpha = 3.

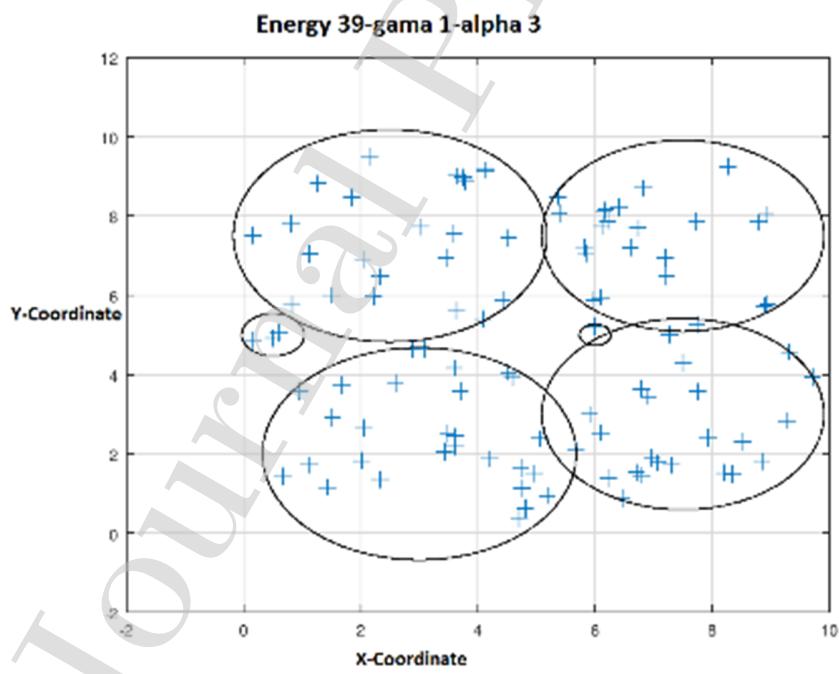


Figure 4 - Energy = 39, gamma = 1, alpha = 3.

More optimal results can be reached by increasing the battery capacity while maintaining the cost of flying horizontally and vertically. Figure 4 shows the least number of drones with wider visibility range than the ones on Figure 5, where the drones' visibility range is smaller. Drones that are flying further away from the hub (zero coordinates) have less visibility range by the time they reach their destination. This is because traveling further diagonally consumes more energy than flying close to the hubs (due to the short distance traveled). Therefore, this leads them to decrease their elevation and their visibilities simultaneously.

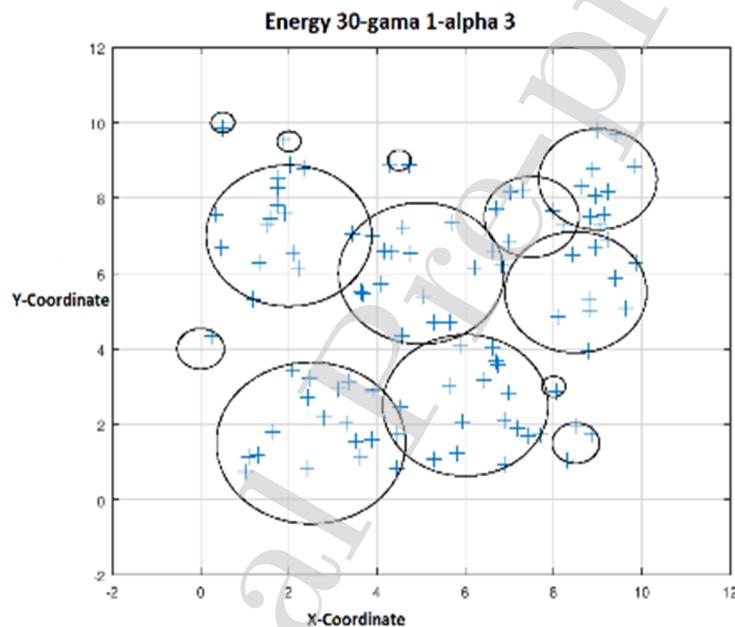


Figure 5 - Energy = 30, gamma = 1, alpha = 3.

In the second scenario regarding static targets, we changed the number of targets (nt) and kept the following energy-related parameters constant; $E = 40$, gamma = 2, and alpha = 4. We then observed the changes in behavior. Figures 6 – 10 represent target coverage by drones with respect to the number of drones. Figure 6 shows three targets in different locations, since they are away from each other; three drones are needed to cover them. If the number of targets is x , then the number

of drones can have a value ranging between a minimum of 1 and a maximum of x . As the number of targets increases, the number of drones increases as seen in Figures 7 – 10.

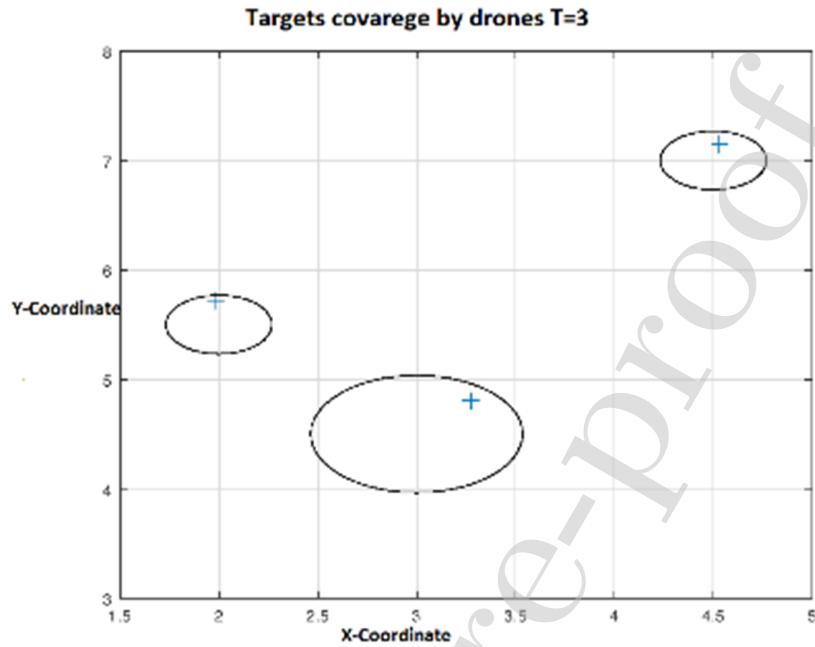


Figure 6 - Targets coverage $nt = 3$.

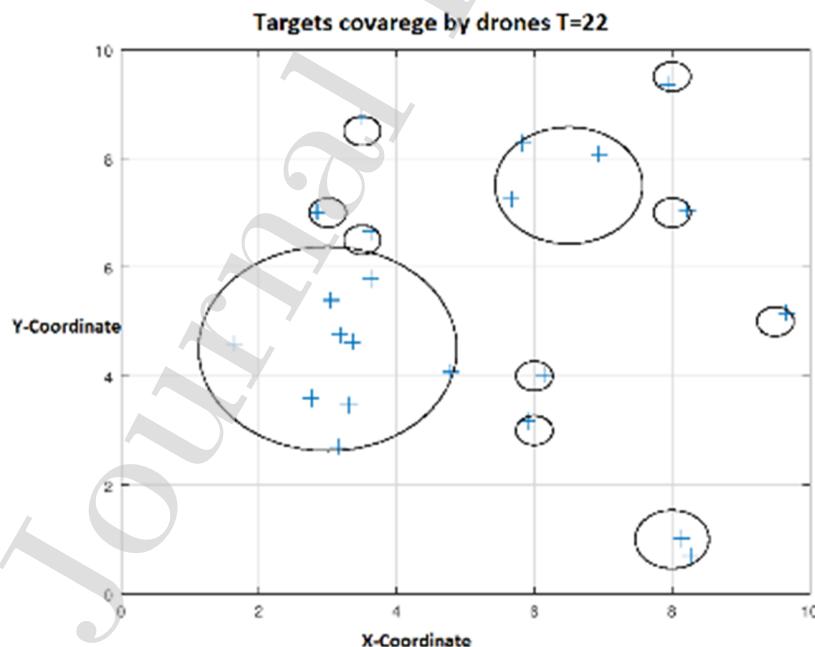


Figure 7 - Targets coverage $nt = 22$.

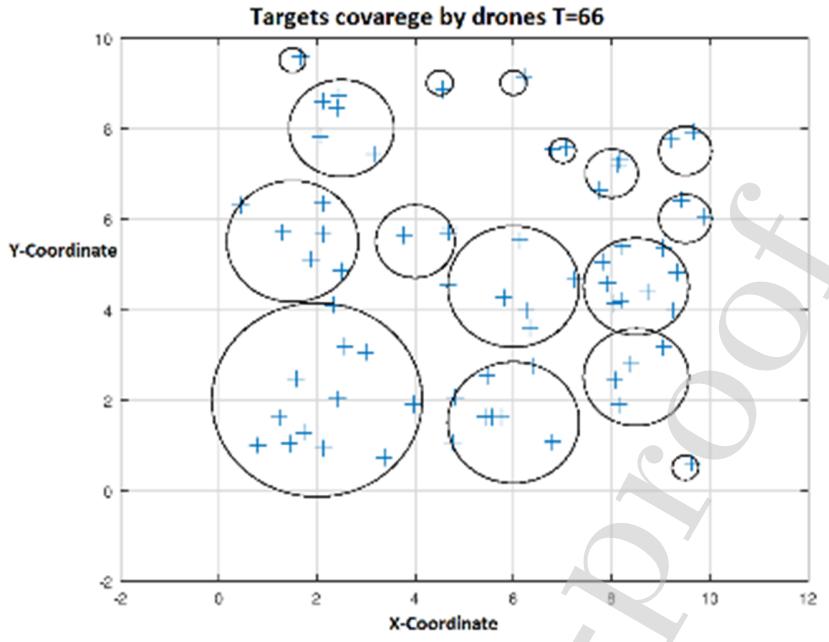


Figure 8 - Targets coverage $nt = 66$.

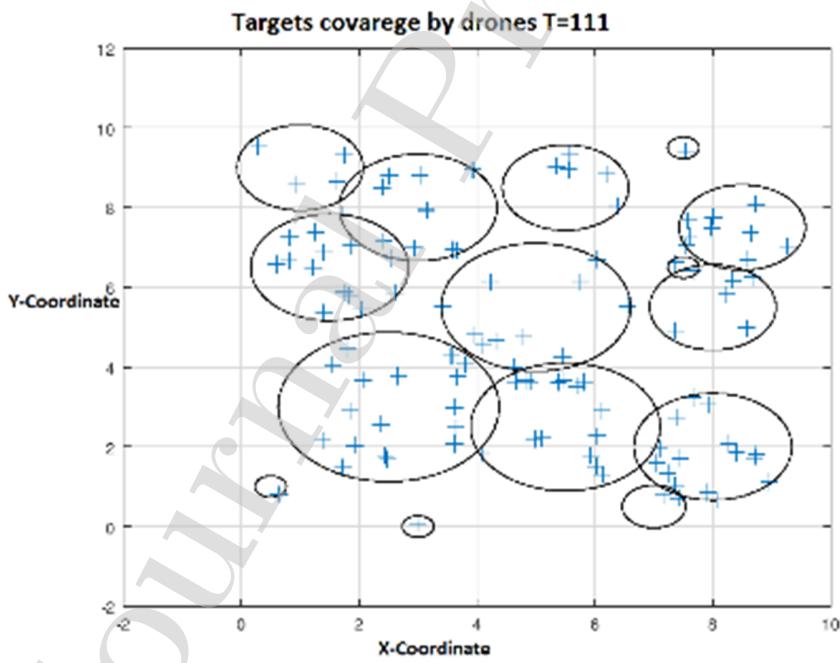


Figure 9 - Targets coverage $nt = 111$.

Figure 11 shows the number of drones with respect to the number of targets. As it is expected, the number of drones needed to monitor a given number of targets increases gradually. However, this

is only true, until a certain number of targets. This number of targets requires the highest number of drones to be monitored. It can be thought to cover several varying regions across the total area being monitored. Later, if the number of targets is further increased, the number of drones does not increase.

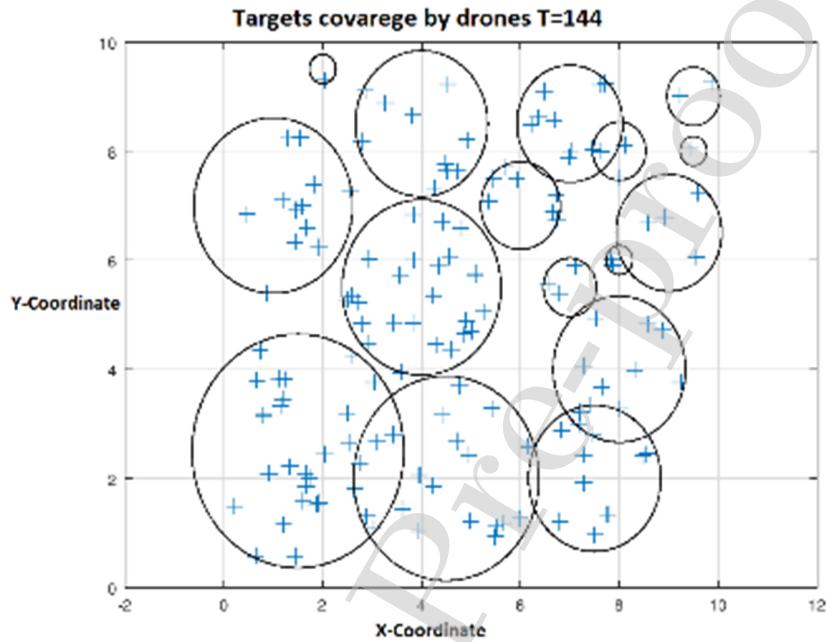


Figure 10 - Targets coverage $nt = 144$.

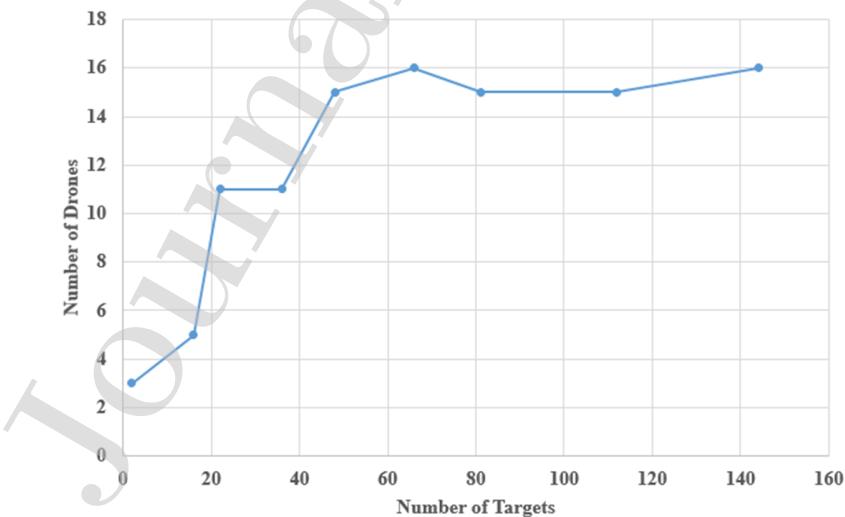


Figure 11 - The relationship between the number of drones and targets.

4.3.2 Dynamic Targets

So far, we have assumed that the targets are static (not moving). However, in this scenario, we assume that the targets are capable of moving throughout the time they are being monitored. To display this effect in the simulation, the $seqLength$ (S_L) and $walkArea$ (W_A) parameters change. As explained earlier, S_L is the number of time intervals in which the target moves (i.e. how many times the target will move), and W_A is how far the target moves within every time interval. For example, if $S_L = 2$ and $W_A = 3$, then each target will move randomly twice, each with three steps. We set the energy-related parameters as $E = 40$, $\gamma = 2$, and $\alpha = 4$ to observe the changes. The script is written in such a way that if the target is static it draws a plus (+) sign, otherwise, it draws a minus (-) sign, and displays the trajectories of movement of the drones. Figure 12 shows static and non-moving targets, whereas in Figure 13, the targets are moving slightly. The visibility range of drones is slightly overlapping. This is due to the fact that the drones are trying to cover the moving targets.

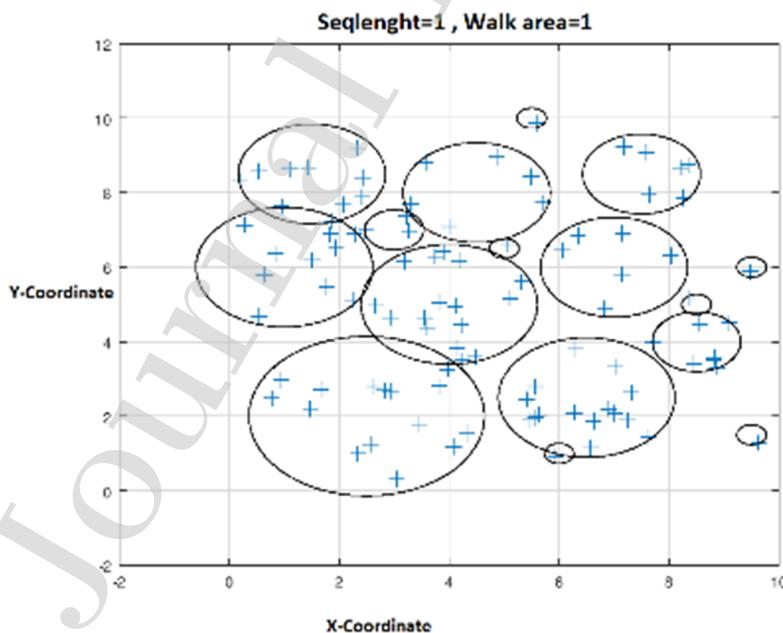


Figure 12 - Targets coverage with $S_L = 1$, $W_A = 1$.

In Figure 14, each target moved five times, each time with five steps, where each colored segment shows the trajectory of a certain target. As it can be seen in Figure 14, the targets are moving a lot more than the previous ones. Therefore, the overlapping region between the drones' visibility range is higher. The reason for this overlapping is the long trajectories of the targets, where they get too close to each other.

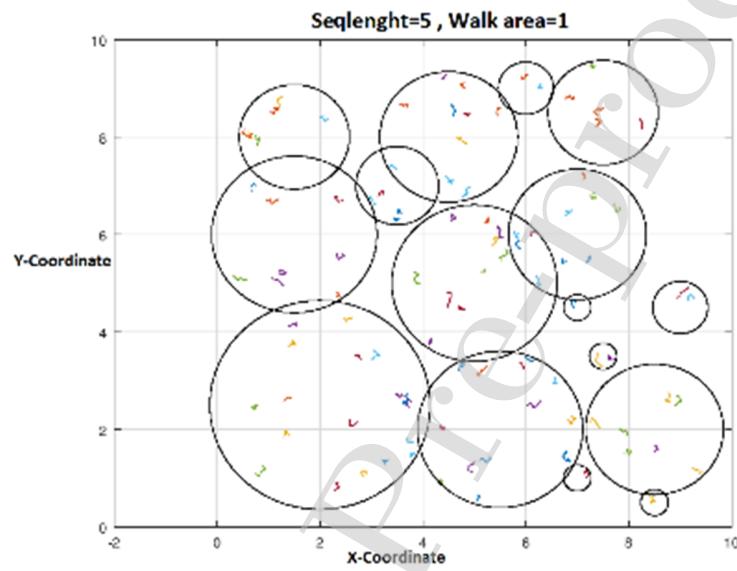


Figure 13 - Targets coverage with $S_L=5$, $W_A = 1$.

Observing the change in the target's behavior and the number of targets with respect to S_L and W_A parameters, we notice that the longer path the trajectory target takes, the closer the targets get to each other. Therefore, less number of drones is needed to cover them. Figure 15 shows how the number of drones decreases as the trajectory length increases. The trajectory length (T_J) was calculated using the following equation:

$$T_J = W_A * S_L \quad (12)$$

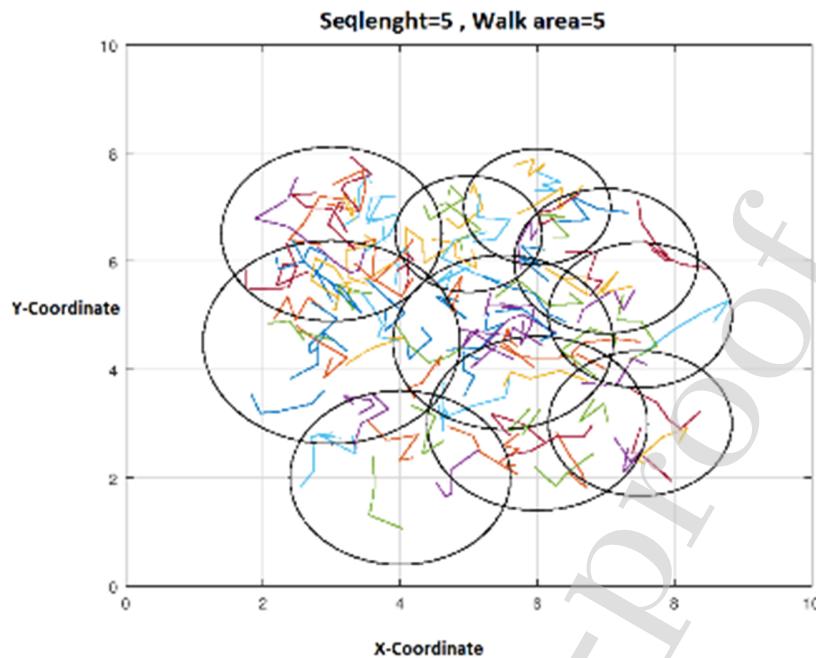


Figure 14 - Targets coverage with $S_L = 5$, $W_A = 5$.

As it can be seen in Figure 15, the curve clearly displays the inverse relationship between the number of drones and the corresponding lengths of the trajectories of targets. This is clearly observed as the highest number of drones corresponds to the lowest trajectory length and vice versa.

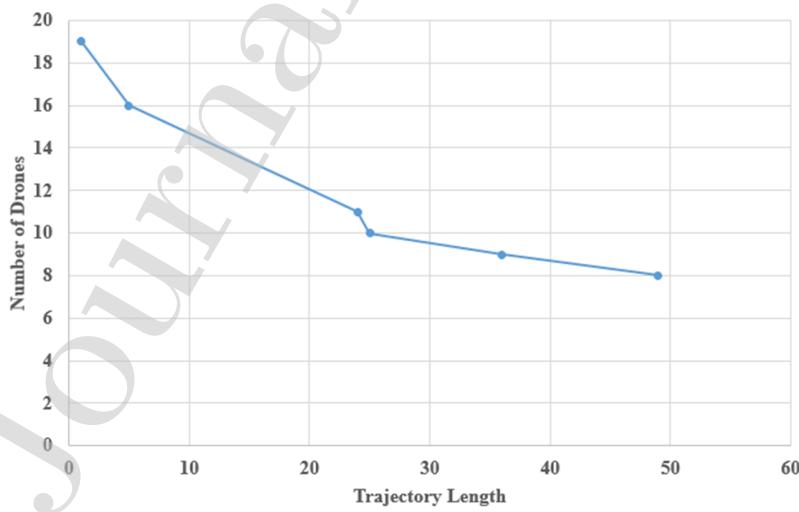


Figure 15 - The relationship between trajectory length and the number of drones.

5. Conclusion

In this study, a cost minimization problem is considered which is related to the optimal placement of drones to monitor a set of static or dynamic targets. Our minimization problem aims to minimize the number of drones, given a constant value of battery capacity. The problem stated earlier is formulated, and the mathematical models were provided accordingly. The simulation results obtained from different variations in changing the parameters reveal that increasing the battery capacity leads to an increase in the drone's visibility range, and thus, a decrease in the number of drones. This effectively provides a better solution for our minimization problem. Moreover, when dynamic targets are considered, moving with higher W_4 leads to targets ending up in locations close to each other. In actuality, almost an inversely proportional linear relation exists, as can be seen in Figure 15. Therefore, the drones' visibility areas will be overlapping, which may cause a number of drones to be considered as redundant, leading to a smaller number of drones. Finally, there exists a limit where the number of drones no longer proportionally increases in relation to the number of targets. This is because the limit exhibits a case where the targets are distributed across a large number of different regions in the area monitored, rendering a further increase to the targets that does not require an increase in the number of drones needed to monitor them.

References

1. R. Sánchez-Corcuera, A. Nuñez-Marcos, J. Sesma-Solance, A. Bilbao-Jayo, R. Mulero, U. Zulaika, G. Azkune, and A. Almeida, "Smart cities survey: Technologies, application domains and challenges for the cities of the future", *International Journal of Distributed Sensor Networks*, 15(6), p.1550147719853984, 2019.

2. Y. Liu, B. Bai, and C. Zhang, UAV image mosaic for road traffic accident scene, in *32nd IEEE Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, May 2017, pp. 1048–1052.
3. J. S. Álvares, D.B. Costa, and R. R. S. D. Melo, Exploratory study of using unmanned aerial system imagery for construction site 3D mapping, *Construction Innovation*, vol. 18, 301–320 2018.
4. M. Micheletto, V. Petrucci, R. Santos, J. Orozco, D. Mosse, S. F. Ochoa, and R. Meseguer, Flying real-time network to coordinate disaster relief activities in urban areas, *Sensors*, vol. 18, no. 5, 1662, 2018.
5. L. Zhu, F. R. Yu, Y. Wang, B. Ning, and T. Tang, Big data analytics in intelligent transportation systems: A survey, *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, 383–398, 2018.
6. R. Tariq, M. Rahim, N. Aslam, N. Bawany, and U. Faseeha, DronAID: A smart human detection drone for rescue, In *15th IEEE International Conference on Smart Cities: Improving Quality of Life Using ICT & IoT (HONET-ICT)*, October 2018, pp. 33–37.
7. A. Gao, S. Wu, F. Wang, X. Wu, P. Xu, L. Yu, and S. Zhu, A newly developed unmanned aerial vehicle (UAV) imagery based technology for field measurement of water level, *Water*, vol. 11, no. 1, 124, 2019.
8. V. Sharma, G. Choudhary, Y. Ko, and I. You, Behavior and vulnerability assessment of drones-enabled industrial internet of things (IIoT), *IEEE Access*, vol. 6, 43368–43383, 2018.
9. J. C. Padró, V. Carabassa, J. Balagué, L. Brotons, J. M. Alcañiz, and X. Pons, Monitoring opencast mine restorations using Unmanned Aerial System (UAS) imagery, *Science of the Total Environment*, vol. 657, 1602–1614, 2019.

10. N. Al Amir, A. Marar, and M. Saeed, Eye in the Sky: How the Rise of Drones will Transform the Oil & Gas Industry. in Abu Dhabi International Petroleum Exhibition & Conference. *Society of Petroleum Engineers*, November 2018.
11. J. Seo, L. Duque, and J. Wacker, Drone-enabled bridge inspection methodology and application, *Automation in Construction*, vol. 94, 112–126, 2018.
12. J. Xia, K. Wang, and S. Wang, Drone scheduling to monitor vessels in emission control areas, *Transportation Research Part B: Methodological*, vol. 119, 174–196, 2019.
13. V. K. Shetty, M. Sudit, and R. Nagi, Priority-based assignment and routing of a fleet of unmanned combat aerial vehicles, *Computers & Operations Research*, vol. 35, no.6, 1813–1828, 2008.
14. C. C. Murray, and M. H. Karwan, An extensible modeling framework for dynamic reassignment and rerouting in cooperative airborne operations, *Naval Research Logistics (NRL)*, vol. 57 no. 7, 634–652, 2010.
15. F. Mufalli, R. Batta, and R. Nagi, Simultaneous sensor selection and routing of unmanned aerial vehicles for complex mission plans, *Computers & Operations Research*, vol. 39, no. 11, 2787–2799, 2012.
16. Y. Xia, R. Batta, and R. Nagi, Controlling a fleet of unmanned aerial vehicles to collect uncertain information in a threat environment, *Operations Research*, vol. 65, no. 3, 674–692, 2017.
17. C. C. Murray, and A. G. Chu, The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery, *Transportation Research Part C: Emerging Technologies*, vol. 54, 86–109, 2015.
18. X. Wang, S. Poikonen, and B. Golden, The vehicle routing problem with drones: Several worst-case results, *Optimization Letters*, vol. 11, no. 4, 679–697, 2017.

19. J. G. Carlsson, and S. Song, Coordinated logistics with a truck and a drone, *Management Science*, vol. 64, no. 9, 4052–4069, 2017.
20. A. Pulver, and R. Wei, Optimizing the spatial location of medical drones, *Applied Geography*, vol. 90, 9–16, 2018.
21. W. Chung, V. Crespi, G. Cybenko, and A. Jordan, Distributed sensing and UAV scheduling for surveillance and tracking of unidentifiable targets, in Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense IV, *International Society for Optics and Photonics*, vol. 5778, May 2005, pp. 226–236.
22. S. Simi, R. Kurup, and S. Rao, Distributed task allocation and coordination scheme for a multi-UAV sensor network, in *Tenth IEEE International Conference on Wireless and Optical Communications Networks (WOCN)*, July 2013, pp. 1–5.
23. J. Kim, and Y. Kim, Moving ground target tracking in dense obstacle areas using UAVs, *IFAC Proceedings Volumes*, vol. 41, no. 2, 8552–8557, 2008.
24. F. Guerriero, R. Surace, V. Loscri, and E. Natalizio, A multi-objective approach for unmanned aerial vehicle routing problem with soft time windows constraints, *Applied Mathematical Modelling*, vol. 38, no. 3, 839–852, 2014.
25. D. Zorbas, L. D. P. Pugliese, T. Razafindralambo, and F. Guerriero, Optimal drone placement and cost-efficient target coverage, *Journal of Network and Computer Applications*, vol. 75, 16–31, 2016.
26. D. Zorbas, T. Razafindralambo, and F. Guerriero, Energy efficient mobile target tracking using flying drones, *Procedia Computer Science*, vol. 19, 80–87, 2013.

27. L. D. P. Pugliese, F. Guerriero, D. Zorbas, and T. Razafindralambo, Modelling the mobile target covering problem using flying drones, *Optimization Letters*, vol. 10, no. 5, 1021–1052, 2016.
28. E. Yanmaz, S. Yahyanejad, B. Rinner, H. Hellwagner, and C. Bettstetter, Drone networks: Communications, coordination, and sensing, *Ad Hoc Networks*, vol. 68, 1–15, 2018.
29. GNU Octave, Available at: <https://gnu.org/software/octave/>.
30. A. Fotouhi, H. Qiang, M. Ding, M. Hassan, L.G. Giordano, A. Garcia-Rodriguez, and J. Yuan, Survey on UAV cellular communications: Practical aspects, standardization advancements, regulation, and security challenges. *IEEE Communications Surveys & Tutorials*, 2019.
31. F. Al-Turjman, J.P. Lemayian, S. Alturjman, and L. Mostarda, “Enhanced Deployment Strategy for the 5G Drone-BS Using Artificial Intelligence”, *IEEE Access*, 7, pp.75999-76008, 2019.
32. F. Al-Turjman, “A novel approach for drones positioning in mission critical applications”, *Transactions on Emerging Telecommunications Technologies*, p.e3603, 2019.
33. F. Al-Turjman, and S. Alturjman, “5G/IoT-enabled UAVs for multimedia delivery in industry-oriented applications”, *Multimedia Tools and Applications*, pp.1-22, 2018.
34. E. Tuba, R. Capor-Hrosik, A. Alihodzic, and M. Tuba, “Drone placement for optimal coverage by brain storm optimization algorithm”, In *International Conference on Health Information Science*, Springer, Cham, pp. 167-176, December 2017.
35. H. Shakhatreh, and A. Khreishah, “Optimal Placement of a UAV to maximize the lifetime of wireless devices”, In *14th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pp. 1225-1230, June 2018.

36. E. Kalantari, H. Yanikomeroglu, and A. Yongacoglu, “On the number and 3D placement of drone base stations in wireless cellular networks”, In *84th IEEE Vehicular Technology Conference (VTC-Fall)*, pp. 1-6, September 2016.
37. M. Alzenad, A. El-Keyi, and H. Yanikomeroglu, “3-D placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements”, *IEEE Wireless Communications Letters*, 7(1), pp.38-41, 2017.
38. A. Al-Habashna, and G. Wainer, “QoE awareness in progressive caching and DASH-based D2D video streaming in cellular networks”, *Wireless Networks*, 2019.
39. Y. Wang, Y. Chen, J. Choi, and C.C.J. Kuo, “Towards Visible and Thermal Drone Monitoring with Convolutional Neural Networks”, *APSIPA Transactions on Signal and Information Processing*, 8, 2019.
40. S. Otoum, B. Kantarci, and H.T. Mouftah, “On the feasibility of deep learning in sensor network intrusion detection”, *IEEE Networking Letters*, 1(2), pp.68-71, 2019.
41. S. Otoum, B. Kantarci, and H. Mouftah, “Empowering Reinforcement Learning on Big Sensed Data for Intrusion Detection”, In IEEE International Conference on Communications (ICC), pp. 1-7, May 2019.
42. Q. Zhang, W. Saad, M. Bennis, X. Lu, M. Debbah, and W. Zuo, “Predictive Deployment of UAV Base Stations in Wireless Networks: Machine Learning Meets Contract Theory”, *arXiv preprint arXiv:1811.01149*, 2018.
43. Y. Yayeh Munaye, H.P. Lin, A.B. Adege, and G.B. Tarekegn, “UAV Positioning for Throughput Maximization Using Deep Learning Approaches”, *Sensors*, 19(12), p.2775, 2019.

44. C. Ruiz-Martin, A. Al-Habashna, G. Wainer, and L. Belloli, “Control of a quadcopter application with DEVS”, In *Proceedings of the Theory of Modeling and Simulation Symposium*, Society for Computer Simulation International, p. 4, April 2019.

The authors declare that they have no conflict of interest.

Journal Pre-proof



An Optimized Communication Scheme for Energy Efficient and Secure Flying Ad-hoc Network (FANET)

Mayank Namdev¹ · Sachin Goyal² · Ratish Agarwal²

Accepted: 13 April 2021 / Published online: 24 April 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

FANET (flying ad-hoc network) has provided broad area for research and deployment due to efficient use of the capabilities of drones and UAVs (unmanned aerial vehicles) in several military and rescue applications. Drones have high mobility in 3D (3 dimensional) environment and low battery power, which produce various problems such as small journey time and infertile routing. The optimal routing for communication will assist to resolve these problems and provide the energy efficient and secure data transmission over FANET. Hence, in this paper, we proposed a whale optimization algorithm based optimized link state routing (WOA-OLSR) over FANET to provide optimal routing for energy efficient and secure FANET. The efficiency of OLSR is enhanced by using WOA and evaluated performance shows the better efficiency of WOA-OLSR in terms of some parameters such as a packet delivery ratio, end to end delay, energy utilization, throughput, and time complexity against the previous approaches OLSR, MP-OLSR, P-OLSR, ML-OLSR-FIFO and ML-OLSR-PMS.

Keywords Drones · FANET · Whale Optimization · UAVs · OLSR · Throughput · Optimal Routing

1 Introduction

The multifarious and rising UAVs [1, 2] are utilized for the feasible and consistent communication over FANET. A multifarious system structure is used to provide a UAV abundant network communicating with various functional applications such as military, rescue, and disaster management on the basis of several topological investigations [3]. UAVs can be seen as a future of IoT (internet of things) communication in which 5G technology is used to enhance the performance of audio and video data transmission. The problems of UAVs like privacy and transmission volatility are diminished by using multiple UAVs depend on on-demand routing over FANET [4]. The transmission cost and capriciousness of the UAV is enhanced due to packet collision in 3D environment. The packet collisions are reduced

✉ Mayank Namdev
mayank.namdev@gmail.com

¹ Department of CSE, University Institute of Technology, RGPV, Bhopal 462023, India

² Department of IT, University Institute of Technology, RGPV, Bhopal 462023, India

by developing a decentralized 3D-SWAP approach in a simulator with unrestrained conditions such as strident positioning and breeze gusts [5]. UAV's ecological information confirms the collection and communication of data packets for exploration and rescue processing in delay tolerant networks (DTN) concept [6]. Proactive coverage areas (PCAs) based DBS (drone base station) is developed to perform dynamic load processing on the basis of supply demand. The exposure region and elevation of DBS is obtained minimum energy utilization and momentous complexity diminution [7]. Drones are a unique kind of UAVs with higher velocity and dynamic environment utilizing for moving desired object detection in the minimum time interval [8]. The wireless sensor network (WSN) is integrated with the help of UAVs to enhance the area of network and overall performance. The clustering is processed with cluster heads and members selection for route distance and transmission time generation in WSN [9].

Efficient clustering also provides the hierarchical routing to diminish the overhead of mobile ad-hoc network (MANET) [10]. The offloading approach is used in drones to obtain privacy and minimum time utilization for data packet transmission over [11, 12] based clustering network (GWOCNET) based on grey wolf nature and node's mobility in vehicular ad-hoc network (VANET). The outputs describe the better efficiency of GWOCNET against CLPSO (comprehensive learning particle swarm optimization) and MOPSO (Multi Objective PSO) [13]. K-Means is one of the popular clustering methods based on dynamic transmission ranges and applied on energy utilization of UAVs. The output shows the better performance of ECRNET (energy based clustering for network) based on number of clusters, cluster lifetime and creation time and power utilization versus CACONET (ant colony optimization based clustering for network) and GWOCNET approaches [14]. Numerous routing protocols [15, 16], which are responsible for information exchanging, path generation and secure communication, are developed for FANETs in the recent past years. The unpredictable and arbitrarily velocity of UAV are general challenges of FANET. These issues should be reduced by using proper routing between UAV through optimization approach [17].

The FANET [18] is a scalable and a feasible communication network among high velocity drones (UAVs) [8, 19] in the 3 D structure. Routing is a most significant issue in FANET and most of the routing problems are resolved by using different topological architectures enhancing the throughput, and packet delivery ratio on OPNET (optimized network engineering tools) platform [20]. Routing is performed on the basis of AODV (ad-hoc on-demand distance vector) and OLSR (optimized link state routing) routing protocols for calculating the shortest paths from sources to destinations over FANET for higher packet delivery ratio (PDR) against number of UAVs, velocity and transmission ranges [21]. Another routing protocol ADRP (Adaptive density based routing protocol) is performed the qualitative [22] and effective data transmission over FANET. ADRP is implemented on a NS-2 simulator and the results are analyzed on the basis of the packet delivery ratio, and end to end delay against the AODV. Multi path OLSR (MP-OLSR) is also used for efficient routing among UAVs [23]. It is an enhanced version of OLSR which is capable

of exchanging fire exposure data from GPS through a Fog or Edge environment for security and disaster management [24].

A stochastic model is combined with dynamic delay inhibited routing in which sender transmits the data packets utilizing local information only [25]. The routing of video and audio transmission has utilized large energy for path failures and network extrication. UAVs are used to charge the IoT things in less time slice maximum possible energy. Wireless technology is used to charge and recharge the IoT devices [26]. The ground and flying ad hoc network are combined to form an ambient network G-FANET which is used neural, genetic and fuzzy strategies to perform complex tasks. The feedback and learning rate of network is calculated for UAVs in terms of simulation time [27]. P-OLSR (predictive OLSR) is used to improve the quality of wireless link based on speed, direction and location of UAVs [28]. A fluid topology is implemented for selecting the neighbours of UAVs utilizing a metric SW-ETX (Speed Weighted Expected Transmission Count). Another improvement in OLSR, called ML-OLSR (mobility and load aware OLSR) [29], is introduced which selects the minimum mobility nodes and reduces the transmission loads and delay with enhancing the packet delivery. The performance of ML-OLSR is enhanced by using FIFO (first in first out) (ML-OLSR-FIFO) and PMS (prioritizing based message scheduling) (ML-OLSR-PMS) for providing reliable and flexible message transmission. The ML-OLSR-PMS approach combines the message priority assignment and scheduling modules in which message priority is decided, after that message is rescheduled in a desired output row [29].

In above survey, several approaches like OLSR, MP-OLSR, P-OLSR, ML-OLSR-FIFO, and ML-OLSR-PMS are introduced to perform optimal routing. The best drones are selected for routing on the basis of some parameters like energy utilization, mobility, and neighbourhood degree, but all the above proposed approaches is not utilized all the parameters together. The security is also not considered proper in proposed approaches. Hence, we proposed a whale optimization algorithm on optimized link state routing (WOA-OLSR) in which WOA is applied to multi objective function combining the several parameters such as neighbourhood benefaction, energy, stability time and key utilization of drones. The results are compared against previous approaches OLSR, MP-OLSR, P-OLSR, ML-OLSR-FIFO and ML-OLSR-PMS on the basis of packet delivery ratio, throughput, and end to end delay and energy utilization.

The remaining paper is as follows. The 2nd part illustrates the WOA approach mathematically with algorithm. The 3rd part describes the proposed WOA-OLSR in brief utilizing flowchart and algorithms. The 4th part illustrates the results obtained from the MATLAB simulation of WOA-OLSR. The 5th part explains conclusion of research paper.

2 Whale Optimization Algorithm (WOA)

The WOA is a freshly anticipated optimization technique imitating humpback whales the hunting procedure and behaviour. Whales are very intellectual emotive animal due to spindle cells and survive unaccompanied or in groups. The bigger humpback whale is as long

as a school bus and prey the krill and tiny fish crowd. The hunting mechanism of humpback whales are also known as bubble net feeding, which is mathematically explained to perform optimization.

2.1 Encircling Prey

The preys positions can be discriminated by Humpback whales and enclose them. WOA supposes that the present best solution (search agent) represents as target prey and other search agents can be modified their locations to optimal solution. The activities are illustrated by the equations (Eqs. (1) and (2)).

$$\vec{D} = \left| \vec{C} \cdot \vec{X}_p(t) - \vec{X}(t) \right| \quad (1)$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \quad (2)$$

where t =present iteration, \vec{X}_p =prey position vector, \vec{X} =Whale position vector, \vec{D} =distance between prey and position of whale and \vec{A}, \vec{C} are coefficient vector calculated by using Eq. (3) and (4).

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \quad (3)$$

$$\vec{C} = 2\vec{r}_2 \quad (4)$$

Here the coefficients of \vec{a} are linearly reduced from 2 to 0 and \vec{r}_1, \vec{r}_2 indicates the random value vectors in $[0, 1]$.

2.2 Bubble-Net Feeding Mechanism (Exploitation State)

It describes two approaches as follows:

- I. Shrinking encircling method: This is done by reducing the value of \vec{a} . The variation assortment of \vec{A} is also reduced by \vec{a} to generate the value in interval $[-a, a]$. Assuming the values of \vec{A} in $[-1, 1]$, the new location of whale (search agent) can be explained somewhere in between actual location of the agent and location of the present best agent.
- II. Spiral modifying location: Initially it evaluates the location of prey (\vec{X}^*, \vec{Y}^*) and whale (\vec{X}, \vec{Y}). After that a spiral equation is generated between location of prey and whale to imitate the humpback whales helix wrought movement (Eq. (5)).

$$\vec{X}(t+1) = \vec{D}' \cdot e^{bt} \cos(2\pi t) + \vec{X}^*(t) \quad (5)$$

Here $\vec{D}' = |\vec{X}^*(t) - \vec{X}(t)|$ which denotes the distance between i^{th} whale and current best prey, $b=$ constant for explaining the logarithmic spiral shape and $t=$ arbitrary number in interval $[-1, 1]$.

The humpback whales spin in the region of prey inside a flinch circle and beside a spiral-shaped route concurrently. Therefore, we supposed that whales are modifying their locations during optimization with 50% probability of selection flinch circle or spiral model (Eq. 6).

$$\vec{X}(t+1) = \begin{cases} \vec{X}^*(t) - \vec{A} \cdot \vec{D} & p < 0.5 \\ \vec{D}' e^{bt} \cos(2\pi t) + \vec{X}^*(t) & p \geq 0.5 \end{cases} \quad (6)$$

Here $p=$ arbitrary number in $[0, 1]$.

The humpback whales are explored for prey arbitrarily as follows:

2.3 Prey Exploration

\vec{A} is used to explore the prey and random exploration is utilized by whales for finding the locations of each other. Hence \vec{A} is utilized (with arbitrary numbers < -1 & > 1) for exploration and $|\vec{A}| > 1$ is used for global search (Eqs. (7) and (8)).

$$\vec{D} = |\vec{C} \cdot \vec{X}_{rand} - \vec{X}| \quad (7)$$

$$\vec{X}(t+1) = \vec{X}_{rand} - \vec{A} \cdot \vec{D} \quad (8)$$

Here $\vec{X}_{rand}=$ arbitrary location vector (arbitrary whale).

The WOA initiates by utilizing a group of arbitrary solutions. In every repetition, whales (search agents) modify their locations according to arbitrary select whale or the best optimal whale. The value of parameter \vec{a} is reduced from 2 to 0 instructing to obtain exploration and exploitation. If $|\vec{A}| > 1$, then arbitrary whale (search agent) is selected and the best solution is obtained for $|\vec{A}| < 1$ to modify the location of whales.

The WOA (algorithm 1) is described under the input parameters $N^i=$ number of iterations, $P^w=$ Whales population, $N^w=$ number of whales and output optimal whale (drone) (X^*).

Algorithm 1. WOA Algorithm	Number of Operations
1. START	
2. WHILE concluding condition is not satisfied	$(N^i + 1)$
3. Given initial population values of whales (drones) $X_i (i = 1, 2, 3, \dots, N_d)$	
4. Given initial values of $a, A, l, c, p & N^i$	
5. Calculate the fitness of all whales	$N^i * P^w$
6. X^* = The best whale (search agent)	$N^i * N^w$
7. WHILE ($i < N^i$)	$(N^i + 1)$
8. FOR all whales (search agents)	$N^i * (N^w + 1)$
9. If ($p < 0.5$)	$N^i * N^w$
10. If ($ A < 1$)	$N^i * N^w$
11. Modify the location of present whale (search agent) (eq. (1) (2) (3) & (4))	$N^i * N^w$
12. Else if ($ A \geq 1$)	$N^i * N^w$
13. Choose an arbitrary search agent (\vec{X}_{rand})	$N^i * N^w$
14. Modify the location of present whale (search agent) (eq. (7) & (8))	$N^i * N^w$
15. END IF	
16. Else if ($p \geq 0.5$)	$N^i * N^w$
17. Modify the location of the present whale (search agent) (eq. (5))	$N^i * N^w$
18. END IF	
19. END FOR	
20. Calculate the fitness of all whales	$N^i * N^w$
21. Modify X^* if there exists a better location	$N^i * N^w$
22. $i = i + 1$	N^i
23. Modify $a, A, l, c & p$	N^i
24. END WHILE	
25. END WHILE	
26. Return X^*	N^w .
27. STOP	

3 Whale Optimization Algorithm on Optimized Link State Routing (WOA-OLSR)

The FANET is explained as a combination of drones, which are communicated with each other in a network area. Drones have a battery with fixed energy and moved into a 3D environment for data transmission. In our methodology, if a drone wants to send a packet to another neighbouring drone, then both of them have same key using for data packet encryption between them. It means a drone is used a single key for a single packet encryption and if drone want to send more than one packet, then it uses more than one key (one key for one packet). It enhances the security of data transmission through drones over FANET. The Whale Optimization Algorithm (WOA) is utilized to perform the optimal

routing with optimized link state routing (OLSR) in FANET. WOA determines the optimal drones on OLSR routing with the help of the final significant weight of all drones on the basis of some parameters such as Drone's Neighbourhood Benefaction, Drone's Stability Time, Drone's Energy and Drone's Key Utilization. After that optimal drones have been utilized for maximum packet transmission to perform optimal routing over FANET.

3.1 System Model

The WOA-OLSR is performed in following steps:

3.1.1 Drone's Neighbourhood Benefaction

Neighbourhood benefaction of each drone is evaluated by obtaining the number of neighbour drones existing in every drone's transmission range over FANET. If a drone has a large number of neighbour drones, then it is selected for maximum packet transmission with minimum energy consumption and negligible packet drop. Drone's Neighbourhood Benefaction (DNB) is obtained by using Eqs. (9) and (10).

$$N(d_a) = \{d_b, \exists \text{distance}(d_a, d_b) < TR_{d_a}\} \quad (9)$$

$$DNB(d_a) = |N(d_a)| \quad (10)$$

where d_a and $d_b = a^{\text{th}}$ and b^{th} drone, $\text{distance}(d_a, d_b)$ =distance between d_a and d_b , TR_{d_a} =transmission range of drone d_a , $|N(d_a)|$ =Number of neighbours of drone d_a , and $DNB(d_a)$ =DNB value of drone d_a .

3.1.2 Drone's Stability Time

Drone stability time (DST) is also related to the drone's mobility. The drones are connected with other drones by several links and the estimated time of these links durations are used to obtain the stability time of each drone in an OLSR routing over FANET. It shows that larger the DST provides lesser the drone mobility with optimal routing. Firstly, the estimation time of link (ETL) is calculated by using Eq. (11), after that DST is evaluated by using Eq. (12).

$$ETL(d_a, d_b) = \frac{-(pq + rs) + \sqrt{(p^2 + r^2)(TR)^2 - (ps - qr)^2}}{p^2 + r^2} \quad (11)$$

where $ETL(d_a, d_b)$ =ETL between a^{th} and b^{th} drone, $p = S_{d_a} \cos \theta_{d_a} - S_{d_b} \cos \theta_{d_b}$, $q = d_{x_a} - d_{x_b}$, $r = S_{d_a} \sin \theta_{d_a} - S_{d_b} \sin \theta_{d_b}$, $s = d_{y_a} - d_{y_b}$ and TR =Transmission Range, S_{d_a} & S_{d_b} =Speed of a^{th} and b^{th} drone in the direction of θ_{d_a} and θ_{d_b} , $d_{x_a}, d_{x_b}, d_{y_a}, d_{y_b}$ = x^{th} and y^{th} component of a^{th} and b^{th} drone.

$$DST(d_a) = \frac{\sum_{d_b=1}^{DNB(d_a)} ETL(d_a, d_b)}{DNB(d_a)} \quad (12)$$

where $DST(d_a)$ =DST of a^{th} drone.

3.1.3 Drone's Energy

A drone has optimally utilized for OLSR routing having maximum remaining energy over FANET. Drone's Energy (DE) is calculated as a ratio of remaining energy to the DNB value of a drone at a particular time over FANET (Eq. (13)).

$$DE(d_a) = \frac{\text{Remaining_Energy}(d_a)}{\text{DNB}(d_a)} \quad (13)$$

where $\text{Remaining_Energy}(d_a)$ =remaining energy of a^{th} drone and $DE(d_a)$ =DE of a^{th} drone.

3.1.4 Drone's Key Utilization

A drone has maximum number of keys utilizing for optimal OLSR routing. A single key is used to encrypt a single packet among drones over FANET. Different keys are used to encrypt the different packets between two drones. So Drone's Key Utilization (DKU) is calculated by using Eqs. (14) and (15).

$$DKU(d_a) = \begin{cases} \sum_{k=1}^K DK_{d_a,k} & \text{If } \exists \text{keys } \in d_a \\ 0 & \text{Otherwise} \end{cases} \quad (14)$$

where

$$DK_{d_a,k} = \begin{cases} 1 & \text{If } K_k \in d_a \\ 0 & \text{Otherwise} \end{cases} \quad (15)$$

Here $DK_{d_a,k}$ =drone key relation which denotes the relation between a^{th} drone and k^{th} key, $K_k=k^{th}$ key from key group K, $DKU(d_a)$ =DKU value of a^{th} drone.

3.1.5 Drone's Final Significant Weight

Drone's Final Significant Weight (DFSW) is obtained for each drone for OLSR routing over FANET by combining all the four parameters (DNB, DST, DE and DKU) to generate a multi objective function by Eq. (16).

$$\text{Maximize DFSW} = (\omega_1 * \text{DNB}) + (\omega_2 * \text{DST}) + (\omega_3 * \text{DE}) + (\omega_4 * \text{DKU}) \quad (16)$$

where $\omega_1, \omega_2, \omega_3$ and ω_4 =weights of each parameter, so $\omega_1 + \omega_2 + \omega_3 + \omega_4 = 1$.

After obtaining multi objective function DFSW, WOA is applied on DFSW to determine optimal drones from the existing drones for optimal OLSR routing over FANET, which maximize the value of DFSW to obtain the best results. The drones are represented as whales in WOA. At last WOA is concluded by satisfying a concluding condition and optimal drones are found from giving drones over FANET.

3.2 Working and Operation

The flow chart of WOA-OLSR is shown in Fig. 1 and steps of WOA-OLSR working as follows:

Step 1: A Drone's neighbourhood benefaction (DNB) is evaluated for a drone by using Eqs. (9) and (10).

Step 2: An estimation time of link (ETL) is generated for a drone by using Eq. (11), after that DST is calculated for a drone by using Eq. (12).

Step 3: A Drone energy (DE) is calculated for a drone using Eq. (13).

Step 4: A Drone's key utilization (DKU) is calculated for a drone by using Eqs. (14) and (15).

Step 5: A Drone's final significant weight (DFSW) is obtained for a drone by combining DNB, DST, DE and DKU generate a multi objective function by Eq. (16).

Step 6: Step 1 to step 5 is repeated for all drones over FANET and DFSW multi objective function for all drones is obtained.

Step 7: The WOA-OLSR is applied using a WOA algorithm (algorithm 1) to generate the optimal drones for optimal OLSR routing over FANET on the basis of multi objective DFSW function.

The Complete WOA-OLSR (algorithm 2) is illustrated with input parameters N_d =Number of Drones as follows:

Algorithm 2. The WOA-OLSR Routing Algorithm	Number of Operations
1. START	
2. FOR every drone	$(N_d + 1)$
3. Calculate DNB value using eq. 9 & 10	N_d
4. Evaluate DST value using eq. 11 & 12	N_d
5. Evaluate DE value using eq. 13	N_d
6. Calculate DKU value (eq. 14 & 15)	N_d
7. Obtain DFSW function by combining DNB, DST, DE and DKU value (eq. 16)	N_d
8. END FOR	
9. FOR all drones	$(N_d + 1)$
10. Apply WOA (algorithm 1)	
11. END FOR	
12. STOP	

4 Result and Analysis

The proposed WOA-OLSR is developed through MATLAB 2018a environment to reasonably evaluate performance of OLSR routing over FANET. The WOA-OLSR performance is analyzed on the basis of several parameters listed in Table 1. The N_d number of drones (30, 50, 80, 100, 150) is distributed over 1000 m * 1000 m and 2000 m * 2000 m area having 100–600 m transmission range and 0–60 m/s speed over FANET. The WOA-OLSR runs over 150 iterations with 150 population size of WOA.

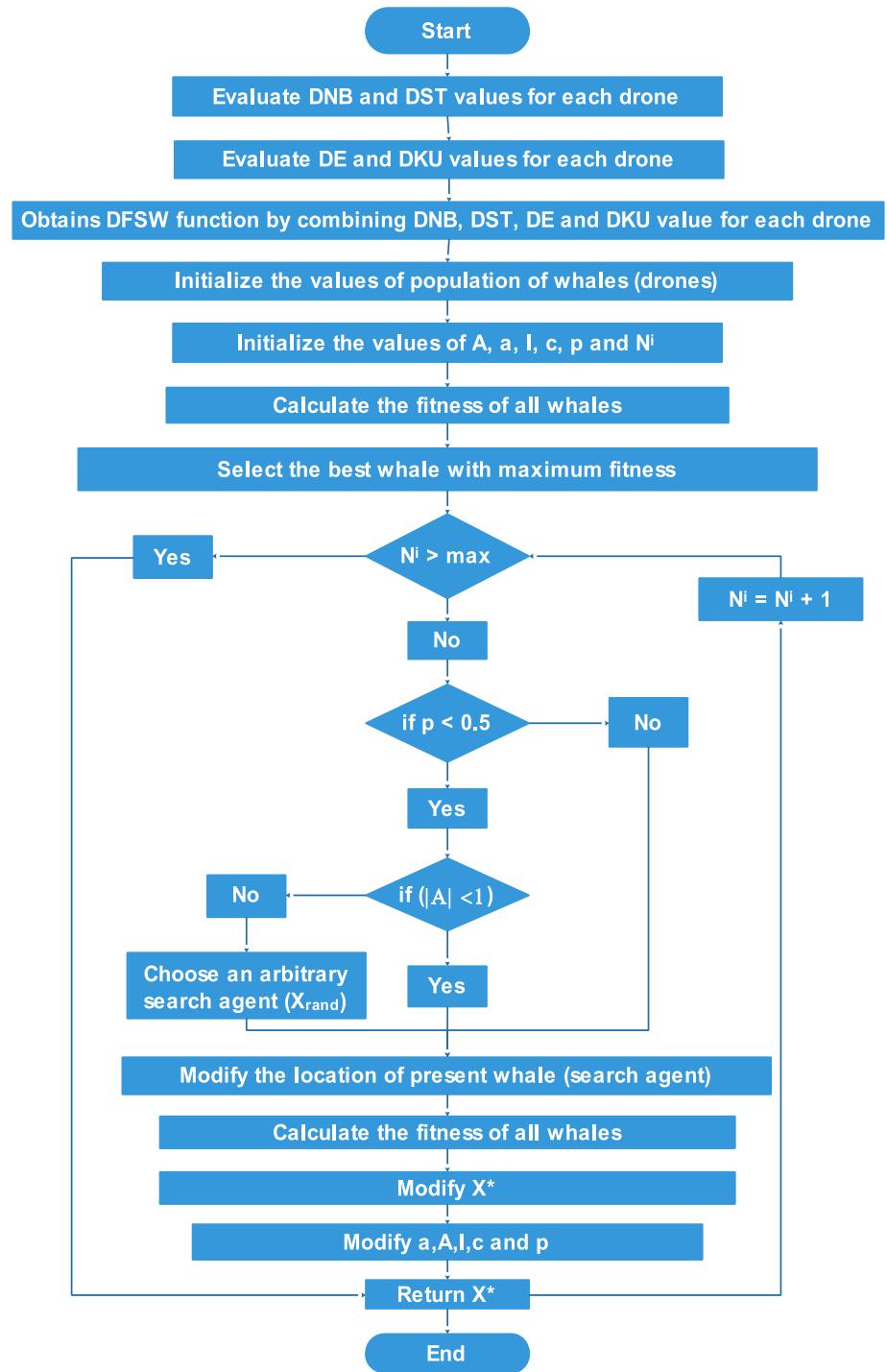


Fig. 1 Flow chart of WOA-OLSR

Table 1 Parameters of simulation

Parameters of simulation	Values
Number of drones	30, 50, 80, 100, 150
Area of network	1000 m * 1000 m and 2000 m * 2000 m
Drone's speed	0–60 m/s (randomly)
Transmission range	100–600 m
Weights ($\omega_1, \omega_2, \omega_3$ and ω_4)	0.25, 0.25, 0.25, 0.25
Size of WOA population	150
Number of iterations	150

The WOA-OLSR is analyzed in terms of the packet delivery ratio, throughput, and end to end delay, energy utilization and time complexity against the results of other routing approaches OLSR, MP-OLSR, P-OLSR, ML-OLSR-FIFO, and ML-OLSR-PMS.

4.1 Packet Delivery Ratio (PDR)

PDR is defined as the fraction of sending data arrived at the base station in the data transmitted by all drones through optimal drones using OLSR in FANET. (Eq. 17)

$$PDR^t = \frac{Data_{BS}^t}{\sum_{i \in O_d} \sum_{j \in N_{d_i}} Data_j^t} \quad (17)$$

where PDR^t =PDR (time t), $Data_{BS}^t$ =sending data arrived to base station (time t), O_d =group of optimal drones, N_{d_i} =group of all drones in i^{th} optimal drone, & $Data_j^t$ =data transmitted by j^{th} drone to base station (time t) using OLSR over FANET.

It shows that the WOA-OLSR generates the maximum PDR value of 93% for 1000 m \times 1000 m (Figs. 2, 3 and 4) and 91% for 2000 m \times 2000 m (Figs. 5, 6 and 7) network area vs. the number of drones, transmission ranges and velocity ranges of drones as compare to several approaches OLSR (75% & 73%), MP-OLSR (81% & 79%), P-OLSR (83% & 81%), ML-OLSR-FIFO (85% & 83%), and MP-OLSR-PMS (87% and 85%). If the network area is increased, then value of PDR is reduced, because data a retransmitting more

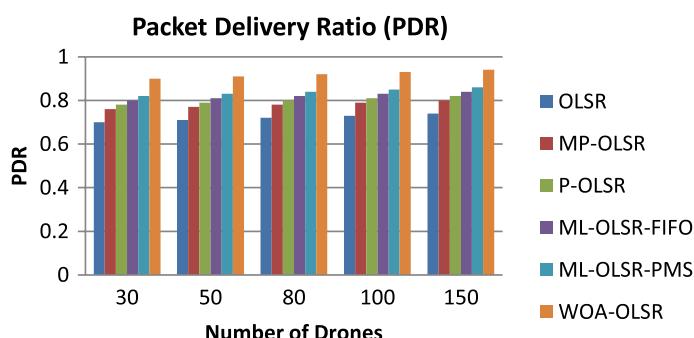


Fig. 2 Packet delivery ratio vs. number of drones (1000 m \times 1000 m network area)

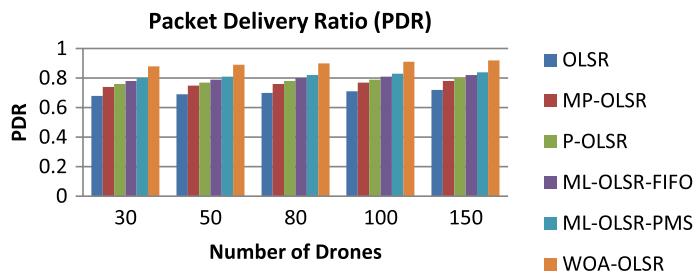


Fig. 3 Packet delivery ratio vs. transmission ranges (1000 m × 1000 m network area)

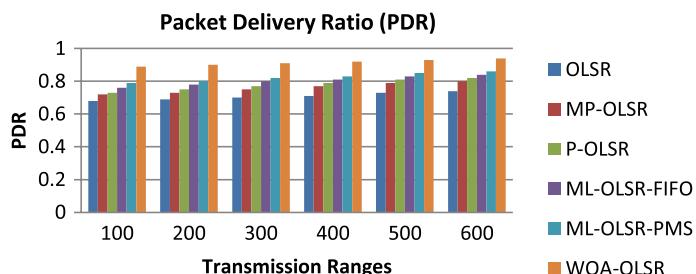


Fig. 4 Packet delivery ratio vs. velocity ranges (1000 m × 1000 m network area)



Fig. 5 Packet delivery ratio vs. number of drones (2000 m × 2000 m network area)

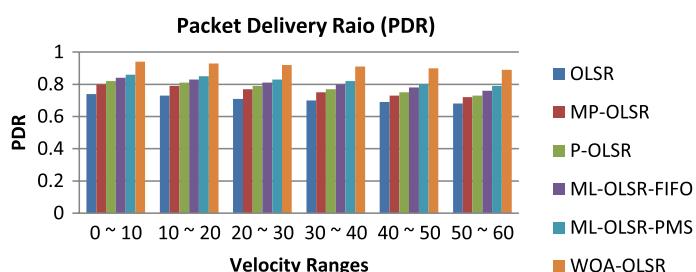


Fig. 6 Packet delivery ratio vs. transmission ranges (2000 m × 2000 m network area)

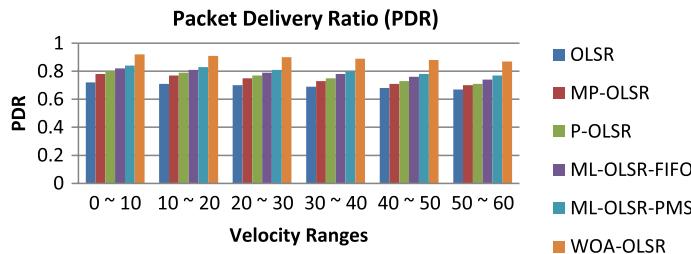


Fig. 7 Packet delivery ratio vs. velocity ranges (2000 m × 2000 m network area)

distance decreasing the probability of delivery in larger network area. If the number of drones is enhanced, then value of PDR is enhanced, because having more drones to send the data. If the transmission ranges are increased, then value of PDR is enhanced, because having least numbers of drones to require for sending data increasing probability of packet delivery. If the velocity ranges are increased, then value of PDR is decreased, because having a maximum velocity of drones to require for sending data decreasing probability of packet delivery.

4.2 End to End Delay (EED)

EED is defined as a combination of path discovery and communication time through OLSR describing the association and ability of FANET. The minimum value of EED is selected for optimal OLSR routing.

It shows that the WOA-OLSR generates the minimum EED value (Seconds) of 0.2914 for 1000 m × 1000 m (Figs. 8, 9 and 10) and 0.3554 for 2000 m × 2000 m (Figs. 11, 12 and 13) network area vs. the number of drones, transmission ranges and velocity ranges of drones as compare to several approaches OLSR (0.4412 & 0.4689), MP-OLSR (0.4234 & 4521), P-OLSR (0.4004 & 0.4321), ML-OLSR-FIFO (0.3895 & 0.4195), and ML-OLSR-PMS (0.3843 and 0.4134). If the network area is increased, then value of EED is enhanced, because data are transmitting more distance in larger network area. If the number of drones is enhanced, then value of EED is enhanced, because having more drones to send the data. If the transmission ranges are enhanced, then value of EED is decreased, because having least numbers of drones to require for sending data. If the velocity ranges are increased,

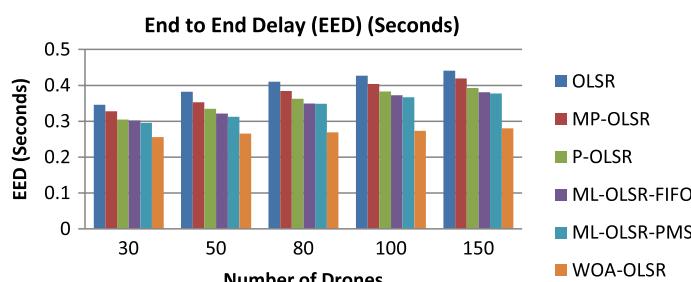


Fig. 8 End to end delay vs. number of drones (1000 m × 1000 m network area)

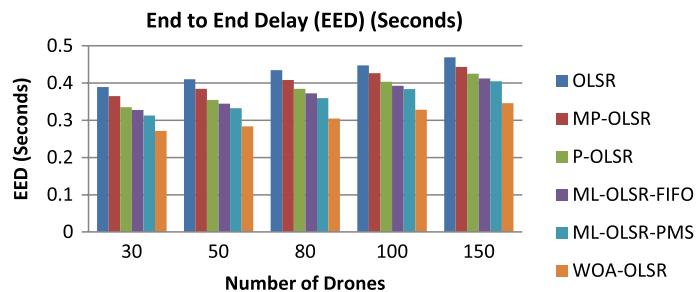


Fig. 9 End to end delay vs. transmission ranges (1000 m × 1000 m network area)

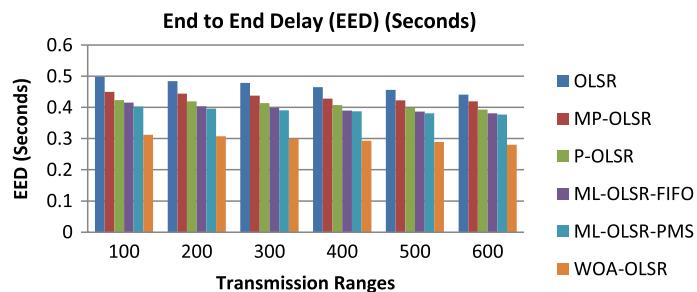


Fig. 10 End to end delay vs. velocity ranges (1000 m × 1000 m network area)

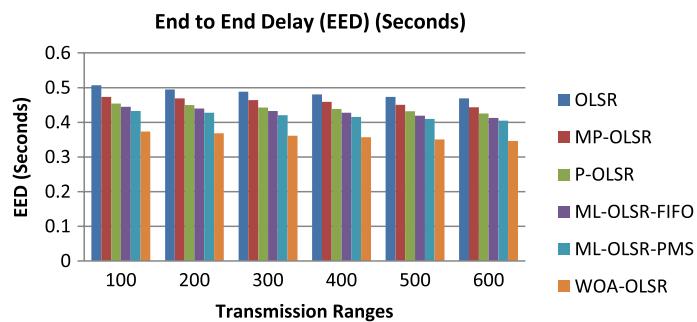


Fig. 11 End to end delay vs. number of drones (2000 m × 2000 m network area)

then value of EED is increased, because having a maximum velocity of drones to require for sending data.

4.3 Energy Utilization

It is the energy utilized by drones for transmitting the data packets at a time through OLSR routing over FANET. The life and stability of optimal drones are increased by decreasing the energy utilization.

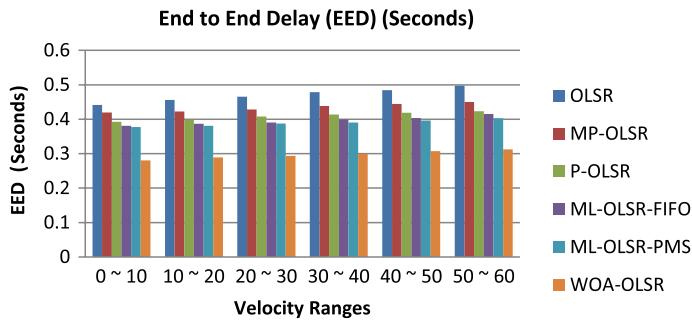


Fig. 12 End to end delay vs. transmission ranges (2000 m × 2000 m network area)

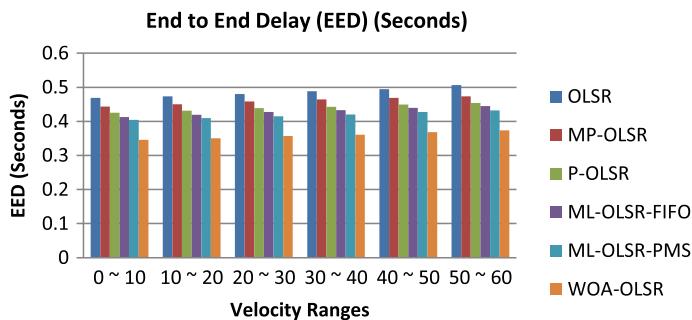


Fig. 13 End to end delay vs. velocity ranges (2000 m × 2000 m network area)

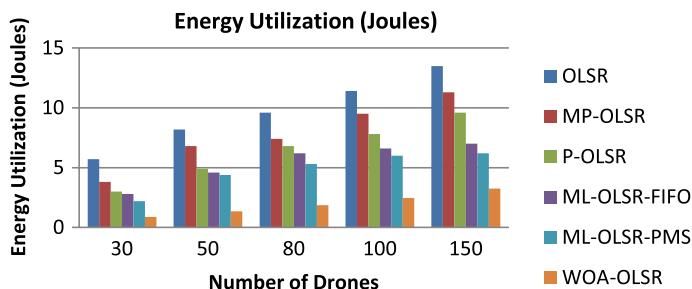


Fig. 14 Energy utilization vs. number of drones (1000 m × 1000 m network area)

It shows that the WOA-OLSR generates the minimum Energy Utilization value (Joules) of 3.385 for 1000 m × 1000 m (Figs. 14, 15 and 16) and 3.684 for 2000 m × 2000 m (Figs. 17, 18 and 19) network area vs. the number of drones, transmission ranges and velocity ranges of drones as compared to several approaches OLSR (13.5 & 13.8), MP-OLSR (11.2&11.6), P-OLSR (9.8&9.3), ML-OLSR-FIFO (7.4 & 8.1), and MP-OLSR-PMS (6.4 and 7.4). If the network area is increased, then value of energy utilization is enhanced, because data are transmitting more distance in larger network area. If the number of drones

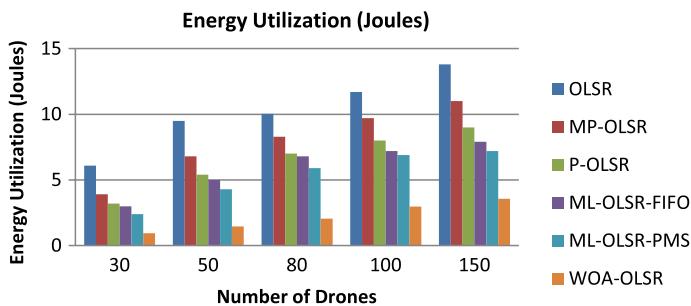


Fig. 15 Energy utilization vs. transmission ranges (1000 m × 1000 m network area)

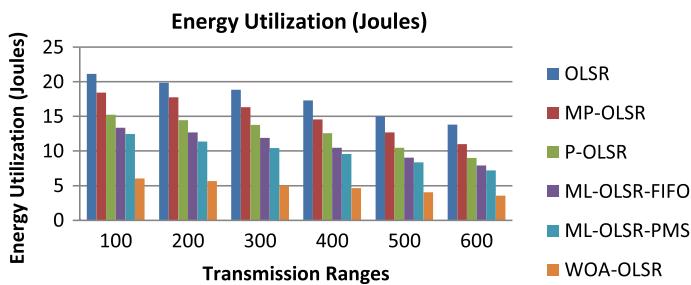


Fig. 16 Energy utilization vs. velocity ranges (1000 m × 1000 m network area)

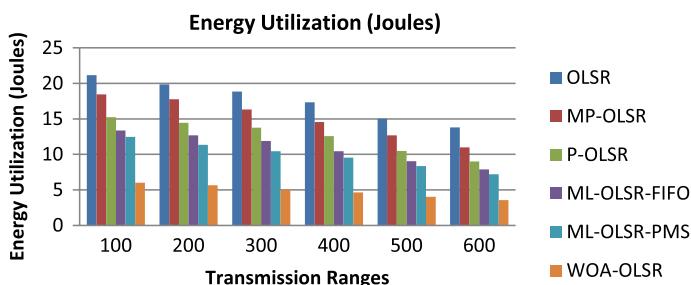


Fig. 17 Energy utilization vs. number of drones (2000 m × 2000 m network area)

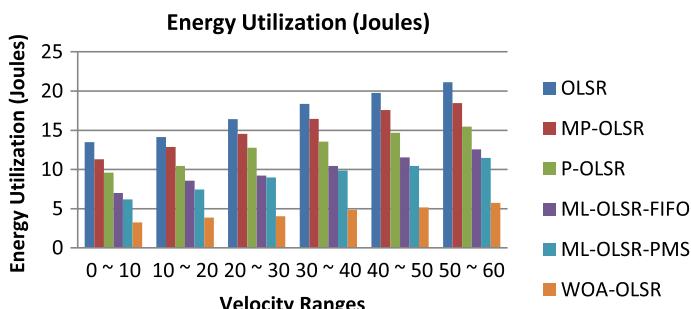


Fig. 18 Energy utilization vs. transmission ranges (2000 m × 2000 m network area)

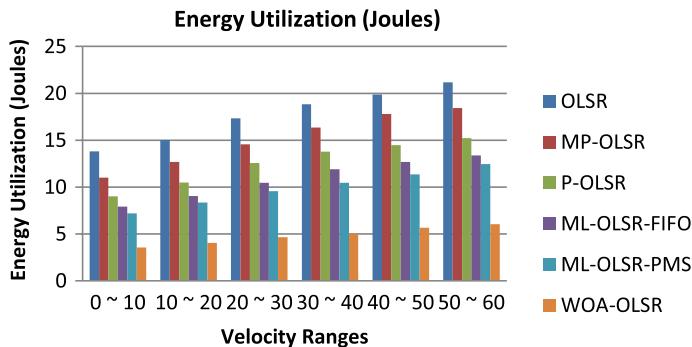


Fig. 19 Energy utilization vs. velocity ranges (2000 m × 2000 m network area)

is enhanced, then value of energy utilization is enhanced, because having more drones consume the more energy. If the transmission ranges are enhanced, then value of energy utilization is decreased, because having more drones as neighbours. If the velocity ranges are increased, then value of energy utilization is increased, because having maximum velocity of drones to consume the more energy over FANET.

4.4 Throughput

It is the transferred data arrived to the base station per unit time slice and data is transferred by all drones through optimal drones using OLSR over FANET.

It shows that the WOA-OLSR generates the maximum throughput (bits per seconds) value of 11,657 for 1000 m × 1000 m (Figs. 20, 21 and 22) and 10,853 for 2000 m × 2000 m (Figs. 23, 24 and 25) network area vs. the number of drones, transmission ranges and velocity ranges of drones as compare to several approaches OLSR (6675 & 6435), MP-OLSR (7466 & 7234), P-OLSR (7857 & 7657), ML-OLSR-FIFO (8046 & 7845), and MP-OLSR-PMS (8435 and 8046). If the network area is increased, then value of throughput is decreased, because data are transmitting more distance decreasing the probability of delivery in larger network area. If the number of drones is enhanced, then value of throughput is increased, because having more drones to send the data. If the transmission ranges are increased, then value of throughput is increased, because having least numbers of drones to require for sending data increasing probability of packet delivery. If the velocity ranges

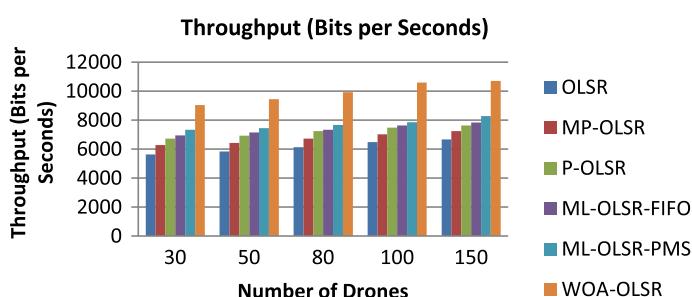


Fig. 20 Throughput vs. number of drones (1000 m × 1000 m network area)

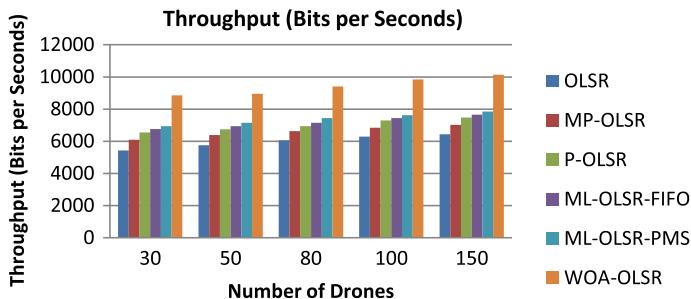


Fig. 21 Throughput vs. transmission ranges (1000 m × 1000 m network area)

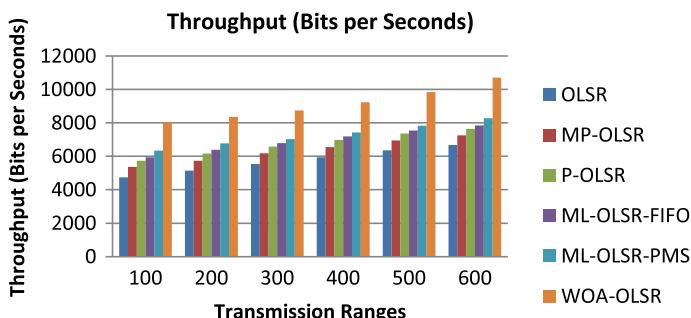


Fig. 22 Throughput vs. velocity ranges (1000 m × 1000 m network area)

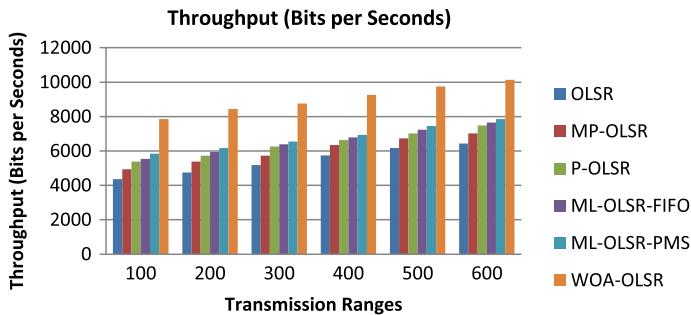


Fig. 23 Throughput vs. number of drones (2000 m × 2000 m network area)

are increased, then value of throughput decreased, because having a maximum velocity of drones to require for sending data decreasing probability of packet delivery.

4.5 WOA-OLSR Time Complexity

The input values based running time of WOA-OLSR is evaluated as follows: N^i =number of iterations, P^w =Whales population, N^w =number of whales, N_d =number of

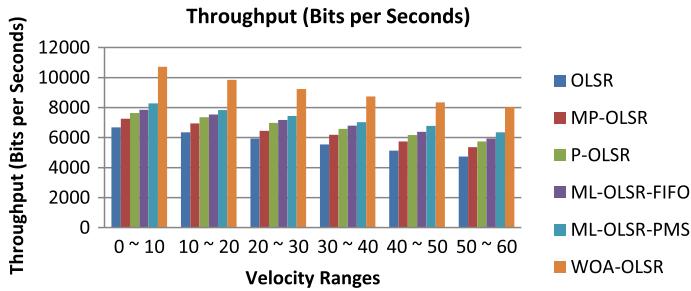


Fig. 24 Throughput vs. transmission ranges (2000 m × 2000 m network area)

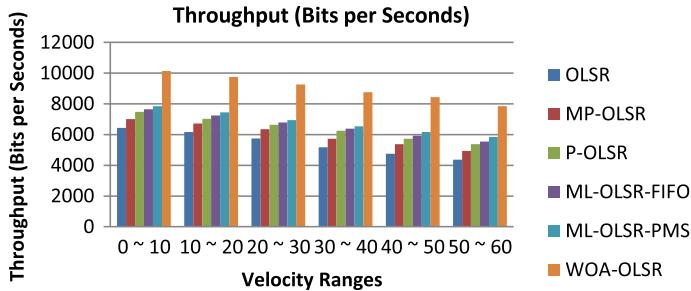


Fig. 25 Throughput vs. velocity ranges (2000 m × 2000 m network area)

drones, D^e = elements of drone and step cost = 1 unit. Hence the total operation of WOA-OLSR is generated from algorithm 1 (Sect. 2.3) and algorithm 2 (Sect. 3.2).

$$\begin{aligned}
 \text{Total - Operation} = & N_d + 1 + N_d + N_d + N_d + N_d + N_d + N^i \\
 & + 1 + N^i * P^w + N^i * N^w + N^i + 1 + N^i * (N^w + 1) \\
 & + N^i * N^w \\
 & + N^i * N^w + N^i * N^w + N^i * N^w + N^i * N^w \\
 & + N^i * N^w + N^i + N^i + N^w
 \end{aligned}$$

$$\text{Total - Operation} = N^i * P^w + 12N^i * N^w + 6N_d + 5N^i + N^w + 3 \quad (18)$$

There is further $N_d * D^e$ operations are utilized to employ N_d number of drones and D^e elements of drones. Hence, Eq. 18 is converted to Eq. 19.

$$\text{Total - Operation} = N^i * P^w * N_d * D^e + 12N^i * N^w + 6N_d + 5N^i + N^w + 3 \quad (19)$$

The time complexity in the worst case is calculated by assuming the entire input values same in Eq. 19, so Eq. 20 is obtained.

$$\text{Total - Operation} = n^4 + 12n^2 + 3 \quad (20)$$

The time complexity of approaches is OLSR ($O(n^2)$), MP-OLSR ($O(n^2)$), P-OLSR ($O(n^2)$), ML-OLSR-FIFO ($O(n^3)$), and MP-OLSR-PMS ($O(n^3)$) and WOA-OLSR ($O(n^4)$). Hence, the solution of all approaches like OLSR, MP-OLSR, P-OLSR, ML-OLSR-FIFO, and MP-OLSR-PMS and WOA-OLSR are evaluated in polynomial time.

5 Conclusion

The capabilities of drones and UAVs are efficiently utilized for various military and rescue applications to enhance the research and development of FANET. The small journey time and infertile routing are basic problems of drones due to high mobility in a 3D environment and low battery power. The secure and energy efficient data transmission will be provided through optimal routing by resolving the problems of drones over FANET. Therefore, a whale optimization algorithm based optimized link state routing (WOA-OLSR) is proposed to obtain energy efficient and secure optimal routing in FANET. The calculated performance illustrates the better efficiency of WOA-OLSR based on some parameters such as a packet delivery ratio, end to end delay, energy utilization, throughput, and time complexity as compared to the previous approaches OLSR, MP-OLSR, P-OLSR, ML-OLSR-FIFO and ML-OLSR-PMS. In the future, developers can advance more than one objective as per their needs for energetic drones with maximum mobility. The routing mechanism in FANET is also improved by utilizing other bio inspired approaches like Fruit Fly Optimization, Moth-Flame Optimizer, Dragonfly Optimization, and Grey Wolf Optimization for future methodical studies.

Funding Not Applicable.

Code Availability Custom Code.

Declarations

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Sharma, V., Sabatini, R., & Ramasamy, S. (2016). UAVs assisted delay optimization in heterogeneous wireless networks. *IEEE Communication Letters*, 20, 2526–2529
2. Khan, N. A., Jhanjhi, N. Z., Brohi, S. N., & Nayyar, A. (2020). Emerging use of UAV's: Secure communication protocol issues and challenges. In *Drones in Smart-Cities* (pp. 37-55). Elsevier.
3. Ahn, T., Seok, J., Lee, I., & Han, J. (2018). Reliable flying iot networks for UAV disaster rescue operations. *Mobile Information Systems*. <https://doi.org/10.1155/2018/2572460>
4. Erdelj, M., Uk, B., Konam, D., & Natalizio, E. (2018). From the eye of the storm: An IoT ecosystem made of sensors, smartphones and UAVs. *Sensors*, 18, 1–20. <https://doi.org/10.3390/s18113814>
5. Ferrera, E., Alcántara, A., Capitán, J., Castaño, A. R., Marrón, P. J., & Ollero, A. (2018). Decentralized 3D collision avoidance for multiple UAVs in outdoor environments. *Sensors*, 18, 1–20. <https://doi.org/10.3390/s18124101>
6. Bujari, A., Calafate, C. T., Cano, J. C., Manzoni, P., Palazzi, C. E., & Ronzani, D. (2018). A location-aware waypoint-based routing protocol for airborne DTNS in search and rescue scenarios. *Sensors*, 18, 1–14. <https://doi.org/10.3390/s18113758>
7. Hu, B., Wang, C., Chen, S., Wang, L., & Yang, H. (2018). Proactive coverage area decisions based on data field for drone base station deployment. *Sensors*, 18, 1–14. <https://doi.org/10.3390/s18113917>

8. Kim, B., Min, H., Heo, J., & Jung, J. (2018). Dynamic computation offloading scheme for drone-based surveillance systems. *Sensors*, 18, 1–10. <https://doi.org/10.3390/s18092982>
9. Popescu, D., Dragana, C., Stoican, F., Ichim, L., & Stamatescu, G. (2018). A collaborative UAV-WSN network for monitoring large areas. *Sensors*, 18, 1–25. <https://doi.org/10.3390/s18124202>
10. Huang, J., Fan, X., Xiang, X., Wan, M., Zhuo, Z., & Yang, Y. (2016). A clustering routing protocol for mobile ad hoc networks. *Mathematical Problems in Engineering*. <https://doi.org/10.1155/2016/5395894>
11. Ganeshan, R., Raajini, X. M., Nayyar, A., Sanjeevikumar, P., Hossain, E., & Ertas, A. H. (2020). BOLD: Bio-inspired optimized leader election for multiple drones. *Sensors*, 20, 1–20
12. Valentino, R., Jung, W. S., & Ko, Y. B. (2018). A design and simulation of the opportunistic computation offloading with learning-based prediction for unmanned aerial vehicle (UAV) clustering networks. *Sensors*, 18, 1–14. <https://doi.org/10.3390/s18113751>
13. Fahad, M., Aadil, F., Khan, S., Shah, P. A., Muhammad, K., Lloret, J., Wang, H., Lee, J. W., & Mehmood, I. (2018). Grey wolf optimization based clustering algorithm for vehicular ad-hoc networks. *Computers and Electrical Engineering*, 70, 853–870
14. Aadil, F., Raza, A., Khan, M. F., Maqsood, M., Mehmood, I., & Rho, S. (2018). Energy aware cluster-based routing in flying ad-hoc networks. *Sensors*, 18, 1–16. <https://doi.org/10.3390/s18051413>
15. Albu-Salih, A. T., Seno, S. A. H., & Mohammed, S. J. (2018). Dynamic routing method over hybrid SDN for flying ad hoc networks. *Baghdad Science Journal*, 15(3), 361–368
16. Hong, J., Zhang, D., & Niu, X. (2017). Impact analysis of node motion on the performance of FANET Routing protocols. In: *14th International Conference on wireless Communications, Networking and Mobile Computing (WiCOM)*, pp. 147–162.
17. Zheng, X., & Qi, Q., Wang, Q., Li, Y. (2017). An adaptive density based routing protocol for flying Ad Hoc networks. In: *2nd International Conference on Materials Science, Resource and Environment Engineering (MSREE) AIP Conf. Proc.*, pp. 1–8. <http://doi.org/https://doi.org/10.1063/1.5005315>
18. Perez, A. G., & Cano, M. D. (2018). Flying ad hoc networks: A new domain for network communications. *Sensors*, 18, 1–23. <https://doi.org/10.3390/s18103571>
19. Sharma, V., Kumar, R., & Rathore, N. (2018). Topological broadcasting using parameter sensitivity-based logical proximity graphs in coordinated ground-flying ad hoc networks. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 6(3), 54–72
20. Khan, M. A., Khan, I. U., Safi, A., & Qureshi, I. M. (2018). Dynamic routing in flying ad-hoc networks using topology-based routing protocols. *Drones*, 2, 1–15. <https://doi.org/10.3390/drones2030027>
21. Leonov, A. V., & Litvinov, G. A. (2018). Simulation-based packet delivery performance evaluation with different parameters in flying ad-hoc network (FANET) using AODV and OLSR. In: *International Conference Information Technologies in Business and Industry, IOP Conf. Series, Journal of Physics*, (pp. 1–16). Doi:<https://doi.org/10.1088/1742-6596/1015/3/032178>
22. Khan, M. A., Qureshi, I. M., & Khanzada, F. (2019). A hybrid communication scheme for efficient and low-cost deployment of future flying ad-hoc network (FANET). *Drones*, 3(16), 1–20. <https://doi.org/10.3390/drones3010016>
23. Yi, J., Adnane, H. A., David, S. & Parrein, B. (2018). Multipath optimized link state routing for mobile ad hoc network. *HAL*, 1–17.
24. Radu, D., Cretu, A., Parrein, B., Yi, J., Avram, C., & Astilean, A. (2018). Flying ad hoc network for emergency applications connected to a fog system. *HAL*, 1–13. (<https://hal.archives-ouvertes.fr/hal-01763827>)
25. Wen, S., & Huang, C. (2018). Delay-constrained routing based on stochastic model for flying ad hoc networks. *Mobile Information Systems*. <https://doi.org/10.1155/2018/6056419>
26. Arabi, S., Sabir, E., Elbiaze, H., & Sadik, M. (2018). Data gathering and energy transfer dilemma in UAV-assisted flying access network for IoT. *Sensors*, 18, 1–25. <https://doi.org/10.3390/s18051519>
27. Sharma, V., & Kumar, R. (2017). G-FANET: An ambient network formation between ground and flying ad hoc networks. *Telecommunication Systems*, 65(1), 31–54
28. Wei, Z., Liu, X., Han, C., & Feng, Z. (2018). Neighbor discovery for unmanned aerial vehicle networks. *IEEE Access*, 6, 68288–68301. <https://doi.org/10.1109/ACCESS.2018.2871132>
29. Li, J., Chen, M., Dai, F., & Wang, H. (2018). Prioritizing-based message scheduling for reliable unmanned aerial vehicles ad hoc network. *International Journal Perform ability Engineering*, 14(9), 2021–2029. <https://doi.org/10.23940/ijpe.18.09.p10.20212029>



Mayank Namdev was born in India in 1985. He has done his BE in Information Technology from BIST, RGPV Bhopal, M.Tech in Computer Science and Engineering from RKDFIST RGPV Bhopal and currently Ph.D. research scholar at the department of Computer Science Engineering in UIT RGPV Bhopal (MP), India. His research interest includes Ad Hoc Network, Data Base Security, Network Management and algorithm analysis.



Sachin Goyal was born in India in March 1979. He has done his BE in Computer Science & Engg From ITM, RGPV, Bhopal, M.Tech in Artificial Intelligence from SATI, RGPV, Bhopal and have done PhD in IT from RGPV, Bhopal. His area of Interest includes Digital watermarking, Theoretical Computer science and computer network. He published More than 30 papers in Journals and Conferences (Including SCIE and SCOPUS Journals). He is presently working as a Associate professor in Department of Information Technology, UIT, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal.



Ratish Agarwal was born in India in May 1979. He has done his BE in Electronics and Communication with honours from Rajiv Gandhi Technological University, Bhopal [M.P.], Qualified GATE in 2001 with 94%, MTech. [Digital Communication] First division with honours (80%) from MANIT, Bhopal [M.P.] and have done PhD from RGPV, Bhopal. His area of Interest includes Mobile Ad Hoc Network, Mobile Computing, Computer Architecture. He published More than 50 papers in Journals and Conferences (Including SCI and SCOPUS Journals). He is presently working as a Associate professor in Department of Information Technology, UIT, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal.

Received April 15, 2019, accepted May 30, 2019, date of publication June 7, 2019, date of current version June 25, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2921729

Enhanced Deployment Strategy for the 5G Drone-BS Using Artificial Intelligence

FADI AL-TURJMAN¹, (Member, IEEE), JOEL PONCHA LEMAYIAN¹,
SINEM ALTURJMAN¹, AND LEONARDO MOSTARDA²

¹Department of Computer Engineering, Antalya Bilim University, 07190 Antalya, Turkey

²Computer Science Division, University of Camerino, 62032 Camerino, Italy

Corresponding author: Leonardo Mostarda (leonardo.mostarda@unicam.it)

ABSTRACT The use of drones to perform various task has recently gained a lot of attention. Drones have been used by traders to deliver goods to customers, scientists, and researchers to observe and search for endangered species, and by the military during critical operations. The flexibility of drones in remote controlling makes them ideal candidates to perform critical tasks with minimum time and cost. In this paper, we use drones to setup base stations that provide 5G cellular coverage over a given area in danger. The aim of this paper is to determine the optimum number of drones and their optimum location, such that each point in the selected area is covered with the least cost while considering communication relevant parameters such as data rate, latency, and throughput. The problem is mathematically modeled by forming linear optimization equations. For fast optimized solutions, genetic algorithm (GA) and simulated annealing (SA) algorithms are provisionally employed to solve the problem, and the results are accordingly compared. Using these two meta-heuristic methods, quick and relatively inexpensive feedback can be provided to designers and service providers in 5G next generation networks.

INDEX TERMS Genetic algorithm, simulated annealing, UAV, smart city, IoT.

I. INTRODUCTION

There is a high demand for provisioning high quality of services (QoS) due to recent massive growth in everything, especially in the telecommunication sector. The rapid population growth has brought a number of challenges in telecommunications, including coverage and data traffic capacity. One promising way to mitigate some of these challenges is the utilization of intelligent systems towards smart projects such as smart cities, smart building, smart vehicles, smart grids, etc. Internet of Things (IoT) is the interconnection of these smart projects with sensing, actuation and computing capabilities via the internet. It is used to provide better services and resource management for the general population. However, the vast amount of data generated and collected requires the use of a powerful communication paradigm in order to guarantee the QoS in all these services. 3G and 4G have a few QoS advantages, such as low deployment cost, simplicity in management, extensive coverage and high security. However, they do not support Low-Cost

The associate editor coordinating the review of this manuscript and approving it for publication was Wael Guibene.

Machine-Type Communications (MTC) with high efficiency [1]. This is an important feature for the future telecommunication because 3G and 4G have been designed mainly for optimised broadband communication [2], [3]. On the other hand, 5G is specifically designed to provide QoS to users, which means that it is capable of providing the maximum bandwidth, and reduced latency, error rate, and uptime. Additionally, 5G have increased data rate, reduced delay, as well as, enhanced cellular coverage [4]. In health care, for instance, these advantages are useful in improving the system for millions of people. Chen *et al.* [5] designed a personalised emotion-aware healthcare care system using 5G. It focuses on the emotional care, particularly for children, and mentally ill and elderly people. The proposed system uses various IoT devices to capture images and speech signals from a patient in an intelligent environment such as a smart home. This data is fed into an emotion detection module, which processes speech and image signals separately. Then it merges the obtained results to produce a final score of the emotion. The score is further analysed to determine if the patient requires attention. And if so, medics are alerted immediately. Furthermore, Poncha *et al.* [6] state that the

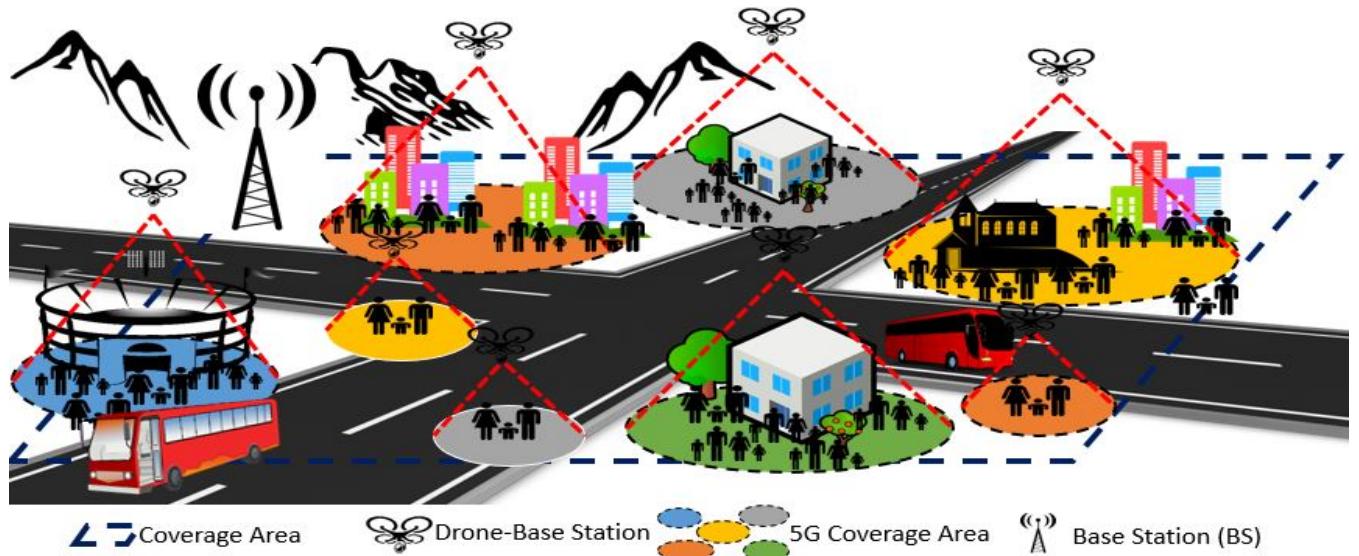


FIGURE 1. UAV-based 5G coverage in urban area.

ability of 5G to focus on heterogeneous access technology has opened a plethora of possibilities. 5G has the ability to create an interconnected world using IoT. Skouby *et al.* [7] add that such a linked system must connect smart cities, smart homes and IoT in one cohesive paradigm. 5G technology will not only offer high-speed broadband Internet connectivity, but will also support e-payments, e-transactions and other fast electronic transactions [8]. Moreover, 5G focusses on Voice over Internet Protocol (VoIP) devices. This in turn leads to high levels of data transmissions and call volumes [8].

In this paper, we work on maximizing the 5G coverage for the aforementioned applications using Unmanned Aerial Vehicles (UAVs) in urban settlements as shown in Figure 1. Wireless users expect to have unlimited and affordable internet access all the times. Increasing the number of Base Stations (BSs) in a given area is a potential way for satisfying users and providing extended 5G coverage. However, this is not an easy task. Because a few of these BSs can have light or no load at all at a particular time, while other BSs might experience very high data traffic and unnecessary overhead. The unpredictability characteristic of the user makes it hard to know exactly where and when a base station should be located. We use UAVs to counter this problem by designing drone-BSs as depicted in Figure 1. The drone-BS is flexible, and able to be placed where it is needed most, and at any particular time. And hence, it efficiently provides 5G coverage for the users at all times. Kalantari *et al.* [9] state that drone-BS can be used to provide assistance to the ground BSs with high data rates as well when additional space and time is required. There is a growing number of research work being done on drone-BS in cellular networks. However, one critical challenge that has not been given much attention is finding the lowest number of drone-BS and their respective positions in a given 3D space, required to provide maximum

5G coverage with guaranteed QoS. The main contribution of this paper is therefore to model this challenge into a linear optimization problem, and use Simulated Annealing (SA) and Genetic Algorithm (GA) metaheuristic algorithms to solve it. GA and SA algorithms have been chosen due to their ability in providing fast and efficient solutions to service providers. These two algorithms are applied in a system of UAVs communicating in a multi-hop fashion. This system helps in reducing the amount of energy consumed by the Drone-BS since we do not require wide-range transmitters, which are high power consumers. The two algorithms are used in extensive simulations, where coverage graphs are drawn and numerical results are compared in order to determine which algorithm can provide quick and more accurate solutions.

The rest of the paper is organized as follows. Section 2 talks about some of the works that have been done relating to this study. Section 3 discusses some of the main challenges faced by aerial sensor networks while Section 4 presents the model of the system. Section 5 discusses the findings of this study. Finally, Section 6 presents our conclusions and future work. In order to further assist the reader, a list of used abbreviations in this article and their definitions are presented in Tables 1 and 2.

II. RELATED WORK

The efficiency of optimal drone positioning has attracted a lot of interest among researchers and academicians. Zorbas *et al.* [10] introduces a minimum cost drone location problem. In their work, Zorbas *et al.* use a two dimensional terrain to find the optimal location and number of drones/UAVs to observe given targets, which could be mobile or static in a given area, the authors develop linear and non-linear optimization equation by considering the coverage of the drones and the energy consumed.

TABLE 1. Abbreviations and definitions.

Abbreviation	Description
SA	Simulated Annealing
GA	Genetic Algorithm
UAV	Unmanned Aerial Vehicle
WSN	Wireless Sensor Networks
MWSN	Multimedia Wireless Sensor Network
MGSAA	Modified Genetic and Simulated Annealing Algorithm
BS	Base Station
ILP	Integer Linear Program
MDLP	Mobile Drone Location Problems
PM	Probability of Mutation
PC	Probability of Crossover
SDLP	Static Drone Location Problems
5G	5 th Generation of cellular networks
BS	Base Station
Drone-BS	Drone Base Station

TABLE 2. Notations and descriptions used in this study.

Notation	Description
T_0	Initial temperature
δ_0	Initial solution
α	Cooling factor
m	Number of stages
n	Number of moves
δ_F	Final solution
δ	Current solution
δ_{Temp}	Temporary solution
σ	move operator
f	Fitness function
T_t	Temperature at time t
PM	Probability of Mutation
PC	Probability of Crossover
G_{max}	maximum generation number
PS	Population size
BFS	Best fitness solution

Moreover, Tuba *et al.* [12] present a study in which they look into a recent brainstorm optimization algorithm. It aims at finding the optimal positions for static drones in a monitored area such that their coverage is maximized. The algorithm was used to solve the placement problem for both uniformly and clustered targets. Obtained results showed that the proposed algorithm is very efficient for solving drone placement problems. In [10]–[12], authors try to find the optimal drone locations for observational and monitoring purposes only. Unlike our work, in which we target a new trend in the 5G coverage.

Furthermore, we can consider UAVs as aerial wireless base stations when cellular networks are out of service [13].

This system can be used when disasters such as flood and earthquake affect the existing communication system. Shakhatreh *et al.* [14] talk about finding an optimal position for the UAVs such that the sum of time durations of uplink transmissions is maximized. They use a gradient projection-based algorithm to find the optimal placement of a single drone-BS by considering the uplink scenario as a constraint. Authors prove their hypothesis by presenting detailed simulation results for the optimization problem under different cases.

Kalantari *et al.* [9] present a study on the number of 3D placement of drone base stations. In this study, the authors use a heuristic algorithm to optimally place drone base station in a region with different target densities. The goal of the study is to find the minimum number of drones and their 3D placement such that all users are served. The simulation results obtained from the study showed that the proposed system can yield QoS constraint of the network. Unlike the attempts in [11]–[14], our work considers time-efficient solutions. We use heuristic GA and SA to find the optimal drone locations for 5G coverage, while considering energy and cost constraints.

Numerous works have been done to compare the different results obtained by different heuristic optimization algorithms. In [15], Rodriguez *et al.* compare four studies that have been done on routing and wavelength assignment with the aim of supporting and improving traffic related problems. Moreover, the authors perform various simulations using the optimizing algorithms, Simulated Annealing (SA) and Genetic Algorithm (GA). The results obtained revealed that the optimizing algorithm produced better results compared to the other algorithms.

Yu *et al.* [16] propose a new heuristic algorithm used to test generation of data during software testing process. Modified Genetic and Simulated Annealing Algorithm (MGSAA) was used to perform different experiments. Yu et al presents the simulation results and conclude that the proposed method generates high quality results compared to Genetic Algorithm (GA). In [17], Thompson *et al.* used GA and SA metaheuristic algorithms to optimise a topological design network and compare the results. The authors concluded that the average GA solution costs less than the average SA solution.

However, Thompson *et al.* [17] and other overviewed studies in this section compared the cost of GA and SA algorithms to determine the optimal solution for a topological network design. In our work, we use GA and SA to determine the optimal position for 5G drone base stations given the constraints of coverage, energy and cost. Hence, we aim at improving parameters such as the data rate, latency and throughput.

III. CHALLENGES IN UAVS NETWORKS

Drone base-stations have been proven to be a good candidate in providing high-throughput in wireless communications for situations requiring moderately stable links and network topologies. This is due to their unique ability in

hovering/moving with the target at close distances. However, there are few challenges that need to be addressed when it comes to aerial sensor deployment. Aerial sensor network face numerous challenges, especially in the monitoring of outdoor critical situations where the severity of the environment such as high temperatures, heavy rains, storms and the likes, destroy the installed aerial sensors [18].

In this work, flying drones are not only used as aerial sensors, but also as access points (BSs). Consequently, new challenges are expected to rise in comparison to conventional sensor networks. Resource allocation is one of the most important aspects. Sample limited resources for sensor nodes include power, memory and communication bandwidth. Usually sensor nodes consume little power while performing some activities such as sensing, data storage and simple data aggregation. However, there are other operations, which consume a significant amount of power such as the image analysis in multimedia and cellular applications. Hence, further research should be focused on determining the trade-offs between locally storing, communicating and processing data, and consequently develop energy-efficient sensory paradigms.

In aerial sensing platform, most of the power is consumed during UAV propulsion, power consumed during sensing, processing, and communication is usually relatively negligible, and hence can be ignored [19]. Therefore, for efficient power consumption, one has to plan the flight path of the UAV. For instance, ascending consumes more power than flying at a constant altitude [10]. Moreover, weather conditions can have a significant effect on the UAV's power consumption. Sensors on the UAV can be used to send back information about direction and speed of the wind during the flight for instance, and adapt accordingly.

Moreover, aerial WSN communication is different in comparison to other communication networks. When UAVs are flying, they need to exchange data (current position, speed, direction, etc.) with each other as observed from Figure 2. Individual UAVs need to exchange their information after only a few seconds. However, multiple UAVs flying simultaneously need to know and transmit their position more accurately. Hence, UAVs' position data is exchanged every few milliseconds. This necessitates a link with low latency and a wide communication range.

In [20], Asadpour *et al.* acknowledge the importance of aerial sensor networks. A few challenges/issues incurred in this paradigm have been addressed, in addition to discussing a few possible solutions, as well. It has been reported that mobility and heterogeneity of the utilized nodes (or BSs), can cause connectivity problems because of their severe influence on the distance between the intra-communicating nodes. This can significantly change the flying network topology. And hence, effective routing, scheduling, and data forwarding techniques must be further investigated in this area. Mitchell *et al.* [24] proposed a scheduling algorithm, which can be applied in such dynamic topologies. Their algorithm computes the shortest path to the sink node dynamically, and

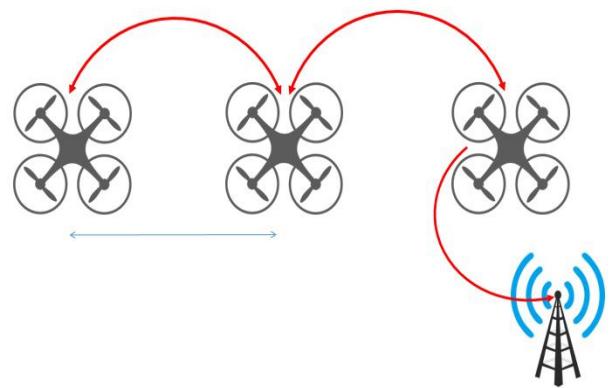


FIGURE 2. UAV communication with each other and BS.

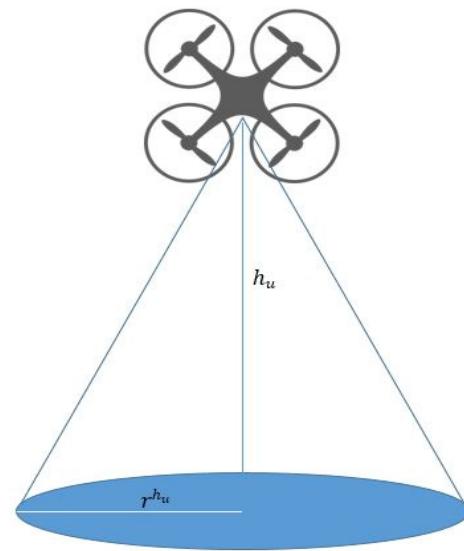


FIGURE 3. UAV cellular cover.

hence, it reduces latency. This approach can also be applied in this study.

IV. SYSTEM MODELS

In this work, the considered system consists of a common sink (BS), to which information is sent and dispatched. Since a UAV might be out of the communication range of the BS, the UAV can send/relay its data to the nearest UAV. The latest can in turn forward the information to the next available one until it reaches an UAV in the communication range with the BS [18]. Figure 2 below demonstrates the network architecture of the considered UAVs system. In Table 2, a summary of the assumed/used notations in this paradigm is provided.

In order to obtain an optimal number of drones to be used in maximizing the 5G coverage, it is imperative that drones are located in the correct position. This is of utmost importance so that it obtains the maximum coverage while minimizing the number of UAVs. This in turn can reduce the cost. Younis *et al.* [21] claims that sensor placement is a challenge by itself. Considering the limited sensing and communication sensor range, as well as the restricted resources such as energy

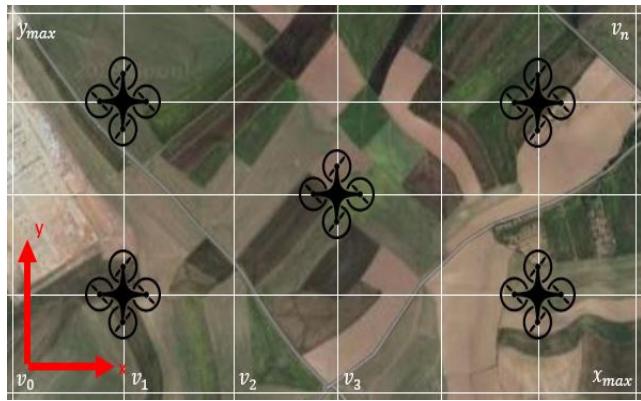


FIGURE 4. Virtual coordinates inside coverage area.

and coverage, make it more complicated problem. In their study, Quaritsch *et al.* [19] investigate the use of UAVs in disaster management. They discuss the challenges facing the networked UAVs as well as focusing on their optimal placement. In doing so, Quaritsch *et al.* mathematically formulated the coverage problem and presented potential assessment results. Authors took into account two optimization criteria, the first one is the quality of the image taken, which refers to the coverage quality of the UAV. The second one is the consumption of resources, which involves the communication bandwidth and the energy used in flying. The observation area and the forbidden area are drawn by the user using worldwide coordinates, namely the longitude and latitude. However, as described by Quaritsch et al, the entire process of optimizing sensor placement is done using relative coordinates. Therefore, the first step is to transform the worldwide coordinates into the relative coordinates by selecting an arbitrary origin inside the observed region. Hence formulating the x- and y- axis to go eastwards and northwards respectively, as shown in Figure 4.

Zorbas *et al.* [10] present a study that determines the optimal static and dynamic drone positioning in a selected area, to minimize cost and maximize coverage. It was shown that drones must have a maximum and a minimum observation altitude. That is because the height of the UAV is directly proportional to the coverage area it can observe. However, the higher the UAV is, the more energy it consumes. Therefore, there must be a threshold on the maximum height/altitude an UAV can be placed at. Park *et al.* [11] propose a coverage decision algorithm, which aims at solving handover problems caused by time-varying aerial environments. The algorithm takes into account the height of the drone needed to provide a better coverage. According to [11], controlling the height of the drone helps to provide better drone coverage.

The relationship between the targeted area coverage and the height of the drone was formulated by $A = \pi (R^2 - h^2)$, where A is the coverage area of the drone, h is the drone's height, and R represents the radius of the drone's wireless transmitter. Obviously, this area A is equal to πR^2 when the

height h is equal to 0. The main focus in [10] and [11], was to minimize the cost, and hence, the number of drones and energy consumed. Accordingly, our assumed UAVs can fly to a maximum height equal to h_{max} , and a minimum height equal to h_{min} , that maintain a specific coverage radius r^{h_u} [10]. Figure 4 shows a rectangle with length x_{max} and width y_{max} , which represent the area of interest. Therefore, targets could be in any arbitrary location in an area of $x_{max} * y_{max}$. We assume that there is a position (x, y, h) that a drone can be located at instantaneously. Let \mathbf{U} denote a set of available drones, and \mathbf{T} is the set of targets.

Each target $t_i \in \mathbf{T}$ has position (X_{t_i}, Y_{t_i}) . Drone $u \in \mathbf{U}$ has position (X_u, Y_u, h_u) . For $h = 0$, the distance between the target and the drone is:

$$D_{t_i}^{u_x, u_y} = \sqrt{(X_{t_i} - X_u)^2 + (Y_{t_i} - Y_u)^2} \quad (1)$$

Each drone u , has a communication range θ in form of a disc in area $x_{max} * y_{max}$ as shown by the blue area in Figure 3. And it has a radius of r^{h_u} , which depends on the height of the drone h_u . The larger the value of h_u , the longer radius r^{h_u} we have. There are two important decisions that must be made at this point, the first one is to determine the position (X_u, Y_u, h_u) of the drone $u \in U$ (coordinates) and the second one is to find the target $t_i \in T$ in the area of interest.

For the first problem (Position of drone):

$$\delta_{xyh}^u = \begin{cases} 1, & \text{if the drone } u \text{ is located at } (x, y, h) \\ 0, & \text{other wise} \end{cases} \quad (2)$$

And for the second problem (Target observed):

$$\gamma_{t_i}^u = \begin{cases} 1, & \text{if the target } t_i \text{ is in the range of drone } u \\ 0, & \text{other wise} \end{cases} \quad (3)$$

The objective is to cover all the targets using at least one drone. Each drone consumes a total energy E formulated as:

$$E = (\beta + \alpha h) t + P_{max} (h/s), \quad (4)$$

where β is the minimum power needed to hover at almost zero altitude, α is the motor speed multiplier, P_{max} is the maximum motor power, and s and t are speed and operating time, respectively. Also, h represents the drone's height. The term $P_{max} (h/s)$ is used to show the power used to rise the drone to a height h at speed s . It is worth pointing out here that β and α depend on the weight of the drone and the used motor characteristics. Therefore, we can formulate our placement problem as follows.

Minimize $f(\delta)$

Subject to ,

$$\sum_{(x,y,h)} \delta_{xyh}^u \leq 1 \text{ and } D_{u'}^{u_x, u_y} \leq r^{h_u} \quad \forall u, u' \in \mathbf{U} \quad (5)$$

Knowing that each drone u can be located in at most one position that is with the communication range of at least one

neighbouring drone. Where $D_{u'}^{u_x, u_y}$ is the Euclidian distance to the nearest neighbouring drone u' .

$$\gamma_{t_i}^u \leq \sum_{(x,y,h)} \delta_{xyh}^u \left(\frac{r^{h_u}}{D_{t_i}^{u_x, u_y}} \right) \quad \forall u \in \mathbf{U}, t_i \in \mathbf{T} \quad (6)$$

With the above constrain, we set the value for $\gamma_{t_i}^u$. If r^{h_u} (radius_range) is less than $D_{t_i}^{u_x, u_y}$ (distance), then $\gamma_{t_i}^u$ is equal to 0. In other words, if the target is outside the communication range of the 5G transmitter mounted on the drone, then the target cannot use that drone to access 5G. Therefore, the variable $\gamma_{t_i}^u$, can get either the value 0 or 1.

$$\sum_{u \in \mathbf{U}} \gamma_{t_i}^u \geq 1 \quad t_i \in \mathbf{T} \quad (7)$$

The above constrain ensures there exists at least one drone observing each target. The following equations show the solution space of the aforementioned $\gamma_{t_i}^u$ and δ_{xyh}^u decision variables.

$$\delta_{xyh}^u = \{0, 1\}, \quad \forall (x, y, h), 1 \leq x \leq x_{max} \quad (8)$$

$$1 \leq y \leq y_{max},$$

$$h_{min} \leq h \leq h_{max}, \quad u \in \mathbf{U} \quad (9)$$

$$\gamma_{t_i}^u = \{0, 1\}, \quad \forall t_i \in \mathbf{T}, u \in \mathbf{U} \quad (10)$$

And hence, $f(\delta)$, to be minimized, can be formulated as follows:

$$f(\delta) = A - \sum_{u \in \mathbf{U}} \delta_{xyh}^u * A'_i \quad (11)$$

where A is the total area to be covered, and A'_i is the area covered by the i^{th} UAV. By integrating Eqs. (11) and (4), to minimize the total energy consumed, while considering the movement time of the drone, $f(\delta)$ becomes:

$$f(\delta) = \beta \sum_{(x,y,h)} \sum_{u \in \mathbf{U}} \delta_{xyh}^u t + \alpha \sum_{(x,y,h)} \sum_{u \in \mathbf{U}} h \delta_{xyh}^u t + \frac{p_{max}}{s} \sum_{(x,y,h)} \sum_{u \in \mathbf{U}} h \delta_{xyh}^u \quad (12)$$

We propose two alternatives to solve the placement problem. Genetic algorithm and Simulated Annealing would be used to calculate the number of drones and their respective position in a given area while maintaining coverage and lifetime constraints in the 3D deployment area.

In Algorithm 1, we used SA to find the minimum number of drones such that line 1 is initializing the aforementioned placement problem parameters. T_0 is the selected initial temperature of the system. We use this parameter to enable us to accept or reject certain drone placement solutions. The higher the value of T_0 , the higher the probability of accepting a bad solution. Hence, we start by allocating a maximum temperature to T_0 . We gradually reduce the temperature of the system using the cooling factor α , which was selected in this work as 0.95. As the temperature reduces, so does the probability of accepting bad solutions. In line 1, we also initialize the initial solution δ_0 , which is heuristically selected for better results. Moreover, m is representing the

Algorithm 1 Simulated Annealing Pseudo code

1. Initialize: $T_0, \delta_0, \alpha, m, n$
 2. $\delta = X_0, \delta_F = X_0, T_1 = T_0$
 3. **For i=1 to m**
 4. **For j=1 to n**
 5. $\delta_{Temp} = \sigma(\delta)$
 6. **If:** $f(\delta_{Temp}) \leq f(\delta)$ **then**
 7. $\delta = \delta_{Temp}$
 8. **End If**
 9. **Else if:** $U(0, 1) \leq e^{-\left(\frac{f(\delta_{Temp}) - f(\delta)}{T_1}\right)}$ **then**
 10. $\delta = \delta_{Temp}$
 11. **End Else if**
 12. **If:** $f(\delta) \leq f(\delta_F)$ **then**
 13. $\delta_F = \delta$
 14. **End If**
 15. **End For**
 16. $T_{t+1} = \alpha \cdot T_t$
 17. **End For**
 18. **Return** δ_F
-

Algorithm 2 Genetic Algorithm Pseudo code

1. Initialize: PS, G_{max} , PC, PM
 2. $\delta = (\delta_{xyh}^u)^+$: Generate initial random solutions
 3. $f(\delta)$: Calculate fitness for random solutions
 4. Select BFS
 5. **For g = 1 to G_{max}**
 6. **For i = 1 to PS/2**
 7. Select two parents
 8. Crossover with PC
 9. Mutate with PM
 10. **End For**
 11. Replace parents with children
 12. Update BFS
 13. **End For**
 14. **Return** BFS
-

number of stages and n is the count of moves per stage with a certain temperature in SA algorithm. The number of moves allows us to explore the neighbourhood for possible solutions (i.e., UAVs' locations). Therefore, it is important that this value is carefully chosen. Line 2 assigns the initial solution to δ and to the final solution δ_F . It assigns also the initial temperature to the current temperature T_1 . Line 3 – 17 iterates over the initialized number of stages, where we decrement the temperature value after every stage. Lines 4 – 15 iterates over the number of moves at a given stage, where we get to explore the neighbouring solutions under a constant temperature. In line 5, we find a neighbouring solution using the move operator $\sigma(\delta)$, where $\sigma(\delta) = \delta + N(0, 1)$. We assign this solution to a temporary solution δ_{Temp} . Line 6 checks if the temporary drone placement solution is better than the current one. To achieve this, we substitute both the temporary and the current solution to the fitness function shown in Eq. (11). Line 7 assigns the temporary solution to

the current solution, if the condition in line 6 is satisfied. Line 8 ends the “If” statement. Line 9 to 11 covers an “Else if” statement. Line 9 uses the current temperature T_t , which represents the temporary solution and the current solution, to find an exponential value. The value is compared with a random number (between 0 and 1 exclusive) to determine whether the temporary bad solution shall be accepted or not. Line 10 assigns the temporary solution to the current solution, if the condition in line 9 is true. Line 11 ends the “Else if” statement. Line 12 – 14 represents another “If” statement. Line 12 checks if the current drone placement solution is better than the final solution using the fitness function. Line 13 assigns the current solution to the final one, if the condition in line 12 is true. Line 14 ends the “If” statement while line 15 ends the second “for” loop. Line 16 computes the next stage temperature of the system T_{t+1} using the cooling factor. Line 17 ends the first “for” loop and finally line 18 returns the selected final solution δ_F after all iterations have been completed. In Algorithm 2, we apply GA on the same problem to find the minimum number of drones and their optimal positions for the maximum coverage. We begin by initializing the aforementioned parameters in line 1. PS is the population size, which represents the count of the initial solutions to be selected. G_{max} is the maximum generation number for which an optimal solution is obtained. PC and PM are the probability of crossover and probability of mutation, respectively. These parameters are selected in order to evolve from one generation to the next. In line 2, we generate the initial solution in accordance with PS. This solution is represented by the set of 0’s and 1’s. In line 3, we compute the fitness of all initial solutions. And in line 4, we select the solution with the best fitness value. Lines 5 – 10 iterate over a specific generation number, while line 6 – 13 iterates for number of times equal to half of the population size. We iterate over half the population size because at every generation we select two parents for the crossover operation. In line 7, we select two parents. Then, in line 8, we produce two children by applying the crossover operation. In line 9, we mutate the produced children using a probability equal to PM. In this case, we consider each element in each solution. In line 10, we end the second “for” loop. In line 11, we replace all the parents with the newly produced children, forming the next generation of the evolved drone placement solutions. In line 12, we update the best-found solution (BFS) by substituting the newly produced solutions in the fitness function (i.e., Eq. (11)) so that we can find the best solution. This solution is compared with the previous BFS. If it is better, we update the BFS. In line 13, we end the first “for” loop, and in line 14, we return the BFS.

V. RESULTS & DISCUSSIONS

In this section, an in-depth analysis of the simulated results is presented. Java and Python were used to execute SA and GA respectively. An area of 80 kilometres squared was selected to be observed, with each drone having a 5G transmitter with an average range equal to 10 kilometres squared. For Simulated Annealing, initialization was done as follows, an initial

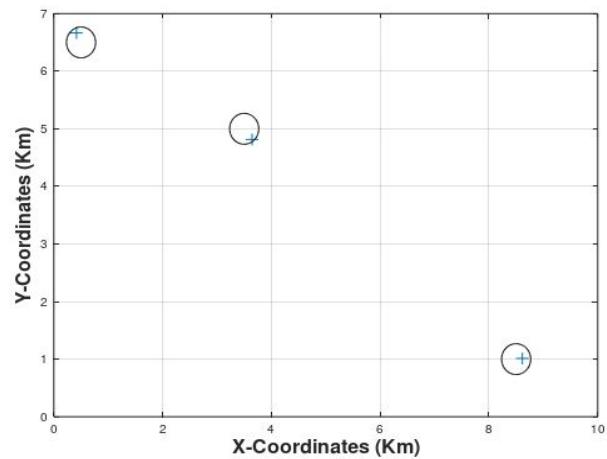


FIGURE 5. Three targets to cover.

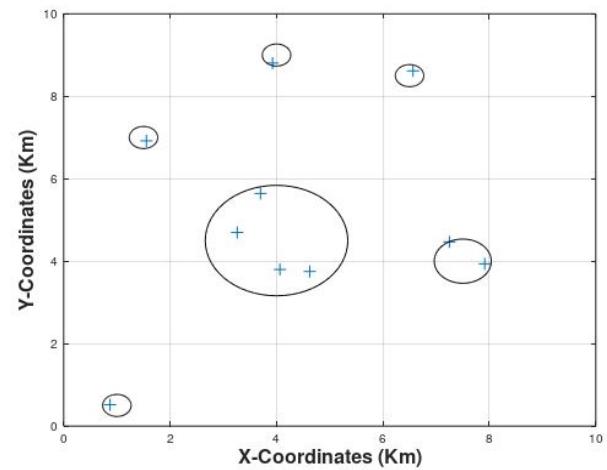


FIGURE 6. Ten targets to cover.

temperature of 300 was chosen, and an initial solution in terms 0s and 1s was chosen (1 indicating the presence of a drone in that vertex, 0 indicating its absence). The movement operator (α) in SA was set to be equal to 0.95, while m and n were set as 500 and 200 respectively. Additionally, initialization for genetic algorithm was done as follows: a population size of 8 was selected, stopping criteria (i.e. G_{max}) as 50, PC (Probability of Crossover) of 0.5 and PM (Probability of Mutation) was chosen as 1.

In Figures 5 – 7, we observe the number of drones required for a randomly generated count of targets on the ground. Figure 5 shows three targets that need to be observed. We notice that these targets have been located far away from each other. Therefore, one drone is not enough to cover all of them and three drones have been used in order to cover all targets. In Figures 6 and 7, we increase the randomly distributed number of targets to be equal to 10 and 22, respectively. We notice that the optimal number of drones required for this configuration increases to six in both figures. Accordingly, we remark that if the number of targets is equal to x, then the required number of drones to cover all targets

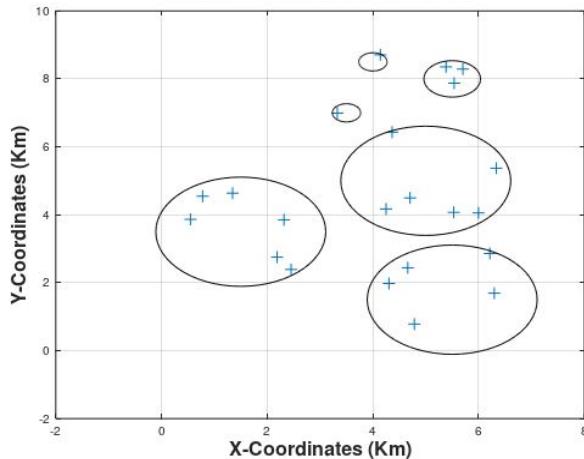


FIGURE 7. 22 targets to cover.

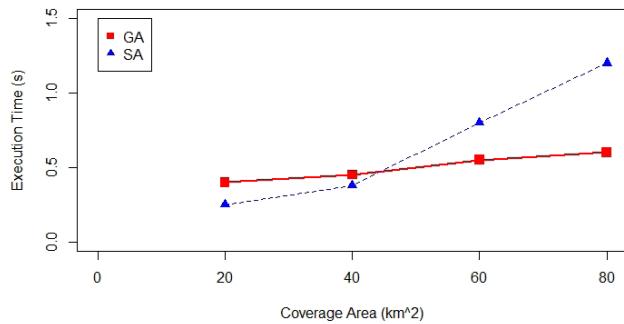


FIGURE 8. Execution time vs coverage area.

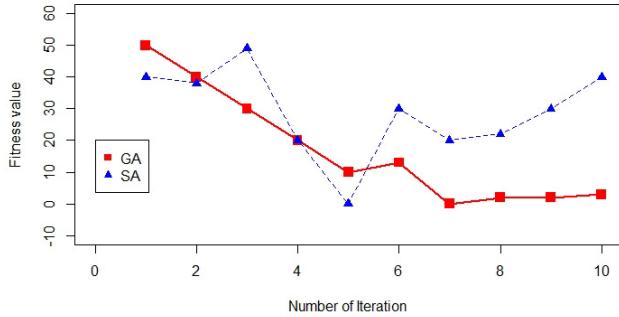


FIGURE 9. Fitness function behaviour.

can range from 1 to x . For example, if two targets are out of the communication range of a single drone, then we need two drones. However, if the two targets are within the communication range of a single drone, then we only need one drone. Therefore, we conclude that the configuration (distribution) of the targets in the covered area has a key influence on the minimum number of drones required to cover these targets.

Figures 8 and 9 below show the average execution time/fitness value over 100 runs for both SA and GA. The relative precision stopping criterion is used. Simulation runs are stopped at the first checkpoint when the condition

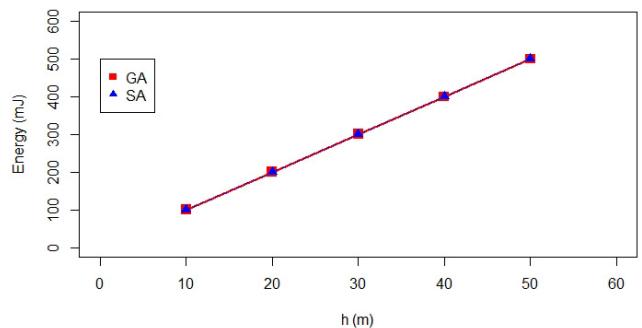


FIGURE 10. Energy consumed vs. the average height (h) of the UAVs.

$\delta \leq \delta_{\max}$ is met. Where δ_{\max} , which can have a value between 0 and 1, is the maximum acceptable value of the relative precision for confidence intervals at the $100(1-\alpha)\%$ significance level. All obtained results from the simulations are within the confidence interval of 5 % with a confidence level of 95%. And thus, both default values for α and δ are set to 0.05. This can help in assessing the evolutionary convergence for algorithms. Figure 8 below shows the execution time for both SA and GA with a varied coverage area. In this setup, the area covered by each drone is held constant, while the total area of interest is increased from 20 to 80 kilometres square. We can observe that the execution time for both algorithms lie between approximately 0.29 seconds and 1.2 seconds, with SA recording the fastest and GA recording the slowest time. From the obtained graph we also see that SA records the fastest time until the total area of interest is equal to 44 kilometres square, where both algorithms have the same execution time. However, when we increase the coverage area further, the execution time for SA slows drastically, while that of GA also slows but not as fast as that of SA. Consequently, GA realizes a faster execution time than SA for a coverage area greater than 44 kilometres square.

Therefore, we can clearly state that SA is capable of generating relevant solutions faster than GA when the coverage area is small. However, for larger areas to be covered by the UAVs, it is efficient to use GA as its time to generate optimal solutions is much shorter than SA.

In Figure 9, we analyse how both algorithms produce optimal solutions by tracking the fitness functions against the number of iterations. From the figure, we observe that genetic algorithm consistently produces a better fitness function output than the previous one until we get to the fifth iteration, where we see a slight deterioration. However, the general form of the GA function depicts that the parent selection and replacement method used in our algorithm, produced optimal solutions in each iteration. On the other hand, we observe that SA is more unpredictable in comparison to GA. This instability can be attributed to the nature of SA algorithm in finding the optimal value. That is why SA requires more computation power in comparison to GA as reported in [25]. Where the major drawback of SA is its slow convergence towards an optimal value [26]. This appears clearly in Figure 9, where

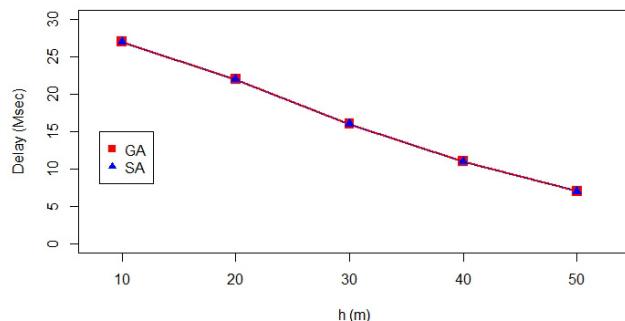


FIGURE 11. Average delay vs. the average height (h) of the UAVs.

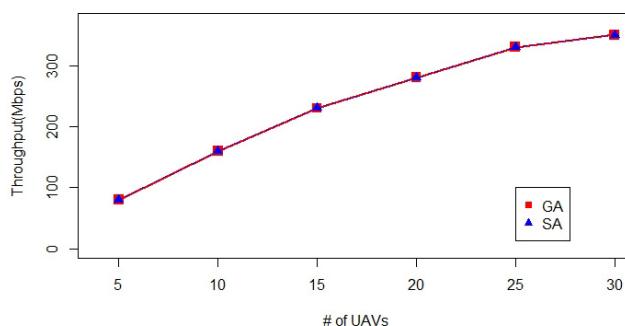


FIGURE 12. Throughput vs. the count of used UAVs.

SA algorithm experiences more local optimal values than GA. Hence, it is more likely to get stuck on a local optimal value in SA than in GA. The figure, therefore, suggests that we have better chances in reaching the global optimal value, when GA is applied rather than SA.

In Figures 10 – 12, we examined more communication-relevant parameters such as the average packet delay, energy consumption, and network throughput, while varying the average height and count of the utilized UAVs in the network. We applied this experimental work on the optimal solutions found by GA and SA. Since both algorithms were able to find the same optimal solution, both of them, GA and SA, have experienced identical behaviour.

In Figure 10, the average energy consumed versus the height of the positioned UAVs has been reported. Obviously, there is a linear relationship between the average consumed energy per delivered data packet and the height of the UAVs. This can be returned to the proportional relationship between the distance and the required transmission power [27].

In Figure 11, we testify the average experienced delay per packet while varying the average height of the UAV. In line with the aforementioned height discussions, when we increase the UAVs height values from 10 – 50m, the delay is decreasing monotonically. This is because of the coverage increment in the 3D space that allows lower number hops between the source user equipment (target) and the final destination (BS).

We examined also another critical communication metric in Figure 12, which is the overall network throughput measured in Megabytes per second (Mbps). This metric represents the amount of useful work a number of connected/networked UAVs can perform per the time unit in terms of the total data bytes that have been successfully delivered at the BS. We notice that as the number of used UAVs increases, the network throughput increases, as well. This makes sense because the more UAVs we have, the more alternative routes towards the BS will evolve also. This leads to better data delivery chances. However, this increment in terms of throughput reaches to a saturation level after a specific number of UAVs, where it stays in a steady state no matter how much extra UAVs are added.

VI. CONCLUSIONS

In this paper, we propose a framework for the optimal number of drone-BS and their positions determination. This framework is needed to provide the 5G cellular coverage to a given region, while considering the 5G transmitter's coverage range and energy constraints of the drones. The framework is very useful in providing coverage for outdoor critical events such as hurricane disasters, fire accidents, and densely populated areas such as urban areas and stadiums. We used two meta-heuristic algorithms, SA and GA written in two different languages, to find an optimised solution. The results from both algorithms were obtained, graphed, and analysed. The results obtained in this study show that SA takes precedence when the coverage area is small. However, for the extended coverage area, faster results are obtained using GA rather than SA. Moreover, our simulation results show that it is commonly possible to settle on a local optimal value when SA is applied, which is not the case with GA. Generally, we conclude that using GA can provide better results in timely manner for outdoor UAV critical applications. In the future work of this study, we would like to analyse the optimal deployment problem in indoor environments, while assuming dynamic UAVs.

REFERENCES

- [1] F. Al-Turjman, C. Altrjman, S. Din, and A. Paul, "Energy monitoring in IoT-based ad hoc networks: An overview," *Comput. Electr. Eng. J.*, vol. 76, pp. 133–142, 2019.
- [2] F. Al-Turjman, L. Mostarda, E. Ever, A. Darwish, and N. S. Khalil, "Network experience scheduling and routing approach for big data transmission in the Internet of Things," *IEEE Access*, vol. 7, pp. 14501–14512, 2019.
- [3] S. A. Alabady and F. Al-Turjman, "Low complexity parity check code for futuristic wireless networks applications," *IEEE Access*, vol. 6, pp. 18398–18407, 2018.
- [4] J. P. Lemayian and F. Al-Turjman, "Intelligent IoT communication in smart environments: An overview," in *Proc. Artif. Intell. IoT*. Cham, Switzerland: Springer, 2019, pp. 207–221.
- [5] M. Chen, J. Yang, Y. Hao, S. Mao, and K. Hwang, "A 5G cognitive system for healthcare," *Big Data Cognit. Comput.*, vol. 1, no. 1, p. 2, Mar. 2017.
- [6] L. J. Poncha, S. Abdelhamid, S. Alturjman, E. Ever, and F. Al-Turjman, "5G in a convergent Internet of Things era: An overview," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, May 2018, pp. 1–6.

- [7] K. Skouby and P. Lynggaard, "Smart home and smart city solutions enabled by 5G, IoT, AAI and CoT services," in *Proc. Int. Conf. Contemp. Comput. Inform. (ICI)*, Nov. 2014, pp. 874–878.
- [8] R. S. Sapakal and S. S. Kadam, "5G mobile technology," *Int. J. Adv. Res. Comput. Eng. Technol.* vol. 2, no. 2, pp. 568–571, Feb. 2013.
- [9] E. Kalantari, H. Yanikomeroglu, and A. Yongacoglu, "On the number and 3D placement of drone base stations in wireless cellular networks," in *Proc. IEEE 84th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2016, pp. 1–6.
- [10] D. Zorbas, L. D. P. Pugliese, T. Razafindralambo, and F. Guerrero, "Optimal drone placement and cost-efficient target coverage," *J. Netw. Comput. Appl.*, vol. 75, pp. 16–31, Nov. 2016.
- [11] K.-N. Park, B.-M. Cho, K.-J. Park, and H. Kim, "Optimal coverage control for net-drone handover," in *Proc. 7th Int. Conf. Ubiquitous Future Netw.*, Jul. 2015, pp. 97–99.
- [12] E. Tuba, R. Capor-Hrosik, A. Alihodzic, and M. Tuba, "Drone placement for optimal coverage by brain storm optimization algorithm," in *Proc. Int. Conf. Health Inf. Sci.* Cham, Switzerland: Springer, 2017, pp. 167–176.
- [13] 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.
- [14] H. Shakhreh and A. Kherishah, "Optimal Placement of a UAV to maximize the lifetime of wireless devices," 2018, *arXiv:1804.02144*. [Online]. Available: <https://arxiv.org/abs/1804.02144>
- [15] A. Rodriguez, A. Gutierrez, L. Rivera, and L. Ramirez "RWA: Comparison of genetic algorithms and simulated annealing in dynamic traffic," in *Proc. Adv. Comput. Commun. Eng. Technol.* Cham, Switzerland: Springer, 2015, pp. 3–14.
- [16] L.-Y. Yu and L. Lu, "Research on test data generation based on modified genetic and simulated annealing algorithm," in *Proc. 8th Int. Conf. Supply Chain Manage. Inf. (SCMIS)*, Oct. 2010, pp. 1–3.
- [17] D. R. Thompson and G. L. Bilbro, "Comparison of a genetic algorithm with a simulated annealing algorithm for the design of an ATM network," *IEEE Commun. Lett.*, vol. 4, no. 8, pp. 267–269, Aug. 2000.
- [18] F. M. Al-Turjman, H. S. Hassanein, and M. A. Ibnkahla, "Efficient deployment of wireless sensor networks targeting environment monitoring applications," *Comput. Commun.*, vol. 36, no. 2, pp. 135–148, Jan. 2013.
- [19] M. Quaritsch, K. Kruggl, D. Wischounig-Strucl, S. Bhattacharya, M. Shah, and B. Rinner, "Networked UAVs as aerial sensor network for disaster management applications," *E & I Elektrotechnik und Informationstechnik*, vol. 127, no. 3, pp. 56–63, Mar. 2010.
- [20] M. Asadpour, B. Van den Bergh, D. Giustiniano, K. A. 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.
- [21] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: A survey," *Ad Hoc Netw.*, vol. 6 no. 4, pp. 621–655, Jun. 2008.
- [22] M. Z. Hasan, F. Al-Turjman, and H. Al-Rizzo, "Analysis of cross-layer design of quality-of-service forward geographic wireless sensor network routing strategies in green Internet of Things," *IEEE Access*, vol. 6, pp. 20371–20389, 2018.
- [23] T. Pino, S. Choudhury, and F. Al-Turjman, "Dominating set algorithms for wireless sensor networks survivability," *IEEE Access*, vol. 6, pp. 17527–17532, 2018.
- [24] P. D. Mitchell, J. Qiu, H. Li, and D. Grace, "Use of aerial platforms for energy efficient medium access control in wireless sensor networks," *Comput. Commun.*, vol. 33, no. 1, pp. 500–512, Mar. 2010.
- [25] A. E. Gamal L. Hemachandra, I. Shperling, and V. Wei, "Using simulated annealing to design good codes," *IEEE Trans. Inf. Theory*, vol. 33, no. 1, pp. 116–123, Jan. 1987.
- [26] G. Storvik, "A Bayesian approach to dynamic contours through stochastic sampling and simulated annealing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 10, pp. 976–986, Oct. 1994.
- [27] F. Al-Turjman, "A novel approach for drones positioning in mission critical applications," in *Transactions Emerging Telecommunications Technologies*. Hoboken, NJ, USA: Wiley, 2019. doi: [10.1002/ett.3603](https://doi.org/10.1002/ett.3603).



FADI AL-TURJMAN (M'07) received the Ph.D. degree in computer science from Queen's University, Canada, in 2011. He is currently a Professor with Antalya Bilim University, Turkey. He is also a leading authority in smart/cognitive, wireless and mobile networks' architectures, protocols, deployments, and performance evaluation. His record spans more than 200 publications in journals, conferences, patents, books, and book chapters, in addition to numerous keynotes and plenary talks at flagship venues. He has authored or edited more than 12 published books about cognition, security, and wireless sensor networks' deployments in smart environments with Taylor & Francis, and the Springer (Top tier publishers in the area). He was a recipient of several recognitions and best papers' awards at top international conferences. He also received the prestigious Best Research Paper Award from Elsevier COMCOM Journal for the last three years prior to 2018, in addition to the Top Researcher Award for 2018 at Antalya Bilim University, Turkey. He led a number of international symposia and workshops in flag-ship IEEE ComSoc conferences. He is serving as the Lead Guest Editor in several journals, including the *IET Wireless Sensor Systems*, Springer EURASIP, MDPI Sensors, Wiley&Hindawi WCM, and the Elsevier COMCOM, and *Internet of Things*.



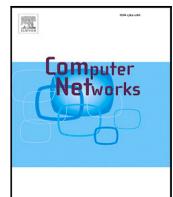
JOEL PONCHA LEMAYIAN received the B.Sc. degree in electrical engineering from Middle East Technical University, Cyprus, in 2017, where he is currently pursuing the master's (M.Sc.) degree in computer and electrical engineering. He is currently with Antalya Bilim University, Turkey. His research interests include 5G Communication networks, and the Internet of Things (IoT) applications.



SINEM ALTURJMAN received the B.Sc. degree in mathematics from Akdeniz University, Turkey, in 2014, where she is currently pursuing the master's (M.Sc.) degree in computer engineering. She is currently with Antalya Bilim University, Turkey. Her research interests include mathematical modeling, and statistical analysis in computer networks, and the Internet of Things (IoT) era.



LEONARDO MOSTARDA received the Ph.D. degree from the Computer Science Department, University of L'Aquila, in 2006. Afterwards, he cooperated with the European Space Agency (ESA) on the CUSPIS FP6 Project to design and implement novel security protocols and secure geo tags for works of art authentication. To this end, he was combining traditional security mechanisms and satellite data. In 2007, he was a Research Associate with the Distributed System and Policy Group, Computing Department, Imperial College London, where he was working on the UBIVAL EPRC Project in cooperation with Cambridge, Oxford, Birmingham, and UCL for building a novel middleware to support the programming of body sensor networks. In 2010, he was a Senior Lecturer with the Distributed Systems and Networking Department, Middlesex University, where he founded the Senso LAB, an innovative research laboratory for building energy efficient wireless sensor networks. He is currently an Associate Professor and the Head of the Computer Science Department, University of Camerino, Italy.



Q-FANET: Improved Q-learning based routing protocol for FANETs

Luis Antonio L.F. da Costa ^{a,*}, Rafael Kunst ^b, Edison Pignaton de Freitas ^a

^a Federal University of Rio Grande do Sul, Brazil

^b University of Vale do Rio dos Sinos (UNISINOS), Brazil

ARTICLE INFO

Keywords:

Flying Ad-Hoc Networks

Routing protocols

Reinforcement learning

Network performance evaluation

ABSTRACT

Flying Ad-Hoc Networks (FANETs) introduce ad-hoc networking into the context of flying nodes, allowing real-time communication between these nodes and ground control stations. Due to the nature of this kind of node, the structure of a FANET is dynamic, changing very often. Since it has applications in military scenarios and other mission-critical systems, an agile and reliable network is essential with robust and efficient routing protocols. Nonetheless, maintaining an acceptable network delay generated by the selection of routes remains a considerable challenge, owing to the nodes' high mobility. This article addresses this problem by proposing a routing scheme based on an improved Q-Learning algorithm to reduce network delay in scenarios with high-mobility, called Q-FANET. This proposal has its performance evaluated and compared with other state-of-the-art methods using the WSNET simulator. The experiments provide evidence that the Q-FANET presents lower delay, a minor increase in packet delivery ratio, and significant lower jitter compared with other reinforcement learning-based routing protocols.

1. Introduction

The technological advances in the last decades, especially the development and miniaturization of electronic components, lead to the popularization and the decrease in production costs of Unmanned Air Vehicles (UAV) [1]. Consequently, UAVs became applied in many different military and civilian domains, such as surveillance [2] and monitoring tasks [3], provision of communications networks in natural disasters [4] and conflict regions [5], or as general purpose aerial data collectors [6].

The use of a single UAV is already well understood and even considered ordinary. However, the use of multiple simultaneous UAVs, which can provide a significant advantage over the option of a single UAV, is still a research area with many possibilities. Despite its usefulness, these multiple UAV setup scenarios pose a challenge regarding communication, which is not a trivial task [7]. It is necessary to exchange packets between UAVs and base station(s) in situations with unique characteristics such as nodes that are continually moving in regions with artificial and natural obstacles, in addition to the most varied types of climate conditions in which they operate. Dealing with these difficulties demand the proposal of robust routing protocols, which is critical for the deployment of high level networked services, such as [8].

FANETs (Flying Ad-Hoc Networks) [9], composed of many UAVs, the nodes' high mobility creates a highly dynamic network topology [10]. Such scenario demands an adaptive and autonomous protocol

to address this issue, meaning that the protocol for routing in FANETs should be able to discover a stable neighbor to send the data by detecting changes in the environment. In this context, Q-learning is an adaptive reinforcement learning technique that receives input feedback from the environment – contributing to provide a routing design focused on adaptation – and presents a promising approach for a routing protocol scheme [11].

The premise of routing protocols that are based in Q-learning usually takes into account the data provided from the neighboring nodes, not making any assumption about any other network aspect. Most of them work by making the most suitable decision among the neighbors to forward a packet until it reaches the destination. Since multiple UAVs systems require real-time data transmission, a routing protocol must have a low delay to support several applications.

FANETs are highly dynamic and, if parameters from Q-learning, such as the learning rate and the discount factor, are fixed, the efficiency of the selection of the best action declines, making the selected link present a minimum possibility of establishing a connection to a neighbor node. This strategy applies to the majority of the known routing protocols that are based on Q-Learning and may limit their performance. Based on these constraints, this paper proposes a novel Q-Learning based routing protocol called Q-FANET, which addresses the mentioned limitations and combines positive features of existing approaches that use reinforcement learning to create an optimized

* Corresponding author.

E-mail addresses: lalfcosta@inf.ufrgs.br (L.A.L.F. da Costa), rafaelkunst@unisinos.br (R. Kunst), edison.pignaton@inf.ufrgs.br (E. Pignaton de Freitas).

routing protocol able to address the tough requirements presented by FANETs. The main contributions reported in this work are:

- **Delay and Jitter decrease:** Without a fixed routing table, Q-Learning can be used with specific rules and mechanisms to choose the optimal routing path based on a low delay constraint;
- **Exploration of last episodes with different weights:** Standard Q-Learning approaches always consider the most recent last episode to update the Q-Values what may lead to imprecise decisions. Therefore, the proposed solution considers a finite amount of last episodes;
- **Enhanced protocol parametrization based on channel conditions:** The transmission quality is an important element that can directly impact the delay of data transmission in a FANET, even when the optimal route is selected. The proposed solution also considers the channel conditions as a new metric to calculate the Q-Values parameter of the proposed approach.

The remainder of this paper is organized as follows. Section 2 reviews essential background concepts on routing protocols for FANETs, as well as the characteristics of Reinforcement Machine Learning and the relevant literature in this area. The proposed Q-FANET architecture is presented in Section 3. The simulation scenario, performed experiments, and obtained results are presented and discussed in 4. Finally, concluding the paper, Section 5 presents final remarks and directions for future investigations.

2. Background and related works

This section presents background aspects on the routing protocols for FANETs and the Reinforcement Learning paradigm, including the favored technique of Q-Learning. The section also discusses relevant related work in the area.

2.1. Routing protocols for FANETs

The existing routing protocols used in Mobile Ad Hoc Networks (MANETs) and Vehicular Ad Hoc Networks (VANETs) are not entirely suitable to be directly applied in UAVs networks, as they must adapt to the higher degree of mobility that characterizes FANETs and the consequently more frequent changes in the topology [12]. According to the literature, one can organize the routing protocols used in FANETs into two categories: single-hop routing [13] and multi-hop routing [14].

For the single-hop routing protocols, a static routing table defines the transmission paths, being computed and loaded before the start of the UAV nodes' operation and cannot be changed. In the multi-hop routing protocols, packets are forwarded hop by hop towards the destination. The selection of the proper hop node is the core issue of the route discovery. Usually, one can classify these protocols into two categories: topology-based and position-based routing [15]. Furthermore, the first category consists of three specific types of protocols: proactive protocols, reactive protocols, and hybrid protocols.

2.1.1. Static protocols

Lightweight and designed for fixed topologies, these protocols are not fault-tolerant, since, in case of failure, it is mandatory to wait until the end of the operation to update the routing table, which makes them not suitable for dynamic environments. Examples of these protocols are Load-carry-and-deliver (LCAD) [16] and Data Centric Routing (DCR) [17].

2.1.2. Proactive protocols

These protocols record and store the routing information in each UAV belonging to the network, with each node updating its routing table to meet changes in the network topology. Therefore, the routing paths can be chosen to send packets with minimum waiting time [18]. Although highly used due to its characteristics of serving high-mobility network scenarios, this type of protocol presents several disadvantages, such as the number of control packets necessary for the route establishment, increasing communication overhead. Examples of proactive protocols include Destination Sequenced Distance Vector (DSDV) [19] and Optimized Link State Routing Protocol (OLSR) [20].

2.1.3. Reactive protocols

This class of routing protocols presents low overhead since they create routing information only when there is a communication between two nodes. However, the overhead reduction comes at the cost of increasing the end-to-end delay, due to the processing time required to establish a path [21]. Examples of reactive protocols include Dynamic Source Routing (DSR) [22] and Ad-hoc On-demand Distance Vector (AODV) [23].

2.1.4. Hybrid protocols

Representing a combination of proactive and reactive routing protocols, these protocols are used to overcome the limitations of each isolate approach, i.e., time demanded to find routes and control messages overhead [24]. Examples of hybrid protocols include Zone Routing Protocol (ZRP) and Temporarily Ordered Routing Algorithm (TORA) [25].

2.1.5. Position-based protocols

They overcome the limitations of proactive and reactive protocols, specifically with the static routing tables and the establishment of the route before the transmission of each packet, correspondingly [26]. Examples of position-based protocols include Greedy Perimeter Stateless Routing (GPSR) [27].

2.1.6. Hierarchical protocols

The last class of routing protocols explores cluster-based approaches to perform route discovery. Examples of hierarchical protocols include Mobility prediction clustering (MPC) [28] and Clustering Algorithm of UAV Networking [29].

2.2. Reinforcement learning

Reinforcement Learning (RL) is another crucial paradigm of the learning process in Artificial Intelligence [30]. A simple analogy that is possible to imagine is a person that does not know the flavor of specific food and tries it for the first time. This individual may identify the food as something good or bad, and this acquired knowledge may apply to decide next time if this individual should eat or do not eat that food. In the context of Computer Science, RL applies to algorithms with some knowledge about the task that they should perform and can use it to make better choices to complete the task. As shown in [31], ML can deal with several challenges involving the communication in FANETs, as well as improving different design and functional aspects such as channel modeling, resource management, positioning, and security. The main components of an RL algorithm are the agent, the environment, the state, the action, and the reward.

The agent learns over time to behave in an optimized manner in an environment by interacting continuously with it. During its learning course, the agent experiences various scenarios in the environment, which are called states. While in a particular state, the agent may choose from a set of allowable actions and, depending on the result of each action, it receives rewards or penalties. Overtime, the agent learns

ways to increase these rewards with the goal of behaving optimally at any given state.

Q-learning is a basic form of RL that makes use of Q-values to improve the learning agent's behavior iteratively. The algorithm defines values for states and actions. $Q(S, A)$ represents an estimation of how good it is to perform action A at a given state S . In Q-Learning, an agent starts from a given state and performs several transitions from its current state to the next one. Every transition happens due to an action considering the environment with which the agent is interacting. In each step of the transition, the agent performs an action, receives a reward, then transitions to another state until it reaches the goal. This situation is called the completion of an episode. The algorithm estimates $Q(S, A)$ using a specific rule that calculates the value of Q at every interaction of an agent with the environment, expressed by (1):

$$Q(S, A) \leftarrow Q(S, A) + \alpha(R + \gamma \max Q(S', A') - Q(S, A)) \quad (1)$$

S represents the current state of the agent, A is the current action chosen in accordance to a specific policy, S' represents the next state that the agent is going to transition, A' is the next best action to pick using the current Q-value estimation, and R is the reward received from the current action. Other important parameters of this update function are:

- **Discounting factor for future rewards (γ):** a value set between 0 and 1. Common Q-learning approaches consider future rewards less valuable than current ones, therefore they must be discounted;
- **Learning rate (α):** step length taken to perform the update of the estimation $Q(S, A)$;
- **ϵ -greedy policy:** a simple method to select actions using the current Q-value estimations. The probability of selecting the action with the highest Q-value is given by $(1-\epsilon)$ while selecting a random action is (ϵ) .

2.3. Related work

Routing protocols based on Q-Learning methods are promising to deal with the dynamic changes in FANETs. Q-Grid [32] is a protocol designed for VANETs that makes use of macroscopic (optimal next-hop grid by querying Q-value table learned offline) and microscopic aspects (specific vehicle in the optimal next-hop grid) to perform the routing decision, dividing the region into different grids. With this approach, Q-Grid calculates the Q-values of various movements between neighboring grids for a specific destination using the Q-learning algorithm. The performed simulations have shown that Q-Grid presents advantages compared to other existing position-based routing protocols.

Q-learning based Adaptive Routing model (Q-LAR) [33] detects the level of mobility at each node of the network and proposes a metric, entitled Q metric, which accounts for both the static and dynamic routing metrics to respond to topology changes. Simulation results show that Q-LAR is more effective than the standard OLSR protocol. Q-Learning based geographic routing (Q-Geo) [34] proposes a system to minimize the network overhead in high mobility scenarios. The authors compare the performance of Q-Geo with other approaches utilizing the NS-3 simulator, with the results showing that Q-Geo presents a higher packet delivery ratio and also a lower network overhead than the compared solutions.

Q-Fuzzy [35] uses fuzzy logic considering parameters as transmission rate, energy state, and flight status between neighbor UAVs to determine the optimal routing path to the destination. The algorithm updates these parameters dynamically using an RL method. The results show that, compared with distance vector routing based on Q-Values, Q-Fuzzy can maintain low hop count and energy consumption and extend the network lifetime. Q-Learning Multi-Objective Routing (QMR) [36] is a routing protocol that uses adaptive parameters (as the learning rate and the mechanism of exploration) combined with link

conditions and specific constraints to provide low delay and low energy consumption. The performed simulations compared QMR with Q-Geo, and showed that the proposed routing scheme presents a higher packet arrival ratio, lower delay, and energy consumption.

In the context of MANETs, [37] propose a Q-learning based CSMA/MAS protocol. In this method, every node in the network is able to be synchronized and then attend in a round-robin way without have to deal with contention collisions. At the network layer, the approach performs several modifications to Q-Geo and Q-Grid. The results have shown that this transmission protocol approach provides a higher packet arrival ratio and lower end-to-end delay than the existing transmission protocols. QNGPSR [38] is a routing protocol which is inspired on the GSPR protocol for the ad-hoc network. It aims at reducing the network delay by using reinforcement learning to perform the next-hop selection. Results show that QNGPSR provides a higher packet delivery ratio and a lower end-to-end delay when compared to the performance of GSPR. In the context of cognitive sensor networks, Q-Noise+ [39] proposes three improvements to algorithms which are based on reinforcement learning and used for dynamic spectrum allocation. Simulation results show that Q-Noise+ allows better quality in the allocation of channels (up to 6 dB), and also presents 4% higher efficiency compared to the standard Q-Learning.

Table 1 presents a summary of the main aspects and gaps of the works found in the literature, highlighting the contribution of this present one. The proposal here presented considers the techniques behind QMR and Q-Noise+ providing an improved solution with optimal channel selection to provide low delay and low jitter in the challenging environment of UAV-networks, using a finite number of last episodes to calculate the Q-Values.

3. Q-FANET protocol

This section introduces and describes Q-FANET, an improved Q-learning based routing protocol for FANETs. This solution takes into account two Q-Learning methods: Q-Noise+ and QMR. In Q-FANET, nodes use a reinforcement learning algorithm without knowing the entire network topology to perform optimal routing decisions focusing in delivering low-delay service.

3.1. Q-FANET design overview

Q-FANET proposes changes and improvements in several components of QMR and other techniques suitable with Q-Learning to propose a novel approach to deal with FANET routing. The proposal consists of two major modules, the Neighbor Discovery and the Routing Decision. Fig. 1 presents an overview of Q-FANET's architecture. The elements C1 and C2 are connectors that serve only as visual tools, not representing any important component of the algorithm's architecture.

An important assumption is that this proposal considers FANETs composed of low to medium speed UAVs, such as multi-rotors and aerostats [40,41], flying with speeds under 20 m/s, such as the 3DR Iris+ quadcopter, a mini-UAV manufactured by 3-D Robotics [42]. Considering these types of UAVs, it is reasonable to assume that the proposed RL algorithm successfully converges to useful solutions.

3.2. Routing neighbor discovery

The Neighboring Discovery module is a control structure used to maintain the routing information updated. Q-FANET updates the location of the nodes regularly. The updating frequency is one of the parameters of the proposal, and its default value is 100 ms. In cases where a node does not inform its location within a specific expiration time of 300 ms, its neighbors remove the specific route from their routing tables.

This updating process in Q-FANET relies on the exchange of *HELLO* packets. In this case, a given network node broadcasts these packets

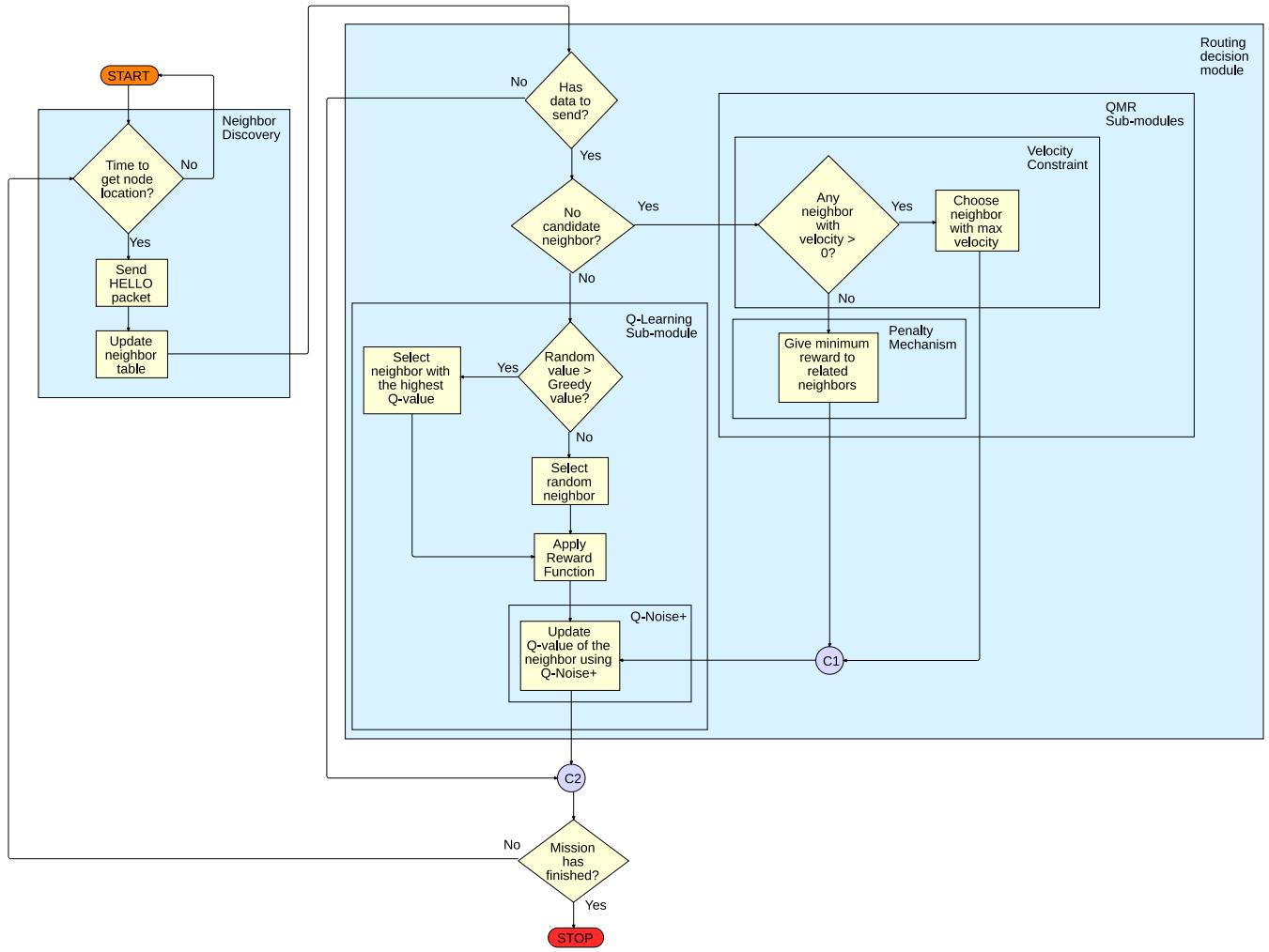


Fig. 1. Q-FANET flowchart exposing its internal modules.

aiming to discover its neighbors. This packet exchange happens periodically and carries the following information: node's geographic location, energy level, mobility model, queuing delay, learning rate, and Q-value. When a node receives *HELLO* packets, the node will use the packet information to establish and maintain its neighbor table.

The idea behind this module is to keep the network ready for transmission at any time. Therefore, its functionalities are always running, despite the existence of a current transmission session. Whenever a transmission is necessary, the Neighbor Discovery module communicates with the Routing Decision, the other module of Q-FANET, to provide information regarding the best routes.

3.3. Routing decision module

The Routing Decision Model receives information about the available routes and selects the one for a given node to transmit data. To do that, it counts with two sub-modules: (I) QMR and (II) Q-Learning that consider the background of existing approaches but modify them to improve the routing response. The flow of the model will proceed to these two sub-modules in a scenario where the list of possible neighbor candidates is not empty. This situation corresponds to the nonexistence of the routing hole problem [43], i.e., all the neighbors of a node are distant than the distance from this node to the destination.

3.3.1. Q-learning sub-module

The standard Q-Learning algorithm applies a reward-based approach that considers two main aspects: (I) the successful transmission

rate in the last episode and (II) the sum of the success rates of all past episodes. However, except for the most recent episode, the remaining episodes generally possess the same weight in the decision process. The problem is that this approach may lead to imprecise decisions, especially when considering scenarios where a high number of episodes is necessary, such as in military, surveillance, and rescue missions, where the dynamism of the situation implies high mobility of the nodes.

Taking the mobility problem into consideration, an extension to the Q-Learning, called Q-Learning+, was proposed by [39]. This extension of the algorithm considers a finite amount of past episodes, defined by a lookback value (l). In this approach, the newer the episode, the higher its weight will be. Consequently, the Q-value at time $t + 1$ is obtained as shown in (2), where w_i states for the weight of the last l instants of time and r_i represents the calculated reward based on $l + 1$ actions. Also, Q-Learning+ evaluates each of its actions based on the total sum of all of its future rewards, therefore the discount factor (γ) is set to 1 and is omitted from the equation, since its usage does not change the calculated Q-value.

$$Q_{t+1}(a_t) = (1 - \alpha) \sum_{i=1}^l [w_{t-i} r_{t-i}] (a_t) + \alpha r_t (a_t) \quad (2)$$

Although Q-Learning+ improves the original approach's efficiency, it only considers the number of successful transmissions to perform its decision, ignoring the channel's propagation conditions. In order to also consider the channel conditions to perform routing-related decisions, another algorithm called Q-Noise+ is available. This algorithm takes

Table 1
Related work summary.

Ref.	Main approach	Discussed research gap
[32]	Use of macroscopic and microscopic aspects to perform the routing decision, dividing the region into different grids	When the time slot is long, the prediction of Q-Grid by Markov chain may be inaccurate
[33]	Q-Learning technique that detects the level of mobility of each node and uses a special metric to account for the static and dynamic route metrics	The algorithm is too much dependent of the metric and could use different approaches, such as energy-aware metrics
[34]	A machine-learning-based geographic routing scheme to reduce network overhead in high-mobility scenarios	Underutilization of network resources and capacity
[35]	Q-learning based fuzzy logic in a multi-objective routing algorithm using link and path-related parameters	Method performance is not outstanding when the number of nodes is small because of power consumption
[36]	QMR is able to provide low-delay and low-energy service guarantees by adaptively adjusting Q-Learning parameters	No implementation and testing in physical UAV networks
[37]	The cross-layer protocol for MANETs uses CSMA/MAS and modified Q-Learning based routing	The experiments do not take into account node failures or the routing hole problem
[38]	A Q-Learning network enhanced GSPR routing protocol that reduces end-to-end delay and packet delivery ratio	The method is not optimized for feature extraction and does not take into account the link status
[39]	It improves the Q-Learning method for routing protocols in CR sensor networks by considering the channel conditions and the SINR of the channels	It lacks analysis of the trade-off between more precise decisions and network overhead
[This work]	A Q-Learning mechanism that combines features of existing approaches with channel conditions analysis	(Covered Gap) Combination of QMR and Q-Noise+ features into an approach that provides lower end-to-end delay, jitter and higher packet delivery ratio than other methods in UAV dynamic networks

into account the quality of the transmission as a secondary metric, calculated considering the Signal-to-Interference-plus-Noise Ratio (SINR) measured in a given channel. This approach tries to avoid selecting an available channel that can be noisy — a situation that might occur when using the Q-Learning+ approach.

Two aspects are considered for the decision taken by Q-Noise+: (I) the learning rate considering the reward obtained in an episode T and (II) a quality status that considers both the SINR level of the channel. These values are parameters to calculate a weighted reward for the most recent episode. Eq. (3) presents details regarding this calculation.

$$Q_{t+1}(a_t) = (1 - \alpha) \sum_{i=1}^l [w_{t-i} r_{t-i}](a_t) + \alpha r_t(a_t) + (S_w * \eta) \quad (3)$$

In (3), S_w ($0 \leq S_w \leq 1$) represents the weight of the SINR in the calculated reward. η corresponds to SINR ranges (see Table I [44]). This weight is set as a parameter that defines the importance given to the quality of transmission. This means that, by increasing the value of S_w , it also increases the SINR impact on the Q-value of the channel. On the other hand, the η parameters defines a set of SINR values that have been chosen to change the Q-value according to the channel conditions.

Table 2
SINR ranges.

SINR value	η
$SINR < 15$ dB	0
$15 \text{ dB} \leq SINR < 17$ dB	0.25
$17 \text{ dB} \leq SINR < 20$ dB	0.5
$20 \text{ dB} \leq SINR < 25$ dB	0.75
$SINR \geq 25$ dB	1

It is expected that, when in a scenario with favorable propagation conditions, the η value will be higher, which will increase the Q-value of the channel. Consequently, as the channel conditions become worse, η decreases, not changing the Q-value.

Q-FANET also makes use of an ϵ -greedy policy for exploration and exploitation [45]. The exploration consists in searching for unknown actions (i.e. obtain new knowledge). Nevertheless, exploration in excess makes it complicated to maintain some better actions. On the other hand, the exploitation aims to create advantages by exploring actions, which have a chance to generate high rewards. Although, if exploitation is used in excess, it may become difficult to choose some undiscovered potential actions that might be optimal.

With the goal of keeping the trade-off between exploration and exploitation well balanced, the ϵ -greedy policy instructs the Q-Learning Sub-module to explore by choosing a random path with probability ϵ (usually 10%) and exploit by choosing the option which offers the highest Q-Value.

Q-FANET benefits from these approaches as building blocks of the Q-Learning Sub-module. One crucial adaptation proposed in Q-FANET regards the reward function, which is discussed in the next section.

3.3.2. Reward function

In Q-FANET, a data structure called R-Table (Reward Table) is proposed to store reward cells. The initial value of the reward cell values is zero. After each forwarded data from node i to node j , the R-table values are updated according to the logic expressed in (4):

$$R(s, a) = \begin{cases} r_{\max} = 100, & \text{if link}(i, j) \text{ leads to destination} \\ r_{\min} = -100, & \text{if link}(i, j) \text{ is local minimum} \\ r = 50, & \text{otherwise} \end{cases} \quad (4)$$

where s represents a state, and a represents an action taken by the agent, which in the network scenario, corresponds to a node. Therefore, when a packet is at node i the current state associated with this packet is s_i . An action $a_{i,j}$ represents the forwarding of the packet from node i to neighbor node j , using link (i, j) , changing the state s_i to s_j . For each change of state, the reward function $R(s, a)$ is applied.

The maximum reward value r_{\max} will be applied to a link (i, j) when the next-hop j is the destination node. On the other hand, the minimum reward value r_{\min} will be used when the node i is defined as a local minimum, meaning that all its neighbors are set farther away from the destination than itself. In any other situation, Q-FANET provides a reward of 50. For example, this situation occurs when node j is a relay node in the path to the destination.

This function structure is based on the binary reward function approach [46] and the values for the rewards were empirically chosen taking into account that they must clearly represent a difference between the maximum and the minimum reward, i.e., when the hop is leads to the destination node, to a local minimum or it is a general link.

3.3.3. QMR sub-module

The QMR sub-module is responsible for the penalty mechanism and it controls the constraint regarding the nodes' velocity to support the best decision. This sub-module is going to be used in scenarios where

the list of possible candidate neighbors is empty, i.e., a routing hole problem happened in the network.

Penalty Mechanism :

The occurrence of routing holes increases the delay of transmitting a data packet. Q-FANET proposes a modification to the penalty mechanism of QMR, aiming to reduce the existence of routing holes. This mechanism applies to the following cases:

- **Routing hole:** when a node j discovers that all of its neighbors are further than itself from the destination, then it sends a feedback to the previous node i .
- **Not-ACK:** when a node i does not get an ACK packet from next-hop node j , meaning that node j may be in a failure state.

In both scenarios, the action taken by the penalty mechanism will be that node i will give the r_{min} for the link i,j and update the corresponding Q-Value of the link. Nevertheless, even if there is a case where both a routing hole and a Not-ACK happen in the network, the penalty mechanism will have the same behavior it would have in each separated case.

Velocity Constraint :

A velocity constraint is necessary to obtain the minimum delay between the hops. Q-FANET adapts this constraint by simplifying the one defined in QMR. In Q-FANET, the velocity constraint of link i,j is defined in (5), observing (6):

$$\text{Velocity}_{(i,j)} = \frac{d_{(i,D)} - d_{(j,D)}}{\text{delay}_{(i,j)}} \quad (5)$$

$$\begin{cases} \text{Velocity}_{(i,j)} < 0, \text{ if } d_{(j,D)} > d_{(i,D)} \\ \text{Velocity}_{(i,j)} > 0, \text{ if } d_{(j,D)} < d_{(i,D)} \end{cases} \quad (6)$$

where d represents the distance in meters, and D stands for the final destination node. Therefore $d_{(i,D)}$ describes the distance between nodes i and D , and $d_{(j,D)}$ represents the distance between nodes j and D .

Also, (5) and (6) show that higher delays will lead to lower velocity constraint values. Moreover, velocities below zero indicate that the distance between node j and the destination is bigger than the one from i to the destination. Q-FANET will obtain this velocity information during the routing neighbor discovery process, where each node will create its routing table.

Since Q-FANET is a reinforcement learning algorithm, one of the concerns regards the possibility of the algorithm not converging. We deal with this potential issue by prioritizing that transmission occurs instead of guaranteeing an optimized link selection. The implementation of this feature occurs within the “Routing Neighbor Discovery” function, which keeps an updated list of nodes’ locations within the network. This module works along with the “Route Decision” function in the transmission route selection. “Route Decision” implements a timer to act as a deadline for Q-FANET to converge. Although this timer is a parameter of Q-FANET, in our simulations, we consider a maximum convergence delay of 10 s. We set this value considering the UAVs’ velocity and their impact on the network topology. In an unlikely situation in which Q-FANET does not converge, the route decision module selects the last successful calculated route. If there is no successful route available, the algorithm deals with the transmission considering that it is the first transmission, i.e., Q-FANET randomly selects a route to start the reinforcement learning process.

3.4. Q-FANET working example

Fig. 2 shows a simple network topology as an example scenario for the application of the proposed Q-FANET. In this network there are the source node (S), destination node (D) and relay nodes (1, 2, 3, 4, 5). In the scenario represented in **2a**, it is assumed that at a current time t , there are two data packets in the network, i.e., two agents. Packet 1 (P1) and Packet 2 (P2) are in nodes S and 1, and their states are S_S

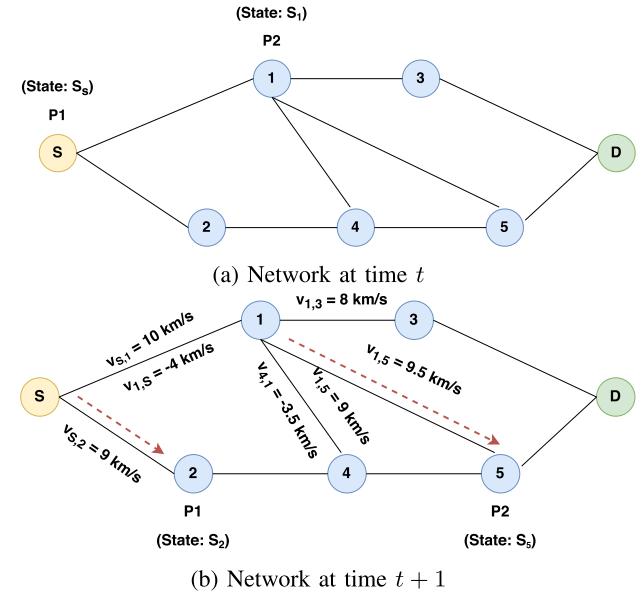


Fig. 2. Network topology example.

and S_1 , respectively. Also, it is possible to determine that the set of neighbors of node S is node 1, node 2 and of node 1 is node S, node 3, node 4, node 5. Node S and node 4 need to select one of their neighbors as the next hop to forward the packet until it reaches the destination node.

By using the Velocity Constraint sub-module from QMR, it is possible to determine the velocity of the links described in **2b**. Since the velocity of the candidate neighbors must be greater than zero, the set of candidate neighbors is then obtained. Therefore, the set of neighbors of node S and node 1 are node 1, node 2 and node 3, node 4, node 5. Suppose that at time t , the Q-value using only the Q-learning+ approach (without considering the channel conditions), SINR and the Q-value when using the Q-Noise+ approach for the links in the network are shown in **Table 3**. It is important to note that the SINR information was obtained in a time $t-1$, since channel conditions may vary through time.

For Packet 1 (agent 1), although the Q-value of the link (S,1) is bigger than the Q-learning+ Value of the link (S,2), the link (S,2) presented a greater SINR than the link (S,1), therefore the updated Q-value by Q-Noise+ of the link (S,2) will be higher than the value of the link (S,1) and the link (S,2) will be selected to forward the packet. Also, the Q-learning+ Value of the link (S,2) is equal to the initial value of 0.5, which implies that now Packet 1 (agent 1) will explore a new link, meaning that previously undiscovered links might be explored.

For Packet 2 (agent 2), the link (1,5) presents the higher Q-learning+ Value between the possible forwarding neighbor candidates. Even so, notice that this link had at time $t-1$ the highest SINR value amongst all possible links and it was the best channel to forward the packet. Therefore, the updated Q-value by Q-Noise+ of the link (1,5) shows that this is the most suitable link to forward Packet 2. Also, the link (1,5) was the best forwarding link in the past (even considering the channel conditions, with its high SINR). Since Packet 1 (agent 1) is choosing the best link in the past to forward data, Q-FANET is exploiting the knowledge that has been previously learned.

Q-FANET contributions include implementing reinforcement learning in a challenging mobile environment. Even though this work considers UAVs that do not reach very high speeds, such as multi-rotors and aerostats, the mobility raises several challenges. For example, it is important to deal with the convergence uncertainty problem. Dealing with this problem demanded the proposal of a non-convergence detection

Table 3
Network link information.

Link	(S,1)	(S,2)	(1,3)	(1,4)	(1,5)
Q-learning+ value	0.63	0.5	0.52	0.61	0.7
SINR	16.2 dB	17.5 dB	17.8 dB	15.5 dB	20.3 dB
Q-Noise+ value	0.8	0.85	0.87	0.78	1.22

Table 4
Simulation parameters setup.

Parameters	Settings
Area size	500 m × 500 m
Number of nodes	25
Radio propagation	Propagation range, rang = 180 m
Interferences	Interferences orthogonal
Modulation	Modulation bpsk
Antenna	Antenna omnidirectional
Battery	Energy linear
HELLO interval	100 ms
Expire time	300 ms
Initial Q-value	0.5
minspeed	0 m/s
maxspeed	15 m/s
Data packet	127 Bytes
SINR weight	0.7
Look back for Q-Noise+ (l)	10
w	0 < w < 1
α	0.6
ϵ	0.1

mechanism. Another contribution of Q-FANET is the “Routing Neighbor Discovery” function and its integration with the “Routing Decision” module.

Besides these original contributions, Q-FANET presents side contributions by combining and modifying features of QMR and Q-Noise+ approaches, developing a strategy that benefits from several aspects of Machine Learning and Channel Occupation techniques. The penalty mechanism and the velocity constraint are adaptations of the QMR approach. Q-FANET also simplifies QMR’s reward function. Finally, Q-FANET combines its ability to discover neighbors with Q-Noise+’s strategy to evaluate the channel conditions. In this sense, the function from Q-Noise+ works as another constraint to the Q-FANET algorithm to decide which is the most suitable link to perform the transmission.

4. Experiments and results

Experiments use simulations to compare the performance of Q-FANET with other existing approaches, namely, Q-Geo, Q-Noise+, and QMR, using the event-driven wireless networks simulator WSNet [47]. The WSNET simulator generally applies to the simulation of large-scale sensor networks’ behavior but was adapted to the FANET evaluated in this article. The simulation scenario consists of 25 nodes (representing the UAVs), randomly distributed in an area of 500 m × 500 m. The simulation tool randomly selects the source, which transmits a time-varying data flow. Table 4 summarizes the simulation parameters. These parameters were selected following the values used in work that reported QMR [36], for comparison purposes.

In the simulations performed, all of the mobile nodes move according to the Random Waypoint Mobility Model [48], and follow the study of [49]. In this manner, a mobile node makes a movement from its current location to a new random location by selecting a direction and speed (in [minspeed, maxspeed]). In this mobility model, after the mobile node moves to the new destination, it will pause for a certain period of time, and then resume moving to another new location. For the experiments, this period is set to 0. Also, this approach considers a lookback value of ten episodes to parametrize Q-Noise+. The simulation tool generates random weights for each episode at the beginning of the simulation. Different from the original Q-Noise+ approach, the most recent episode does not receive the higher weight.

Therefore, in this manner, Q-FANET can assign random weights to each episode, not prioritizing a specific one. It is also important to highlight that, following the approach used on Q-Noise+, both the values of the learning rate (α) and discount factor (γ) remain unchanged (0.6 and 1, respectively) throughout the whole experiment.

Two sets of simulations were performed for each algorithm. In the first one, all 25 nodes work correctly, while the second one simulates the existence of ten faulty nodes. This last set evaluates the protocols’ capability to overcome the unfavorable conditions of a network with faulty nodes. The data transmission intervals vary between 10 ms and 50 ms, with an increasing pace of 10 ms [36]. This approach allows comparing the results of QMR and Q-FANET in the same conditions. For each transmission interval, initially 100 simulations runs were performed, with the final values being represented as the results’ average. Then, the confidence interval was calculated using the t-student distribution and performed additional runs, if necessary, to reach a confidence interval of 95%. The evaluation considers the following metrics.

- **Maximum end-to-end delay:** the maximum delay of a data packet transmission made from the source node to the destination node.
- **Jitter:** The degree of change in the delay of data packets transmitted from the source node to the destination node.
- **Packet delivery ratio:** The ratio of the number of received data packets by the destination node in relation to the number of data packets transmitted by the source node.

As most of the reinforcement learning techniques, it is recommended that Q-learning uses a training time or number of cases to compose a training set, which would be executed until a convergence of results was obtained. Nevertheless, since 100 simulations are executed for each time interval – with the final average representing the result – the training aspect of this algorithm is not used in the simulation.

This simulation parameters were chosen according to the ones stated in [36] in order to compare the results obtained with Q-Noise+ and Q-FANET in an equivalent test environment. Also, the SINR values vary during the experiments, and it determines the value of the η parameter according to Table 2.

4.1. Scenario without faulty nodes

In this simulation, the source node sends one thousand data packets considering different transmission intervals. Figs. 3 to 5, show the results of Q-FANET for the different data intervals, in comparison with QGeo, Q-Noise+, and QMR, considering the performance metrics mentioned above.

In Figs. 3 and 4 it is possible to observe that Q-FANET presents a lower max end-to-end delay, as well as a lower jitter than QGeo, Q-Noise+ and QMR. There are two reasons for this better performance: the first is the use of the velocity constraint adapted from QMR, and the second is the channel selection from Q-Noise+. The velocity constraint always select the routing path with the lowest delay from source to destination. Furthermore, the use of Q-Noise+ features gives a higher weight to the channels with a good SINR value. Exploring advantageous features of both algorithms, the new proposal surpass them two. Besides, the standard deviation error bar shows that the results of Q-Noise+ and Q-FANET are inside an acceptable error margin.

Fig. 5 shows that Q-FANET increases the packet delivery ratio compared to the other algorithms. This improvement mainly occurs because the weighted last ten episodes change the learning rate and discount factor in the Q-Learning sub-module of Q-FANET. The SINR-based selection of the best channel, in the QMR sub-module, also collaborates for this result.

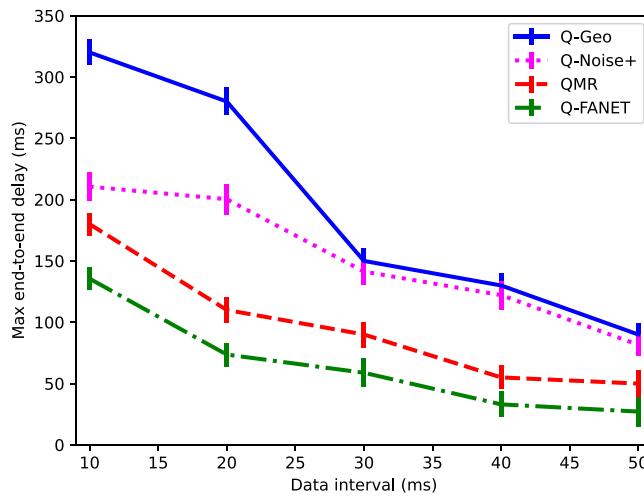


Fig. 3. Max end-to-end delay for the first scenario with all nodes working properly.

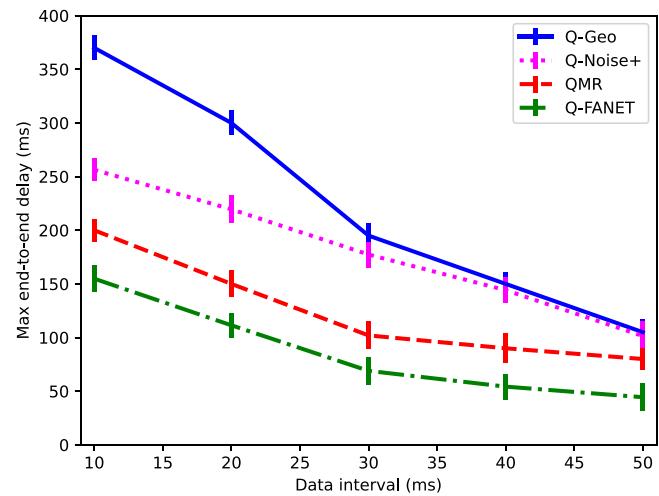


Fig. 6. Max end-to-end delay for the scenario with faulty relay nodes.

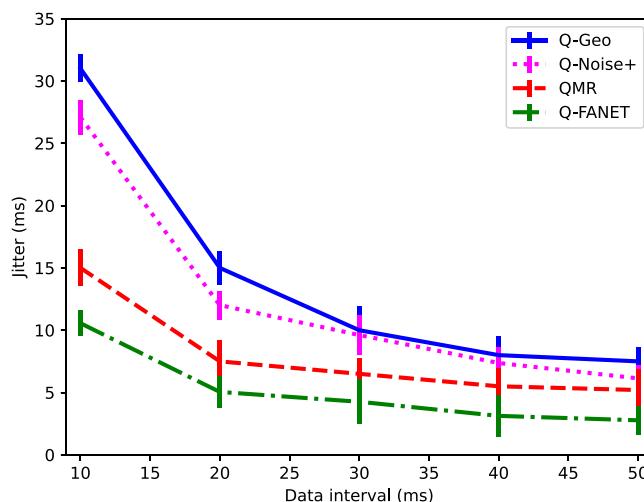


Fig. 4. Jitter for the first scenario with all nodes working properly.

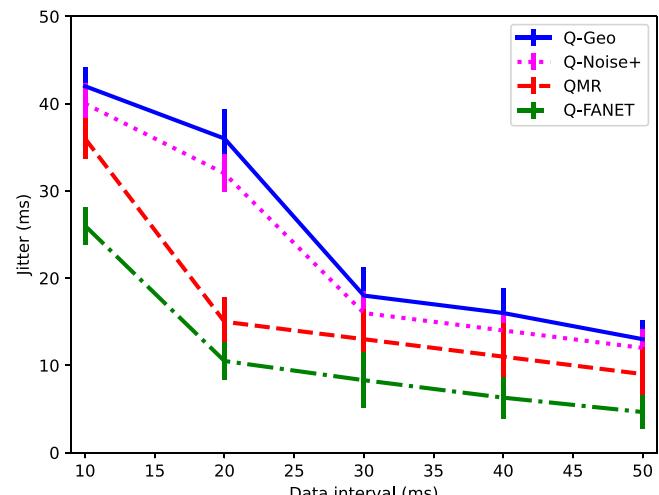


Fig. 7. Jitter for a scenario with faulty relay nodes.

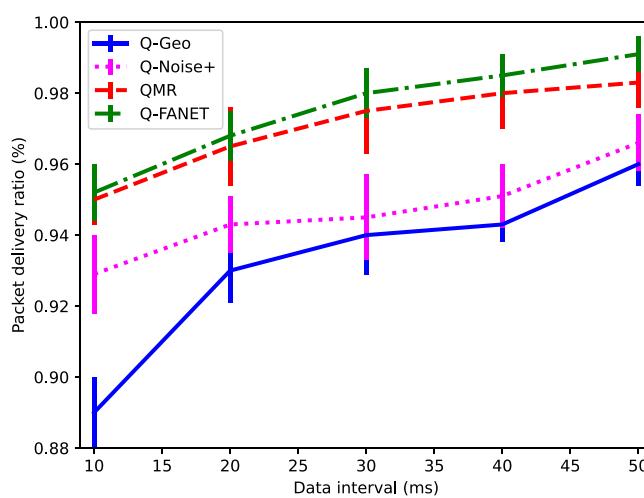


Fig. 5. Packet delivery ratio for the first scenario with all nodes working properly.

4.2. Scenario with faulty nodes

In this second set of simulations, 10 out of the 25 nodes stop working by powering them off 1 s after starting the simulation. As in the first set of simulations, Q-FANET was compared with QGeo, Q-Noise+, and QMR under different data intervals.

From Figs. 6 to 8, it is still possible to observe that Q-FANET presents a better performance in all the evaluation metrics. As observed in Figs. 6 and 7, Q-Geo, and Q-Noise+ are greatly affected by the presence of the faulty nodes while both QMR and Q-FANET present a better adaptive behavior, and can overcome the problem by selecting better routes to transmit. The packet delivery is also more affected in Q-Geo and Q-Noise+ compared to Q-FANET and QMR. The difference between the latter ones is smaller in this situation with faulty nodes, but still significant, particularly considering applications such as video streaming, which are very sensitive to the Quality of Service (QoS) degradation, as discussed by [50].

4.3. Discussing the improvements of Q-FANET

Figs. 9 and 10 show that Q-FANET can enhance routing performance and presents a significant improvement over QMR, which is the best among the other three protocols tested in the performed experiments.

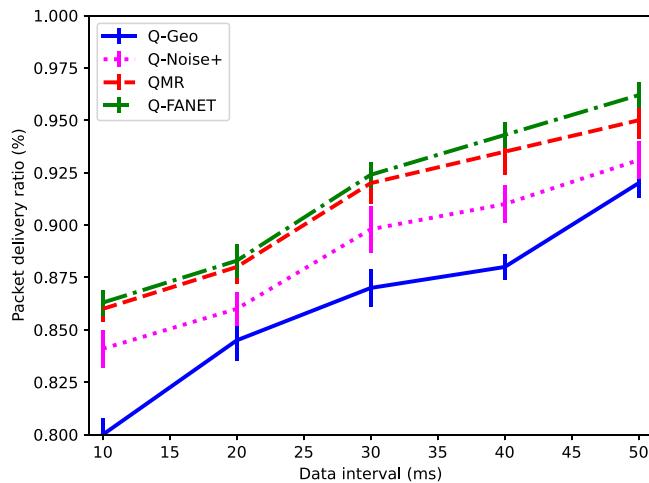


Fig. 8. Packet delivery ratio for the scenario with faulty relay nodes.

Q-FANET presents an increasing improvement in terms of maximum delay and jitter over QMR as the time intervals between the data transmission increase, achieving between 45.71% and 46.75%, respectively, of better performance than QMR for the 50 ms data sending time interval. Even in the scenario with the faulty nodes, Q-FANET shows improvements of 44.49% and 48.28% for the maximum delay and jitter. It is crucial to observe that the results under tight intervals are not as good as those obtained for larger intervals, which happens because the Q-FANET algorithm improves its knowledge of the links' status and channel conditions as the experiment runs. Variable intervals result in network topology changes that lead to adjustments in the algorithm's behavior, consequently improving the results. However, even in these challenging conditions, the results in terms of delay and jitter are significant and represent an important contribution to the type of video-stream based applications UAV-networks usually are employed to, which are very sensitive to QoS degradation.

It is worth mentioning that Q-FANET has a minor improvement over QMR in terms of packet delivery ratio, showing a better performance of 0.81% and 1.26%, for the scenario with all the nodes working and the scenario with the faulty nodes, respectively. Under tight intervals, the improvement is still smaller and they can be explained by the same reasons mentioned above, i.e. the learning process of the algorithm. However, considering the possible video streaming applications, this small improvement can represent a significant result for the final user, considering Quality of Experience (QoE) evaluations, as discussed in [51]. Analyzing these results in light of the discussions provided by [51] and [50], it is possible to assess that the improvements provided by Q-FANET can benefit end-users of video streaming applications through lowering the number and length of stalls in the videos, particularly considering that these QoE metrics are directly affected by the QoS metrics delay and jitter.

The realism of the adopted nodes' mobility model could be argued as a threat to the validity of the obtained results. However, the adopted Random Waypoint Mobility Model is realistic enough for the scenarios considered in this present study, as analyzed in [49]. Thus, no bias in the result is expected due to the used mobility model. Anyway, other mobility models can be considered in the continuation of this study, such as the Gaussian-Markov explored in [52].

Still about the mobility, as stated in 3.1, the considered UAVs are those flying with low to medium speed. However, studying adaptations in the proposed solution to consider faster UAV platforms, flying at high speeds, is also possible with the developed simulation framework setup. Despite of the importance of this subject, it is out of the scope of this current study.

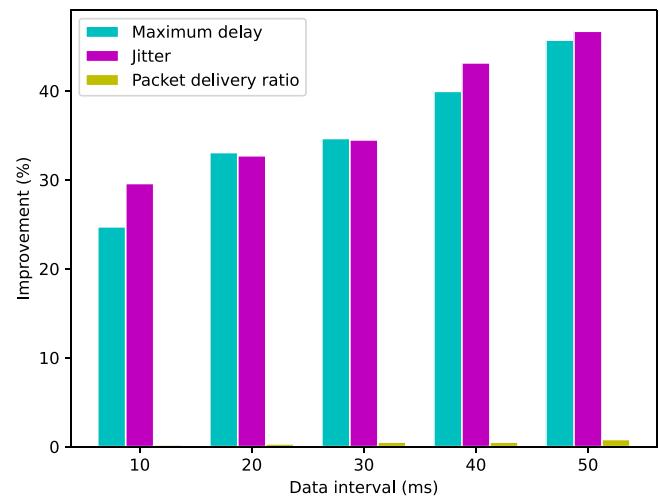


Fig. 9. Improvement percentage of Q-FANET over QMR.

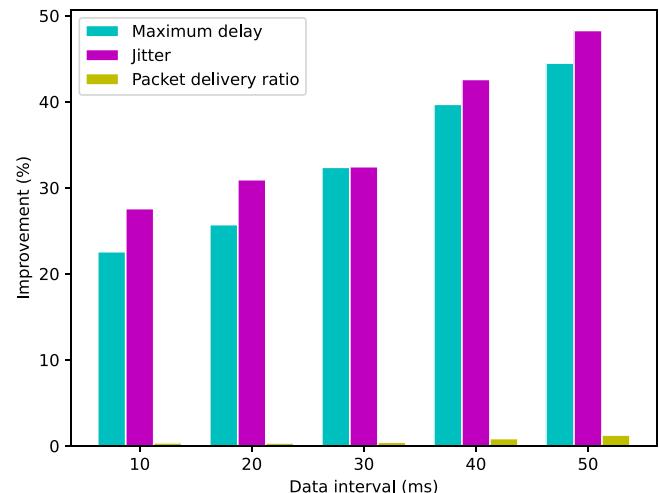


Fig. 10. Improvement percentage of Q-FANET over QMR in the faulty nodes simulation scenario.

Finally, Fig. 11 shows that Q-FANET respects the timer limit for convergence in both simulation scenarios. Following Q-FANET knowledge and performance improvement over time, the convergence time for the algorithm slowly decreases as it learns the most optimal paths to transmit data, even with higher transmission intervals and the adjustments of the network topology.

5. Conclusion and future work

This paper proposed Q-FANET, an improved Q-learning based routing protocol for FANETs. The proposed approach has brought together the leading techniques and elements used in two different routing protocols that make use of Reinforcement Learning: QMR and Q-Noise+ in a new protocol. By combining and adapting elements of these base protocols into the new conceived protocol architecture, the goal was to propose a protocol that better suites the dynamic behavior of FANETs, improving the network reliability and performance.

The proposal was evaluated, having its performance compared with QMR, Q-Noise+ and Q-GEO protocols in two scenarios. In the first one, all the nodes were up and running, while the second one considered the presence of faulty nodes. Q-FANET obtained significantly lower maximum end-to-end delay and jitter than the competitors in both scenarios. There was also a minor increase in the packet delivery ratio.

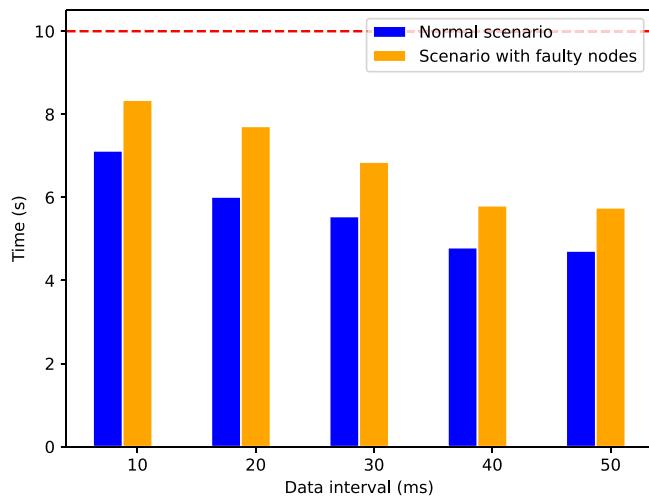


Fig. 11. Time for the convergence of Q-FANET in both simulation scenarios.

Directions for future investigations include dealing with other issues, like energy consumption, an essential concern regarding small UAVs with constrained energy resources. Online inner parameter adaptation is a possible direction to further enhance the proposed solution. Particular movement patterns are also of interest for future exploratory experiments as well as further adaptations of the proposed solution to support networks composed of high-speed UAVs.

Declaration of competing interest

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

Acknowledgment

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 and in part by Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brasil (CNPq) Projects 309505/2020-8 and 420109/2018-8, and Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul (FAPERGS).

References

- [1] A. Merwaday, I. Güvenç, Uav assisted heterogeneous networks for public safety communications, in: 2015 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), 2015, pp. 329–334.
- [2] R.S. de Moraes, E.P. de Freitas, Distributed control for groups of unmanned aerial vehicles performing surveillance missions and providing relay communication network services, *J. Intell. Robot. Syst.* 92 (3) (2018) 645–656.
- [3] R.S. de Moraes, E.P. de Freitas, Multi-uav based crowd monitoring system, *IEEE Trans. Aerosp. Electron. Syst.* 56 (2) (2020) 1332–1345.
- [4] L.F.F.G. de Assis, L.P. Behnck, D. Doering, E.P. de Freitas, C.E. Pereira, F.E.A. Horita, J. Ueyama, J. Porto de Albuquerque, Extending sensor web for near real-time mobile sensor integration in dynamic scenarios, in: 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA), 2016, pp. 303–310.
- [5] D. Orfanus, E.P. de Freitas, F. Eliassen, Self-organization as a supporting paradigm for military uav relay networks, *IEEE Commun. Lett.* 20 (4) (2016) 804–807.
- [6] C. Caillouet, F. Giroire, T. Razafindralambo, Efficient data collection and tracking with flying drones, *Ad Hoc Netw.* 89 (2019) 35–46, [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1570870518305985>.
- [7] O.K. Sahingoz, Mobile networking with uavs: Opportunities and challenges, in: 2013 International Conference on Unmanned Aircraft Systems (ICUAS), 2013, pp. 933–941.
- [8] E. Pignaton de Freitas, L.A.L.F. da Costa, C. Felipe Emgydio de Melo, M. Basso, M. Rodrigues Vizzotto, M. Schein Cavalheiro Corrêa, T. Dapper e Silva, Design, implementation and validation of a multipurpose localization service for cooperative multi-uav systems, in: 2020 International Conference on Unmanned Aircraft Systems (ICUAS), 2020, pp. 295–302.
- [9] İlker Bekmezci, O.K. Sahingoz, Şümail Temel, Flying ad-hoc networks (fanets): A survey, *Ad Hoc Netw.* 11 (3) (2013) 1254–1270, [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1570870512002193>.
- [10] M.B. Yassein, N.A. Damer, Flying ad-hoc networks: Routing protocols, mobility models, issues, *Int. J. Adv. Comput. Sci. Appl.* 7 (6) (2016) [Online]. Available: <http://dx.doi.org/10.14569/IJACSA.2016.070621>.
- [11] C.J. Watkins, P. Dayan, Q-learning, *Mach. Learn.* 8 (3–4) (1992) 279–292.
- [12] J. Jiang, G. Han, Routing protocols for unmanned aerial vehicles, *IEEE Commun. Mag.* 56 (2018) 58–63.
- [13] Y. Fu, Ding Prof. M., C. Zhou Prof., H. Hu, Route planning for unmanned aerial vehicle (UAV) on the sea using hybrid differential evolution and quantum-behaved particle swarm optimization, *IEEE Trans. Syst. Man Cybern.: Syst.* 43 (6) (2013) 1451–1465.
- [14] M.S.-H. Jean-Daniel Medjo Me Biomo, Thomas. Kunz, Routing in Unmanned Aerial Ad hoc Networks: A Recovery Strategy for Greedy Geographic Forwarding Failure, 3, (2014) 2236–2241. [Online]. Available: <https://goo.gl/aBgvQ>.
- [15] A. Boukerche, B. Turgut, N. Aydin, M.Z. Ahmad, L. Böloni, D. Turgut, Routing protocols in ad hoc networks: A survey, *Comput. Netw.* 55 (2011) 3032–3080.
- [16] C.M. Cheng, P.H. Hsiao, H.T. Kung, D. Vlah, Maximizing throughput of UAV-relaying networks with the load-carry-and-deliver paradigm, in: IEEE Wireless Communications and Networking Conference, WCNC, 2007, pp. 4420–4427.
- [17] J. Ko, A. Mahajan, R. Sengupta, A network-centric UAV organization for search and pursuit operations, *IEEE Aerosp. Conf. Proc.* 6 (2002) 2697–2713.
- [18] P. long Yang, C. Tian, Y.H. Yu, Analysis on optimizing model for proactive ad hoc routing protocol, in: MILCOM 2005-2005 IEEE Military Communications Conference, Vol. 5, 2005, pp. 2960–2966.
- [19] M. Bani, N. Alhudha", Flying ad-hoc networks: Routing protocols, mobility models, issues, *Int. J. Adv. Comput. Sci. Appl.* 7 (6) (2016) 162–168, [Online]. Available: <https://goo.gl/Tu95TD>.
- [20] D.S. Vasiliev, A. Abilov, V.V. Khvorenkov, Peer selection algorithm in flying ad hoc networks, in: 2016 International Siberian Conference on Control and Communications (SIBCON), 2016, pp. 1–4.
- [21] S. Habib, S. Saleem, K.M. Saqib, Review on manet routing protocols and challenges, in: 2013 IEEE Student Conference on Research and Development, 2013, pp. 529–533.
- [22] D.B. Johnson, D.A. Maltz, Dynamic source routing in ad hoc wireless networks, 1996.
- [23] V.A. Maistrenko, L.V. Alexey, V.A. Danil, Experimental estimate of using the ant colony optimization algorithm to solve the routing problem in fanet, in: 2016 International Siberian Conference on Control and Communications (SIBCON), 2016, pp. 1–10.
- [24] J. Jailton, T. Carvalho, J. Araújo, C. Frances, Relay positioning strategy for traffic data collection of multiple unmanned aerial vehicles using hybrid optimization systems: A fanet-based case study, *Wirel. Commun. Mob. Comput.* 2017 (2017) 1–11.
- [25] V. Park, S. Corson, Temporally-ordered routing algorithm (tora) version 1, [Online]. Available: <https://tools.ietf.org/html/draft-ietf-manet-tora-spec-00>, 0000.
- [26] D.K.L. Ram Shringar Raw, S. Das, An analytical approach to position-based routing protocol for vehicular ad hoc network, 2012, p. 9.
- [27] B. Karp, H.T. Kung, Gpsr: greedy perimeter stateless routing for wireless networks, in: MobiCom, 2000.
- [28] C. Zang, S. Zang, Mobility prediction clustering algorithm for uav networking, in: 2011 IEEE GLOBECOM Workshops (GC Wkshps), 2011, pp. 1158–1161.
- [29] L. Kesheng, Z. Jun, Z. Tao, The clustering algorithm of uav networking in near-space, in: 2008 8th International Symposium on Antennas, Propagation and EM Theory, 2008, pp. 1550–1553.
- [30] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, MIT press, 2018.
- [31] P.S. Bithas, E.T. Michailidis, N. Nomikos, D. Vouyioukas, A.G. Kanatas, A survey on machine-learning techniques for uav-based communications, *Sensors* (Basel, Switzerland) 19 (2019).
- [32] R. Li, F. Li, X. Li, Y. Wang, Ogrid: Q-learning based routing protocol for vehicular ad hoc networks, in: 2014 IEEE 33rd International Performance Computing and Communications Conference (IPCCC), 2014, pp. 1–8.
- [33] A. Serhani, N. Naja, A. Jamali, Qlar: A q-learning based adaptive routing for manets, in: 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), 2016, pp. 1–7.
- [34] W.-S. Jung, J. Yim, Y.-B. Ko, Qgeo: Q-learning-based geographic ad hoc routing protocol for unmanned robotic networks, *IEEE Commun. Lett.* 21 (2017) 2258–2261.

- [35] Q. Yang, S. Jang, S. Yoo, Q-learning-based fuzzy logic for multi-objective routing algorithm in flying ad hoc networks, *Wirel. Pers. Commun.* (2020) 1–24.
- [36] J. Liu, Q. Wang, C. He, K. Jaffrèse-Russer, Y. Xu, Z. Li, Y. Xu, Qmr: Q-learning based multi-objective optimization routing protocol for flying ad hoc networks, *Comput. Commun.* 150 (2020) 304–316.
- [37] C. He, Q. Wang, Y. Xu, J. Liu, Y. Xu, A q-learning based cross-layer transmission protocol for manets, in: 2019 IEEE International Conference on Ubiquitous Computing & Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS), 2019, pp. 580–585.
- [38] N. Lyu, G. Song, B. Yang, Y. Cheng, Qngpsr: A q-network enhanced geographic ad-hoc routing protocol based on gpsr, in: 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall), 2018, pp. 1–6.
- [39] L.R. Faganello, R. Kunst, C.B. Both, L.Z. Granville, J. Rochol, Improving reinforcement learning algorithms for dynamic spectrum allocation in cognitive sensor networks, in: 2013 IEEE Wireless Communications and Networking Conference (WCNC), 2013, pp. 35–40.
- [40] K. Dalamagkidis, Classification of UAVs, Springer Netherlands, Dordrecht, 2015, pp. 83–91, [Online]. Available: https://doi.org/10.1007/978-90-481-9707-1_94.
- [41] D. of Defense, Lighter-than-air vehicles, 2012, [Online]. Available: <https://apps.dtic.mil/cgi-bin/GetTRDoc?Location=GetTRDoc&docType=GetTRDoc&docID=GetTRDoc&GetTRDocID=a568211.pdf>.
- [42] 3DRobotics, iris plus quadcopter, 2014, [Online]. Available: <http://3dr.com/support/articles/207358106/iris/>.
- [43] R.E. Mohamed, A.I. Saleh, M. Abdelrazzak, A.S. Samra, Energy-efficient routing protocols for solving energy hole problem in wireless sensor networks, *Comput. Netw.* 114 (2017) 51–66.
- [44] T.S. Rappaport, *Wireless communications - principles and practice*, 1996.
- [45] M. Wunder, M.L. Littman, M. Babes, Classes of multiagent q-learning dynamics with epsilon-greedy exploration, in: Proceedings of the 27th International Conference on Machine Learning (ICML-10), Citeseer, 2010, pp. 1167–1174.
- [46] L. Matignon, G. Laurent, N.L. Fort-Piat, Reward function and initial values: Better choices for accelerated goal-directed reinforcement learning, in: ICANN, 2006.
- [47] Wsnet simulator, 2019, <http://wsnet.gforge.inria.fr/>, (accessed: 2019-09-30).
- [48] T. Camp, J. Boleng, V. Davies, A survey of mobility models for ad hoc network research, *Wirel. Commun. Mob. Comput.* 2 (2002) 483–502.
- [49] D. Orfanus, E.P. de Freitas, Comparison of uav-based reconnaissance systems performance using realistic mobility models, in: 2014 6th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2014, pp. 248–253.
- [50] I. Zacarias, J. Schwarzkopf, L. Gaspary, A. Kohl, R. Fernandes, J. Stocchero, E.P. de Freitas, Enhancing mobile military surveillance based on video streaming by employing software defined networks, *Wirel. Commun. Mob. Comput.* 2018 (2018) 1–12.
- [51] Z. Zhao, P. Cumino, A. Souza, D. Rosário, T. Braun, E. Cerqueira, M. Gerla, Software-defined unmanned aerial vehicles networking for video dissemination services, *Ad Hoc Netw.* 83 (2019) 68–77, [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1570870518306231>.
- [52] X. Li, T. Zhang, Stgm: A spatiotemporally correlated group mobility model for flying ad hoc networks, in: ChinaCom, 2016.



Luis Antonio L.F. da Costa is currently an Assistant Researcher at the Applied Computing Group of the University of Vale do Rio dos Sinos (Unisinos). He also holds a M.Sc. degree in Computer Science from the Institute of Informatics of the Federal University of Rio Grande do Sul (UFRGS). His research interests include computer networks, optimization, artificial intelligence and wireless networks.



Rafael Kunst is a professor and Researcher at the Applied Computing Graduate Program of the University of Vale do Rio dos Sinos (Unisinos), Brazil, where he is also a member of the Software Innovation Laboratory — SOFTWARELAB. He is also an ad-hoc consultant for the Brazilian Ministry of Education. He holds a Ph. D. and an M.Sc. degree in Computer Science; both received from the Federal University of Rio Grande do Sul (UFRGS). His current research interests involve next-generation mobile communications, such as 5G and 6G, military communications, Industry 4.0, smart cities, Internet of Things, and the application of Big Data Analytics and Machine Learning to optimize telecommunications. He has extensive experience as a consultant, coordinating and participating in projects with companies from Brazil and abroad.



Edison Pignaton de Freitas Computer Engineer by the Military Institute of Engineering in Brazil (2003), he received the M.Sc. in Computer Science by Federal University of Rio Grande do Sul (UFRGS) in Brazil (2007), and the Ph.D. degree in Computer Science and Engineering from Halmstad University in Sweden in 2011. He is currently associate professor at UFRGS, developing research in the Graduate Program on Computer Science and in the Graduate Program in Electrical Engineering. His research interests are mainly in the following areas: Computer Networks, Internet of Things, Real-Time Systems, Multiagents Systems and Unmanned Aerial Vehicles.



Local Mutual Exclusion algorithm using fuzzy logic for Flying Ad hoc Networks



Ashish Khanna ^a, Joel J.P.C. Rodrigues ^{b,c,*}, Naman Gupta ^a, Abhishek Swaroop ^d, Deepak Gupta ^a

^a Maharaja Agrasen Institute of Technology, GGSIP University, India

^b Federal University of Piauí (UFPI), Teresina - PI, Brazil

^c Instituto de Telecomunicações, Portugal

^d BPIT, GGSIPU, Delhi, India

ARTICLE INFO

Keywords:

Resource allocation problem

Local Mutual Exclusion

Leader election

Dynamic topology

ABSTRACT

The Local Mutual Exclusion (LME) problem is a variant of classical Mutual Exclusion (ME) problem and can be considered as an extension of dining philosopher problem. In LME, no two neighboring nodes can enter the critical section (CS) simultaneously, whereas two non-neighboring nodes can be in their CS simultaneously. The resource allocation problem in Flying Ad hoc Networks (FANETs), is relatively an unexplored area despite having several potential applications. The present paper proposes LME problem for FANETs and provides a leader-based algorithm named as Request Collector Local Mutual Exclusion (RCLME) for the same. To the best of our information, LME problem is introduced first time in Flying Ad hoc Networks. The striking feature of the proposed algorithm is the introduction of a fuzzy logic-based leader election that considers the node speed, node direction, link quality, and the distance from the resource. The correctness proof of the RCLME algorithm has been presented. The simulation results show that RCLME algorithm significantly outperforms other related algorithms available in the literature; specially, when the number of nodes is large. The use of fuzzy logic and request collector improves the efficiency, fault tolerating capacity and ability to handle volatility.

1. Introduction

The classical mutual exclusion (ME) problem [1] is one of the most important problems of distributed systems and its variants where a number of nodes compete to have mutually exclusive access of a shared resource, and at an instance, a single node is permitted to enter critical section (CS). The shared resource can be a file, a database chunk, or a printer. In the last decade, several academicians have proposed numerous algorithms for the classical ME problem for various types of distributed systems such as Mobile Ad hoc Networks (MANETs) [2–4] and Vehicular Ad hoc Networks (VANETs) [5,6].

Unlike the classical ME problem, in the dining philosopher problem [7], which is also one of the variants of ME problem, two non-neighboring philosophers can be in a CS state simultaneously whereas the neighboring philosophers cannot enter the CS state simultaneously. The solution to dining philosopher's problem exhibits a special property that the entire system is not affected by the failure of a node. This property is analogous to the characteristics of MANETs in which most of the activities are local; hence, the impact of any failure must be local and should not affect the entire system. Therefore, the dining philosopher's

problem was solved in MANETs under the title local mutual exclusion (LME) problem by Attiya–Kogan–Welch [8]. In [8], the LME problem was defined using the communication range of a node for the neighborhood. Later on, Khanna–Singh–Swaroop presented the concept of resource-centric neighborhood and proposed various algorithms [9–11] for MANETs with resource-centric neighborhood.

In recent past, there has been a tremendous amount of development in the technologies involving networks and hardware, resulting in the emergence of Flying Ad hoc Network (FANET), a special subclass of MANET in which Unmanned Aerial Vehicles (UAVs) acts as communicating nodes. UAV is a drone or any other aerial vehicle that can fly and navigate autonomously or can be managed from a distance with the help of a remote control. The nodes in FANETs have a higher mobility rate and higher communication range in comparison to MANETs and VANETs. Recently, several applications of FANETs for military and civilian purposes have emerged due to their adaptability, easy configurability, and flexibility. However, efficient resource sharing without conflict is also a major research challenge in FANETs. The ME problem is relatively an unexplored area in FANETs. Recently in [12],

* Correspondence to: Federal University of Piauí (UFPI), Campus Petrônio Portela; Av. Ministro Petrônio Portela, S/N - Bloco 8, Centro de Tecnologia; 64049-550, Ininga, Teresina - PI, Brazil.

E-mail addresses: ashishkhanna@mait.ac.in (A. Khanna), joeljr@ieee.org (J.J.P.C. Rodrigues), namangupta0227@gmail.com (N. Gupta), asa1971@gmail.com (A. Swaroop), deepakgupta@mait.ac.in (D. Gupta).

a ME algorithm for FANETs under the title of Mobile Resource Mutual Exclusion (MRME) procedure has been proposed. However, due to the availability of several localized operations in FANETs, the viability of the LME problem is more significant than a conventional ME problem.

In FANETs, the resource enabled UAV can have swift mobility. Furthermore, all mobile resources have a defined communication range. Therefore, the concept of resource-centric neighborhood [9] is suitable for FANETs. Due to their specialized characteristics, the solutions proposed for MANETs cannot be applied for FANETs without compromising on the efficiency. Hence, in the present exposition, LME problem has been proposed for FANETs. To solve the LME problem, a leader-based algorithm has been presented and named as Request collector Local Mutual Exclusion (RCLME) algorithm.

The role of a leader in the presented solution is played by a request collector node instead of a token node because of which the message overhead of the algorithm significantly reduces. If the token is given the responsibility of collecting CS requests of the nodes, the information about the new token holder has to be broadcasted after every CS execution. However, the request collector usually needs to be changed after several executions. This reduces the number of broadcast messages and over all message complexity of the algorithm.

The striking feature of the RCLME algorithm is the introduction of a fuzzy logic-based scheme to select a leader node. The selection function is based upon the speed of the node, direction, distance from the resource, and link quality. The concept of fuzzy logic is used to elect a node which will probably remain in the neighborhood for the maximum amount of time. Hence, the frequency of change of request collector is significantly reduced. Consequently, the number of messages exchanged per critical section executions and waiting time for requests is reduced drastically, making the system energy efficient.

Conventionally, the solutions to the ME problem have been categorized into two classes: permission-based and token-based. The permission-based algorithms require a great number of message exchanges, which make them unsuitable for wireless networks. In order to reduce the communication overhead in permission-based algorithms, Maekawa [13] introduced the concept of quorum. However, the quorum-based algorithms are prone to deadlock. Moreover, the dynamic topology in FANET may compel frequent quorum reconstruction, which is a non-trivial task. Constructing fixed size quorums is equivalent to finding a finite projective plane, which is complex, computation intensive and memory consuming for dynamic networks [14]. Additionally, the permission-based algorithms are poorly fault-tolerant and thus the possibility of deadlock cannot be denied completely. Though, several permission-based distributed ME algorithms exist, very few of them can adapt well to dynamic topologies. The permission-based algorithms require strong connectivity among nodes, which is difficult to guarantee in ad hoc networks.

On the other hand, the token-based algorithms are deadlock free and it is easy to ensure safety in token-based algorithms. Additionally, the token-based algorithms outperform permission-based algorithm as far as the communication overhead is concerned [14–16] which is an essential requirement for resource starved nodes of FANETs. Thus, we preferred token-based approach for solving LME problem in resource neighborhood based FANETs. However, the token-based approaches face the token-loss problem [16] as token may be lost in transit or the node-possessing token may crash or move out of the system. In the event of token-loss, the token needs to be regenerated which is handled in the proposed solution.

To the best of our knowledge, the proposed algorithm is the pioneer solution to solve LME problem in resource-centric neighborhood based FANETs.

1.1. Applications

There are several potential applications of resource-centric neighborhood based FANETs. Some of these applications are discussed below.

One possible application can be the health monitoring of endangered species in a forest. The monitoring of endangered species may be performed using biosensors and a UAV moving across the forest. The wearable biosensors implanted on animal, monitors their health and calculates their vitals like heart rate, body temperature and blood pressure, etc. These biosensors transmit their data when they come into the communicating range of a UAV. The entire forest may be divided into regions with one UAV in each region. An LME algorithm will be required when data from more than one animal needs to be transferred safely.

Another major application is in the field of disaster rescue operations where UAV is a resource and ground nodes (survivors) are static/dynamic nodes. Each survivor needs to be rescued or requires medical aid. Therefore, each ground node is competing for the resource. It is possible that there are multiple survivors distributed over the entire region, and there are multiple UAV (resource) employed for a rescue operation. An LME algorithm can be perfectly applied in such a scenario to fulfill the needs of ground nodes (survivors) where a copy of the proposed algorithm is running in multiple available neighborhoods.

LME in FANETs can be useful in a scenario where multiple UAVs are deployed on a mega airport/port or railway station where the nodes may be officers, passengers, some sensors installed at the premise, or airplane/train/ships. Each UAV can be the center of a neighborhood and passengers/officers/planes/ships/trains in the range of it are mobile nodes in the neighborhood. The application may require data transfer from UAV to nodes and vice versa. Only the nodes within the range of UAVs will be able to compete for exclusive access to UAVs. Therefore, the LME problem considering resource centric neighborhood is required in this situation.

In all of the above-mentioned applications, the resources available in each resource centric neighborhood must be accessed by the nodes present in that neighborhood in a mutually exclusive way. Hence, the LME problem in resource centric neighborhood for FANETs is the central point of this research.

The main contributions of this paper are the following:

- The present exposition introduces LME problem in FANETs.
- A leader-based algorithm is used to solve the LME problem for FANETs is proposed.
- The concept of Request collector has been used to reduce the message overhead.
- A fuzzy logic-based scheme to select the leader is applied which further improve the overall efficiency and waiting time.
- The correctness properties of the proposed algorithm are defined and proved.
- The static performance of the algorithm is analyzed.
- The dynamic performance analysis of the algorithm along with comparison with the closest works has been shown.

The remaining paper is structured in the following way. Section 2 elaborates on the related work whereas, in Section 3, the proposed system model along with the data structure and working of RCLME is presented. The correctness proof is addressed in Section 4, and the analysis of static performance of the RCLME algorithm has been shown in Section 5. Section 6 includes dynamic analysis of the proposed approach and performance comparison with closest work. In the end, Section 7 concludes the work with the future scope.

2. Related work

The ME problem [7] along with variants such ask-mutual exclusion (KME) [17] and Group Mutual Exclusion (GME) [18] has been actively persuaded as an area of research since its inception. The researchers have proposed several algorithms for ME and its variants in distributed computing system and its variants like MANET [3], VANET [6]. The wireless networks are becoming popular day by day,

and the resource sharing problem has tremendous significance due to lot of possible applications in wireless networks. The algorithms developed for static networks cannot be efficiently applied for wireless networks. The first ME methodology in wireless networks was proposed by Walter-Kini [2], who leveraged the logical tree structure to develop a token-based algorithm an adaptive to node mobility. Later on, several algorithms have been proposed by the researchers for MANETs [8,19]. Sharma-Bhatia-Singh [20] presented a comprehensive survey of all the ME algorithms for MANETs from 2004 to 2010.

The dining philosopher's problem is a prototypical resource allocation problem in distributed systems and its variants [2,10,11,20–24]. In the dining philosopher's problem, non-neighboring philosophers (nodes) may enter in eating state (Critical Section) simultaneously, whereas neighboring philosophers (nodes) are not allowed to be in eating state (Critical Section) simultaneously. Moreover, the failure of a node affects only the neighboring nodes, not the entire system. The major characteristics of the dining philosopher's problem are concurrency, increased resource utilization, and limited failure locality. These characteristics make the dining philosopher's problem applicable in MANETs. Therefore, Attaiya-Kogan-Welch [8] leveraged the aforementioned property of the dining philosopher's problem and proposed LME problem, an attention-grabbing variant of ME problem. They presented the correctness requirements for LME problem and presented two door-way based solutions to solve LME problem in MANETs [8].

Attaiya-Kogan-Welch [8] applied the concept of neighborhood using the location of nodes in which the nearby nodes are considered to be neighbors, and they make a neighborhood. In [9], Khanna-Singh-Swaroop presented the concept of resource centric neighborhoods in which the nodes in the communication range of the shared resource are considered in one neighborhood and may compete for the access of shared resource in an exclusive manner. The LME problem [9] and its variants such as KLME [10], GLME [11] have also been solved in recent years for MANETs considering resource centric neighborhood.

Flying Ad hoc network (FANET) is a subclass of MANETs in which the nodes and resources may be placed on a UAV. Due to the involvement of UAV, the dynamism in the topology of the network is high, and the area under consideration is large in comparison to other categories of MANETs. The efficient and accurate resource sharing is required in applications involving FANETs. Carfang-Frew-Kingston [25] used UAV to transfer data between two nodes. Bekmezci-Sahingoz-Temel [26] presented a comprehensive survey on FANETs and its applications in various areas of life. Additionally, they tabulated the similarities and dissimilarities between FANETs and its variants, along with some of the recently developed routing protocols for FANETs. Yan-Mostofi [27] used a robot similar to UAV and deployed it to gather data by following a fixed set of paths. Ho-Grotli-Johansen [28] presented a heuristic algorithm for the optimization of energy spent in data gathering using UAV. Jawhar-Mohamed-Jaroodi [29] briefly explained the usage of UAV and satellite nodes for data collection from deployed sensor nodes. Due to its linear topology, use of UAV is very efficient and logical in case of Linear Wireless Sensor Networks, especially when the network is spread in one dimension for hundreds of kilometers. Turzman in [30] used machine learning techniques to predict the UAV trajectory path in the absence of GPS signal for accurate positioning of UAVs in mission critical applications. Also, 5G enabled efficient UAV communication using soft computing techniques [31], and artificial intelligence [32] has become an active research area in recent years. Recently, Turzman et al. [33] used UAVs for static and mobile target monitoring.

Recently, in [12] the first token-based algorithm under the title of Mobile Resource Mutual Exclusion (MRME) to solve the classical ME problem in FANETs has been proposed. This is the closest work to the proposed problem. The above-mentioned discussion clearly signifies the importance of the availability of a large number of localized operations in FANETs. Furthermore, classical ME algorithms cannot be efficiently applied to utilize the full potential of the available localized operations and applications in FANETs. Therefore, in the presented exposition, the authors have introduced the problem of LME in FANETs. To the best of our knowledge, the LME problem for FANETs has been solved for the first time in the present paper.

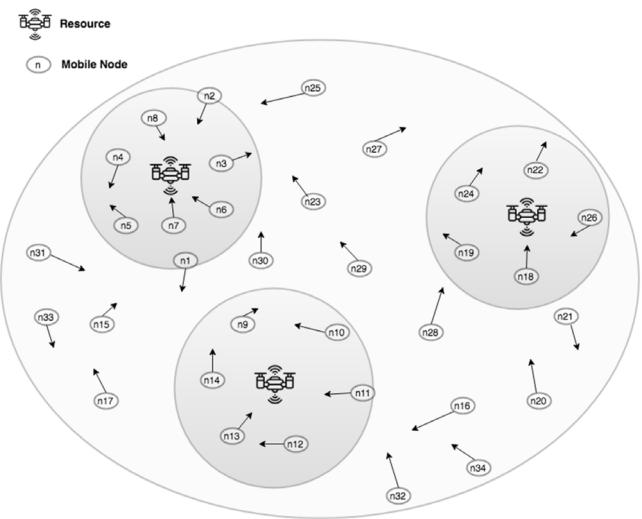


Fig. 1. Multiple Neighborhoods in FANETs.

3. System model

The authors assume asynchronous distributed system which comprises of n dynamic or (and) static nodes, having numbered from 1 to n . The network is separated into non-overlapping regions based on the location of a resource on a mobile UAV. Fig. 1 shows an illustration of such a system with three non-overlapping regions.

Fig. 1 represents a system in which the region of interest is divided into neighborhoods depending upon communication range and location of UAV. The gray portions illustrate the independent non-overlapping neighborhoods which are defined using the communication range of a UAV. The small oval shapes with arrows are the mobile nodes which are close to the resource on an Unmanned Aerial Vehicle. The nodes in the range of an Unmanned Aerial Vehicle can access it in a mutually exclusive way. The nodes may join or leave the neighborhood according to their location with respect to a particular UAV.

Assumptions

1. A wireless resource is placed at a UAV flying and moving at a constant altitude and velocity.
 2. UAV has a limited communication range which defines the neighborhood of mobile resource.
- Rest of the assumptions are similar to be used for MRME algorithm [12].

Rest of the assumptions are similar as used in [12].

3.1. RCLME algorithm

3.1.1. Data structure of RCLME

At every node i

$state_i$: It represents current state of process i . There are mainly four states, namely Requesting token (RT), Holding idle token (HIT), Remainder (RM), and Critical Section (CS).

$local_req_q_i$: A queue to store pending requests locally.

$possess_token_i$: It is a Boolean value to confirm the possession of token by a node.

$reqNo_i$: Array of request number of all nodes at node i .

$request_collector_i$: Node id of Request collector according to node i .

$req_collector_no_i$: Latest request collector number according to node i .

- i. It is incremented when a new request collector is declared.

$req_ack_status_i$: Boolean variable specifying whether request of node i has been received or not.

dup_token : Replica of token stored at token transferring node.

token_transfer_status: It represents the status of token transfer. Successful = 1 and Failure = 0.

Data Structure at Token

last_reqNo: Array containing the information of last requests served for each node.

token_q: It is a queue that contains the CS requests to be served of nodes present in the neighborhood.

req_collector: Current request collector information according to current token holder node.

req_collector_no: Its value is incremented by 1 every time request collector is changed.

Messages

req_collector_info (*req_collector*, *req_collector_no*): It is the message sent by token-holder to update all nodes about new request collector.

request_CS_i(request_no): It is a request message sent by a node requesting CS to request collector.

req_ack(i): It is an acknowledgment sent by token holder or request-collector on successfully receiving node *i*'s request.

token(token_q, last_reqNo, req_collector): A node getting this message becomes token holder node.

i_am_leaving(n_i, local_req_q_i): A node leaving neighborhood broadcasts this message.

i_have_join(n_i, state_i): Broadcast message having information of a node joining neighborhood.

token_info(n_j): The information sent by the node holding the token to updates the new token information at node *i*.

token_ack_i(j): New token holder sends acknowledgment to node *j*, i.e., previous token holder when node *i* has received the token.

3.2. Request Collector Election Method (RCE)

In the proposed system, two roles of a node are defined, namely, a token holder and a request collector. A node possessing the token is called token holder node, and only a token holder node is allowed to enter CS. Request collector node is responsible for collecting CS requests from other nodes. The token holder node is responsible for electing a request collector node every time the request collector node is changed. The request collector can be either a node holding the token or a node whose request is added in the token queue. Request collector node is elected by the process called Request Collector Election (RCE). Election is done among the nodes that are already in the token queue. Each node is assigned a weighted value (fitness value), taking into consideration several important factors. Mobility of node i.e., speed of moving node, the distance of a node from the resource, direction of movement of node with respect to moving resource, and link quality are the major factors considered in this research work. These factors are given the appropriate weights according to their significance. A fitness value is calculated using fuzzy logic and is assigned to every node in the token queue. The fittest node is elected as the request collector node by the token holder. Here fitness of a node represents how long a node will stay in the resource-centric neighborhood as compared to other nodes. Initially, all four factors, i.e., distance, speed, the direction of movement, and link quality, are given equal weights. The fitness value for the leadership of each node is calculated by Eq. (1) given below:

$$\text{fitness}_i = 0.25 * \left[\frac{1}{v} \right] + 0.25 * [\theta] + 0.25 * \left[\frac{1}{x} \right] + 0.25 * q \quad (1)$$

where *v* is speed of node *i*, *x* is the distance between anode and the resource, *q* is the link quality, which is the absolute value of received signal strength from the resource, and *θ* is the direction of node defined as follows:

$$\theta = \begin{cases} -1, & \text{if node is moving away from there source} \\ 1, & \text{if node is moving towards the resource} \end{cases} \quad (2)$$

The fitness value of each node in the token queue is updated at regular intervals considering the dynamic nature of the FANET environment.

3.3. Working of RCLME algorithm

In the initialization phase, the node with lowest id in the resource-neighborhood say node *i*, is elected as token holding node, which also acts as a request collector. The node holding the token is the only privileged node to enter CS whereas the request collector is responsible for collecting requests. The request collector is changed in the following three scenarios. (a) When the request collector exits from CS, if the token queue has no request, the current request collector continues as request collector; otherwise, the node selected by RCE method becomes a new request collector. (b) When the token-holder node moves out of current resource neighborhood, if there are no requests in token queue, the lowest id node in the neighborhood is assigned token along with the responsibility of the request collector; however, if the token queue is not empty, the fittest node according to RCE method becomes new request collector. (c) When the request collector decides to leave the current resource neighborhood or forced to leave the neighborhood due to movement of UAV, the token holder becomes new request collector if the token queue is empty. Otherwise, the fittest node in the token queue (RCE) becomes the new request collector. The information about the election of a new request collector is broadcasted to all other nodes in the neighborhood.

A requesting node *i* forwards its request to the request collector and waits for the acknowledgment. When node *j* receives a request from node *i*, three cases are possible. (i) Node *j* is neither the token-holder nor the request collector: Node *j* appends the request in its local request queue. (ii) Node *j* is the request collector but not the token-holder: The request is added in the local request queue, and an acknowledgment is sent to the node *i*. (iii) Node *j* is request collector as well as token-holder: If node *j* is holding idle token, it transfers the token to node *i* and node *i* becomes new request collector else the request is appended to the token queue.

On receiving the token, node *i* updates the token queue from its local request queue. Now, there are two possibilities. (a) If node *i* is in the remainder state, it sends the token to the front node of the token queue and declares the node given by RCE as new request collector. However, in case the token queue is empty, node *i* holds the token. (b) If node *i* is in requesting state, it enters CS. On exiting from CS, node *i* updates the token queue from its local request queue. If the token queue is empty, node *i* holds the idle token. Otherwise, if node *i* is not the request collector, it forwards the token to the node at the head of the token queue else the node elected by RCE is declared as new request collector.

The issues like sensitivity to node mobility, link breaking, and link forming are important in mobile environment. These issues are addressed adequately in proposed algorithm. The events like node joining the neighborhood, node leaving the neighborhood, and the effect on the token ownership have been discussed below.

In case, any node in CS leaves the resource-neighborhood, it changes its state to requesting (RT) before moving to the next resource-neighborhood. In addition, the token is transferred to the node at the head of token-queue. However, if the token queue is empty or node *i* was holding idle token, the token is transferred to the lowest id node of the resource-neighborhood. The outgoing node broadcasts the information about the change in request collector by node *i*, if any, along with its local request queue. On receiving this broadcast, the token-holder node updates token queue. Further, if the outgoing node was the request collector, the token holder declares the node selected by RCE as a new request collector; however, if the token queue is empty, the token-holder node declares itself as the request collector.

When a node *i* joins new resource-neighborhood, it may be either in remainder state or in requesting state. Node *i* broadcasts its joining information along with its state information. When a request collector node receives this broadcast and finds that the incoming node is in requesting state, the request collector appends this request to its local request queue. Moreover, in this case, request acknowledgment is also

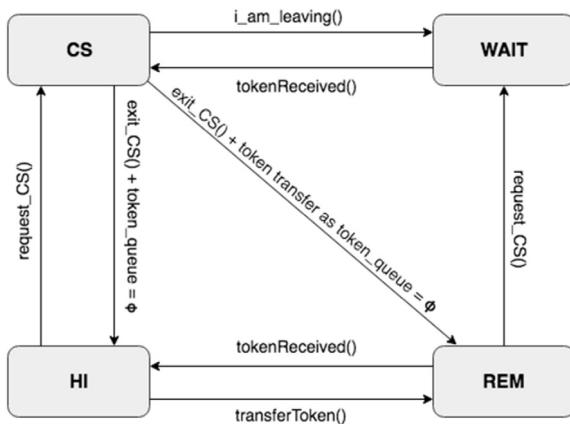


Fig. 2(a). State Transition Diagram.

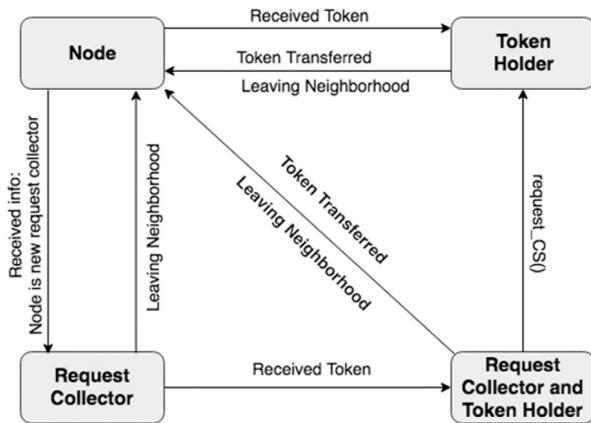


Fig. 2(b). Activity Diagram of node i.

sent to the incoming node. However, if the newly joined node is in the remainder state, the token holder informs it about the request collector node. The transitions between different states of the node upon various events have been presented in Fig. 2(a).

The node in RCLME may play various role. The responsibility of a node during execution changes with role. In Fig. 2(b), the transition of a node from one role to another has been shown for the convenience of the reader. The detailed explanation of the responsibilities of a node in various roles has been explained in upcoming paragraph.

In the following paragraph, we discuss how the proposed algorithm handles token loss, thereby making the algorithm fault tolerant. A copy of the token is stored by the node holding the token every time it transfers the token. A node exiting the neighborhood always broadcasts *i am leaving* message along with its residual information. Therefore, if the token is transferred to any node which decides to leave the neighborhood before the token reaches to it, the token can be lost. Since, as the outgoing node broadcasts its leaving information, after receiving the leaving information of the token receiving node, token forwarding node will announce itself as the new token holder. Now, if the token successfully reaches the destined node, the node forwarding the token will delete the saved old token copy to maintain the safety of the system.

3.4. Pseudo code of RCLME algorithm

Helper Functions:

Request Collector Election:

RCE(token_q)

fittest=-1

k=0

w_x : weight of x

for each node i in token_q

fitness_value_i = wspeed + wdistance + wdirection + wenergy

if(fittest < fitness_value_i)

fittest=fitness_value_i

k=i

return node with fittest value

Update Request Collector:

URC(i, new_req_collector)

req_collector_no = req_collector_no + 1

req_collector_no_i = req_collector_no

request_collector_i=new_req_collector

req_collector=new_req_collector

Initialization:

Initialization at node i

Let N be total number of nodes in the resource-neighborhood.

for every node i inside neighborhood

state_i = RM

local_req_q_i = \emptyset

possess_token_i = 0

node_id_i = i

request_collector_i = -1

reqNo_i[i] = 0

req_collector_no_i = 0

Initialization at token node

Let token holder be the lowest id node j in a neighborhood

req_collector_j = j

req_collector = j

req_collector_no = 0

req_collector_no = req_collector_no + 1

for node i=1 to N

last_reqNo[i] = 0

token_q = \emptyset

Broadcastreq_collector_info

A. Received req_collector_info (j,k) by node i

if(req_collector_no_i < k)

request_collector_i = j

if(state_i = RT&&req_ack_i = 0)

sendrequest_CS torequest_collector_i

B. Node i sends request for CS

state_i = RT

reqNo_i[i] = reqNo_i[i] + 1

if (possess_token_i)

enterCS

state_i=CS

call exit_CS(i)

elseif (request_collector_i ≠ -1)

send request_CS torequest_collector_i

req_ack_i= 0

else wait for Request collector information

C. When Node i receives request_CS(req_no) from node j

if (reqNo_i[j] < req_no)

if (possess_token_i)

if (state_i = HIT)

newReqCol = j

URC(i,newReqCol)

possess_token_i=0

initializedup_token = token

send token to j

Broadcast Request collector info

else

```

addrequest of node j in token_q
send req_ack to node j
elseif (request_collectori = i)
reqNoi[j] = req_no
add request to local_req_qi
sendreq_ack to node j
else
add request to local_req_qi
else reject the request message

```

D. Node i receives token from node j

```

possess_tokeni=1
sendtoken_received_ack to thenode j
Remove node i request from token_q
if (i = request_collectori)
update the token_q with the local_req_qi
Send req_ack to all nodes added in the token_q
if (statei = RT)
statei = CS
enter CS
call exit_CS(i)
elseif (state"-ei = RM&&token_q ≠ Φ)
newReqCol= RCE(token_q)
URC(i,newReqCol)
possess_tokeni=0
initializedup_token = token
Send token to front of token_q
Broadcast Request collector information
//it means that state is remainder but token_q is empty
else
stat"-ei = HIT
newReqCol = i
URC(i,newReqCol)
Broadcast Request collector information

```

E. Node i exits CS

```

last_reqNo[i] = reqNoi[i]
update the token_q with the local_req_qi
send ACK to the added nodes in the token_q
if(req_collector = i )
if (token_q ≠ Φ)
newReqCol = RCE(token_q)
URC(i,newReqCol)
possess_tokeni= 0
initializedup_token = token
send token to node at the front position of token_q
Broadcast Request collector information
else
statei = HIT
else // token has not reached to the last id
if (token_q ≠ Φ)
initializedup_token = token
possess_tokeni= 0
send token to node at the top of the token_q
else statei = HIT

```

F. Received request acknowledgment by the node i

```
req_ack_statusi = 1
```

G. Node i is about to leave resource neighborhood

```

X= -1
if (statei = CS)
if(token_q ≠ Φ)
req_collector_no = req_collector_no + 1
possess_tokeni= 0
initializedup_token = token
Send token to front of token_q
request_collectori=RCE(token_q)

```

```

X= request_collectori
statei= RT
else
possess_tokeni= 0
initializedup_token = token
send token to lowest id node
request_collectori = lowest node id in neighborhood
X= request_collectori
elseif (statei = HIT)
possess_tokeni= 0
initializedup_token = token
send token to lowest id node
request_collectori = lowest node id
X= request_collectori
else
statei= RM
Broadcast i_am_leaving along withhold token info

```

H. Node i receives the i_am_leaving (local_req_q_j, X) message from the node j

```

if( req_collector_noi < req_collector_no)
req_collector_noi = req_collector_no
if(statei = CS || statei = HIT)
Updatetoken_qof node with local_req_qi
Remove j from token_q and local_req_qi
if(request_collectori = j)
if(token_q ≠ Φ)
req_collector = request_collectori=RCE(token_q)
Broadcast Request collector info
else
req_collector = request_collectori= i
Broadcast Request collector info
elseif (statei = RT)
if (X ≠ -1)
request_collectori= X
if (req_ack_statusi = 0)
Sendrequest CSto X
elseif (statei = RM)
if(X ≠ -1)
request_collectori = X
updatelocal_req_qi with local_req_qj

```

I. Node i joins the system

```

Initialize all the data structure
if (statei = RT)
Broadcast i_have_join info with the requesting state
else
Broadcast i_have_join with remainder state

```

J. node i has received the i_have_join (state_j) message from the node j

```

if(request_collectori = i || possess_tokeni)
add request of node j in node i data structure
send the req_ack message to the node j
else
add request in local request queue of node i
else
if(possess_tokeni)
send request collector info to node j

```

K. Node i received token_received_ack message from node j

```

token_transfer_status = 1
deletedup_token
statei = RM

```

4. Proof of correctness

4.1. Mutual exclusion

The proof of mutual exclusion is trivial as the proposed algorithm is token-based.

4.2. Safety

Theorem 4.1. *Two nodes in the same neighborhood can never be in their CS concurrently at any instance of time.*

Proof. In the proposed system, in a neighborhood only single token is only generated and initialized at the beginning of RCLME algorithm. A node can only enter CS after receiving or possessing the token. The node holding token transfers the token to the next requesting node after exiting from its CS. Therefore, the safety of the system can be violated only if, there are two tokens simultaneously in the same neighborhood.

Let us assume that there are two tokens in the same neighborhood simultaneously. It is possible only if node forwarding the token and the node accepting the token, both declare themselves as the node holding the token. Whereas, the node forwarding the token can never announce itself as the token holder node, until the node gets *i_am_leaving()* from token receiving node. Additionally, the node forwarding the token also removes the copy of token at its end as soon as it receives an acknowledgment from token receiving node. Therefore, our assumption that two nodes in the same neighborhood, will be able to declare themselves as token holder simultaneously holds invalid.

4.3. Starvation freedom

In case, a node is waiting to enter CS, it should be able to move into the CS in finite amount of time.

Lemma 4.1. *Every request for the resource reaches to the corresponding request collector eventually.*

Proof. In case, we assume the opposite that if request of a node i never reaches to request collector. As per RCLME algorithm, a request is re-sent until it is acknowledged. Further, a non-request collector node stores the request locally, so that, it may send an acknowledgment, in case it becomes the request collector. Hence, a new node is required to be selected as the request collector infinitely many times to ensure that the assumption holds true. For this to be true, infinite nodes must be present in the system. However, this violates assumed system model. Hence, our assumption is invalid, and [Lemma 4.1](#) holds.

Lemma 4.2. *Every request is eventually added to token queue.*

Proof. In case, we assume the opposite that a particular request is never added to the token queue. [Lemma 4.1](#) proves that every request eventually reaches request collector's request queue. Further, the request may not be added to the token queue in following cases. Case (i) Request Collector remains in resource-neighborhood without receiving the token. However, the number of requests in token queue is finite, and a node remains in CS for a finite amount of time. Therefore, this case is not validated. Case (ii) Request collector leaves resource-neighborhood, before receiving the token. Since, request collector broadcasts its local node queue; the token-holder appends all requests in the token queue after having received the leaving information of request collector node. Therefore, case (ii) also leads to an infeasible state. Thus, [Lemma 4.2](#) holds.

Theorem 4.2. *Each request is eventually served.*

Proof. It is apparent from [Lemma 4.2](#) that every request for CS will be added to token queue eventually. Moreover, each node cannot have at more than one pending requests. Hence, the number of entries in the token queue cannot be more thann. additionally, the execution time of a node in CS is bounded, and the token queue is FCFS. Therefore, every request of token queue will eventually be served.

5. Performance analysis

In the proposed system for FANETs which is highly dynamic in nature, both resource and nodes are moving, and any node or a process can join and leave the neighborhood. We analyze the performance of RCLME algorithm in this section.

5.1. Response time

Response time is defined as the time delay between when a node requests an exclusive access of a resource and when it actually allowed to enter the CS. It is generally calculated under light load conditions.

Case 1: Node i is present in the neighborhood:

Under light load conditions i.e. when non-CS time is greater than the CS time, the token holding node can be holding an idle token. Therefore, the request will arrive in time 'T' ($T = \text{maximum message transmission time}$), and the token will be reassigned to requesting node in T time. Hence, under the light load settings, the time of response turns out to be $2T$.

Case 2: A newly arriving node entering neighborhood in case it is in requesting state:

In this case T time is required by the neighboring nodes to receive the joining information of a new node and T time to forward token from the token-holder node in ideal state. Therefore, response time will be $2T$.

However, if the request collector moves out of neighborhood just before receiving the request, the request is resent by the node on receiving the request collector information. On an average, the request may need to resend m_1 -1 times, and each such iteration takes $2T$ time. Hence, $(m_1-1) * 2T$ has to be added in all the above cases.

5.2. Synchronization delay

Synchronization delay is significant if the token queue always has outstanding requests, i.e. in heavy load. Therefore, upon exiting from CS, the token-holder will be able to send the token to the head node of token-queue. It involves simple message propagation. Therefore, in this case, the synchronization delay will only be T .

6. Performance evaluation

"The Opportunistic Network Environment" simulator also known as ONE simulator is used to calculate the performance of RCLME via simulation experiments. It is an open source, Java based network simulator. It provides a rich set of node mobility capabilities along with integrated visualization. Main modules used in the simulation are movement, data, report, and core. For the proposed system, a UAV is presumed to be flying at a constant altitude in a3D space. Due to the swift dynamic proposed system, certain parameters were updated at regular intervals to ensure the dynamism of the system. The CS duration of a node is presumed to be exponentially spread with an average value of 500 milliseconds (μ_{cs}). Furthermore, the time for which a node remains idle is also presumed to be exponentially spread with a mean value (μ_{ncs}). The experimentations have been accomplished under medium load (contention level 50%) where contention level = $(\mu_{cs}/(\mu_{cs} + \mu_{ncs})) * 100$. To investigate the performance, RCLME is simulated in the low, medium, and high mobility rate i.e., Churn rate. The simulation parameters have been shown in [Table 1](#). Due to the limited availability of the space, the results with medium load only has been shown in the presented article.

Table 1
Simulation parameters.

Parameter	Value
Area of simulated environment	4500X3400 meter ² (Helsinki map)
Communication range of resource (UAV equipped with resource)	1000 meter
Number of nodes	20 to 100
Mobility of the nodes	Exponential distribution with the average speed between 0 m/s to 4 m/s (Used to adjust churn rate)
Moving time	Exponential distribution with average between 30 s to 50 s (Used to regulate churn rate)
Time spent in CS	Exponential distribution with average of 500 ms
Time spent in non-critical section	Exponential distribution (Used to adjust load)
Mobility speed of a resource equipped UAV	2.8 meter/second
Pause time	Exponential distribution with the average of 4 s.
Movement model	Random waypoint Movement
Routing protocol used	Direct delivery routing

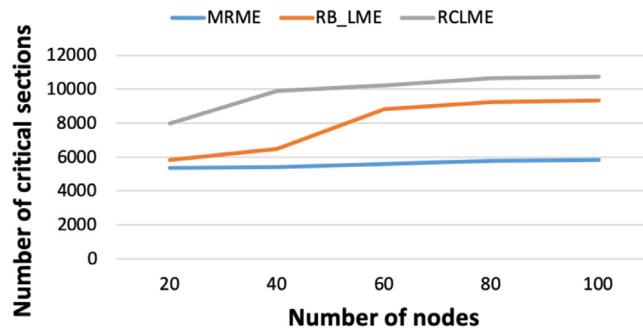


Fig. 3.1. Throughput (Low churn rate).

6.1. Results analysis

To show the dynamic performance of the RCLME algorithm for FANETs with the other contemporary algorithms, two token-based algorithms from the presented literature are carefully chosen, namely RB_LME [9] and MRME [12]. RB_LME is an LME algorithm for MANETs that uses a request collector-based approach where a request collector is always the last node in the token queue. Further, MRME follows a token-based approach to solve the ME problem for FANETs. The RB_LME and MRME algorithms are simulated in the proposed system to compare them with the RCLME algorithm. The performance parameters analyzed are throughput, message overhead, response time, and synchronization delay. The main findings of the comparison are given below and have been displayed in Figs. 3, 4, and 5.

Figs. 3, 4 and 5 shows that RCLME has a clear edge over the compared algorithms for almost all considered parameters.

The primary reason for the observed performance of RCLME is the employing of the request collector node as a leader instead of a conventional token holder node and the usage of fuzzy logic for selecting the request collector node. The applicability of fuzzy fitness function causes the selection of the best leader out of the available candidates. The best leader remains for the maximum amount of time in the highly dynamic environment of the FANETs, and the longer stay ensures better serving to the system comparatively. Furthermore, the introduction of request collector provides controlled broadcast, causing lesser number of message exchanges and also limits the usage of a node's resources. Collectively, the coupling of fuzzy logic along with request collector is a mutually complementary introduction of concepts in the proposed algorithm giving the best results for the RCLME algorithm. Further, in

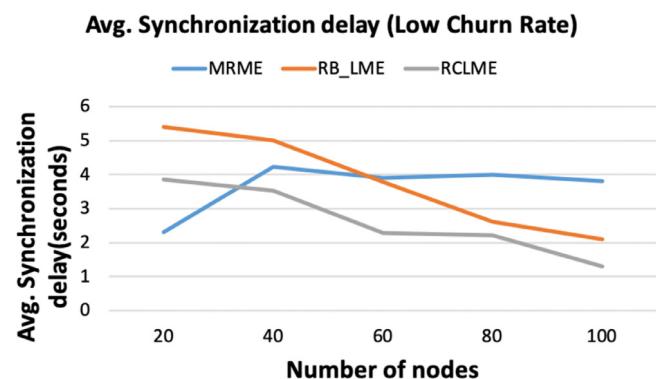


Fig. 3.2. Avg. Synch. Delay (Low churn rate).

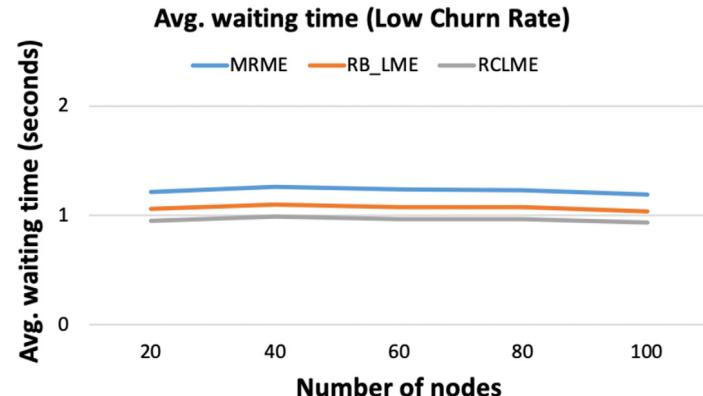


Fig. 3.3. Avg. Waiting time (Low churn rate).

the upcoming paragraphs, the detailed comparative analysis has been discussed.

As shown in Figs. 3.1, 4.1 and 5.1, the throughput of RCLME algorithm is highest followed by RB_LME and MRME algorithm. The reason for the same is the usage of fuzzy and request collector, as explained above. Another finding related to the throughput parameter is that the throughput is decreasing with the increase in the churn rate. The reason for the aforementioned trend can be explained by the fact that an increase in the churn rate of nodes leads to a decrease in the density of nodes in a neighborhood, thus causing a lesser number of fulfilled CS requests.

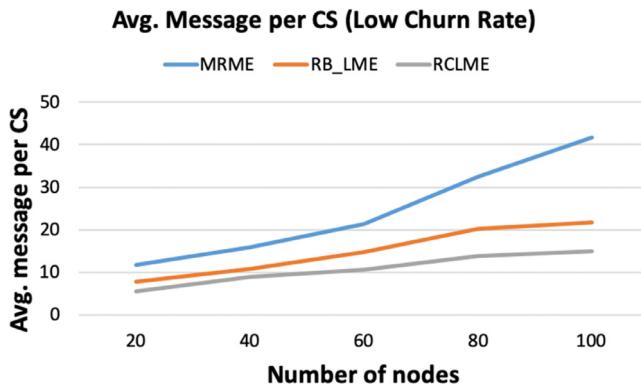


Fig. 3.4. Message complexity (Low churn rate).

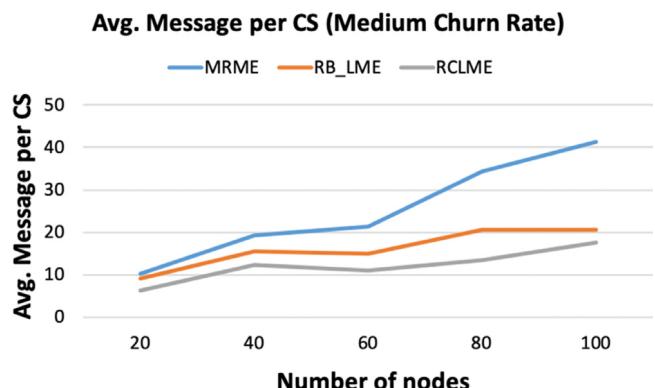


Fig. 4.4. Message Complexity (Medium churn rate).

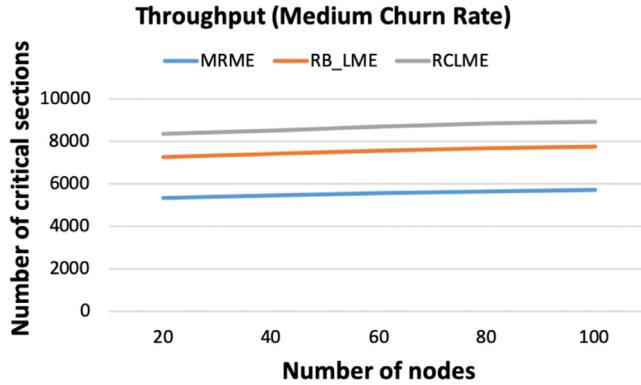


Fig. 4.1. Throughput (Medium churn rate).

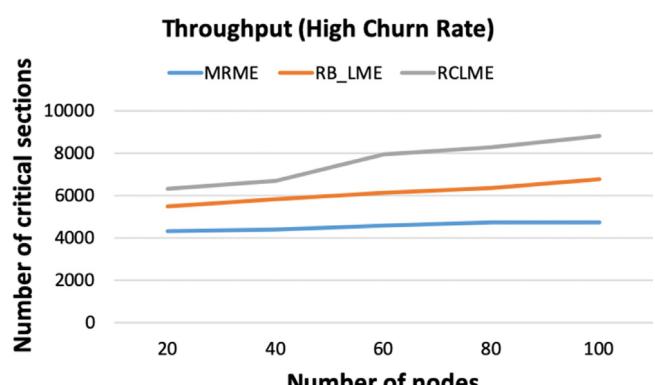


Fig. 5.1. Throughput (High churn rate).

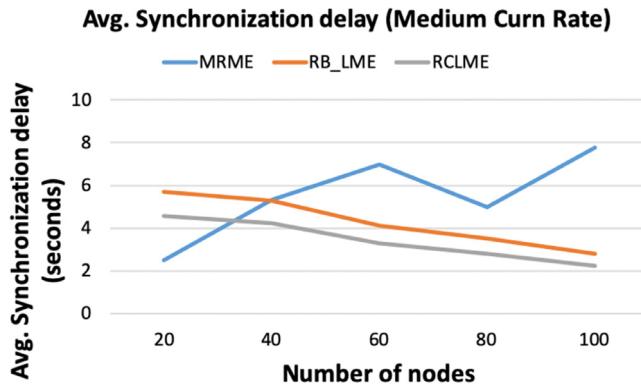


Fig. 4.2. Avg. Syn. Delay (Medium churn rate).

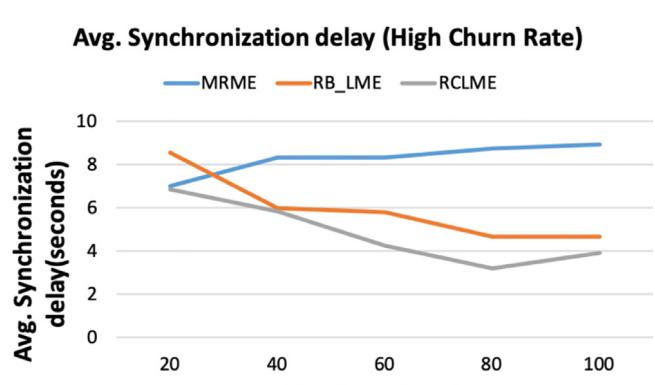


Fig. 5.2. Avg. Syn. Delay (High churn rate).

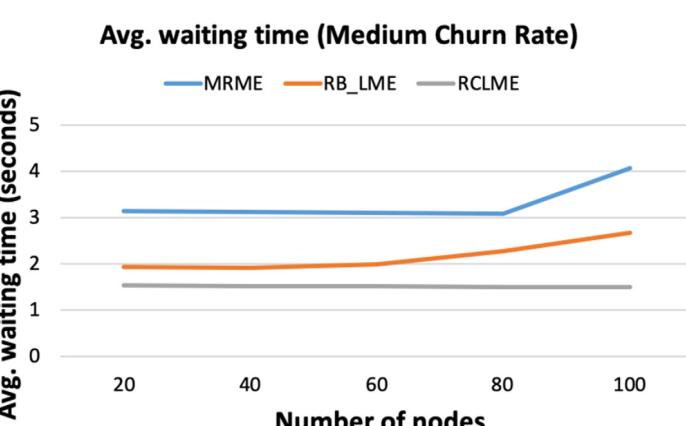


Fig. 4.3. Avg. Waiting time (Medium churn rate).

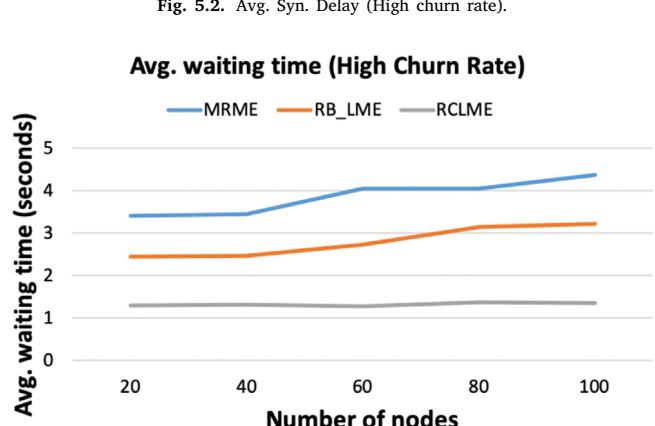


Fig. 5.3. Avg. Waiting Time (High churn rate).

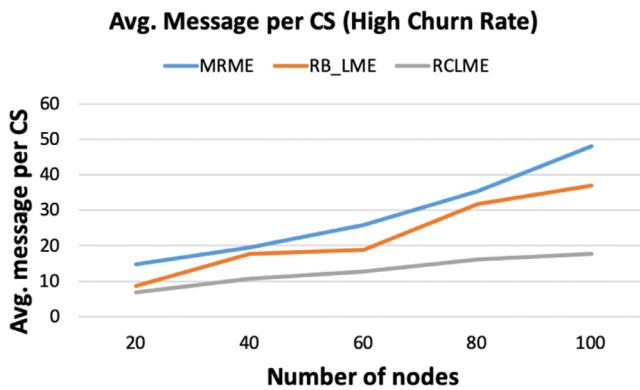


Fig. 5.4. Message complexity (High churn rate).

The synchronization delay of RCLME is notably better than RB_LME and MRME as shown in Figs. 3.2, 4.2 and 5.2. As far as RB_LME and MRME algorithms are concerned, both algorithms show slightly higher (20%–25%) synchronization delay than the RCLME algorithm. Hence, RCLME algorithm exhibits better synchronization delay than all other algorithms under consideration in most of the cases.

It is observed from Figs. 3.3, 4.3 and 5.3 that RCLME has the lowest waiting time as the request collector is infrequently changed due to fuzzy based Request Collector Election method, and the requesting node has the more updated information about the request collector. Hence, the request directly reaches to the request collector's request queue which reduces the waiting time.

The message overhead is least in RCLME, whereas it is maximum in case of MRME, as depicted in Figs. 3.4, 4.4, and 5.4. The primary reason for the highest message overhead in MRME is the broadcast of token information along with pending CS requests every time the token is transferred, or the token holder node leaves the neighborhood, which in turn results in a very high message overhead. However, RCLME and RB_LME use the request collector that ensures the controlled broadcast, which leads to reduced message overhead in comparison to MRME. It is noteworthy to mention that in RCLME, the message overhead increases slightly with the increase in the churn rate, whereas it increases significantly in the case of RB_LME. The reason for this significant increase in RB_LME is that the probability of request collector node moving out of the neighborhood increases with the increase in the churn rate, thereby increasing the number of broadcast messages for request collector information. However, in case of RCLME, the fittest node, i.e., the node which will stay for the maximum amount of time in the neighborhood, is elected as the request collector, hence broadcast overhead of request collector information is reduced in RCLME.

The major trends visible from the simulation results are summarized as follows:

- The average number of messages per CS entry is lowest in RCLME at low, medium and high churn rates.
- RCLME shows significantly better waiting time.
- RCLME outperforms other algorithms as far as synchronization delay is concerned except when there is small number of nodes in the system.
- The throughput of RCLME is better than other algorithms involved in the experiment.

7. Conclusion and future scope

FANETs is relatively an emerging variant of distributed systems with several unexplored applications. However, due to the swift mobility of FANETs environment in comparison to MANETs and VANETs, and to exploit the possible potential of these applications, new solutions need

to be introduced. In the presented exposition, a local mutual exclusion (LME) problem has been introduced in FANETs that has several possible applications. The concept of request collector using fuzzy logic to elect a request collector, handling of token loss make the presented solution efficient, fault-tolerant, and effective in comparison to the algorithms available. The dynamic analysis confirms that RCLME shows scalable performance at the medium load and high churn rate. Furthermore, the algorithm also handles the link breakage and link formation. Although, the algorithm is able to handle token loss, a more fault tolerant version able to handle message loss and other faults may be considered as a future work. Moreover, other variants of the mutual exclusion problem need to be solved in FANETs environments.

Declaration of competing interest

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

CRediT authorship contribution statement

Ashish Khanna: Conceptualization, Methodology, Writing - original draft. **Joel J.P.C. Rodrigues:** Conceptualization, Investigation, Resources, Supervision. **Naman Gupta:** Software, Validation, Formal analysis, Data curation, Writing - original draft. **Abhishek Swaroop:** Visualization, Writing - review & editing. **Deepak Gupta:** Supervision, Project administration.

Acknowledgments

This work has been supported by FCT/MCTES, Portugal through national funds and when applicable co-funded EU funds under the Project UIDB/EEA/50008/2020; and by Brazilian National Council for Scientific and Technological Development (CNPq) via Grant No. 309335/2017-5.

References

- [1] A.D. Kshemkalyani, M. Singhal, *Distributed Computing Principles, Algorithms, Systems*, Cambridge University Press, 2009.
- [2] J.E. Walter, S. Kini, Mutual Exclusion on Multihop Mobile Wireless Networks, Technical Report No. TR-97-014, Texas A & M University, College Station, 1997.
- [3] J. Walter, J. Welch, N.H. Vaidya, A mutual exclusion algorithm for ad hoc mobile networks, *Wirel. Netw.* 7 (2001) 585–600.
- [4] R. Roberto, R. Baldoni, A.A. Virgillito, Distributed mutual exclusion algorithm for mobile ad-hoc networks, in: Proceedings of the 7th International Symposium on Computers and Communications, 2002, pp. 539–548.
- [5] R. Chawla, P. Thakral, A.K. Kaura, K. Gupta, Modified centralized approach for preventing collision at traffic intersection, in: 4th International Conference on Signal Processing, Computing and Control, ISPC, Solan, IEEE, 2017, pp. 35–42.
- [6] J. Lim, Y.S. Jeong, D.S. Park, H. Lee, An efficient distributed mutual exclusion algorithm for intersection traffic control, *J. Supercomput.* 7 (2018) 1090–1107.
- [7] E.W. Dijkstra, Hierarchical ordering of sequential processes, *Acta Inform.* (1971) 115–138.
- [8] H. Attiya, A. Kogan, J.L. Welch, Efficient and robust local mutual exclusion in mobile ad-hoc networks, *IEEE Trans. Mob. Comput.* (2010) 361–375.
- [9] A. Khanna, Local mutual exclusion and its variants in mobile ad hoc networks (Doctoral dissertation), NIT, Kurukshetra, India, 2017.
- [10] A. Khanna, A.K. Singh, A. Swaroop, A leader-based k-local mutual exclusion algorithm using token for MANETs, *J. Inf. Sci. Eng.* 30 (5) (2014) 1303–1319.
- [11] A. Khanna, A.K. Singh, A. Swaroop, Token-based solution to group local mutual exclusion problem in mobile ad hoc networks, *Arab. J. Sci. Eng.* 41 (12) (2016) 5181–5194.
- [12] A. Khanna, Rodrigues. J.P.C., N. Gupta, A. Swaroop, D. Gupta, K. Saleem, V.H.C. Albuquerque, A Mutual Exclusion Algorithm for Flying Ad Hoc Networks, *Computers and Electrical Engineering*, Vol. 76, Elsevier, 2019, pp. 82–93, <https://www.sciencedirect.com/science/journal/00457906/76/supp/C>.
- [13] M. Maekawa, A \sqrt{N} algorithm for mutual exclusion in decentralized systems, *ACM Trans. Comput. Syst.* 3 (2) (1985) 145–159.
- [14] W. Zheng, L.S. Song, L. Mei'an, Ad hoc distributed mutual exclusion algorithm based on token-asking, *J. Syst. Eng. Electron.* (2007) 398–406.

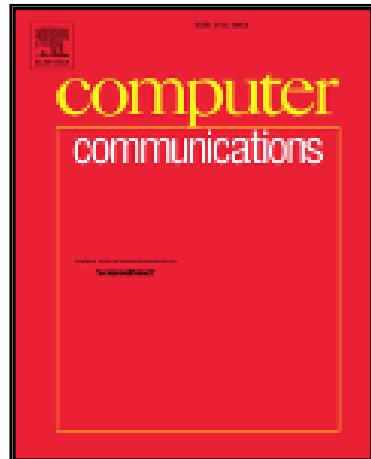
- [15] A. Derhab, N.A. Badache, Distributed mutual exclusion algorithm over multi-routing protocols for mobile ad-hoc networks, *Int. J. Parallel Emergent Distrib. Syst.* 23 (3) (2008) 197–218.
- [16] W. Wu, J. Cao, M. Raynal, A dual-token-based fault tolerant mutual exclusion algorithm for MANETs, in: the Proceedings of International Conference on MSN, 2007, pp. 572-583.
- [17] M.J. Fischer, N.A. Lynch, J.E. Burns, A. Borodin, Resource allocation with immunity to limited process failure, in: Proceedings of the 20th Annual Symposium on Foundation of Computer Science, 1979, pp. 234-254.
- [18] Y.J. Joung, Asynchronous Group Mutual Exclusion (extended abstract), in: Proceedings of 17th annual ACM Symposium on Principle of Distributed Computing, 1998, pp. 51-60.
- [19] A. Khanna, A.K. Singh, A. Swaroop, Dynamic request set based mutual exclusion algorithm in MANETs, *Int. J. Wirel. Microw. Technol. (MECS)* 5 (4) (2015) 1–14.
- [20] B. Sharma, R.S. Bhatia, A.K. Singh, DMX in MANETs: major research trends since 2004, in: the Proceedings of the International conference on advances in computing and artificial intelligence, ACAI, 2011, pp. 50-55.
- [21] M. Papatriantafilou, P. Tsigas, On Distributed Resource Handling: Dining, Drinking and Mobile Philosophers, in: International Conference on Principles of Distributed Systems, OPODIS, 1997, pp. 293-308.
- [22] S.A. Tamhane, M. Kumar, A token-based distributed algorithm for supporting mutual exclusion in opportunistic networks, *Pervasive Mob. Comput.* (2012) 795–809.
- [23] B. Sharma, R.S. Bhatia, A.K. Singh, A token-based protocol for mutual exclusion in mobile ad hoc networks, *J. Inf. Process. Syst.* 10 (1) (2014) 36–54.
- [24] H. Kakugawa, M. Yamashita, Self-Stabilizing Local mutual exclusion on networks in which process identifiers are not distinct, in: the Proceedings of the 21st Symposium on reliable distributed systems SRDS, 2002, pp. 202-211.
- [25] E.W. Carfang, A.J. Frew, D.B. Kingston, CaScaded optimization of aircraft trajectories for persistent data ferrying, *J. Aerosp. Inf. Syst.* 11 (12) (2014) 807–820.
- [26] I. Bekmezci, O.K. Sahingoz, S. Temel, Flying ad-hoc networks (FANETs): A survey, *Ad Hoc Netw.* (2013) 1254–1270.
- [27] Y. Yan, Y. Mostofi, Communication and path planning strategies of a robotic coverage operation, in: American Control Conference, Washington, DC, 2013, pp. 860-866.
- [28] D.T. Ho, E.I. Grotli, T.A. Johansen, Heuristic algorithm and cooperative relay for energy efficient data collection with a UAV and WSN, in: International Conference on Computing, Management and Telecommunications (ComManTel), Ho Chi Minh City, 2013, pp. 346-351.
- [29] I. Jawhar, N. Mohamed, J. Al-Jaroodi, Data communication in linear wireless sensor networks using Unmanned Aerial Vehicles, in: International Conference on Unmanned Aircraft Systems, ICUAS, Orlando, FL, 2014, pp. 43-51.
- [30] F. Al-Turjman, A novel approach for drones positioning in mission critical applications, *Trans. Emerg. Telecommun. Technol.* (2019).
- [31] F. Al-Turjman, S. Alturjman, 5G/IoT-Enabled UAVs for Multimedia Delivery in Industry-Oriented Applications, *Multimedia Tools and Applications*, Springer, 2018, pp. 1–22.
- [32] F. Al-Turjman, J.P. Lemayian, S. Al-Turjman, L. Mostarda, Enhanced deployment strategy for the 5G drone-BS using artificial intelligence, *IEEE Access* 7 (1) (2019) 75999–76008.
- [33] Fadi Al-Turjman, Hadi Al-oqily Zahmatkesh, Daboul Ibrahim, Reda, Optimized unmanned aerial vehicles deployment for static and mobile targets' monitoring, *Comput. Commun.* 149 (2019) 27–35.

Accepted Manuscript

DPTR: Distributed Priority Tree-Based Routing Protocol For FANETs

Vishal Sharma, Ravinder Kumar, Neeraj Kumar

PII: S0140-3664(16)30326-7
DOI: [10.1016/j.comcom.2018.03.002](https://doi.org/10.1016/j.comcom.2018.03.002)
Reference: COMCOM 5659



To appear in: *Computer Communications*

Received date: 10 September 2016
Revised date: 29 October 2017
Accepted date: 3 March 2018

Please cite this article as: Vishal Sharma, Ravinder Kumar, Neeraj Kumar, DPTR: Distributed Priority Tree-Based Routing Protocol For FANETs, *Computer Communications* (2018), doi: [10.1016/j.comcom.2018.03.002](https://doi.org/10.1016/j.comcom.2018.03.002)

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

DPTR: Distributed Priority Tree-Based Routing Protocol For FANETs

Vishal Sharma^a, Ravinder Kumar^b, Neeraj Kumar^{c,*}

^a*Department of Information Security Engineering, Soonchunhyang University, Asan-si 31538, Republic of Korea.*

^b*Computer Science and Engineering Department, Thapar University, Patiala, Punjab, India-147004*

^c*Computer Science and Engineering Department, Thapar University, Patiala, Punjab, India-147004*

Abstract

Collaborative Flying Ad Hoc Networks (FANETs) are mutually operating ad hoc systems comprising ground ad hoc and aerial ad hoc networks as their coordinating units. Coordination amongst different ad hoc networks and their hybridization elongate the application of ad hoc networks. These networks can be utilized for both military and civilian applications. However, despite possessing a lot of advantages, network partitioning and relaying are the major issues that arise when two or more networks coordinate with each other. There subsist a plethora of approaches, which highlighted the issues cognate to network partitioning as well as routing, but these fixated on such quandaries within the same network. Even the subsisting routing protocols counterfeit these problems, but with a scope inhibited to a single network and do not resolve these issues for two different operating networks. In this paper, the problem of network partitioning is considered between the aerial ad hoc network and the ground ad hoc network, and a felicitous routing protocol is proposed that can handle transmission in this mutually coordinated system. The proposed protocol resolves both the issues related to topology formation as well as routing between simultaneously operating nodes of two different ad hoc formations. The proposed protocol is derived over distributed Red-Black (R-B) tree, which forms a priority network that allows selection of an appropriate node and a channel for relaying, and is termed as Distributed Priority Tree-Based Routing Protocol (DPTR). The protocol operability is demonstrated using network simulations in comparison with the existing state-of-the-art routing protocols. The results suggest significant gains in channel utilization, packet delivery ratio, end to end delay, overheads, probability of connectivity and network throughput.

Keywords: FANETs, Routing, UAVs, Coordination, data dissemination

1. Introduction

Multi-Unmanned Aerial Vehicles (UAVs) guided networks are one of the major applications of Flying Ad Hoc Networks (FANETs) [1]. These are formed by collaboration between the aerial ad hoc and the ground ad hoc networks with one of the networks acting as a coordinating unit. These networks form a special class of distributed networks. Network operating in distributed mode has varied applications in areas such as surveillance, territorial-security, disaster management and navigation [2]. Unmanned aerial network operating in coordination with other networks increases maneuverability and provides in-depth navigation that increases the scalability of network and enhances performance in terms of coverage and range [3] [4].

Networking with UAVs has emerged as a new area of research in recent years [5] [6]. Recent surveys have shown that networking using UAVs can increase the applications and will lower the risks involved due to manned aerial vehicles [7]. Unmanned aerial vehicles have the capability to perform maneuverabilities autonomously. These can perform complex tasks based on certain computations that are fixed as on-board processors over these vehicles. Ground coordinated flying ad hoc networks are the set of simultaneously operating two ad hoc systems; one operating on the ground and another in the air. These can be treated as a single ad hoc formation to perform data transmission between them.

*Corresponding author

Email addresses: vishal_sharma2012@hotmail.com (Vishal Sharma), ravinder@thapar.edu (Ravinder Kumar), neeraj.kumar@thapar.edu (Neeraj Kumar)

Single UAV systems that are controlled by operators have already been implemented by various military agencies. However, controlling them from a distance without actually knowing the dynamics at a certain time instance lowers the applications and performance of these systems. The solution to such problem can be obtained using Multi-UAVs in a cooperative mode so as to overcome the limitations of the single UAV system [8] [9]. Many frameworks exist in the literature, which focuses on the formation of aerial swarms or aerial cooperative units for relaying between UAVs [10]. However, despite their varied applications, even these frameworks rely either on the UAV's local frame of reference or some navigation system such as Global Positioning System (GPS), Assisted-GPS or Inertial Navigation System (INS). Also, these multi-flying aerial vehicles neither use an on-ground guidance system nor do they form a mutual ad hoc network with ground nodes, thus, causing a limitation to such ad hoc formations. Recent research has focused on multi-UAV guided systems that comprise multiple ad hoc networks. These networks have reduced the geographical boundaries and have formulated a hybrid ad hoc network. Such networks can also be termed as *Mutual Ad Hoc Networks*.

Mutual ad hoc networks increase the applicability and operability of such dynamic networks. These ad hoc networks operating in coordination with each other enhance the operations of the multi-UAV system and overpower the limitations of a single UAV system. However, data transmission and forwarding is a complex task in such networks as parameters like mobility, location, difference in processing power, delays along with network partitioning cause hindrance in network operations. Despite frameworks that unite these two ad hoc networks, there is a requirement of certain protocols that will provide routing facility within such networks to efficiently transfer data between two simultaneously operating ad hoc units.

Network partitioning¹ leads to different pieces of a network which may inhibit a particular piece from participating in the transmission. A network may become isolated if the partitions remain unidentified as there will be no information regarding the nodes that no longer receives any information. Network partitioning can also result when two differently operating networks are combined together as in the case of mutual ad hoc formations. Such networks are much tedious to manage as these require an effective strategy for managing topology as well as addressing of the nodes. Lack of metadata and need for more tabular information further elongate the issues of transmission in partitioned networks. Routing becomes tedious and excessive overheads are induced in the network causing a high end to end delay while transmitting information over the network. Such scenarios can be seen in patrolling services (surveillance)- which use both ground and aerial nodes in combination with each other, disaster scenarios- where the connecting nodes between the two networks may fail and lead to partitioning, capacity and coverage enhancement of networks- which use UAVs in combination with ground terminals for better services, etc. Thus, it is of utmost importance to understand the risks and management implications associated with the aerial-ground ad hoc formations along with their possible solutions for enhancing their utilization.

The existing solutions to network partitioning rely on the identification of each partition and then use a separate address schema for resolving it [11]. Message ferrying is also one of the efficient solutions, which uses proactive routing for transmitting data in wireless partitioned networks [12]. However, such solutions are more applicable to stationary networks and need a combination of routing schemes for their applicability to general wireless networks. Akkaya *et al.* [13] and Wang *et al.* [14] focused on distributed recovery via controlled mobility during high partitioning in mobile and sensor networks, respectively. Repairing of network partitioning can also be considered, but these are limited to a single network and cannot be considered for distributed networks [15] [16]. Further, genetic algorithms and game theory can also be used for supporting transmissions in highly partitioned networks [17] [18]. All these solutions are efficient subject to their domain, but their scope for handling aerial nodes is limited as these lack parameter configurations for supporting aerial-ground coordination. Further, most of them rely only on resolving partitions and do not ensure relaying between the partitions. Thus, it is necessary to develop a generic solution for aerial-ground coordination, which can handle their dynamic behavior and can support transmissions without failing.

In this paper, the problem of routing between aerial and ground ad hoc networks is addressed along with a solution for topological formations while operating these two networks together. A solution that solves the problem of network partitioning along with the provision of routing is proposed. The protocol is termed as Distributed Priority Tree-based Routing (DPTR) Protocol as it uses the properties of an R-B tree and formulates distributed routing trees by adding-up certain rules in the formation of these trees, thus, providing a solution that allows routing to be performed over

¹<https://www.rfc-editor.org/ien/ien120.txt>

simultaneously operating distributed ad hoc networks. The rules are helpful in building the network and also resolve the network isolations. These rules are further extended for the inclusion of protocol functioning schema followed by addressing schema and path formations. The proposed protocol is evaluated via simulations conducted using Network Simulator (NS-2) and Matlab in three parts. The first part evaluates the performance for aerial ad hoc formations, the second part evaluates it for ground ad hoc formations and the third part analyzes the performance for mutual ad hoc formations. The evaluations are presented for Packet Delivery Ratio (PDR), end to end delays, channel utilization, network throughput, network connectivity time and the probability of connectivity. Initially, the results are presented for the standalone formation of mutual ad hoc network between the aerial and the ground nodes and these are studied in comparison with the existing state-of-the-art routing protocols. A comparative table is also presented and discussed for evaluation of the proposed protocol with the existing solutions, which focus on the issues of network partitioning as well as FANETs.

Remaining paper is structured as follows: Historical perspective and related work are discussed in section 2. Proposed routing scheme is explained in section 3. Section 4 provides the detailed working of the proposed routing protocol. Simulations and results-analysis are provided in section 5 and section 6, respectively. Finally, section 7 concludes the paper with its distinguished features and highlights.

2. Historical Perspective and Related Works

Ad hoc networks have evolved as self-arranging networks that have the capability to form a network in a heterogeneous environment without any usage of heavy gadgets and centralized infrastructure. Ad hoc operating on the ground has seen tremendous research and implementation during the last decade. The applications of such networks have now been extended by evolving other network areas that can be clustered together with traditional ad hoc networks to form newer ad hoc families such as mutual ad hoc networks, flying ad hoc networks or swarm ad hoc networks. Ad hoc network formation between the two varied networks that have implementation barriers and working differences is a complex task. Such network requires a connecting base and then a routing unit that can handle transmission between these simultaneously operating networks [19] [20]. Routing is a simple term with complex tasks. Routing involves pathfinding, path selection and decision regarding transmission, receiving, and forwarding of data. Many routing protocols have been developed during recent years that primarily focused on the provision of routing between the aerial vehicles.

Lin *et al.* [21] have designed a routing strategy for UAVs that utilizes the geographical location of nodes for the purpose of seamless routing. The approach uses GPS for navigation and predicts the mobility of UAVs. Further, the authors used this prediction to estimate the node connectivity before data transmission. Pre-configured beacons are used to exchange location amongst UAVs. The mobility model selected for the approach is based on the Gauss-Markov modeling.

Another formation using UAVs can be aerial swarms. Hauert *et al.* [22] have implemented an aerial swarm that does not rely on positioning system and is capable to provide relaying between the aerial nodes. The approach operates by defining a base launching station that controls the aerial relaying. The artificial neural schema is used to estimate the UAVs maneuvers. A fitness function is defined for the whole network that controls the connectivity on the basis of a number of paths available to each node. Gu *et al.* [23] have proposed a 2-level architecture for UAVs in ad hoc mode. The protocol uses the extension of hierarchical state routing and forms an intelligent state routing that can be used for relaying in UAVs. Protocol divides the network into logical and physical partitioning and then utilizes the message passing to allow routing. GlomoSim is used to evaluate the performance of the protocol.

In another approach by Bellur *et al.* [24], a team of autonomous vehicles is formulated. The global networking paradigm described in the paper utilizes intra-team and inter-team relaying. The authors aimed a provision of architecture that provides flexible transmission between aerial vehicles. The flight experiments in paper use TCP-traffic, generated by the netperf tool. The network is tested over multi-hop paradigms.

Mobile ad hoc networks can prove handy in the implementation of UAV ad hoc networks. This application is identified by Rubin *et al.* [25]. The authors designed a network that utilizes the backbone that focuses on mobility as a major parameter of backbone node. The protocol defined in the paper aims at similarity on-demand operations of AODV along with the support of mobile-backbones in the networks. These backbones are replaced with UAVs that facilitates the localization of unmanned ground vehicles in the network.

Iordanakis *et al.* [26] have focused on another sort of aerial ad hoc network i.e. aeronautical ad hoc network. Unlike to UAV ad hoc network, the traffic requirement and resource utilization are of a varied range that requires more functional and routing units. The protocol ARPAM given in the paper is an extension of AODV protocol that has the capability to provide VOIP service over the aeronautical network. In another approach by Cetin *et al.* [10], the authors aimed at the formation of potential fields that are artificially driven and formed a relaying with UAVs that provides collision avoidance. Artificial field identifies any potential obstacles that hinder the line of sight for communication. The model derived is capable to identify obstacles on the basis of parameters provided for particular objects during simulations. Matlab and NS-2 are used to test the operations on the potential fields.

Shetty *et al.* [27] designed a surveillance protocol that aimed at routing amongst UAVs on the basis of identification of targets in the defined zone. It is more of artillery based combat routing protocol that aims at the efficient utilization of available military resources. Enright *et al.* [28] focused on similar issues and designed a stochastically operating routing protocol that aims at surveillance of pre-defined zone in a stochastic manner. It is more of the algorithmic approach rather than a networking modeling. A comparative analysis of shortest path algorithms for UAVs is presented by Sathyaraj *et al.* [29]. The authors identified networking with UAVs and tested its operability over Dijkstra, distance vector and Astar approaches. The analysis of different pathfinding algorithms over UAVs is presented using Matlab. Pavone *et al.* [30] have identified a vehicle routing problem that aims at providing services at particular time variance. Authors have considered both the time and spatial dependencies to form a request generator stochastic process that creates a requirement for a particular service for a fixed span of time. The approach aims at the selection of minimum resources and agents to handle all the requests.

Zhang *et al.* [31] have given a routing approach that focuses on data collected from the field of wireless sensors. This approach can be treated similarly to the one given by Rubin *et al.* [25]. The approach focuses on the formation of the aerial relaying model using unmanned vehicles and then utilizes the underlying sensor network to transfer the information. Aerial nodes also act as relaying nodes of the sensor network. The analysis shown in the paper demonstrates that only temporarily connected network is possible with their approach as UAV dynamics are not handled using their designed protocol [31]. Li *et al.* [32] have designed a cooperative scheduler for UAVs that exchange token to form a cooperative network amongst UAVs. The approach utilizes the features of CDMA network and similar to that, one of the UAVs always act as connecting the end with the base station. The approach is more of guided approach rather than an independent surveillance system.

There is no direct mechanism in the existing routing protocols for supporting transmission during network partitioning. However, the existing solutions use distributed addressing for supporting such networks. These networks provide addressing by identifying the divided networks and operating each as a different subnet. Such approach can be used for most of the protocols, but lack of topology management during partitioning makes it difficult to sustain data rate, and hence, may lead to failures, frequent interruptions, or end to end delay.

Routing strategies of hybrid ad hoc networks, as well as heterogeneous ad hoc networks, are also capable of resolving issues related to the operations considering distributed ad hoc formations [33]. These networks usually consider different types of nodes in their networks, and thus, can be good candidates for topology formation as well as routing over aerial-ground ad hoc networks. Brahmin *et al.* [34] proposed a routing strategy for connected vehicles in a hybrid ad hoc formation. The authors developed a routing strategy, which selects the best suitable radio technology for transmitting data over the network. Such solutions can be used for identifying the type of technology, which prevails in the mutual ad hoc formations, and then, an applicable strategy can be used for transmissions. Amongst another approach can be the routing strategy given by Lu *et al.* [35]. This approach is also efficient in handling transmissions in hybrid scenarios. However, at the moment, this solution is evaluated particularly for a coal mine-based network and requires sufficient proof-theory for its applicability to aerial guided ad hoc formations.

Similar to hybrid formations, heterogeneous networks operating with sensor nodes can also support topological and routing requirements of aerial ad hoc formations. Kumar *et al.* [36] proposed a routing strategy for heterogeneous Wireless Sensor Networks (WSNs). The approach uses clustering mechanisms for data transmissions. Although efficient for WSNs, yet this approach needs sufficient testing before its applicability to aerial networks. P-SEP routing protocol by Naranjo *et al.* [37], which is based on stable election algorithm, can also be extended for aerial networks considering its performance for battery-limited WSNs. Despite these competitive solutions, there is a limitation of literature, which considers routing as well as topology formation for resolving network partitioning problems in collaborative FANETs.

Apart from these solutions, tree-based approaches are capable of supporting hierarchies, which help the coor-

dination between the multiple networks. The advantages of tree-based routing are its easier support in identifying appropriate nodes as well as determining the precedence in terms of different parameters over which a given protocol is operated and evaluated [38] [45]. Tree-based routing is highly popular and used for the majority of the networks. Comparative information for various tree-based routing protocols developed over the years for different networks is presented in Table 1.

Table 1: Tree-based routing protocols

Protocol	Author (s)	Ideology	Application Network
STOD-RP	Zhu et al. [39]	Spectrum tree-based on demand routing	Cognitive radio networks
SMPC	Sajid et al. [40]	Singular division of multipath power control tree based routing protocol	Underwater WSNs
Optimal tree routing	Sohan et al. [41]	Tree-based routing with particle swarm optimization	WSNs
ETSW	Liang et al. [42]	Encounter history-based routing	Opportunistic networks
Self-organizing protocol	Qiu et al. [43]	Tree-based organization of networked and non-networked nodes	Internet of Things
Self-organized energy balanced protocol	Han et al. [44]	Tree-based minimal energy consumption	WSNs

From the related works, it can be identified that there are limited approaches which focus on relaying between different networks operating simultaneously to form a guidance system. The networks operating in distributed mode face a similar problem of network partitioning. The next section addresses this problem and provides a solution to it along with a routing strategy to form guided ad hoc networks.

2.1. Background

Networks involving highly dynamic environment require data structures that have self-balancing properties. Such data structures can solve the issues involving hybrid network formations. One of the best examples is R-B trees [46]. These are the binary search trees that can organize themselves efficiently and can have well-balanced properties. Govern by the rules, if these data structures are well modified, can provide a solution to network partitioning problem that involves multiple networks operating simultaneously. More about R-B trees and its variants can be studied from literature given by Cormen *et al.* [46] and Kahrs [47]. In this paper, a basic data structure of R-B trees is modified and based on the certain rules, a new routing protocol is proposed for collaborative flying ad hoc networks.

3. DPTR: Protocol Description

For a network comprising two different operating units with a variable number of nodes and diverse parameters, it is an extremely strenuous task to formulate a routing protocol. The major problem is of network partitioning. As discussed previously, network partitioning is not a new problem, it has been there for years and no exact solution has been derived from it. In the multi-UAV guided network, this problem exists because of simultaneously operating two networks. The major task for a routing protocol is to decide the interacting nodes in both the networks which will establish a network corridor to allow routing in diversified mode. The protocol proposed in the paper utilizes the features of the R-B tree, and with some modification and constraints to tree formation, it becomes feasible to find the connecting nodes from both the network, thus, solving the problem of network partitioning. Not only R-B trees make it attainable, IPV6 is also disintegrated to provide an addressing support for such networks. The proposed protocol operates in three parts.

- i. Identification of ground node.
- ii. Identification of aerial node.
- iii. Interfacing using neural structure.

The routing protocol utilizes the neural framework [48] for its operability and senses the tree formation to form a routing table comprising both aerial and ground nodes.

3.1. Identification of Ground Node

The ground node identified acts as a network corridor for the aerial ad hoc via an intermediate neural layer. The primary task is to select the interacting node. The selection is carried out on the basis of certain priority given to each node. The node with a higher priority interacts with the above lying aerial ad hoc network. The nodes are labeled as R_1, R_2, \dots, R_n in order of final selection, where R_1 is the node with the highest priority. This order helps in selection of root for R-B tree. The order of priority is formulated on the basis of weight assigned to each node. These weights are normalized over value of each node for number of configurable parameters for underlying ground networks and number of aerial corridors supported by the node along with the degree of ground connectivity w.r.t. mobility. For example, consider a network with 6 ground nodes marked as N_1, N_2, N_3, N_4, N_5 and N_6 . The configurable parameters considered are:

- x_1 : Minimum achievable rate
- x_2 : Number of aerial corridors
- x_3 : Mobility
- x_4 : Degree of connectivity
- x_5 : Link speed

such that Priority (\mathcal{P}) for these parameters is of the order: $\mathcal{P}(x_4) > \mathcal{P}(x_2) > \mathcal{P}(x_1) > \mathcal{P}(x_5) > \mathcal{P}(x_3)$. Here, priority refers to the precedence of parameters for the proposed protocol. The priority helps in determining the nodes for the developing the proposed protocol by using the R-B tree.

Now, the node data are evaluated in this priority order to formulate the priority order for ground nodes. To explain it further, following example is formulated:

Let for $\mathcal{P}(x_4)$, node order be: $N_6 > N_2 > N_3 > N_1 = N_4 > N_5$. Thus, $\mathcal{P}(x_4)$ does not provide with complete priority orders for ground node. To obtain complete solution, N_1 and N_4 are evaluated over priority $\mathcal{P}(x_2)$, even if it comes equal, continue evaluating for next priority mark. Let say that $N_1 = N_4$ for $\mathcal{P}(x_2)$ and $\mathcal{P}(x_1)$, but for $\mathcal{P}(x_5)$, $N_4 > N_1$, then the final priority order for nodes is: $N_6 > N_2 > N_3 > N_4 > N_1 > N_5$, i.e.

$$\begin{aligned} R_1 &\leftarrow N_6 \\ R_2 &\leftarrow N_2 \\ R_3 &\leftarrow N_3 \\ R_4 &\leftarrow N_4 \\ R_5 &\leftarrow N_1 \\ R_6 &\leftarrow N_5 \end{aligned}$$

In the next step, a distributed R-B priority tree comprising red, black and green nodes is constructed by adding the following rules to its default properties:

- **Rule 1:** The highest priority node cannot be part of inner R-B tree, i.e. R-B tree will be formed from nodes other than R_1 .
- **Rule 2:** The node with the highest priority in set $\{R_2, R_3, R_4, R_5, R_6\}$ should be the rightmost children with both of its children as NULL.
- **Rule 3:** R_1 forms an isolated tree with its connectivity to inner R-B tree.
- **Rule 4:** The inner-tree must preserve the properties of default R-B tree.
- **Rule 5:** Each inner tree must have one guider node colored in green i.e. both of its children should be a network node but not NULL.

- **Rule 6:** Root can act as guider node.
- **Rule 7:** An Isolated tree will be connected to guider node only. It cannot be connected directly to any red node.
- **Rule 8:** The guider node for connectivity must have Priority greater than the root node i.e. $\mathcal{P}_{guider} > \mathcal{P}_{root}$.
- **Rule 9:** If no such node exists as required in Rule 8, then the root node will be selected as connecting guider node.
- **Rule 10:** Priority order amongst guider node is set according to the degree of connectivity $\mathcal{P}(x_4)$ of each guider node.

Data aggregation and map-reassembly are performed on the red colored nodes of the network, black nodes act as simple relaying hops and green node act as guider node that controls the transmission route of the network.

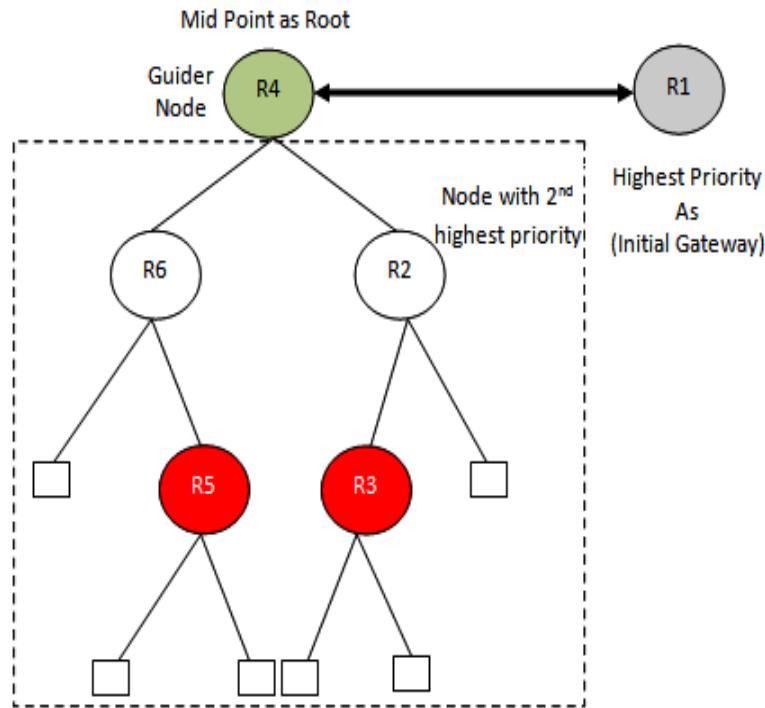


Figure 1: R-B tree with midpoint as root for ground network - violated Rule-3, requires re-adjustment

To form the distributed R-B tree, the midpoint of R series is taken as a root. There might be a case which violates the conditions specified above to form a routing tree by taking midpoint as root. Thus, re-adjustments are then carried to form an R-B tree that follows all the properties prescribed for selection of priority node. Figure 1 shows that the tree formed using midpoint, does not satisfy all the properties prescribed for R-B routing tree. Thus, to solve the problem, either of the adjacent node is selected to form a routing tree and properties are checked again. Re-operations are performed until all the properties are satisfied. The solution to Figure 1 is shown in Figure 2.

3.2. Identification of Aerial Nodes

Let A_1, A_2, \dots, A_n be the priority order for aerial nodes. A_1 to A_n , similar to ground nodes, depend upon certain parameters opted for aerial network, such that for a given aerial network U_1, U_2, U_3, U_4, U_5 and U_6 are the UAVs with following configurable parameters:

- y_1 : Corridors supported

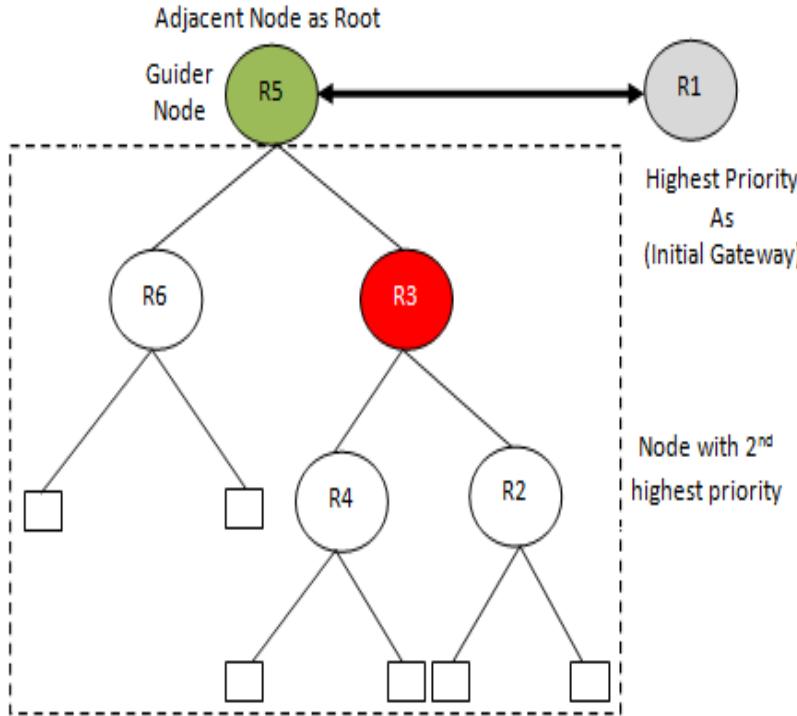


Figure 2: R-B tree with adjacent node as root for ground network - satisfies all properties

- y_2 : Number of cells traversed
- y_3 : Number of cognitive maps
- y_4 : Range

such that Priority (\mathcal{P}) for these parameters is of the order: $\mathcal{P}(y_2) > \mathcal{P}(y_4) > \mathcal{P}(y_1) > \mathcal{P}(y_3)$. The data over this priority order is evaluated to select the highest priority node. Let say for $\mathcal{P}(y_4)$, the node order obtained is : $U_6 > U_3 > U_2 > U_4 > U_5 > U_1$ i.e.

$$A_1 \leftarrow U_6$$

$$A_2 \leftarrow U_3$$

$$A_3 \leftarrow U_2$$

$$A_4 \leftarrow U_4$$

$$A_5 \leftarrow U_5$$

$$A_6 \leftarrow U_1$$

Figure 3 illustrates a distributed R-B priority tree constructed for aerial network using properties defined in subsection 3.1.

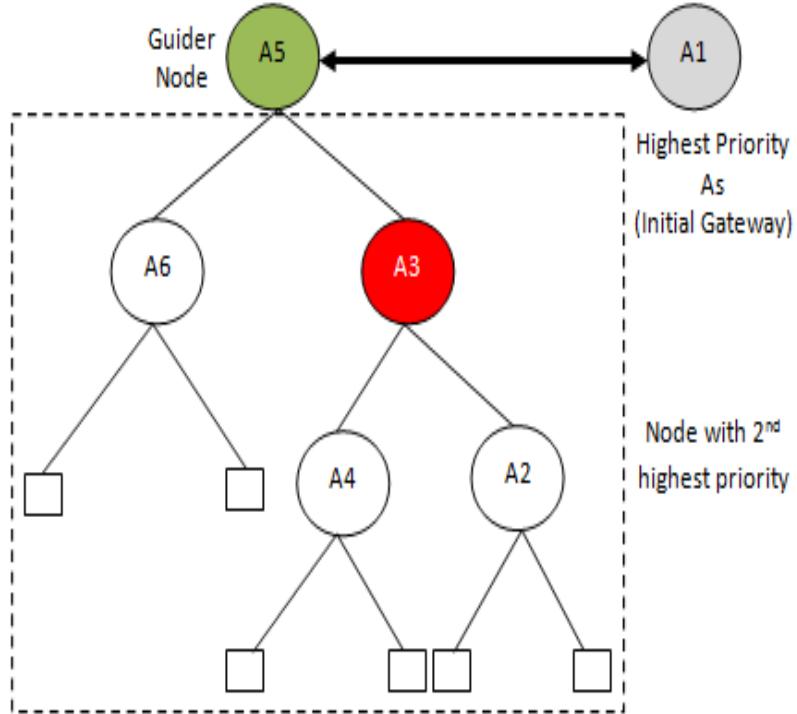


Figure 3: R-B tree for aerial network - satisfies all properties

3.3. Interfacing Using Neural Structure

The primary focus of routing protocol for a multi-UAV guided network is to select the coordinating nodes from both the network and provide them with interfacing for routing between two variable networks. The coordination between the aerial and the ground network is handled using the cooperative framework and routing is supported using neurons of the cooperative framework. The cooperative framework provides network connectivity between the two distinguished ad hoc networks and provides topology management between them [48] [49]. These neurons bridge up the gap between two different networks. Figure 4 shows the structure of routing tree that will be constructed by the protocol during the initial phase. Thus, the final routing tree comprises three R-B trees: R-B for the ground network, R-B for aerial network and R-B for interfacing. The interfacing of the two networks is subject to the following conditions:

- i. Only black to black routing or coordination is allowed for two nodes of different networks.
- ii. Node colored in red cannot directly interact with a node of another network.
- iii. Red nodes will be used to combine the maps obtained during routing. It is better and safer in terms of power consumption to utilize the computation power of red nodes as these do not participate in corridor formations.
- iv. Guider nodes in the network will act as internal corridors for the rest of the network nodes.

It can also be stated that red node in the network acts as a pivot to support all the routing operations of two variably operating ad hoc networks.

3.3.1. Virtual Red-Black Routing Layer

Two routing trees are created in the virtual R-B routing layer from A_1 to R_1 and vice-versa, with k_n number of active neurons that supports routing. $F_1, F_2, F_3 \dots F_{k_n}$ are created on the basis of various logs that are maintained

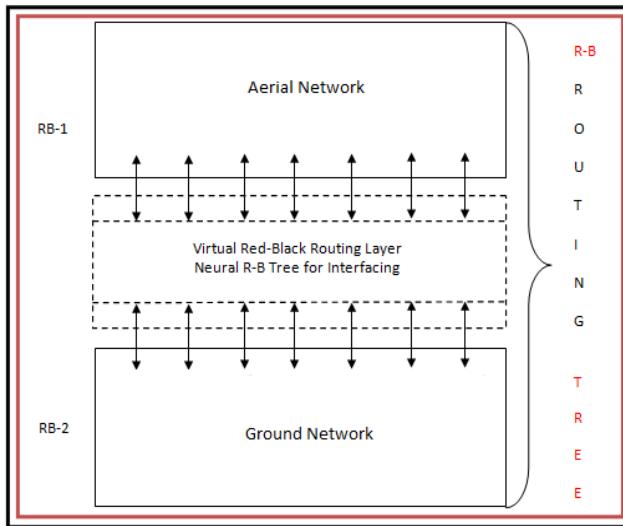


Figure 4: R-B Complete Routing Tree

during routing. To further clarify, consider an interfacing layer that requires seven neural structures to maintain logs, namely, $N_1, N_2, N_3, N_4, N_5, N_6$ and N_7 . Further, to create R-B (1-2) tree, following steps are performed:

- Place the destination Node as a highest priority node.
- Place the source as a midpoint node on the neural list

This forms a tree series given by: $F_1 > F_2 > F_3 > F_4 > F_5 > F_6 > F_7$ i.e.

$$F_1 \leftarrow (\text{mapped}) R_1, F_1 \leftarrow A_1$$

$$F_2 \leftarrow N_1$$

$$F_3 \leftarrow N_2$$

$$F_4 \leftarrow (\text{mapped}) A_1, F_4 \leftarrow R_1$$

$$F_5 \leftarrow N_3$$

$$F_6 \leftarrow N_4$$

$$F_7 \leftarrow N_5$$

By using the following rules, a combined virtual routing layer tree is constructed for multi-UAV guided networks.

- **Rule 1:** Neurons form an inner R-B tree with its default rules given in subsection 3.1.
- **Rule 2:** The node from source network interacts with the root node, which is a guider node for such network in all cases.
- **Rule 3:** The node from receiver network interacts with the rightmost node having both children as NULL.

Similarly, the inverse is performed to obtain the R-B (2-1) tree with A_1 as destination and R_1 as source. Both the trees can be represented by Figure 5, where F_2, F_3, F_5, F_6 and F_7 are log maintenance nodes.

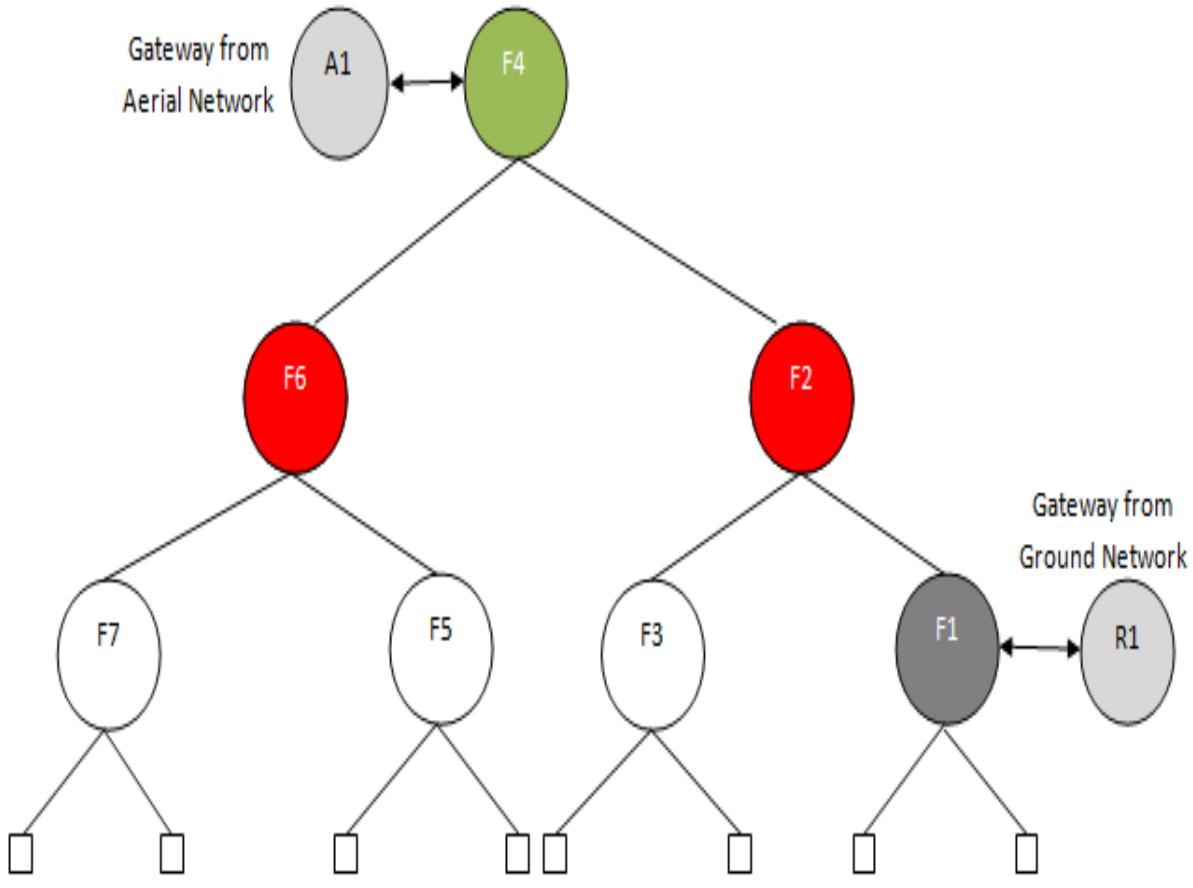


Figure 5: R-B Tree for Virtual Routing Layer

3.4. Final Routing Tree

The three trees formed in process of identification of priority nodes are combined to form a final routing tree that allows the formation of the adjacency matrix to coordinate amongst nodes of the different network, as shown in Figure 6. The internal nodes in the final routing tree are actual routing units and external nodes behave as a reset or exit conditions for the network. The tree suggests that purely randomized structure is obtained during the initial path searching process that can further define randomized graphs. The structure depends on the location of nodes defined on the basis of configurable parameters and searchable area initialized in the form of cells using a cooperative framework for such networks.

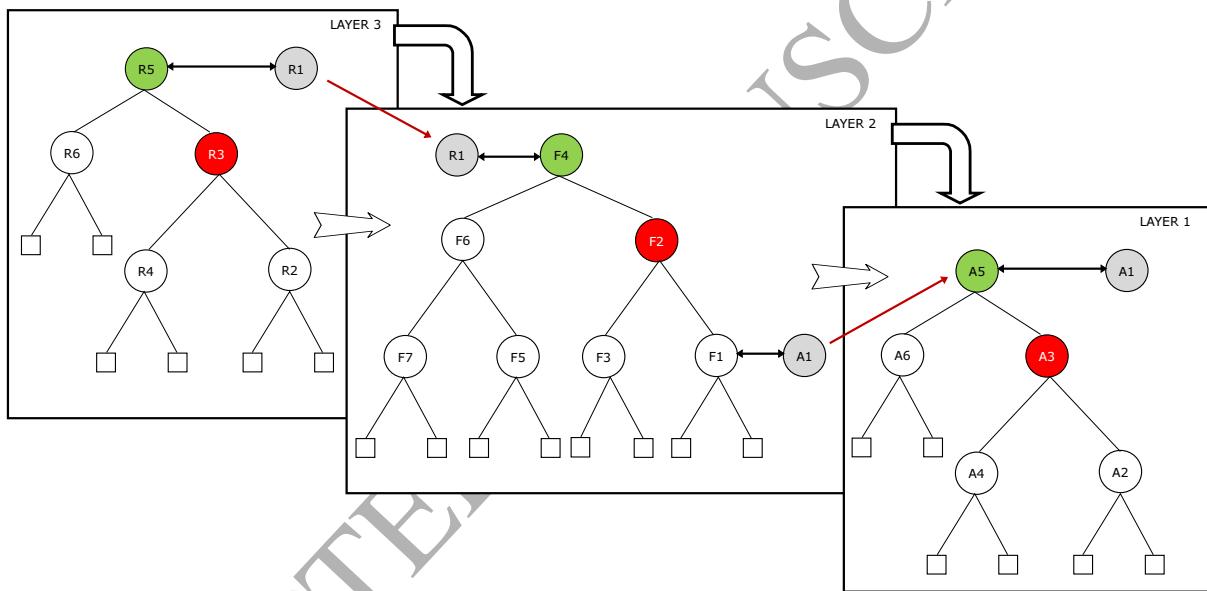


Figure 6: Layered View of Routing Trees

3.5. Proactive Functioning of Protocol

There might be a case where ground node connectivity is defined in the form of some directed or undirected graph. However, pre-connectivity of the aerial node is not suggested as mobility is extremely high and cooperative framework requires continuous updating in the movement of UAVs involved in the guided network. Thus, taking into consideration the pre-designed map, a linked structure of the ground node is considered in the formation of routing tree. Let G be a graph (N, L) where N is the nodes and L is the weighted edge between two nodes defined as

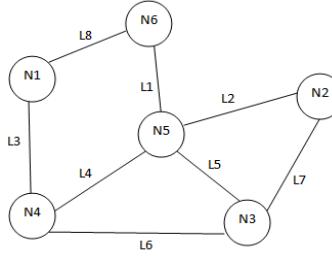


Figure 7: Weighted Graph - Link Speed

normalized link speed, as shown in Figure 7. From the defined graph, a best suited path is formed by constructing the R-B tree for ground nodes. Considering this for a given normalized link speed, the relational series is given as: $L_8 > L_1 > L_2 > L_3 > L_4 > L_5 > L_6 > L_7$. This relation also defines the priority order of the connected nodes with prescribed incident links. Following this, an R-B tree for link speed is constructed as shown in Figure 8.

Two trees can be created on the basis of priority-order; one by taking the first element of the link and another by taking the second element of the link. In order to create the tree, following steps are performed to identify priority set:

- From the priority order of links, construct the node priority set as: $S: \{(N_1, N_6) > (N_5, N_6) > (N_2, N_5) > (N_1, N_4) > (N_4, N_5) > (N_3, N_5) > (N_3, N_4) > (N_2, N_3)\}$, where N_1 and N_6 have equal priority.
- Now, select the first element of each unit from above set S and make priority order according to their first appearance i.e. $N_1 = N_6 > N_5 > N_2 > N_4 > N_3$. In case, if any element does not appear as the first element in the given sequence, then consider the second element sequence only to identify priority order of non-existing elements.
- Construct two R-B priority trees, one with N_1 as highest priority node taking priority order $N_6 > N_5 > N_2 > N_4 > N_3$ and another with N_6 as priority order node with priority order $N_1 > N_5 > N_2 > N_4 > N_3$.

By following the above steps, two R-B priority trees are formed, as shown in Figure 9 and Figure 10. Thus, the above methodology defined to form a routing tree by considering the topology of two different networks creates a path similar to a traversal order. But, for pre-defined graphs, the highest priority node acts as an entry point to a particular network. The approach used for route formation is capable of solving the problem of network partitioning and performing route discovery at the same time.

3.6. Protocol Addressing Schema

From the protocol description, it is clear that two different networks operate in the coordinated network structure. Thus, addressing scheme should be capable enough to handle both networks individually as well as it must provide with proper gateway treating the whole unit as a single network. Thus, for this split network formation, IPV6 is used as it provides an addressing length of 128 bits. The addresses are split that generates two subnets and two identifiers. Aerial ad hoc is given 64-bit identifier with 64-bit subnet to its neural interface. Similarly, the ground unit is given 64-bit subnet with a 64-bit identifier to its neural interface. Thus, forming an addressing schema comprising three IPV6 addressing structures. This provides with a facility to allocate IPV6 addressing scheme to the network by creating multiple virtual layers for IPV6, as shown in Figure 11 and Figure 12.

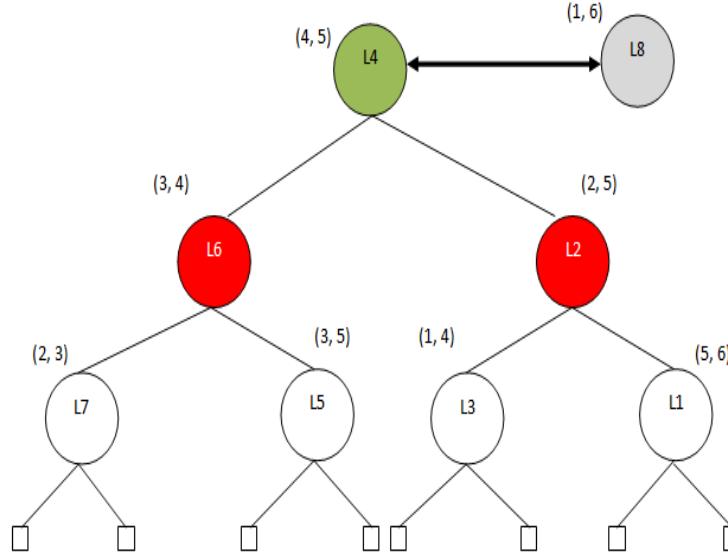
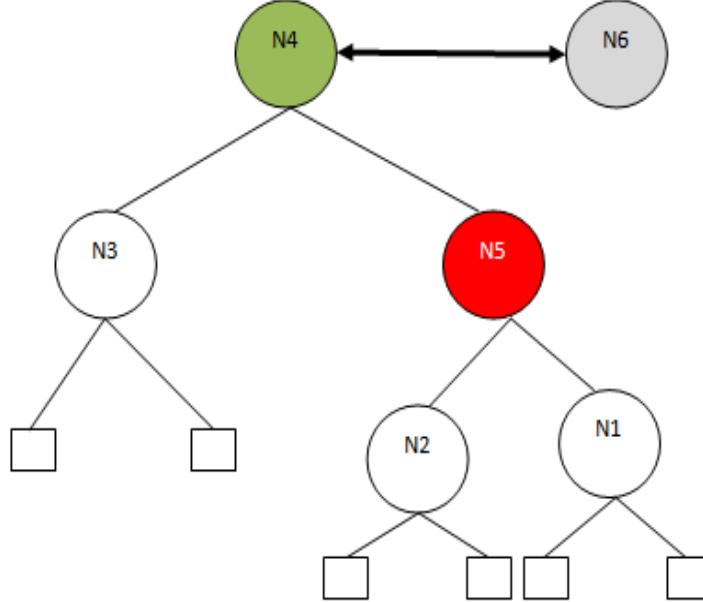
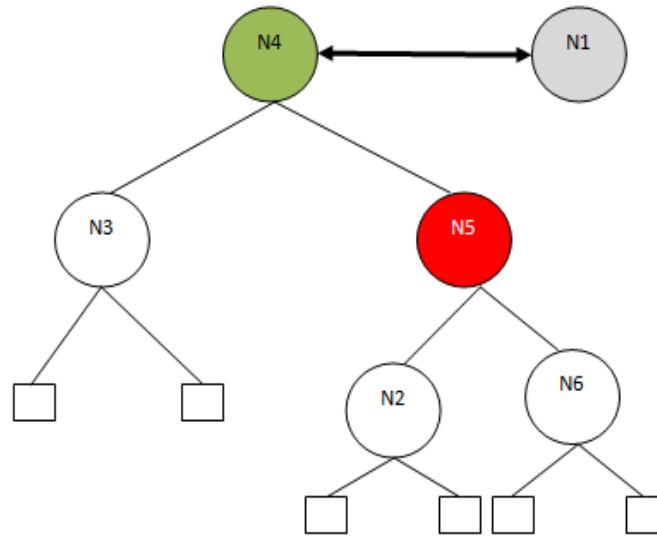


Figure 8: R-B Tree over Link Speed

Figure 9: R-B Tree over Link Speed (N_6 as Priority Node)

3.7. Path Discovery and Route Finalization

The routing tree formulated allows the construction of shortest path between any two nodes at a given time t . The mobility causes the tree to alter after every iteration i.e. routing tree gets updated after every time slot Δt . A new connectivity table is formed during route discovery and this might not be similar to that formed during transmission. Thus, the protocol should be reactive to handle sudden changes. This means that tree needs to be re-computed each time a node transmits the data. The overall complexity of the transmission process is defined as the product of nodes in the formation of route tree and the number of iterations for updating process required at time t for successful transmission i.e.

Figure 10: R-B Tree over Link Speed (N_1 as Priority Node)

$$\mathcal{K} \log(X)$$

where X is the number of nodes and \mathcal{K} is the number of iterations.

3.7.1. Route Discovery Estimation

Route discovery is carried out using pre-existing network operations termed as flooding. The nodes colored in red perform flooding. Flooding allows the identification of connected and active nodes, and also allows the estimation of node connectivity time and evaluates the formation of a randomized graph for transmission. Flooding cannot be performed arbitrarily, rather it is a systematic process as randomized flooding can cause initial network congestion and halt the network operations. The flooding estimations can be carried out using Gauss-Markov model [50] with its integration with Bayesian Kalman Filter [51]. It allows estimation of route discovery time and also defines the complexity of this process. Thus, route discovery time can be defined as the minimum time in which each edge is identified in the network that connects any two nodes or it is the minimum time to form a routing table comprising all active nodes.

Let n be the number of ground nodes, m be the number of aerial nodes, k be the number of neurons in cooperative network layer, t' be the time to exchange beacons between two nodes and t'' be the time to update the routing table

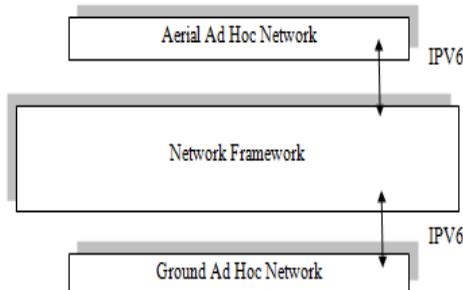


Figure 11: Abstract Virtual Overview : Addressing Schema

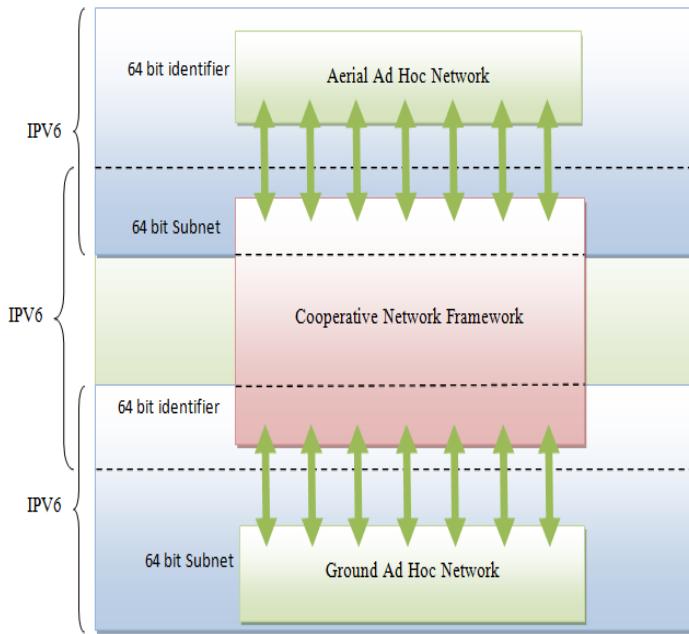


Figure 12: Structural View: Addressing Schema

such that:

$$t_e = (t' + t''), \quad (1)$$

where t_e is the total time for initial route formation. In the worst case, considering that each node exchange beacons with every other node, number of beacons for ground nodes is given as:

$$\frac{n(n-1)}{2}, \quad (2)$$

and for aerial nodes

$$\frac{m(m-1)}{2}. \quad (3)$$

Both the ground and the aerial network operate in parallel to each other. Now, route discovery time for ground nodes t_g and for aerial node t_a are given as:

$$t_g = t_e \frac{n(n-1)}{2}, \quad (4)$$

and

$$t_a = t_e \frac{m(m-1)}{2}, \quad (5)$$

respectively. Therefore, minimum route discovery time D_{min} for parallel operating network will be given as:

$$D_{min} = \max(t_g, t_a), \quad (6)$$

D_{min} is also the initial timings to be set for protocol's route discovery operations. The overall network route discovery time depends on the number of connections identified, the number of aerial and ground nodes and the number of intermediate neurons operating as an interface on the cooperative layer. The route on the cooperative layer depends

on the number of triggers from either of the network. For top-down transmission, the total time to formulate the route is computed as:

$$t_{t-d} = \max(t_a + t_c, t_g), \quad (7)$$

and for bottom-up transmission, it is computed as:

$$t_{b-u} = \max(t_a, t_g + t_c), \quad (8)$$

where,

$$t_c = k_c \times Lg \times t_{lg}. \quad (9)$$

Here, k_c is the number of connected neurons, Lg is the number of logs maintained at cooperative layer and t_{lg} is the time to maintain logs.

3.8. Corridor Identification

Routing protocol aims at establishing multipath hierarchical connectivity between different networks and then identifies the possible network corridors between the networks for supporting heterogeneous relaying. From the formation of routing tree and route discovery estimations, it can be concluded that multi-UAV guided network is purely heterogeneous and form extremely random graphs for path selection. Similarly, network corridors may vary as follows:

- i. Fully Connected Network Corridors: Such network provides more node to node connectivity. Let M and N be the number of aerial and ground nodes, respectively. Such connectivity is formed if there exists $M_{aerial} > \frac{M}{2}$ and $N_{ground} > \frac{N}{2}$. M_{aerial} and N_{ground} are the nodes that can interact with cooperative framework, also such connectivity can exist when $T_{R_n} > \frac{M_x}{2}$ and $T_{R_n} > \frac{N_x}{2}$, where $M_x \leq M$, $N_x \leq N$ and T_{R_n} are the number of trees with no repetition of the source, the sink and the priority nodes. Such orientation is possible if more than $\frac{M}{2}$ and $\frac{N}{2}$ nodes in the network have a similar location in the priority order.
- ii. Sparsely Connected Network Corridors: It includes a network failing to the conditions defined for the fully connected network, but with more than one node on each network to form interaction with cooperative layer or possibility to form more than one routing tree for similar source and sink, i.e. $T_{R_n} > 1$, $T_{R_n} \leq \frac{M}{2}$ and $T_{R_n} \leq \frac{N}{2}$.
- iii. Randomized Connected Network Corridors: If any one of the network corridors does not satisfy the property of being fully connected or sparsely connected with the routing tree being altered after every update, then, such network corridors are considered randomized i.e. either $T_{R_n} \leq \frac{M}{2}$ and $T_{R_n} > \frac{N}{2}$ or $T_{R_n} > \frac{M}{2}$ and $T_{R_n} \leq \frac{N}{2}$.

The categorization allows identification of all possible network corridors that can be used for relaying.

3.8.1. Relaying Gateway Selection

Relaying Gateway Selection (RGS) operates in two parts. The first one is to select the best configuration for network connectivity if more than one network corridors are available for routing. The selection is based upon the maximum number of channels supported by the corridors. But, it is to be taken care that the number of channel utilization refers to distribution of data which requires segmentation and re-assembly that need more number of computations. Therefore, selected channel are the one offering maximum achievable rate.

Let $C_1, C_2, C_3, \dots, C_s$ be the s number of channels available. Thus, a corridor is required that utilizes a less number of channels, but transfers at maximum possible rate. Let $NC_1, NC_2, NC_3, \dots, NC_{\frac{s}{2}}$ be the network corridors such that each network corridor supports $\frac{s}{2}$ channels, i.e.,

NC_1 contains $\{C_1, C_2, C_3, \dots, C_{\frac{s}{2}}\}$ channels,
 NC_2 contains $\{C_1, C_2, C_3, \dots, C_{\frac{s}{2}}\}$ channels,

$NC_{\frac{s}{2}}$ contains $\{C_1, C_2, C_3, \dots, C_{\frac{s}{2}}\}$ channels.

Let channel C offers transmission rate R_t such that,

$$C_1 \longrightarrow R_{t,1}$$

$$C_2 \longrightarrow R_{t,2}$$

.

.

$$C_{\frac{s}{2}} \longrightarrow R_{t,\frac{s}{2}}$$

Channel rate is selected on the basis of maximum value for each network corridor gateway (γ^t) at time t such that:

$$\gamma_{NC_1}^t = \max \{(C_1 R_{t,1}), (C_2 R_{t,2}), \dots, (C_{\frac{s}{2}} R_{t,\frac{s}{2}})\}$$

$$\gamma_{NC_2}^t = \max \{(C_1 R_{t,1}), (C_2 R_{t,2}), \dots, (C_{\frac{s}{2}} R_{t,\frac{s}{2}})\}$$

.

.

.

$$\gamma_{NC_{\frac{s}{2}}}^t = \max \{(C_1 R_{t,1}), (C_2 R_{t,2}), \dots, (C_{\frac{s}{2}} R_{t,\frac{s}{2}})\}$$

Thus, at time t , network corridor selected as gateway is given by:

$$\max \left\{ \gamma_{NC_1}^t, \gamma_{NC_2}^t, \dots, \gamma_{NC_{\frac{s}{2}}}^t \right\} \quad (10)$$

The second part operates in the selection of an inner log tree. RGS also maintains the record of various logs at the cooperative layer. It indirectly controls the connectivity of two networks by identifying the log-neurons that are part of the routing tree. The advantage of using an inner log tree is that it provides inner temporary save locations/units for routing in the defined time interval.

3.9. Collaborative Data Forwarding

For an effective networking, the self included parameter is hierarchy of the network, but to have multiple topology-map formations and head to head decision analysis, it is mandatory that protocol should be capable enough to collaboratively transfer the data. In collaborative mode, number of network corridors can be more than one and uses the maximum rate channel for transfer i.e.

```
if colloborative_mode=True
    G_s={NC_1, ..., NC_{\frac{s}{2}}}
```

where

$$NC_i = \max \{C_1, \dots, C_{\frac{s}{2}}\}, 1 \leq i \leq \frac{s}{2}. \quad (11)$$

This allows multiple channels of different corridors to be utilized for transmission, thus, lowering the chance of bit overlapping as well as interference. When multiple corridors are operating simultaneously, it is required to decide the computational unit and mutual agreement principle that drives the collaborative data forwarding. This is where log-neurons play an active part as they handle multiple network corridors with both aerial and ground network by using the collaborative data forwarding algorithm 1.

```

Require: routing_tree(Initialization)
Ensure: routing_tree = True
if collaborative_mode = True then
    corridor(priority)
     $M_r \Leftarrow \max\{C_1, C_2, C_3, \dots, C_{\frac{s}{2}}\}$ 
     $C_M \Leftarrow \max\{NC_1, NC_2, \dots, NC_{\frac{s}{2}}\}$ 
    sort  $C_M$ (priority)
    while search_map = True &  $R_s \neq NC$  do
        construct RB_priority Tree( $N, M, K$ )
        maintian cognitive_log( $N$ )
        maintian cognitive_log( $M$ )
        map(cognitive_log( $N$ ), cognitive_log( $M$ ))
        route(map)
        transfer(map)
        if ack = true & ( $C_g = N$  &  $C_a = M$ ) then
            reset
        else
            continue : route(map)
        end if
         $R_s \Leftarrow R_s + 1$ 
    end while
    reset_tree  $\Leftarrow$  update(tree)  $\Leftarrow$  route(tree)
else
    Continue : single_mode
end if

```

Algorithm 1: Collaborative Data Forwarding

3.10. Network Leaving and Joining Rate Prediction

The pace at which the nodes join or leave a network is defined as network leaving or joining rate. The number of nodes joining or leaving the network affects the routing tree formation. For a network to sustain, there is always a certain threshold value that gives the minimum number of nodes which should be present in the network to form a guided system. For a distributed parallel operating network, leaving or joining rate at time t' is computed as

$$R_{LJ} = \frac{N(t') - N(t)}{(t' - t)}, \text{ for } t' > t \begin{cases} \text{Join,} & \text{if } R_{LJ} > 1 \\ \text{Leave,} & \text{if } R_{LJ} < 1 \\ \text{Stable,} & \text{if } R_{LJ} = 0 \end{cases} \quad (12)$$

where R_{LJ} denotes the network leaving or joining rate, $N(t)$ denotes number of nodes operating in network at time t . Thus, connectivity at time t can be determined in terms of network leaving and joining rate. If a network protocol is able to predict this rate, it can perform better and can provide enhanced connectivity. Further, P_L and P_J are the probability of leaving and joining a network, respectively, defined over P_C i.e. probability of connectivity. Initially, when the network is in a stable state, P_C is 1. Also,

$$P_C = P_L + P_J, \quad (13)$$

A time span ΔT between any two arbitrary locations during network progression is considered for predicting the leaving and joining rate, as shown in Figure 13. This time can be adjusted depending on the network operational time, network up-time and number of nodes operating in the network.

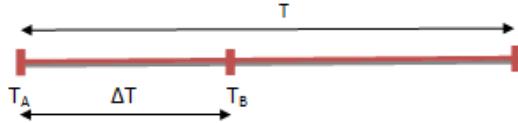


Figure 13: Network Operational Time

Number of nodes joining or leaving the network (N_{LJ}) in this time slot is computed as:

$$N_{LJ} = (N_{T_B} - N_{T_A}) \begin{cases} Join, & \text{if } N_{LJ} > 1 \\ Leave, & \text{if } N_{LJ} < 1 \\ Stable, & \text{if } N_{LJ} = 0 \end{cases} \quad (14)$$

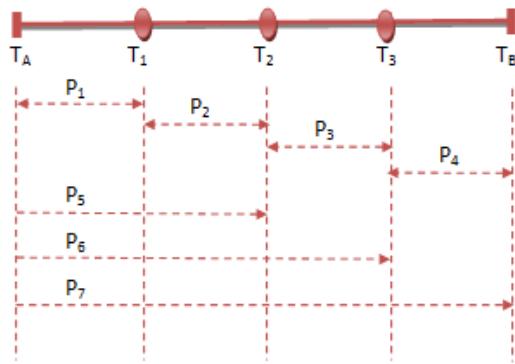


Figure 14: Probability Identification Chart

Now,

$$P_C = \frac{N_{LJ}}{N_{\Delta T}}, \quad (15)$$

where $N_{\Delta T}$ is the number of nodes in time slot ΔT . The time ΔT is further sub-divided into four probabilistic divisions, P_1-P_4 , as shown in Figure 14. Two different approaches are considered for probability chart formation.

i. Divisional Approach

$$\begin{aligned} T_A T_1 &\leftarrow P_1 \\ T_1 T_2 &\leftarrow P_2 \\ T_2 T_3 &\leftarrow P_3 \\ T_3 T_B &\leftarrow P_4 \end{aligned}$$

ii. Progressive Approach

$$\begin{aligned} T_A T_1 &\leftarrow P_1 \\ T_A T_2 &\leftarrow P_5 \\ T_A T_3 &\leftarrow P_6 \\ T_A T_B &\leftarrow P_7 \end{aligned}$$

$P_1 - P_7$ can be evaluated using equation (15) and equation (14). From the system classification shown above, it is possible to perform network prediction using computed probabilities. The model computes the estimated number of nodes that may join or leave the network, thus, giving a certain threshold value that decides the number of active links required to form a guidance system.

3.10.1. Probabilistic Tree Formation

Probabilistic tree formation allows prediction of nodes and time slots that operates in crucial mode in the network, i.e. it identifies the nodes that cause maximum and minimum alterations in a routing tree during network operations. It also provides the support for maintenance of logs for network nodes. For a network which operates at a time slot t , average rate $\text{Avg}_{\text{rate}}(g)$ and $\text{Avg}_{\text{rate}}(a)$ for ground and aerial network are computed as:

$$\text{Avg}_{\text{rate}}(g) = \sum_{i=1}^N x_i - N_{\text{ini}}, \quad (16)$$

$$\text{Avg}_{\text{rate}}(a) = \sum_{i=1}^M x_i - M_{\text{ini}}, \quad (17)$$

where N_{ini} and M_{ini} are the initial number of nodes in ground ad hoc and aerial ad hoc networks, respectively. Using the available time slots, Time-Probability tree is constructed that labels the probabilities and the number of nodes joined or left during particular time slot, as shown in Figure 15.

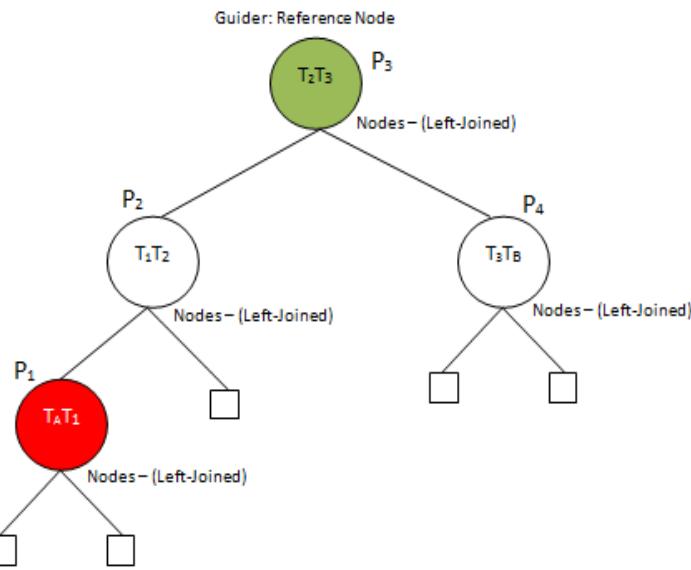


Figure 15: Time-Probability Tree : Divisional Approach

Guider node in Time-Probability tree defines the threshold value for network performance. Probability values greater than the guider value show improved transmission and values lower than the guider value represents degraded network performance. Similarly, a Time-Probability tree can be constructed for a progressive approach as shown in Figure 16.

Using equation (15), a complete-probability tree depending on the probability and priority can be constructed that provides support to analyze the network for its performance by choosing the time slot during which transmission is maximum, as shown in Figure 17. From this figure, it is clear that maximum alterations are shown for probability P_3 i.e. for time slot T_2T_3 . Thus, nodes appearing or disappearing during this time slot can be assigned the highest priority value during route formation despite their ranking in the network. Constructing a Time-Probability tree to identify the nodes left/joined the network at time t allows the prediction of nodes that make a maximum number of shuffles

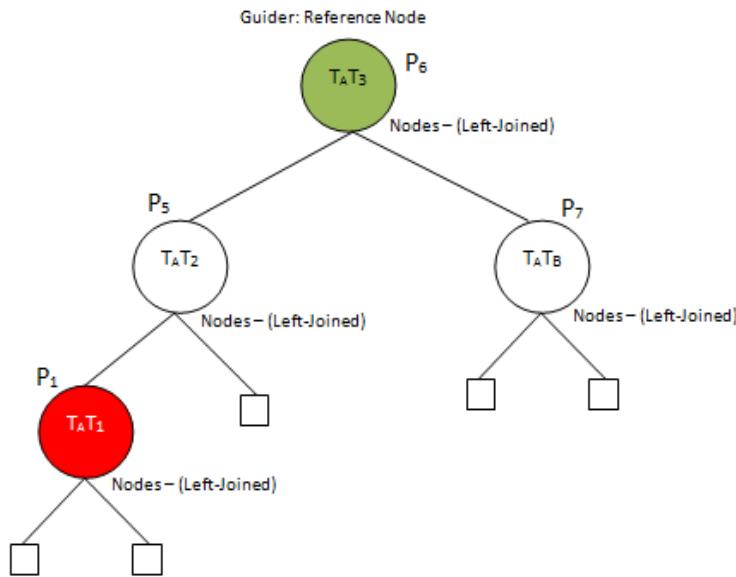
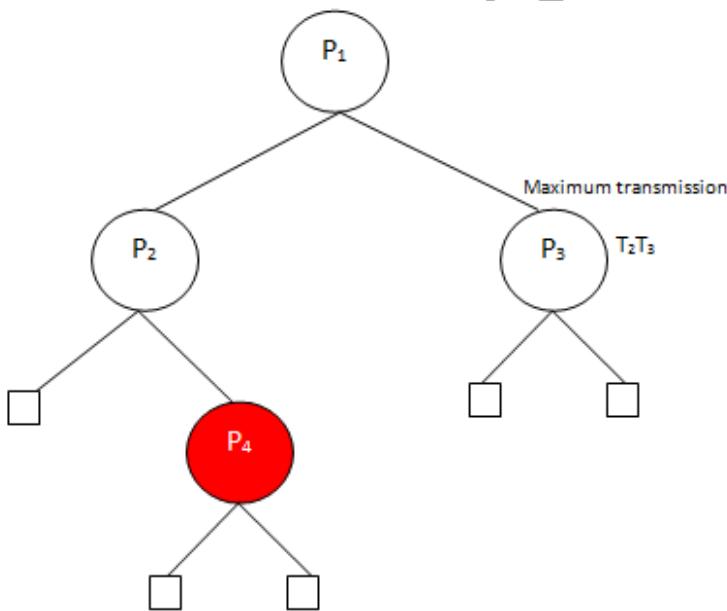


Figure 16: Time-Probability Tree : Progressive Approach

Figure 17: Probability Decision Tree: $P_3 > P_1 > P_4 > P_2$

during transmission. More the number of appearances more is the probability for a node to leave or join. Thus, from prediction, it is evaluated that such nodes should not be considered as a priority node despite their initial ranking in the network as it can adversely affect the network transmission rate. From the probability tree, a minimum number of time span required for the routing protocol to perform updates can also be evaluated. If P_C increases with the time, this means mobility is high and the network is affected by it. Lesser the variation in P_C more stable is the network. It is to be noted that the framework that supports the routing, should have the feature for maintaining the minimum number of nodes in both the network at each time slot to allow the formation of a guided network. The joining and leaving rate of a network defines the spatial dependencies of the network as more logs are to be maintained for more

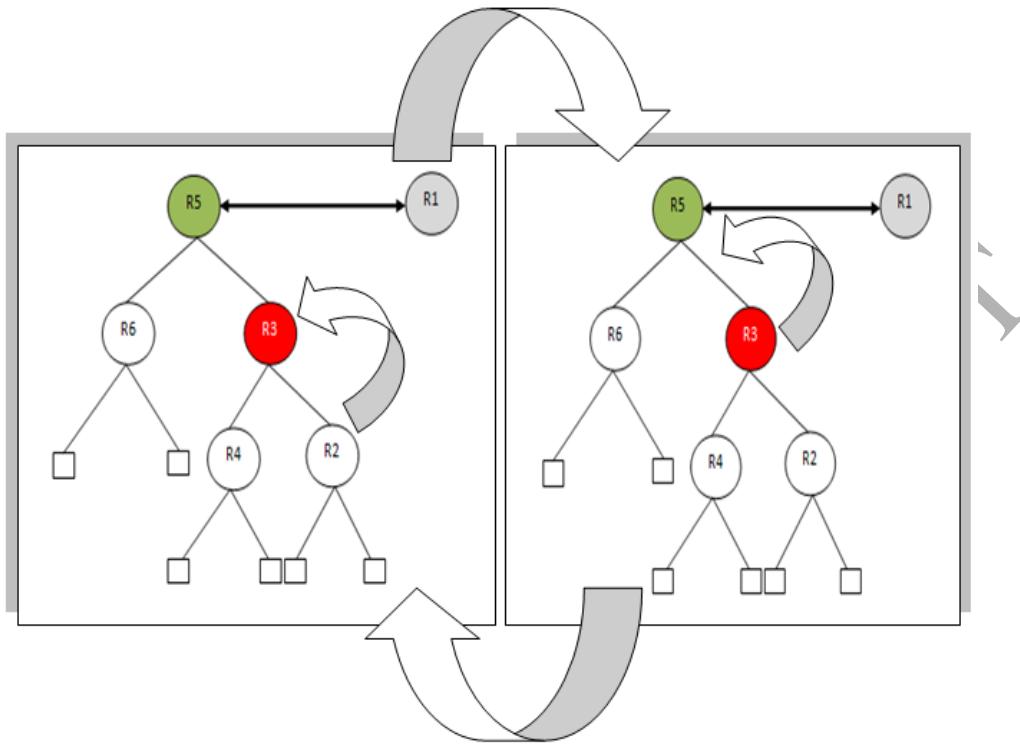


Figure 18: Switching over Handovers

node movement and this requires more memory and computations for their processing.

3.11. Neighbour Identification and Handovers

The proposed routing protocol splits the trees to allow handovers and neighbour identification. R-B tree protocol provides a neighbour identification directly by forming an adjacency matrix from the already created routing tree. The neighbour identification allows seamless handovers with zero-delay switching between the trees. In a guided network consisting of varied mobility ranges, handovers become more complex. For this, pre-existing tree based greedy approach is adopted, i.e. switch if node not found. In this approach, the handovers are confirmed without affecting the number of network corridors. Also, it is to be noted that handovers are already handled using R-B routing tree, however, handovers are allowed only if there is no shortest path between the source and the destination, as shown in Figure 18.

```

if shortest_path=False
    perform switching ← handovers
else
    continue
  
```

4. DPTR: Protocol Functionality

The proposed protocol operates using multiple R-B trees that identify the routing-path for ground and aerial network, respectively. The protocol aims at solving the network partitioning problem and forms a guided network. The protocol is capable of handling multiple transmissions in a distributed environment. It forms a trending curve that helps the protocol to predict the reset conditions, and also analyzes the network performance by identification of time slots during which the network performed to its best. This trending curve is formed using the probability of

connectivity (P_C) computed by using equation (15). The labeling and mark identification of this curve are shown in Figure 19.

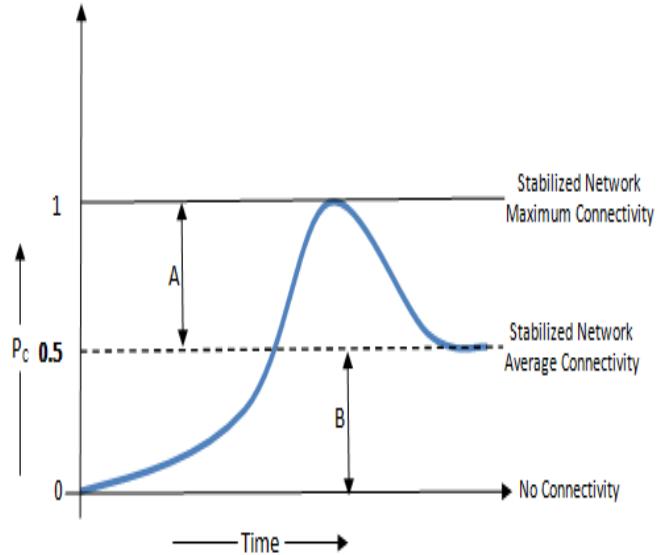


Figure 19: Trending Curve - A: Maximum Path Available, B: Minimum Path Available

4.1. Beacon Configuration and Routing Table

For a network comprising multiple units operating at the same instance, exchange of initial information is of much importance. For distributed routing protocol, class-based beacon configuration is considered, namely, class-A, class-B, class-C as shown in Figure 20. Class-A configuration is used for ground ad hoc network, class-B is used for aerial ad hoc network and class-C defines the configuration messages for the inner neuron layer that supports routing. The parameters and agent values passed using beacons in different class levels of routing protocol are shown in Table 2, Table 3 and Table 4.

PARAMETER	DESCRIPTION
$R_{(min)}$	Rate sustainable to all nodes
$Corridors_{(S)}$	Internet support
$Corridors_{(C)}$	Internet connections
$Max_{(m)}$	Maximum Mobility
$Min_{(m)}$	Minimum Mobility
$Channel_{(S)}$	Number of simultaneous Channels
$Link_{(S)}$	Transmission variations offered
Connectivity	Degree of node
$Priority_{(N)}$	Priority Node selected
$W_{(Link)}$	For proactive approach

Table 2: Beacon - Class A

Routing tables allow dynamic retrieval of useful information required to select the next best hop for data transmission. In the distributed network environment, two types of nodes exist- one is a normal ad hoc node that acts as a transceiver or relaying node and the other is a node that interacts with the nodes from its counterpart network. Thus,

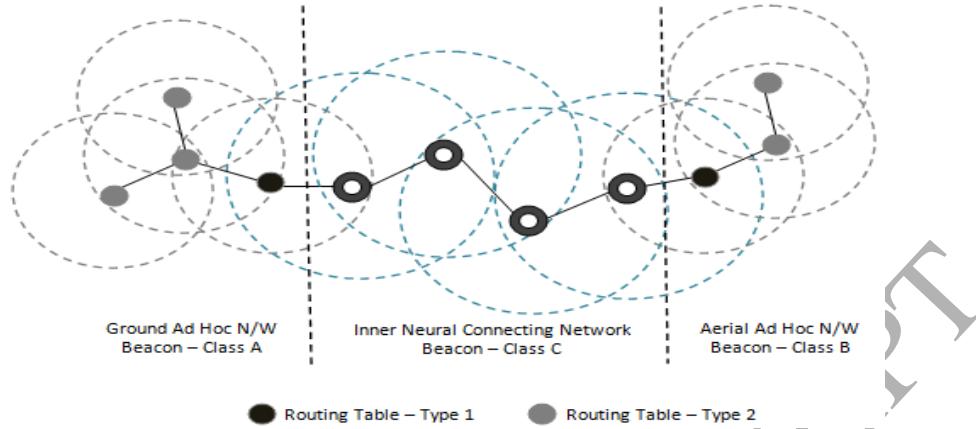


Figure 20: DPTR : Protocol Operations

PARAMETER	DESCRIPTION
Class A	Includes configuration of class A
Range	Node covering distance
Cells	Number of cells traversed
Maps	Number of maps shared

Table 3: Beacon - Class B

PARAMETER	DESCRIPTION
Class A	Includes interfacing for class A
Class B	Includes interfacing for class B
k_n	Number of active neurons
U_t	Network up-time
G_i	Ground interacting node
A_i	Aerial interacting node

Table 4: Beacon - Class C

two types of the routing table are used in UAV guided networks. The proposed DPTR protocol's connectivity and transmission are handled through algorithm 2.

5. Simulations

For analysis of Multi-UAV guided networks, such as FANETs, no direct simulating tool is available. Thus, for analysis of the proposed DPTR protocol, a virtual simulation environment is developed by using NS-2 and Matlab. Two different systems are coded together so as one of them handles simulations related to aerial ad hoc network and other provides support for ground ad hoc network. A routing interface is created comprising neural framework that allows testing and analysis of the proposed protocol. For simulations, a total of 100 runs is performed to track the results over the network operating for 10 slots each of 100 seconds. Number of UAVs considered is 5, 8 and 10 operating over ground ad hoc network consisting of 20, 25 and 30 nodes in an area of 2500x2500 sq.m. Random waypoint mobility model is used for ground nodes and for aerial nodes, existing framework allowed fixed waypoint

```

Require:  $M \geq 3$  &  $N \geq 3$ 
Require:  $Neurons \leftarrow K \geq 3$ 
Ensure:  $Aerial\_Network : A(M) \leftarrow initialized$ 
Ensure:  $Ground\_Network : G(N) \leftarrow initialized$ 
while  $final\_route \neq true$  do
     $fetch\_g \leftarrow node\_data(G)$ 
     $\mathcal{P}_{x_1} \leftarrow max(\mathcal{P}_x[N])$ 
     $P\_Tree(G, \mathcal{P}_{x_1}, Guider) \rightarrow T_1$  /* ground network
     $fetch\_a \leftarrow node\_data(A)$ 
     $\mathcal{P}_{y_1} \leftarrow max(\mathcal{P}_y[M])$ 
     $P\_Tree(A, \mathcal{P}_{y_1}, Guider) \rightarrow T_2$  /* aerial network
     $find(G) \leftarrow priority\_node(G)$ 
     $find(A) \leftarrow priority\_node(A)$ 
     $Re\_initialize \leftarrow neurons\_tree(K)$ 
     $P\_Tree(A, G, K, Guider) \rightarrow T_3$  /* neural tree
    if  $proactive = true$  then
         $identify\_link \leftarrow link\_matrix(weight)$ 
         $update \leftarrow T_3(link)$ 
    else
         $T_3 \leftarrow no\_update$ 
         $N_c \leftarrow corridors$ 
         $set path = true$ 
    end if
     $estimate\_route(N_c)$ 
    if  $collaborative\_mode = true$  then
        Algorithm 1
    else
         $set single\_mode$ 
    end if
     $P_C \leftarrow (N_{LJ}, N_{\Delta T})$ 
     $Tree(P_C) \rightarrow T_4$  /* probability tree
    if  $P_C(node_i) = max \& P_c < 0.5 \& node_i = find(G)$  then
         $reset \leftarrow T_1, T_2$ 
         $priority(find(A), find(G)) \leftarrow node_j \forall j \neq i$ 
    else
         $N_h \leftarrow select next\_hop$ 
    end if
     $flooding \leftarrow beacons(class A, class B, class C)$ 
end while
 $set \rightarrow routing\_table$ 
start transmission

```

Algorithm 2: DPTR : Proposed Connectivity Algorithm

mobility modeling. Neural set created for interfacing comprises 40 to 60 neurons, each denoting a configurable parameter in the network, and supports log maintenance. An exemplary illustration of a network scenario considered for analysis is shown in Figure 21.

Other parameters configured for routing are shown in Table 5. Node arrangement with respect to each other in aerial and ground ad hoc network follows a Poisson distribution for localization. Also, neural connectivity varies similarly to node arrangement and scenario set-up. The distribution of nodes w.r.t. neural framework is shown in Figure 22, Figure 23, and Figure 24.

Node distribution allows identification of nodal connectivity with respect to network connectivity time. It also provides the statistical location of nodes in the network. Statistical modeling of the scenario allows efficient traffic generation and supports estimations regarding node movement during the transmission process. A similar model is adopted for the generation of traffic. A separate GUI is created that provides the R-B guided tree formation, providing the initial guider nodes available in the network according to the configured scenario. Guider nodes are generated for all the three types of nodes operating in the network- aerial nodes, ground nodes, and neural nodes. Neural mapping is

PARAMETER	VALUE	DESCRIPTION
Area	2500x2500 sq. m	Area under observation
Ground Nodes	20,25,30	Node sets for evaluation
UAVs	3,5,8	UAVs for Cooperative network
Traffic Type	CBR/VBR	Traffic over TCP
Packet Size	1024 bytes	Average Packet Size
Propagation Radio Model(Ground)	Two Ray Ground	Topological Propagation Model
Propagation Radio Model(Aerial)	Line of Sight Based	Topological Propagation Model
Mobility Model(Ground)	Random waypoint	Node movement model
Mobility Model(Aerial)	Fixed waypoint	Node movement model
Maximal Speed(Ground)	25m/s	Node mobility
Minimal Speed(Ground)	10m/s	Node mobility
Maximal Speed(Aerial)	35m/s	Node mobility
Minimal Speed(Aerial)	20m/s	Node mobility
Pause Time	2.0s	Halt time for Node
Interval Time To send	2 p/s	Time to wait before sending
Transmission Range	100-500 m	Transmission Distance of Node
Ground Node Mobility Range	0-5.0 m	Node Steps during movement
Simulation Time	100x10s	10 slots each of 100 seconds
Configurable Parameters	2	Neural controlled parameters
Seeds(Aerial)	1,3,5	Source at time instance t
Seeds(Ground)	5,8,10	Source at time instance t
Minimum Rate	128	Minimum Transmitting rate
Neural Set	40,50,60	Neurons for interfacing
Network Corridors	3	Inter-network connections
Maximum Traffic	50	map size for transmission
Simulation Runs	100	Number of analysis runs

Table 5: Parameter Configuration

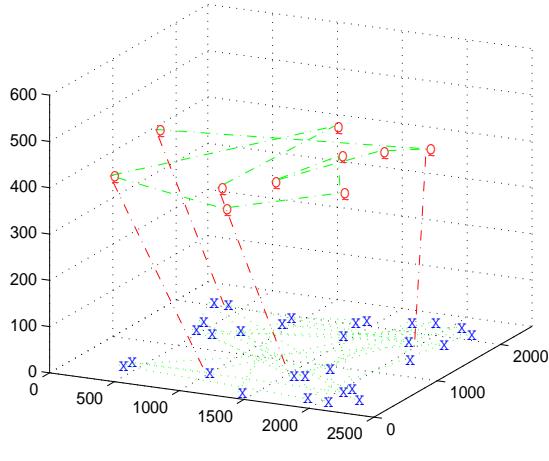


Figure 21: 10 UAVs, 30 Ground Nodes

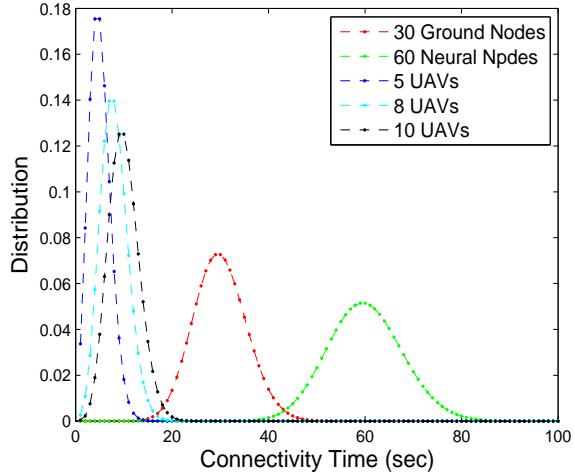


Figure 22: Network Distribution with 30 Ground Nodes, 60 Neurons

performed on the same system that handles aerial and ground ad hoc network. A dedicated neural layer application is used so as to create a virtual network layer that can handle the functionality of the proposed protocol. The simulation results are mapped to log files maintained by red-nodes of the routing protocol. These log files are then traced to analyze the performance of the proposed routing protocol.

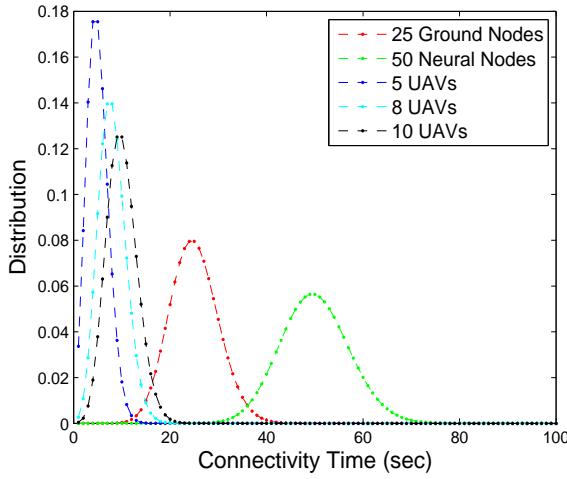


Figure 23: Network Distribution with 25 Ground Nodes, 50 Neurons

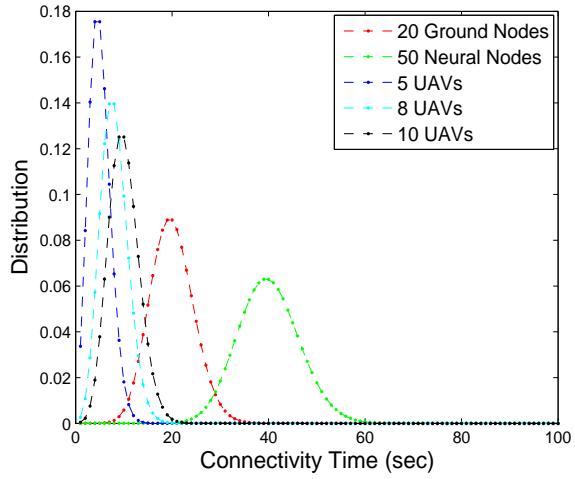


Figure 24: Network Distribution with 20 Ground Nodes, 40 Neurons

6. Results Analysis and Discussions

The proposed DPTR protocol is evaluated by comparing it to its own multiple runs with different scenarios as well as with other existing protocols. An efficient protocol allows higher packet delivery ratio and causes a lower end to end delays during traffic forwarding. For analysis, 3-mode evaluation is performed. The first evaluation is carried for aerial ad hoc network, second for ground ad hoc network and third for the overall guided network. The performance of the proposed distributed tree protocol is evaluated over following metrics:

- i. Packet delivery ratio
- ii. End to end delays
- iii. Channel Utilization
- iv. Throughput
- v. Network connectivity time
- vi. Probability of connectivity

6.1. Packet Delivery Ratio(PDR)

Higher packet delivery ratio directly identifies the efficiency of the protocol and its capability to deliver the data. Maps comprising cognitive data are transferred over the network and PDR is evaluated over successful receiving of maps. Segmentation, re-assembly, and forwarding of maps are performed by red nodes of the network. Figure 25 shows the PDR over an aerial ad hoc network with inputs varying from 5 to 10. These inputs are the initial corridor connectivity carried out by the ground network during ground to aerial transmissions. Similarly, ground ad hoc network is evaluated from 1 to 5 corridor connectivity, as shown in Figure 26.

For overall network analysis, all the corridors are used for evaluation, i.e. total inputs ranging from 6 to 15 as shown in Figure 27. The proposed protocol offers high packet delivery ratio and is capable of sustaining these higher rates even when the network is scaled to a large number of aerial and ground nodes. Thus, cognitive map transfers are provided by the protocol with zero drops during transmission.

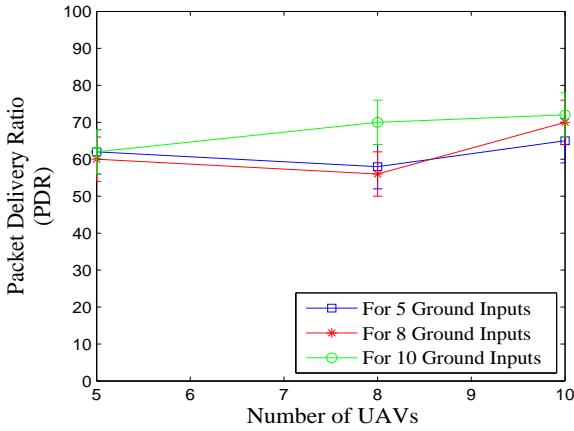


Figure 25: Packet Delivery Ratio - UAVs ad hoc network

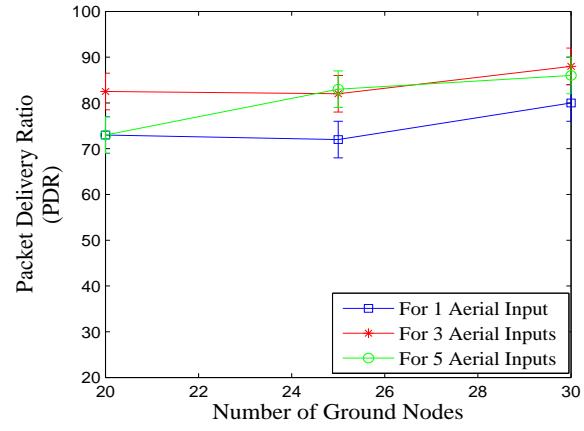


Figure 26: Packet Delivery Ratio - Ground ad hoc network

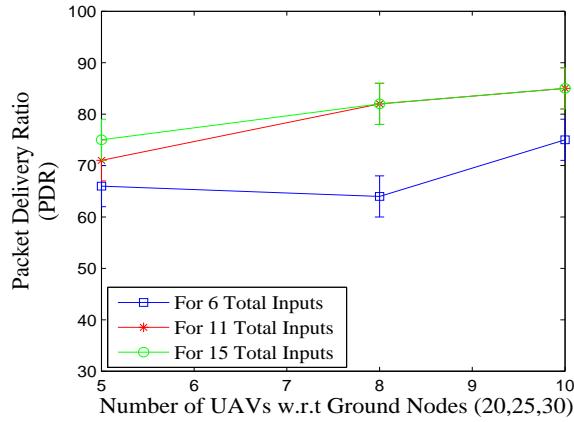


Figure 27: Packet Delivery Ratio - Complete guided ad hoc network

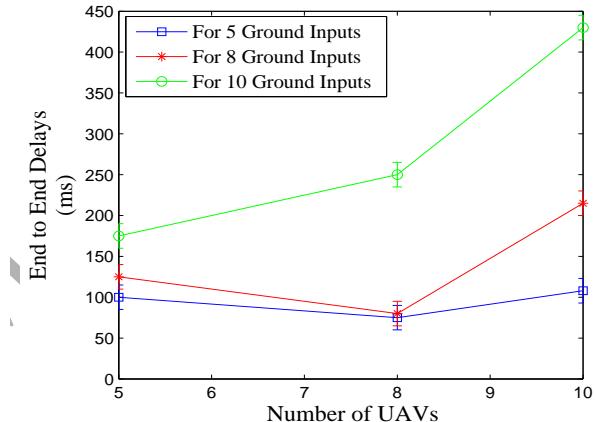


Figure 28: End To End Delays - UAVs ad hoc network

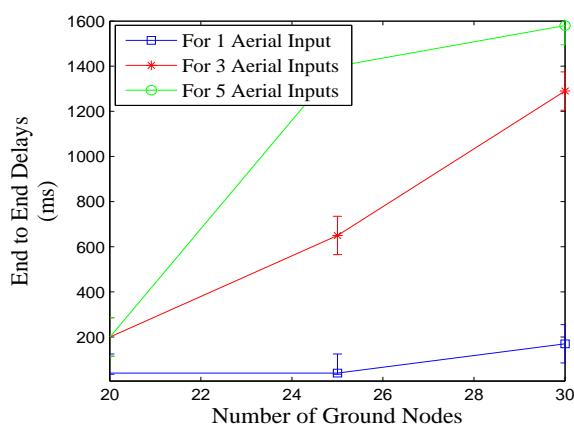


Figure 29: End To End Delays - Ground ad hoc network

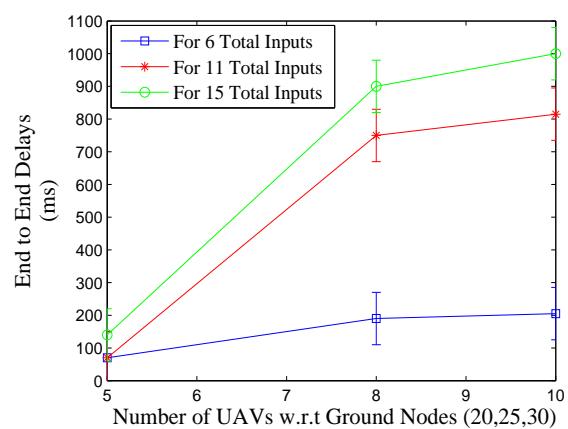


Figure 30: End To End Delays -Complete guided ad hoc network

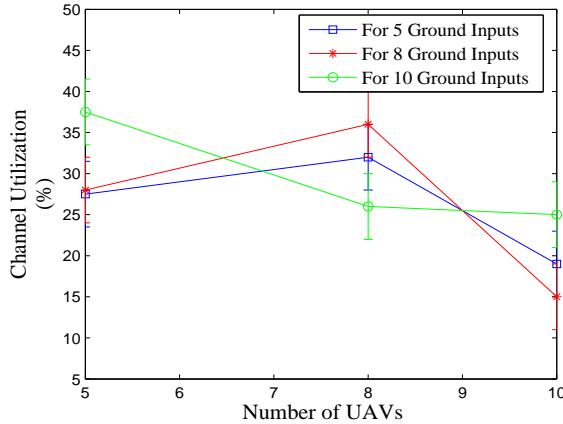


Figure 31: Channel Utilization - UAVs ad hoc network

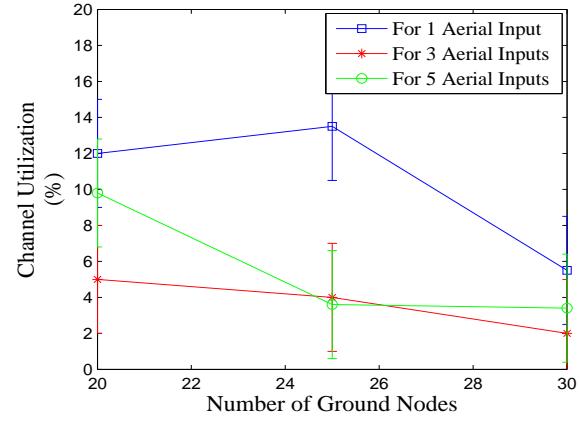


Figure 32: Channel Utilization - Ground ad hoc network

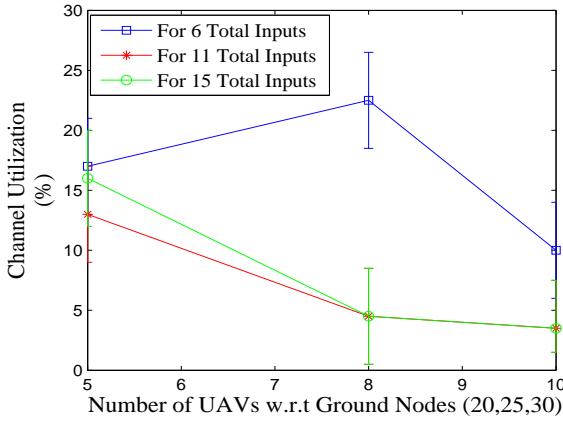


Figure 33: Channel Utilization - Complete guided ad hoc network

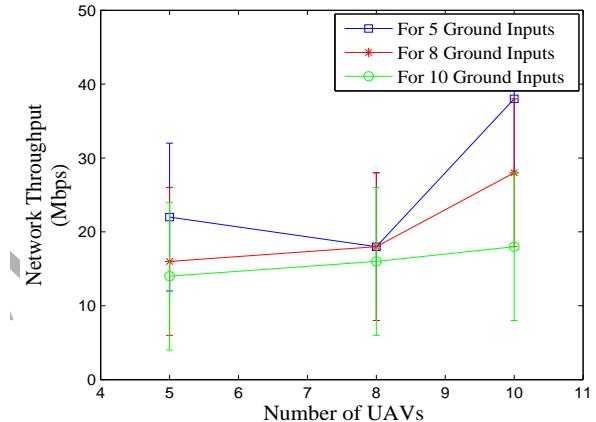


Figure 34: Throughput - UAVs ad hoc network

6.2. End To End Delays

A network operating in two modes of ad hoc network is bound to affect from end to end delays. However, delays should not be beyond a certain limit so as to allow proper and continuous transmission of data. Delays in the multi-guided network would allow incorrect map formation and may cause failure in the network leading to zero maps transfers. Figure 28, Figure 29, and Figure 30 represent the end to end delays recorded for aerial ad hoc network, ground ad hoc network and overall guided network, respectively. The proposed DPTR protocol is capable of maintaining delays at certain threshold values. The end to end delay in the network is negligible and does not affect the transmission.

6.3. Channel Utilization

Channel utilization is the ratio of the connected channels to the number of channels provided in the network. Higher percentage of channel utilization shows large traffic transfer with high bandwidth offered in the network. Most of the ad hoc networks operate using single channel over each node, but with the provision of multiple corridor support, multi-channel transmission is possible using distributed routing protocol. The channel utilization variations in the network are shown in Figure 31, Figure 32, and Figure 33.

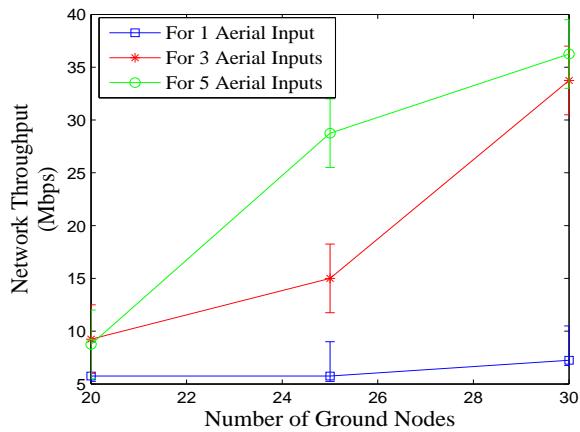


Figure 35: Throughput - Ground ad hoc network

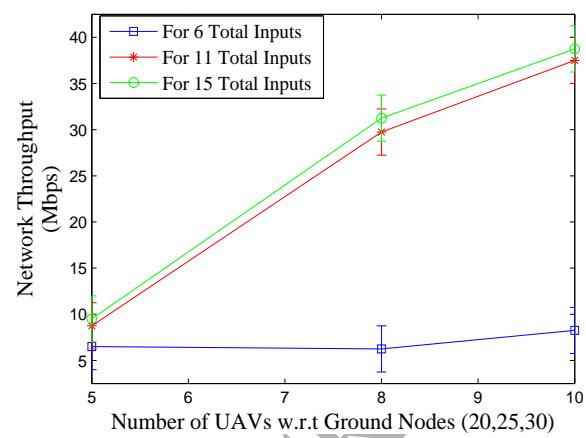


Figure 36: Throughput - Complete guided ad hoc network

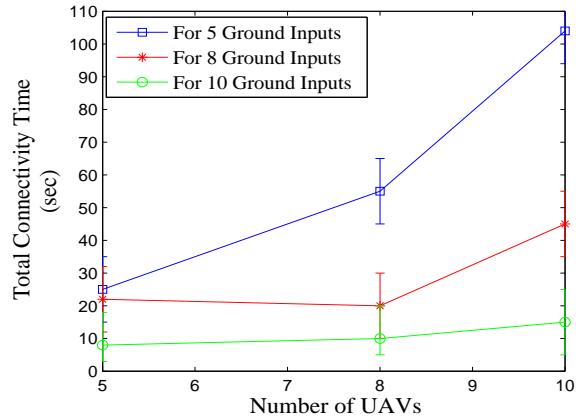


Figure 37: Total Connectivity Time - UAVs ad hoc network

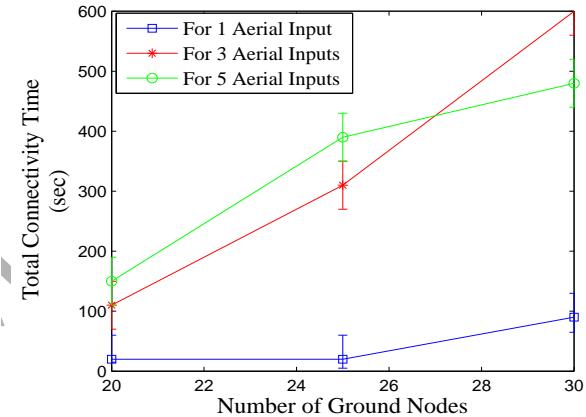


Figure 38: Total Connectivity Time - Ground ad hoc network

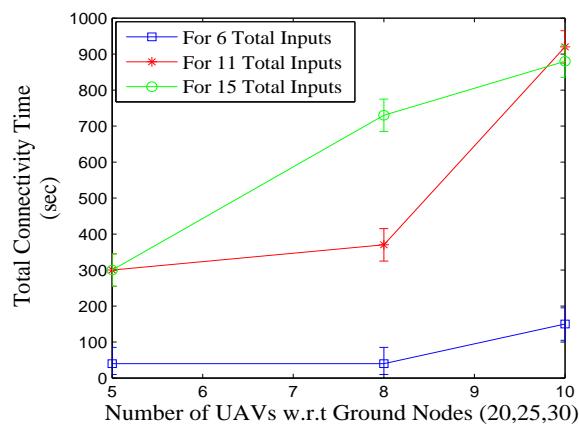


Figure 39: Total Connectivity Time - Complete guided ad hoc network

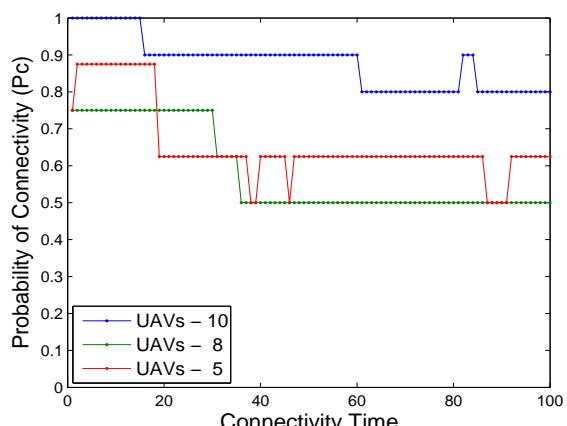


Figure 40: Probability of Connectivity

6.4. Throughput and Network Connectivity Time

Analysis of DPTR protocol is carried to analyze the offered network throughput and total connectivity time. Throughput allows analysis of traffic transmission rate. DPTR provided high throughput during the transmission process with a varied range of provision for data forwarding. The major reason for higher throughput values is support of multiple channels and multiple corridors by the protocol that provided simultaneous transmission of large data. Figure 34, Figure 35, and Figure 36 shows the throughput evaluation of the network structure. Another evaluation metric considered is the total network connectivity time. Higher the connectivity time, more reliable is the network. DPTR offered higher scalability in terms of total connectivity time. By increasing the number of nodes, total connectivity time also increases that provides an indirect support for the protocol to offer higher packet delivery ratio with a controlled end to end delays. Total connectivity time analysis of DPTR are shown in Figure 37, Figure 38, and Figure 39.

6.5. Probability of Connectivity (P_c)

A network can sustain till it maintains the minimum number of nodes in the network for efficient transmission. This has already been diagrammatically represented in Figure 19. For a complete analysis of DPTR regarding its connectivity, P_c is plotted during the complete transmission process. The probability of connectivity can be computed either for a particular time span during transmission or for the overall network operation time. Such evaluations allow prediction of network performance at any instant of time. For proposed protocol, the probability of connectivity is evaluated w.r.t. total connectivity time. Evaluation plot represents that the proposed protocol offers P_c greater than equal to 0.5 which means that it allows stabilized network formation with the higher transmission rate, as shown in Figure 40.

Protocol	Abbreviated For
AODV	Ad Hoc On Demand Distance Vector Routing [52]
DSDV	Destination Sequenced Distance Vector Routing [53]
DSR	Dynamic Source Routing [54]
MDART	Multipath Dynamic Address Routing [55]
OLSR	Optimized Link State Routing [56]

Table 6: Protocols Considered For Comparison With DPTR

6.6. Comparison With Existing Routing Protocols

The proposed DPTR protocol is tested for its effectiveness and efficiency against some of the existing routing protocols. The routing protocols considered are presented in Table 6. The number of aerial vehicles considered for comparison varies between 1 and 10 and maneuvers with a similar configuration as considered for stand-alone simulation of DPTR in Section 5 (Simulations). Protocols are compared for channel utilization, packet delivery ratio, network throughput, total connectivity time, transmission delays and the probability of connectivity. Percentage channel utilization for DPTR is higher than the existing routing protocols during aerial network formation. Also, the proposed DPTR protocol offers better corridor connectivity support, as shown in Figure 41.

Data efficiency of the protocols is analyzed in form of PDR. The comparative plot between the existing and the proposed DPTR protocol presents much improvement as shown in Figure 42. The protocols are also compared for network throughput, total connectivity time, and transmission delays as shown in Figure 43, Figure 44, and Figure 45, respectively. Results in these graphs show that the proposed DPTR protocol offers high network throughput and high connectivity time with lower transmission delays. Further, protocol functionality test for the probability of connectivity (Section 4-Figure 19) is performed to compare the existing protocols with proposed DPTR protocol. Analysis, as shown in Figure 46, proves that none of the existing routing protocols (AODV, MDART, OLSR, DSR, DSDV) is able to sustain the aerial network connectivity for a longer duration, whereas the proposed DPTR offers more stabilized connectivity for a longer duration.

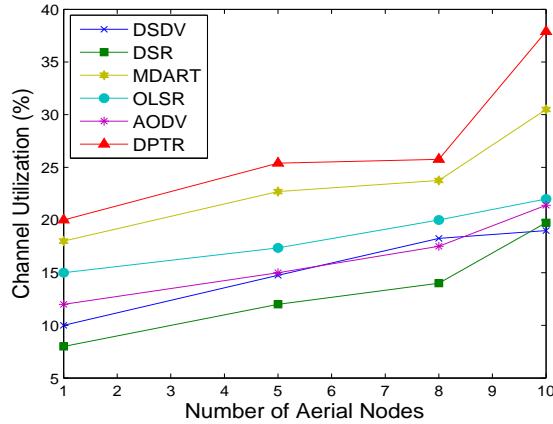


Figure 41: Comparison: Channel Utilization

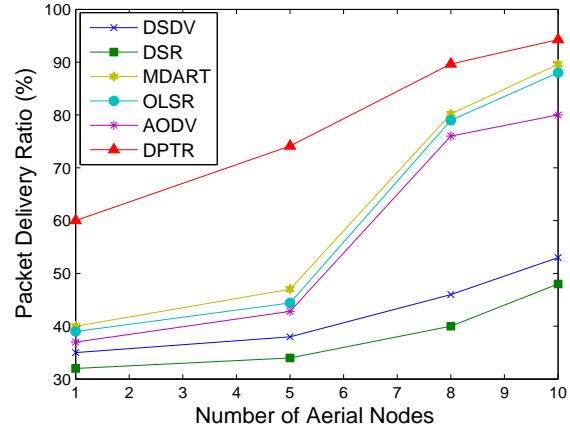


Figure 42: Comparison: Packet Delivery Ratio

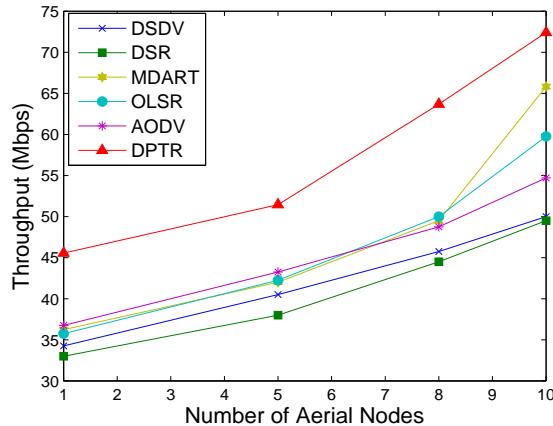


Figure 43: Comparison: Network Throughput

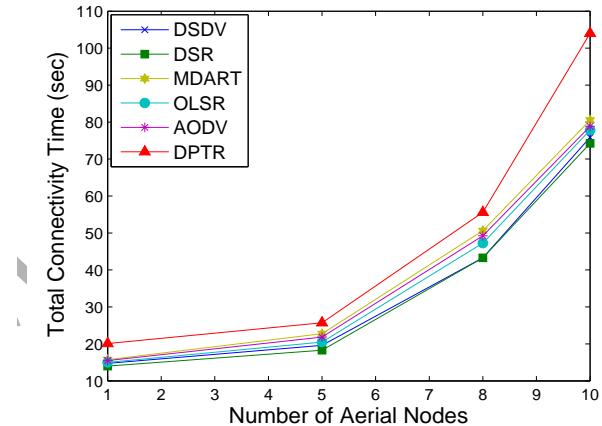


Figure 44: Comparison: Total Connectivity Time

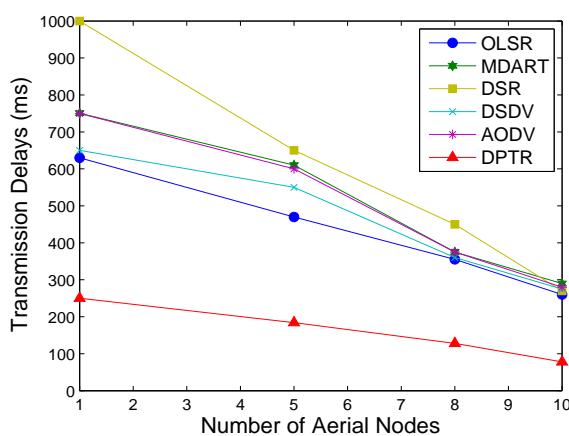


Figure 45: Comparison: Transmission Delays

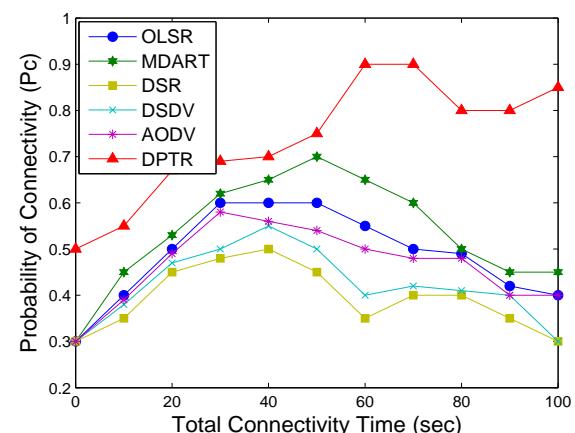


Figure 46: Comparison: Probability of Connectivity

Table 7: Comparison with state-of-the-art solutions for network partitioning and routing.

Approach	Author(s)	Technique	Network Partitioning	Tree-based	Routing	Network	Channel Utilization	Transfer Rate	Analysis	Mutual Network	Overheads	Delays	Throughput	UAVs Support
Distributed recovery	Akkaya et al. [13]	Controlled mobility	Yes	Yes	No	Mobile sensors	-	High	Simulation	No	Low	Low	-	No
Distributed prevention mechanism	Wang and Wu [14]	Critical node detection	Yes	No	No	WSNs	-	High	Simulation	No	-	-	-	No
Optimized repairing	Lalouani et al. [15]	Boundary aware optimization	Yes	Yes	No	WSNs	-	-	Simulation	No	-	-	-	No
SMPC	Sajid et al. [40]	Singular division of multipath power control tree based routing protocol	No	Yes	Yes	Underwater WSNs	-	Medium	Simulation	No	-	Low	-	No
Optimal tree routing	Soham et al. [41]	Tree-based routing with particle swarm optimization	No	Yes	Yes	WSNs	-	-	Simulation	No	Medium	Medium	Medium	No
ETSW	Liang et al. [42]	Encounter history-based routing	No	Yes	Yes	Opportunistic	-	High	Simulation	No	Low	Low	Low	No
Self-organizing protocol	Qiu et al. [43]	Tree-based organization of networked and non-networked nodes	No	Yes	Yes	IoT	-	High	Simulation	No	-	-	High	No
Self-organized protocol	Han et al. [44]	Tree-based minimal energy consumption	No	Yes	Yes	WSNs	-	-	Simulation	No	-	Low	-	No
Inter-UA collaboration	You et al. [57]	Genome-based networking	No	No	Yes	UAVs	Yes	High	Simulation/ Experimental	Yes	Medium	Low	High	Yes
Video transmission	Rosrio et al. [58]	Opportunistic routing	No	No	Yes	FANETs	No	High	Simulation	No	-	Low	-	Yes
Data Dissemination	Sharma et al. [59]	Energy efficient approach	No	No	Yes	UAVs-WSNs	No	High	Simulation	Yes	Medium	Low	High	Yes
Proposed DPTR	sharma et al.	R-B tree based routing	Yes	Yes	Yes	FANETs	Yes	High	Simulation	Yes	Low	Low	High	Yes

From analysis, it is evident that the proposed DPTR protocol is able to sustain the dynamics of hybrid network formation between the aerial nodes and the ground nodes. Also, more stabilized network data support is provided by DPTR. Network resource utilization is efficient enough to maintain connectivity for much longer duration. Based, on this, the proposed DPTR protocol is highly recommendable for a hybrid dynamic network such as “Flying Ad Hoc Networks”.

6.7. Comparison with state-of-the-art solutions

The proposed protocol supports routing while resolving issues related to network partitioning. The proposed DPTR protocol offers a better quality of experience to the users in a network, which operates in a combination of aerial and ground nodes. Apart from the comparisons with the core routing protocols presented in the previous section, the proposed DPTR protocol is also compared with the state-of-the-art solutions, which primarily focus on network partitioning, routing and are tree-based, as shown in Table 7. The comparison helps to understand the features supported by the proposed protocol and its applicability to the mutual ad hoc formations. The existing solutions to network partitioning are efficient, but these do not consider routing and transmission as core functionalities. For the protocols, which are able to support routing in FANETs, there is no provisioning for partitioning issues.

7. Conclusion

In this paper, a distributed priority tree-based routing protocol (DPTR) is proposed for cooperative flying ad hoc networks. The protocol extends the properties of R-B trees by defining governing rules for routing in simultaneously operating distributed ad hoc networks. The proposed DPTR protocol aims at providing a solution to network partitioning problem that arises in such guided ad hoc networks. DPTR is capable of handling two variably operating networks and is able to provide transceiver functionality to each network node. DPTR is evaluated using network simulations determined over scenarios generated using the neural framework.

The proposed DPTR protocol provided high transmission rate with increased scalability, lower latency and higher end to end connectivity. The proposed DPTR protocol also offered higher throughput and increased the delivery ratio by utilizing transmission resources efficiently, such as network channels and corridors as compared to existing routing protocols.

In future, the proposed protocol can be tested by utilizing different mobility models. Further, the proposed protocol can be evaluated under different network conditions, especially fixating on the utilization of UAVs in the next generation of wireless networks.

References

- [1] Ilker Bekmezci, Ozgur Koray Sahingoz, and Åamil Temel. “Flying ad-hoc networks (FANETs): A survey.” *Ad Hoc Networks* 11, no. 3, pp. 1254-1270, (2013).
- [2] Vishal Sharma, and Rajesh Kumar. “An opportunistic cross layer design for efficient service dissemination over flying ad hoc networks (FANETs).” In *Electronics and Communication Systems (ICECS), 2015 2nd International Conference on*, pp. 1551-1557, (2015).
- [3] Mohammad Mozaffari, Walid Saad, Mehdi Bennis, and Mrouane Debbah. “Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage.” *IEEE Communications Letters* 20, no. 8, pp. 1647-1650, (2016).
- [4] Vishal Sharma, and Rajesh Kumar. “Cooperative frameworks and network models for flying ad hoc networks: a survey.” *Concurrency and Computation: Practice and Experience* 29, no. 4, doi: 10.1002/cpe.3931, (2017).
- [5] Haibo Wang, Da Huo, and Bahram Alidaee. “Position Unmanned Aerial Vehicles in the Mobile Ad Hoc Network.” *Journal of Intelligent & Robotic Systems* 74, no. 1-2, pp. 455-464, (2014).
- [6] Nader Mohamed, Jameela Al-Jaroodi, Imad Jawhar, and Sanja Lazarova-Molnar. “A Service-Oriented Middleware for Building Collaborative UAVs.” *Journal of Intelligent & Robotic Systems* 74, no. 1-2, pp. 309-321, (2014).
- [7] M. Liu, J. Lin, and Y. Yuan. “Research of UAV cooperative reconnaissance with self-organization path planning.” In *International Conference on Computer, Networks and Communication Engineering, ICCNCE*, Atlantis Press, pp. 207-213, (2013).
- [8] Min-Hyuk Kim, Hyeyoncheol Baik, and Seokcheon Lee. “Response Threshold Model Based UAV Search Planning and Task Allocation.” *Journal of Intelligent & Robotic Systems*, pp. 1-16, (2013).
- [9] Erik Kuiper and Simin Nadim-Tehrani. “Mobility models for UAV group reconnaissance applications.” In *Wireless and Mobile Communications, IEEE, ICWMC’06*. International Conference on, pp. 33-33, (2006).
- [10] Omer Cetin, Ibrahim Zagli, and Guray Yilmaz. “Establishing Obstacle and Collision Free Communication Relay for UAVs with Artificial Potential Fields.” *Journal of Intelligent & Robotic Systems* 69, no. 1-4, pp. 361-372, (2013).
- [11] Sandip Roy, Yan Wan, and Ali Saberi. “A flexible algorithm for sensor network partitioning and self-partitioning problems.” *Lecture notes in computer science* 4240, pp. 152, (2006).

- [12] Wenrui Zhao, and Mostafa H. Ammar. "Message ferrying: Proactive routing in highly-partitioned wireless ad hoc networks." In Distributed Computing Systems, FTDCS 2003. Proceedings. The Ninth IEEE Workshop on Future Trends of, pp. 308-314, (2003).
- [13] Kemal Akkaya, Fatih Senel, Aravind Thimmapuram, and Suleyman Uludag. "Distributed recovery from network partitioning in movable sensor/actor networks via controlled mobility." IEEE Transactions on Computers 59, no. 2, pp. 258-271, (2010).
- [14] Lili Wang, and Xiaobei Wu. "Distributed prevention mechanism for network partitioning in wireless sensor networks." Journal of communications and networks 16, no. 6, pp. 667-676, (2014).
- [15] Wassila Lalouani, Mohamed Younis, Nadjib Badache. "Optimized repair of a partitioned network topology", doi: <https://doi.org/10.1016/j.comnet.2017.02.003>, Computer Networks, (2017).
- [16] Yaoyao Gu, Doruk Bozdag, Eylem Ekici, Fusun Ozguner, and Chang-Gun Lee. "Partitioning based mobile element scheduling in wireless sensor networks." In Sensor and Ad Hoc Communications and Networks, 2005. IEEE SECON 2005. 2005 Second Annual IEEE Communications Society Conference on, pp. 386-395, (2005).
- [17] Xiao-Ma Huang, Yue-Jiao Gong, Jing-Jing Li, and Xiao-Min Hu. "A novel genetic algorithm based on partitioning for large-scale network design problems." In Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation, pp. 111-112. ACM, (2014).
- [18] Konstantin E. Avrachenkov, Aleksei Yu Kondratev, and Vladimir V. Mazalov. "Cooperative Game Theory Approaches for Network Partitioning." In International Computing and Combinatorics Conference, pp. 591-602. Springer, Cham, (2017).
- [19] Markus Quaritsch, Karin Krugl, Daniel Wischounig-Struel, Subhabrata Bhattacharya, Mubarak Shah, and Bernhard Rinner. "Networked UAVs as aerial sensor network for disaster management applications." e & i Elektrotechnik und Informationstechnik 127, no. 3, pp. 56-63, (2010).
- [20] Byung Duk Song, Jonghoe Kim, Jeongwoon Kim, Hyorin Park, James R. Morrison, and David Hyunchul Shim. "Persistent UAV service: an improved scheduling formulation and prototypes of system components." Journal of Intelligent & Robotic Systems 74, no. 1-2, pp. 221-232, (2014).
- [21] Lin Lin, Qibo Sun, Jinglin Li, and Fangchun Yang. "A novel geographic position mobility oriented routing strategy for UAVs." Journal of Computational Information Systems. v8, pp. 709-716, (2012).
- [22] Sabine Hauert, Jean-Christophe Zufferey, and Dario Floreano. "Evolved swarming without positioning information: an application in aerial communication relay." Autonomous Robots 26, no. 1, pp. 21-32, (2009).
- [23] Daniel Lihui Gu, Guangyu Pei, Henry Ly, Mario Gerla, Beichuan Zhang, and Xiaoyan Hong. "UAV aided intelligent routing for ad-hoc wireless network in single-area theater." In Wireless Communications and Networking Conference, 2000. WCNC. 2000 IEEE, vol. 3, pp. 1220-1225, (2000).
- [24] B. Bellur, M. Lewis, and F. Templin. "An ad-hoc network for teams of autonomous vehicles." In Proceedings of the First Annual Symposium on Autonomous Intelligence Networks and Systems, (2002).
- [25] Izhak Rubin, Arash Behzad, Huei-Jiun Ju, Runhe Zhang, Xiaolong Huang, Yichen Liu, and Rima Khalaf. "Ad hoc wireless networks with mobile backbones." In Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium on, vol. 1, pp. 566-573, (2004).
- [26] Michael Iordanakis, Dimitrios Yannis, Kimon Karras, Georgios Bogdos, Georgios Dilintas, Massimiliano Amirfeiz, Giorgio Colangelo, and Stefano Baiotti. "Ad-hoc routing protocol for aeronautical mobile ad-hoc networks." In Fifth International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP), (2006).
- [27] Vijay K. Shetty, , Moises Sudit, and Rakesh Nagi. "Priority-based assignment and routing of a fleet of unmanned combat aerial vehicles." Computers & Operations Research 35, no. 6, pp. 1813-1828, (2008).
- [28] John J. Enright, and Emilio Frazzoli. "UAV routing in a stochastic time-varying environment." In IFAC World Congress. (2005).
- [29] B. Moses Sathyaraj, Lakhmi C. Jain, Anthony Finn, and S. Drake. "Multiple UAVs path planning algorithms: a comparative study." Fuzzy Optimization and Decision Making 7, no. 3, pp. 257-267, (2008).
- [30] Marco Pavone, Nabhendra Bisnik, Emilio Frazzoli, and Volkan Isler. "A stochastic and dynamic vehicle routing problem with time windows and customer impatience." Mobile Networks and Applications 14, no. 3, pp. 350-364, (2009).
- [31] Jin Zhang, Kai Li, Andreas Reinhardt, and Sali S. Kanhere. "AGRO-Optimal Routing with Low Latency and Energy Consumption in Wireless Sensor Networks with Aerial Relay Nodes.", (2014).
- [32] Jun Li, Yifeng Zhou, Louise Lamont, and Mathieu Dziel. "A token circulation scheme for code assignment and cooperative transmission scheduling in CDMA-based UAV ad hoc networks." Wireless networks 19, no. 6, pp. 1469-1484, (2013).
- [33] Huamiao Hu, Angela Doufexi, Simon Armour, and Dritan Kaleshi. "A Reliable Hybrid Wireless Network Architecture for Smart Grid Neighbourhood Area Networks." In Wireless Communications and Networking Conference (WCNC), 2017 IEEE, pp. 1-6, (2017).
- [34] Mohamed Ben Brahim, Zeeshan Hameed Mir, Wassim Znaidi, Fethi Filali, and Noureddine Hamdi. "QoS-Aware Video Transmission Over Hybrid Wireless Network for Connected Vehicles." IEEE Access 5, pp. 8313-8323, (2017).
- [35] Liansheng Lu, Haifeng Jiang, Guangzhi Han, Shanshan Ma, and Renke Sun. "Multi-criteria routing metric for supporting data-differentiated service in hybrid wireless mesh networks in coal mines." International Journal of Distributed Sensor Networks 13, no. 1, pp. 1-11, (2017).
- [36] Sunil Kumar, Priya Ranjan, Radhakrishnan Ramaswami, and Malay Ranjan Tripathy. "Resource Efficient Clustering and Next Hop Knowledge-Based Routing in Multiple Heterogeneous Wireless Sensor Networks." International Journal of Grid and High Performance Computing (IJGHPC) 9, no. 2, pp. 1-20, (2017).
- [37] Paola G. Vinuela Naranjo, Mohammad Shojafar, Habib Mostafaei, Zahra Pooranian, and Enzo Baccarelli. "P-SEP: a prolong stable election routing algorithm for energy-limited heterogeneous fog-supported wireless sensor networks." The Journal of Supercomputing 73, no. 2, pp. 733-755 (2017).
- [38] Viswanath Kumar, Katia Obraczka, and Gene Tsudik. "Exploring mesh and tree-based multicast. Routing protocols for MANETs." IEEE Transactions on mobile computing 5, no. 1, pp. 28-42, (2006).
- [39] Guo-Mei Zhu, Ian F. Akyildiz, and Geng-Sheng Kuo. "STOD-RP: A spectrum-tree based on-demand routing protocol for multi-hop cognitive radio networks." In Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE, pp. 1-5., (2008).
- [40] Muhammad Sajid, Ayesha Hussain Khan, Saba Gull, Kamran Khan, Muammad Imran, and Nadeem Javaid. "SMPC: Singular division of

- Multipath Power Control tree based routing protocol for Underwater Wireless Sensor Networks." In Wireless Communications and Mobile Computing Conference (IWCMC), 2017 13th International, pp. 1641-1647, IEEE, (2017).
- [41] Radhika Sohan, Nitin Mittal, Urvinder Singh, and Balwinder Singh Sohi. "An Optimal Tree-Based Routing Protocol Using Particle Swarm Optimization." In Nature Inspired Computing, pp. 117-124. Springer, Singapore, (2017).
 - [42] Haoyan Liang, Zhigang Chen, Jia Wu, and Peiyuan Guan. "ETSW: An Encounter History Tree Based Routing Protocol in Opportunistic Networks." In National Conference of Theoretical Computer Science, pp. 46-59. Springer, Singapore, (2017).
 - [43] Tie Qiu, Xize Liu, Lin Feng, Yu Zhou, and Kaiyu Zheng. "An efficient tree-based self-organizing protocol for internet of things." IEEE Access 4, pp. 3535-3546, (2016).
 - [44] Zhao Han, Jie Wu, Jie Zhang, Liefeng Liu, and Kaiyun Tian. "A general self-organized tree-based energy-balance routing protocol for wireless sensor network." IEEE Transactions on Nuclear Science 61, no. 2, pp. 732-740, (2014).
 - [45] Melike Yigit, Ozlem Durmaz Incel, and Vehbi Cagri Gungor. "On the interdependency between multi-channel scheduling and tree-based routing for WSNs in smart grid environments." Computer Networks 65, pp. 1-20, (2014).
 - [46] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. "RedBlack Trees". Introduction to Algorithms (second ed.). MIT Press. pp. 273301. ISBN 0-262-03293-7, (2001).
 - [47] Stefan Kahrs. "Red-black trees with types." Journal of functional programming 11, no. 4, pp. 425-432, (2001).
 - [48] Vishal Sharma, and Rajesh Kumar. "A Cooperative Network Framework for Multi-UAV Guided Ground Ad Hoc Networks." Journal of Intelligent & Robotic Systems, 77, no. 3-4, pp. 629-652, (2015).
 - [49] Vishal Sharma, Rajesh Kumar, "Service-Oriented Middleware for Multi-UAV Guided Ad Hoc Networks." IT CoNvergence PRACTice (IN-PRA), 2(3), pp. 24-33, (2014).
 - [50] Ludwig, Alexander, Klaus D. Schmidt. "GaussMarkov loss prediction in a linear model. In:" CAS E-Forum Fall, pp. 1-48, (2010).
 - [51] Meinholt, R. J., & Singpurwalla, N. D.: Understanding the Kalman filter, *The American Statistician*, 37(2), pp. 123-127, (1983)
 - [52] Charles E. Perkins, and Elizabeth M. Royer. "Ad-hoc on-demand distance vector routing." In Mobile Computing Systems and Applications, Proceedings. WMCSA'99. Second IEEE Workshop on, pp. 90-100. IEEE, (1999).
 - [53] Charles E. Perkins, and Pravin Bhagwat. "Highly dynamic destination sequenced distance-vector routing (DSDV) for mobile computers." In ACM SIGCOMM Computer Communication Review, vol. 24, no. 4, pp. 234-244. ACM, (1994).
 - [54] David B. Johnson, David A. Maltz, and Josh Broch. "DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks." Ad hoc networking 5, pp. 139-172, (2001).
 - [55] Marcello Caleffi, and Luigi Paura. "MDART: multipath dynamic address routing." Wireless Communications and Mobile Computing 11, no. 3, pp. 392-409, (2011).
 - [56] Thomas Clausen, Philippe Jacquet, Cdric Adjih, Anis Laouiti, Pascale Minet, Paul Muhlethaler, Amir Qayyum, and Laurent Viennot. "Optimized link state routing protocol (OLSR)." RFC-3626,(2003).
 - [57] Ihsun You, Vishal Sharma, Mohammed Atiquzzaman, and Kim-Kwang Raymond Choo. "GDTN: Genome-Based Delay Tolerant Network Formation in Heterogeneous 5G Using Inter-UA Collaboration." PLoS one 11, no. 12, pp. e0167913, (2016).
 - [58] Denis Rosrio, Zhongliang Zhao, Torsten Braun, Eduardo Cerqueira, Aldri Santos, and Islam Alyafawi. "Opportunistic routing for multi-flow video dissemination over flying ad-hoc networks." In World of Wireless, Mobile and Multimedia Networks (WoWMoM), IEEE 15th International Symposium on a, pp. 1-6, (2014).
 - [59] Vishal Sharma, Ihsun You, and Rajesh Kumar. "Energy efficient data dissemination in multi-UAV coordinated wireless sensor networks." Mobile Information Systems, 2016, <http://dx.doi.org/10.1155/2016/8475820>, pp. 1-13 (2016).

Evaluation of Flying Ad Hoc Network Topologies, Mobility Models, and IEEE Standards for Different Video Applications

Ghassan A. QasMarrogy

Department of Communication and Computer Engineering, College of Engineering, Cihan University-Erbil,
Kurdistan Region, Iraq

Abstract—Nowadays, drones became very popular with the enhancement of the technological progress of moving devices with a connection to each other, known as Flying Ad Hoc Network (FANET). It is used in most worldwide necessary life scenarios such as video recording, search and rescue, military missions, moving items between different areas, and many more. This leads to the necessity to evaluate different network strategies between these flying drones, which are essential to improve their quality of performance in the field. Several challenges must be addressed to effectively use FANET, to provide stable and reliable transmission for different types of data during vast changing topologies, such as different video sizes, different types of mobility models, recent Wireless Fidelity standards, types of routing protocols used, security problems, and many more. In this paper, a fully comprehensive analysis of FANET will be done to evaluate and enhance these challenges that concern different video types, mobility models, and IEEE 802.11n standards for best performance, by measuring throughput, retransmission attempt, and delay metrics. The result shows that Gauss–Markov mobility model gives the highest result using Ad Hoc On-Demand Vector and lowest delay, whereas for retransmission attempts, 2.4 GHz frequency has the lowest as it can reach more coverage area than 5 GHz.

Index Terms—Video transmission; flying ad hoc network; mobility model; 2.4–5 GHz standards; routing protocols

I. INTRODUCTION

Recently, mobile networks became very popular with the advance in the technology sector, as they can move and fly at different speeds, equipped with multitools to offer different types of application services such as video recording, item carrying, search and rescue, military operations, and many more. There are different types of these mobile networks, namely, ad hoc network, mobile ad hoc network (MANET),

ARO-The Scientific Journal of Koya University
Vol. IX, No.1 (2021), Article ID: ARO.10764, 82 pages
DOI:10.14500/aro.10764



Received: 07 December 2020; Accepted: 14 February 2021
Regular research paper: Published: 08 May 2021

Corresponding author's e-mail: ghassan.qasmarrogy@cihanuniversity.edu.iq
Copyright © 2021 Ghassan A. QasMarrogy. This is an open-access article distributed under the Creative Commons Attribution License.

vehicular ad hoc networks, and flying ad hoc network (FANET) (Kaur and Singh, 2018), all of them use different types of routing protocols to communicate as they change their topology frequently. FANET is a set of moving drones that change its topology consistently without any fixed infrastructure while communicating with a base station from different heights, as shown in Fig. 1 (Marrogy, 2020).

One of the main requirements of FANET is video recording, it is used to capture high-quality videos while flying at different speeds across the area. The formats of the recorded video play a key role in data transmission between the drones, as it needs more time and higher bandwidth B.W to transfer high-quality videos during movement, this also requires better converging between the routing protocols to route the big size packets from node to node (Zheng, Sangaiah and Wang, 2018). Therefore, due to the fast movement of FANET causing the topology to be changed rapidly, it is difficult to capture and transmit high-quality videos between the nodes while movement, which gives FANET limited capability in transmitting. Furthermore, different other challenges and constraints are fronting FANET such as drone's scalability, limited B.W, different types of data transmission, different types of routing protocols, and the time required to finish tasks (Khan, Safi, Qureshi and Khan, 2017).

To communicate between drones, an IEEE 802.11 wireless adapter is needed to transmit with different frequencies such as 2.4 GHz, 5 GHz, and 6 GHz frequency bands. A numbering scheme was used by the Wireless Fidelity (WiFi) Alliance as a WIFI generation 802.11b, 802.11a, 802.11g, 802.11n, 802.11ac, and 802.11ax protocols, respectively (Deng, et al., 2020), whereas some of these generations can support a dual band of 2.4 GHz and 5 GHz together (Karmakar, Chattopadhyay and Chakraborty, 2017). Table I shows the list of WIFI 802.11n standards.

Due to the mentioned challenges, an analysis is needed to test and simulate different types of realistic scenarios for FANET, to determine the best results and parameters for video data transmission.

In this paper, different types of realistic scenarios will be evaluated and analyzed for FANET, to simulate and find the best results for the mentioned challenges, concerning

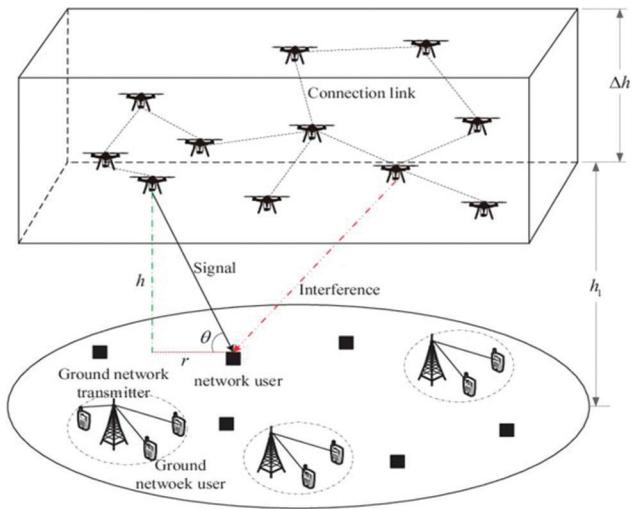


Fig. 1. Flying ad hoc networks [2].

TABLE I
LIST OF WIFI 802.11N STANDARDS (KARMAKAR, CHATTOPADHYAY AND CHAKRABORTY, 2017)

IEEE standard	Frequency/medium	Speed	Transmission range
802.11	2.4 GHz RF	1–2 Mbps	20 feet
802.11a	5 GHz	Up to 54 Mbps	25–75 feet; range can be affected by building materials
802.11b	2.4 GHz	Up to 11 Mbps	Up to 150 feet; range can be affected by building materials
802.11g	2.4 GHz	Up to 54 Mbps	Up to 150 feet; range can be affected by building materials.
802.11n	2.4 GHz/5 GHz	Up to 600 Mbps	175+feet; range can be affected by building materials
802.11ac	5 GHz	Up to 600 Mbps	175+feet; range can be affected by building materials

WIFI: Wireless Fidelity

different video sizes, different type of mobility model, and transmitting with dual bands of IEEE standards by measuring the throughput, retransmission attempts, and delay metrics.

This paper gives extra importance to FANET, as it analyzes different transmission challenges using infrastructure-less FANET flying drones, which is useful for video recording, surveillance, tactical and wireless sensor networks, firefighters, search and rescue teams, and thermal detection which is very important for detecting COVID-19 patients.

The following is the structure of the paper, part two explains the recent related work that analyzed FANET with different scenarios, part three shows and explains the full parameters used for the different scenarios of the paper simulations, part four shows the results and the performance analysis, and finally, part five will conclude the paper with a given future work.

II. RELATED WORK

Different researches were published that analyzed and optimized FANET challenges, using several techniques and methods to enhance the performance of data transmission.

In Mahmud and Cho, 2019, a new technique was proposed for low-energy hello interval adaptive signals by choosing the best route to the receiver and minimizing the number of FANET drones used to establish routes. The new technique decreases up to 25% less from the used energy. In He, Tang, Zhang, Du, Zhou and Guizani, 2019, a new protocol for FANET was proposed course-aware opportunistic routing estimate the best position of the next drone to transmit the data with lower delay and higher throughput, the result shows lower delay with highly performance improvement and better throughput. In Srivastava and Prakash, 2021, a comprehensive survey is presented regarding FANET and discussion about its main critical problems, such as FANET's characteristics, many mobility models, types of possible communication, architecture, categorization, routing protocol, and topologies, also a discussion about FANET's related problems and challenges was analyzed to determine the probabilistic listed points of the research and methods for better results. The researcher in QasMarrogy, 2020, shows an attempt to enhance the video transmission in 5 GHz frequency in IEEE Wifi standard by changing wireless LAN parameters of MANET nodes, the result shows better performance and throughput with lower delay using specific parameters. Finally, AlKhatieb, Felemban and Naseer, 2020, presented an evaluation and comparison about different types of mobility models such as Pursue Mobility Model, Semi-Random Circular Movement (SCRM), Manhattan Grid Mobility Model (MGM), and Random Waypoint Mobility (RWPM), the paper concludes MGM model effect highly on the packet dropping ratio and delay.

III. FANET'S PARAMETER

In this part, a full explanation will be presented and analyzed regarding the calculated FANET parameters in this paper.

A. Routing Protocols

Routing protocols are a set of rules that connect drones to guide the transmitted packets from the source node to the destination. There are different types of routing protocols used to connect the flying drones or moving nodes (QasMarrogy, Alqaysi and Almashhadani, 2017), and they can be classified into three types, namely, reactive, proactive, and hybrid routing protocols, as shown in Fig. 2.

In this paper, two types of routing protocols were selected and simulated as they give the highest result for FANET, namely, **Ad Hoc On-Demand Vector (AODV)** and **Optimized Link State Routing (OLSR) Protocol**.

Due to the reactive nature of AODV, it broadcast discovery messages tagged with sequence numbers to select a recently updated route to the destination. When the sender sends a packet to the destination the discovery mechanism started, and the sender will record all the fresh recent routes in its routing table to the time that the sender finishes the transmission, and the stored routes will be deleted, then when a new packet will be transmitted another discovery message will be broadcast and the same circle will be repeated, which

can give more delay to the packets and higher overhead at the beginning of the transmission (QasMarrogy, 2021).

Furthermore, due to the proactive nature of OLSR, it exchanges the final recent information regarding all the latest routes between a group of selected devices or drones called Multipoint Relays, resulting in lower delays during the process of route discovery. After finishing transmission, all the used routes will be stored and used again for the next transmission while collecting more information about more effective routes all the time, this can cause lower delay during transmission but gives higher load as there is too much information broadcasted all the time to prepare available routes continuously (QasMarrogy, 2021).

B. Mobility Models of FANET

Mobility models show the movement of drones from one location to another location with varying velocity and direction over a specific amount of time. The main challenge in FANET is the varying speed and height of flying drones, which causes the packets to be delayed, failed to be received, or dropped (Chriki, Touati, Snoussi and Kamoun, 2019). Therefore, to break this challenge, the performance of FANET must be evaluated using different mobility models with realistic scenarios that are used in real-life missions. Many researchers use the RWPM to simulate real-life scenarios for FANET, where the drones in this model flies to random directions with random speeds, which lowest the performance as they cause multiple route link breakage between the drones.

In this paper, three different mobility models were simulated, namely, Gauss Markov Mobility (GMM) Model, RWPM, and SCRM Model, as shown in Fig. 3.

The first type of mobility model used in this paper is the RWPM, it uses random times for movements and pauses, and random values for speeds and directions. This model starts when the simulation begins and finishes when the simulation is finished. This randomness in movement and speed will be repeated during simulation time. This model is very important for real-life scenarios such as wireless sensor networks, scanning areas, and search and rescue missions (Sharma and Kim, 2019).

The second type is the SCRM, which uses a route shaped like a hexagon or a circle with a fixed speed to the destination drone, both hexagon area and directions are specified at the beginning of the simulation, and it is used for patrolling surveys and target monitoring (Adya and Sharma, 2020).

The third type is the GMM, it updated the direction and speed of the drone according to their past values during

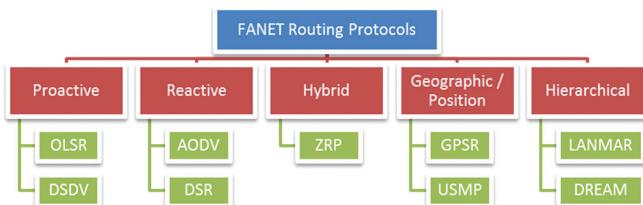


Fig. 2. Flying ad hoc network routing protocols (QasMarrogy, Alqaysi and Almashhadani, 2017).

earlier time intervals. Direction and speed calculations are done with a specific degree of randomness that depends on FANET parameters. Finally, each drone movement is independent of all other drones in the same FANET (Korneev, Leonov and Litvinov, 2018).

C. IEEE WIFI Standards

In the wireless world, WIFI is a standard means of wireless access transmission, by sending the packet without any cable using radio frequencies, which is a trademark owned by the WIFI Alliance that sets the IEEE of 802.11 wireless standards (Deniau, Gransart, Romero, Simon and Farah, 2017).

One of the standards of WIFI is the IEEE 802.11n that uses dual B.W frequencies of 2.4 GHz and 5 GHz, which considered an essential upgrade to the WIFI standards, by increasing data transmission speed while decreasing the delay and interference (Dolińska, Jakubowski and Masiukiewicz, 2017), Fig. 4 shows a comparison between both 2.4 GHz and 5 GHz in terms of speed, delay, interference, and range.

Data transmission in WIFI can affect the performance of FANET greatly, using 5 GHz frequency can increase the speed of transmission, as it has less network congestion, much reliable WIFI connection, more solid, and lower interference, as it uses more than 20 channel to operate than the 2.4 GHz which uses 13 channels with three non-overlapping channels (1, 6, and 11), as shown in Fig. 5 (Qaddus, 2019). Still, as a disadvantage, it has less coverage range as it has shorter radio waves. Nonetheless, WIFI coverage can be extended using high gain directional antennas (Aziz, Abd Razak and Ghani, 2017).

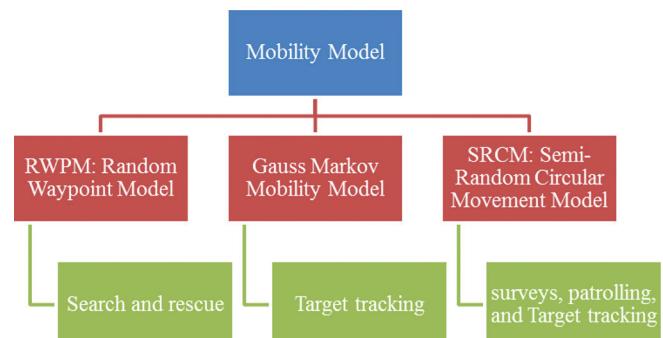


Fig. 3. Real-life mobility model scenarios (Chriki, Touati, Snoussi and Kamoun, 2019).

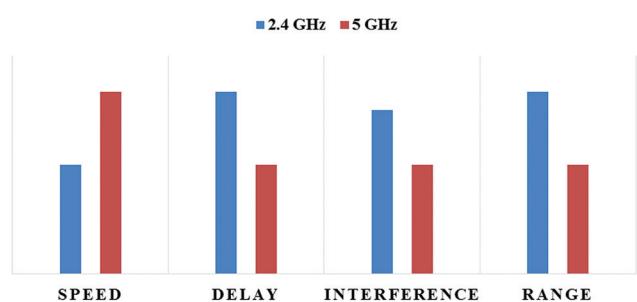


Fig. 4. 2.4 GHz versus 5 GHz comparison (Deniau, Gransart, Romero, Simon and Farah, 2017).

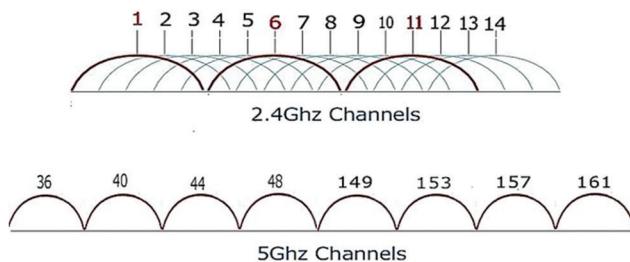


Fig. 5. 2.4 GHz and 5 GHz channel band (Aziz, Abd Razak and Ghani, 2017).

For high-quality video transmission, IEEE 802.11n 5 GHz frequency will be more required as it can support a higher transfer rate of data as possible (Qaddus, 2019).

IV. RESULTS AND ANALYSIS

In this paper, a comprehensive analysis of realistic FANET scenarios will be done, to evaluate different aspects of challenges and parameters that facing FANET, to enhance its performance by comparing the results of the simulations together. A 1500×1500 square meter area was simulated with 30 drones flying at a height of 20 m, with a varying speed from 1.4 m/s (human walking speed) to 20 m/s (drone speed), recording two types of video quality, HD and 2k, moving with three different types of mobility models, namely, RWPM, SRCM, and GMM. These flying drones stream and send the recorded videos from one to one until they reach the destination drone using two types of WIFI IEEE 802.11n standards, 2 GHz and 5 GHz, with the support of two types of routing protocols AODV, and OLSR. All simulations were repeated 10 times with average calculations to get the optimal values by calculating the end-to-end delay, throughput, and retransmission attempt of the transmitted packets metrics. Finally, the NS3 simulator was used to simulate FANET as it is one of the best network simulation tools.

To calculate the required B.W for video quality formats, the following equation will be used Equation (1) (Li, Salehi, Bayoumi and Buyya, 2016).

$$\text{Size of Video} = \text{Color Depth} \times \text{frame Rate}$$

$$\times \text{Streaming Duration} \times (\text{Frame } w \times h)$$

To calculate the two sizes of videos chosen in this paper using Equation (1), with size of 1280×720 pixel for the HD video and 2048×1080 pixel for the 2K video, with a depth color of 240 bits, 30 frame rate, and 60 s streaming duration. The audio size will be neglected as it does not affect the B.W, the final size will be as follows:

$$\text{Size of HD Video} = 240 \times 30 \times 60s \times (1280 \times 720) \approx 47,460 \text{ MB}$$

$$\text{Size of 2K Video} = 240 \times 30 \times 60s \times (2048 \times 1080) \approx 113,906 \text{ MB}$$

The full parameters used in the simulations are shown in Table II.

The results in Fig. 6 demonstrate the streaming of HD video throughput for IEEE 802.11n, routing protocols, and mobility models of FANET. It shows that AODV gives a high

TABLE II
SIMULATION PARAMETERS

Parameters	Values
Video formats	HD (1280×720) pixels, 2K (2048×1080) pixels
Area size	1500×1500 m ²
FANET routing protocols	AODV, OLSR
FANET number	30 Drone
Mobility models	RWPM, SRCM, GMM
Simulation time	1 min
Node speed, height	Varying 1.4–20 m/s, 20 m
WIFI IEEE 802.11n	2.4 GHz, 5 GHz

FANET: Flying ad hoc network

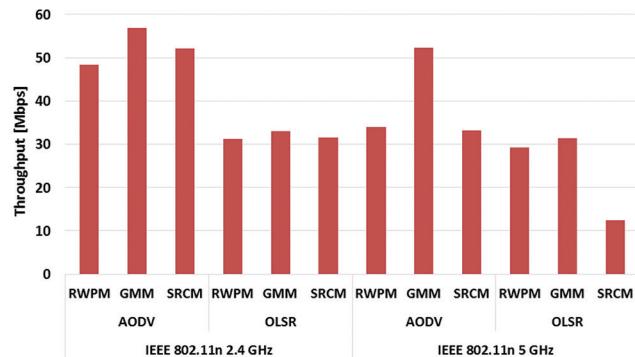


Fig. 6. HD video throughput for IEEE 802.11n, routing protocols, and mobility models.

result in GMM for the 5 GHz and 2.4 GHz whereas OLSR shows less result. As a reactive protocol AODV requires less B.W to discover a recent fresh route to destination, still it overhead high message at the beginning of the transmission to find recent routes, after that the packets transmitted until the simulation ends. Whereas OLSR keeps transmitting advertising messages to fined routes continuously, but due to FANET movement, the topology is changing all the time which is difficult to keep consistent routes for transmission.

GMM in mobility models shows higher throughput as it keeps the distance and speed from one point to another, this keeps the route fresher for a long time for transmission. RWPM gives the lowest results, as its routes keep changing due to high random mobility direction and speed.

Fig. 7 demonstrates the streaming of 2K video throughput for IEEE 802.11n, routing protocols, and mobility models of FANET. It shows that AODV has also higher results as its search for a new route in the beginning of the transmission, therefore, it can keep the route for a longer time. OLSR also shows less result with the RWPM mobility model.

The result in Fig. 8 demonstrates the streaming of HD video delay for IEEE 802.11n, routing protocols, and mobility models of FANET. It shows that the mobility model GMM for AODV gives less delay for both 2.4 and 5 GHz, respectively, as AODV uses a fresh route for more time than OLSR, which keeps the route alive for a long time for transmission, therefore, it lowers the delay to find other routes on request.

SRCM and RWPM in mobility models show higher delay as their movement and speed change all the time, which causes the route to be broken.

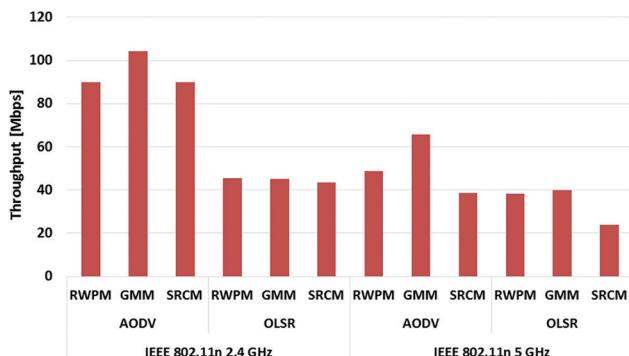


Fig. 7. 2K video throughput for IEEE 802.11n, routing protocols, and mobility models.

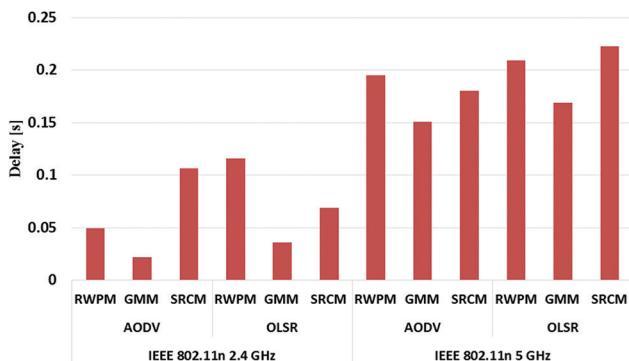


Fig. 8. HD video delay for IEEE 802.11n, routing protocols, and mobility models.

Fig. 9 demonstrates the streaming of 2K video delay for IEEE 802.11n, routing protocols, and mobility models of FANET. It shows that 2.4 GHz gives less delay for all mobility models as the distance between the nodes get higher, which is not convenient for 5 GHz as it has small coverage distance, thus causing a higher delay.

The result in Fig. 10 demonstrates the streaming of HD video retransmission attempts for IEEE 802.11n, routing protocols, and mobility models of FANET. It shows that the mobility model GMM for AODV also gives less retransmission attempt for both 2.4 and 5 GHz, as the route remains longer than OLSR which support the packet delivery, thus decreasing the retransmission attempts from the drone.

RWPM always shows higher retransmission attempts as the drone movement and speed changes all the time, which causes the route to be broken, and the packet to be dropped and retransmit again.

Fig. 11 demonstrates the streaming of 2K video retransmission attempts for IEEE 802.11n, routing protocols, and mobility models of FANET. It shows that 2.4 GHz gives less retransmission attempts for all mobility models as the distance between the nodes get higher, which is not convenient for 5 GHz as it has small coverage distance, thus causing higher route breakage, and more retransmission attempts from the drones.

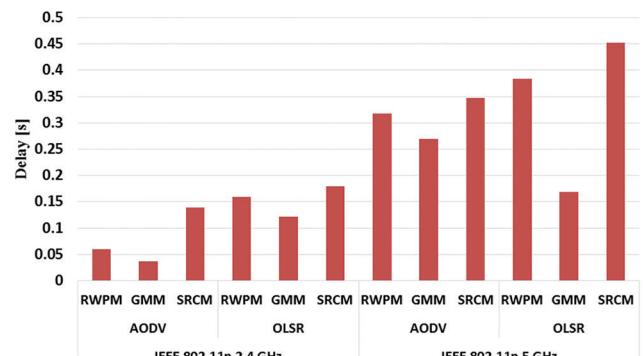


Fig. 9. 2K video delay for IEEE 802.11n, routing protocols, and mobility models.

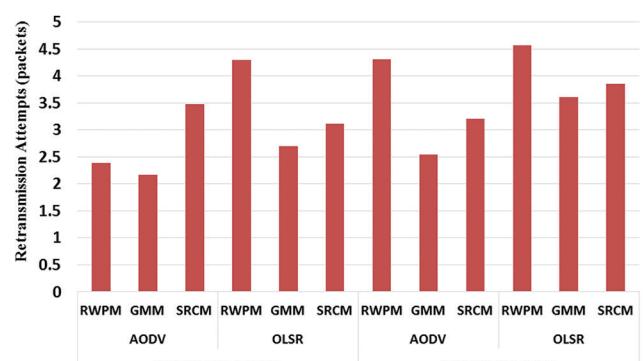


Fig. 10. HD video retransmission attempts for IEEE 802.11n, routing protocols, and mobility models.

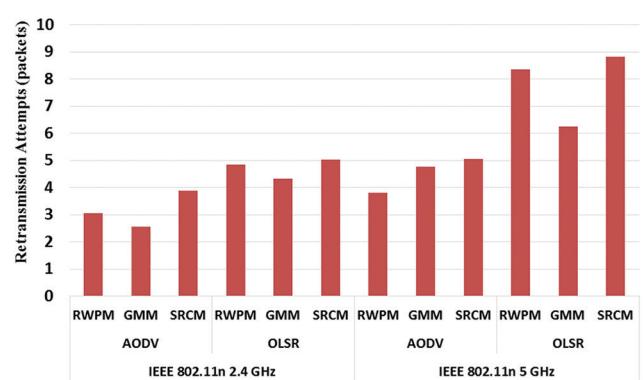


Fig. 11. 2K video retransmission attempts for IEEE 802.11n, routing protocols, and mobility models.

V. CONCLUSION

Recently, FANET is considered one of the most important future technologies to be used, as it can achieve more realistic missions without any physical guidance such as thermal detection for COVID-19, search and rescue, military missions, moving objects between different areas, and many more. Still, FANET is facing different types of challenges to achieve better performance for heavy duty missions such as battery life, transmission rate, and coverage, mobility

models, and more. Thus, it is very important to analyze these challenges of FANET.

In this paper, a comprehensive analysis for FANET was done for 30 drones, flying with varying speed of 1.4, to 20 m/s, using two types of routing protocol, AODV, and OLSR, with three types of mobility models, RWPM, GMM, and SRCM, while transmitting two types of video format HD and 2K, using IEEE 802.11n standards 2.4 GHz and 5 GHz frequencies.

The result shows that the mobility model GMM gives higher throughput and lower delay for both 2.4 and 5 GHz frequencies using the AODV routing protocol, as it keeps the same routes available for a longer time than OLSR, which gives more time for the packets to be transmitted with lower delay, as OLSR keeps changing its routes during topology changes. IEEE 802.11n 2.4 GHz gives a better result as it supports more coverage area from 5 GHz, therefore. When the node changing its speed and direction randomly, their distance becomes larger, which will be out of the area of coverage for 5 GHz frequency. For mobility models, GMM is better for drones as it keeps its speed and direction with the group, therefore, no route breakage or packet dropping happens.

As FANET is upgrading all the time, future work is needed, to test and simulate more frequencies and more mobility models, with different types of applications and metrics, also different sizes of drones are needed to be analyzed.

REFERENCES

- Adya, A. and Sharma, K.P., 2020. Energy Aware Clustering Based Mobility Model for FANETs. In: *Proceedings of ICETIT 2019*. Springer, Cham, pp.36-47.
- AlKhatieb, A., Felemban, E. and Naseer, A., 2020. Performance evaluation of Ad-Hoc routing protocols in (FANETs). In: *2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. IEEE, United States, pp.1-6.
- Aziz, T.A.T., Abd Razak, M.R. and Ghani, N.E.A., 2017. The performance of different IEEE802.11 security protocol standard on 2.4 GHz and 5GHz WLAN networks. In: *2017 International Conference on Engineering Technology and Technopreneurship (ICE2T)*. IEEE, United States, pp.1-7.
- Chriki, A., Touati, H., Snoussi, H. and Kamoun, F., 2019. FANET: Communication, mobility models and security issues. *Computer Networks*, 163, p.106877.
- Deng, C., Fang, X., Han, X., Wang, X., Yan, L., He, R., Long, Y. and Guo, Y., 2020. IEEE 802.11 be Wi-Fi 7: New challenges and opportunities. *IEEE Communications Surveys and Tutorials*, 22(4), pp.2136-2166.
- Deniau, V., Gransart, C., Romero, G.L., Simon, E.P. and Farah, J., 2017. IEEE 802.11 n communications in the presence of frequency-sweeping interference signals. *IEEE Transactions on Electromagnetic Compatibility*, 59(5), pp.1625-1633.
- Dolińska, I., Jakubowski, M. and Masiukiewicz, A., 2017. Interference comparison in Wi-Fi 2.4 GHz and 5 GHz bands. In: *2017 International Conference on Information and Digital Technologies (IDT)*. IEEE, United States, pp.106-112.
- He, Y., Tang, X., Zhang, R., Du, X., Zhou, D. and Guizani, M., 2019. A course-aware opportunistic routing protocol for FANETs. *IEEE Access*, 7, pp.144303-144312.
- Karmakar, R., Chattopadhyay, S. and Chakraborty, S., 2017. Impact of IEEE 802.11 n/ac PHY/MAC high throughput enhancements on transport and application protocols-a survey. *IEEE Communications Surveys and Tutorials*, 19(4), pp.2050-2091.
- Kaur, P. and Singh, A., 2018. Nature-inspired optimization techniques in VANETs and FANETs: A survey. In: *Advanced Computational and Communication Paradigms*. Springer, Singapore, pp.651-663.
- Khan, M.A., Safi, A., Qureshi, I.M. and Khan, I.U., 2017. Flying ad-hoc networks (FANETs): A review of communication architectures, and routing protocols. In: *2017 First International Conference on Latest trends in Electrical Engineering and Computing Technologies (INTELLECT)*. IEEE, United States, pp.1-9.
- Korneev, D.A., Leonov, A.V. and Litvinov, G.A., 2018. Estimation of mini-UAVs network parameters for search and rescue operation scenario with Gauss-Markov mobility model. In: *2018 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO)*. IEEE, United States, pp.1-7.
- Li, X., Salehi, M.A., Bayoumi, M. and Buyya, R., 2016. CVSS: A cost-efficient and QoS-aware video streaming using cloud services. In: *2016 16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*. IEEE, United States, pp.106-115.
- Mahmud, I. and Cho, Y.Z., 2019. Adaptive hello interval in FANET routing protocols for green UAVs. *IEEE Access*, 7, pp.63004-63015.
- Marrogy, G.A.Q., 2020. Enhancing video streaming transmission in 5 GHz fanet drones parameters. *Telecommunications and Radio Engineering*, 79(11), pp.997-1007.
- Qaddus, A., 2019. An Evaluation of 2.4 GHz and 5 GHz ISM radio bands utilization in backhaul IP microwave wireless networks. In: *2019 International Conference on Information Science and Communications Technologies (ICISCT)*. IEEE, United States, pp.1-5.
- QasMarrogy, G.A., 2020. Optimizing video transmission performance in 5GHz MANET. *Journal of Duhok University*, 23(2), pp.402-411.
- QasMarrogy, G.A., 2021. Improving VoIP transmission for IEEE 802.11 n 5GHz MANET. *Zanco Journal of Pure and Applied Sciences*, 33(1), pp.157-162.
- QasMarrogy, G.A., Alqaysi, H.J. and Almarshadani, Y.S., 2017. Comprehensive study of hierarchical routing protocols in MANET using simple clustering. In: *Conference of Cihan University-Erbil on Communication Engineering and Computer Science*, p.62.
- Sharma, P.K. and Kim, D.I., 2019. Random 3D mobile UAV networks: Mobility modeling and coverage probability. *IEEE Transactions on Wireless Communications*, 18(5), pp.2527-2538.
- Srivastava, A. and Prakash, J., 2021. Future FANET with application and enabling techniques: Anatomization and sustainability issues. *Computer Science Review*, 39, p.100359.
- Zheng, Z., Sangaiah, A.K. and Wang, T., 2018. Adaptive communication protocols in flying ad hoc network. *IEEE Communications Magazine*, 56(1), pp.136-142.

Article

A Hybrid Communication Scheme for Efficient and Low-Cost Deployment of Future Flying Ad-Hoc Network (FANET)

Muhammad Asghar Khan ^{1,*}, Ijaz Mansoor Qureshi ² and Fahimullah Khanzada ³

¹ Department of Electronic Engineering, ISRA University, Islamabad 44000, Pakistan

² Department of Electrical Engineering, AIR University, Islamabad 44000, Pakistan; imqureshi@mail.au.edu.pk

³ National Physical and Standards Laboratory (NPSL), Islamabad 44000, Pakistan; fahimullahk@gmail.com

* Correspondence: khayyam2302@gmail.com; Tel.: +92-336-5276465

Received: 26 December 2018; Accepted: 2 February 2019; Published: 11 February 2019



Abstract: In recent years, FANET-related research and development has doubled, due to the increased demands of unmanned aerial vehicles (UAVs) in both military and civilian operations. Equipped with more capabilities and unique characteristics, FANET is able to play a vital role in mission-critical applications. However, these distinctive features enforce a series of guidelines to be considered for its efficient deployment. Particularly, the use of FANET for on-time data communication services presents demanding challenges in terms of energy efficiency and quality of service (QoS). Proper use of communication architecture and wireless technology will assist to solve these challenges. Therefore, in this paper, we review different communication architectures, including the existing wireless technologies, in order to provide seamless wireless connectivity. Based on the discussions, we conclude that a multi-layer UAV ad-hoc network is the most suitable architecture for networking a group of heterogeneous UAVs, while Bluetooth 5 (802.15.1) is the most favored option because of its low-cost, low power consumption, and longer transmission range for FANET. However, 802.15.1 has the limitation of a lower data rate as compared to Wi-Fi (802.11). Therefore, we propose a hybrid wireless communication scheme so as to utilize the features of the high data transmission rate of 802.11 and the low-power consumption of 802.15.1. The proposed scheme significantly reduces communication cost and improves the network performance in terms of throughput and delay. Further, simulation results using the Optimized Network Engineering Tool (OPNET) further support the effectiveness of our proposed scheme.

Keywords: UAVs; FANET; drones; Bluetooth; Wi-Fi; communication architecture; wireless technologies; OPNET

1. Introduction

During the past couple of years, providing access anytime and anywhere to network resources has become an important challenge for FANET. Such a challenge is further agitated in mission-critical applications such as disaster recovery operations, where on-time data transmission is a prerequisite. Therefore, there exists a high demand for wireless communication technologies that can be rapidly deployed to enable data communication services with air-to-air and air-to-ground links. One possible solution for providing wireless connectivity is via high-altitude platforms (HAPs), such as balloons. The balloons generally reside on stratosphere at an altitude of about tens of kilometers above the earth's surface, as shown in Figure 1. However, HAP-based communications are preferred for large geographical areas. On the other hand, low-altitude platforms (LAPs) have several advantages over HAPs [1], primarily in cases of disastrous events when existing communication infrastructure

collapses and where it is difficult to install communication infrastructure in a short amount of time. These on-demand short-range wireless technologies can be used to provide easily deployable and reconfigurable communication services for connecting UAVs with rescue teams on the ground. Moreover, due to the low altitude of UAVs, short-range wireless links lead to substantial performance improvement over the high-altitude wireless links. In addition, performance can further be improved with jointly designed adaptive communications with a UAV mobility model. For example, when a UAV experiences good communication channel with the ground station, the speed to sustain wireless connectivity is reduced. These evident benefits make a short-range wireless system a promising solution for the low-cost deployment of the future flying ad-hoc network.

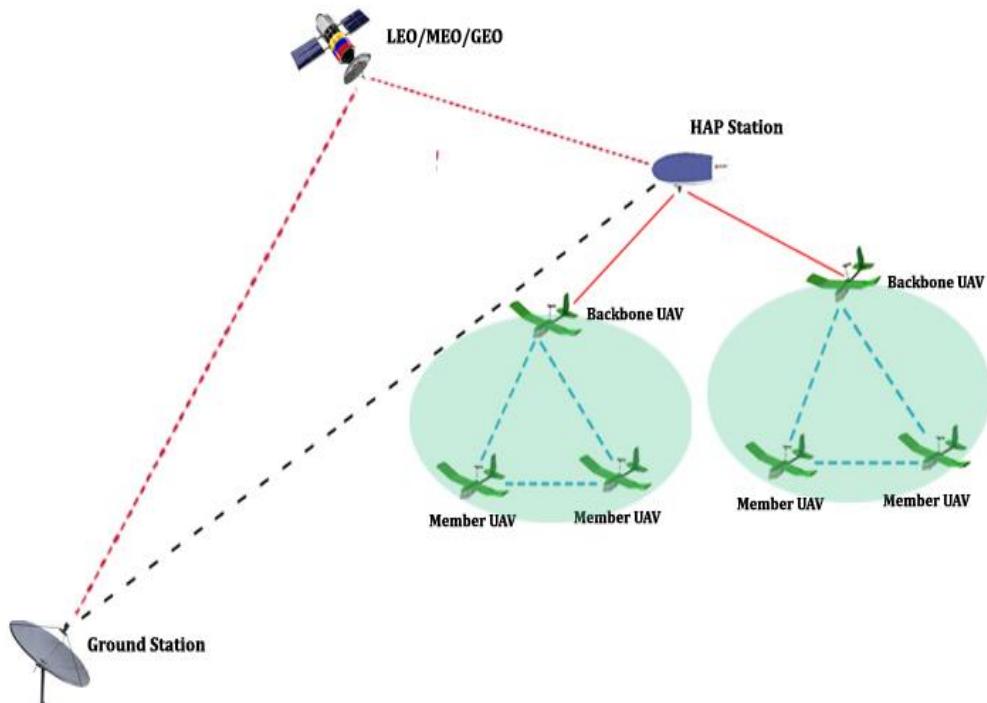


Figure 1. High-altitude platform (HAP) for FANET.

Recently, the need for various types of on-demand wireless communication technologies is increasing for the rapid, efficient, and low-cost deployment of FANET. The technology of choice should be able to support air-to-air and air-to-ground connections in an unhampered manner. It must be able to cater to a wide range of UAV applications, irrespective of significant height and orientation differences of UAVs [2]. In this context, a number of wireless technologies can be exploited for FANET, in order to execute the communication compulsion for all UAV applications. However, a pertinent question needs to be answered about the existing wireless technologies, which is whether they are able to withstand the harsh environment of aerial links or not. Secondly, the doubt behind their applicability for the real-time communication also needs to be cleared. One of the major purposes of our work is to come forth with answers to the requirements, constraints, and shortcomings of these technologies for FANET-focused communication and networking schemes.

Since wireless communication technologies are progressing at a rapid pace, UAVs with reduced size are being introduced with increased incorporation in FANET applications. Thus, the installation and maintenance of the networked communication among multiple UAVs and the ground station has emerged as a crucial task [3]. In addition, accomplishing the complex tasks, the UAVs need to assure simultaneous coordination with each other and with the ground station. For this reason, to realize a well-coordinated UAV ad-hoc network, an efficient architecture is required [4].

Inspired by the aforementioned observations, in this work, with the aim to address the communication and networking needs of FANET, we present a comprehensive review of

communication architectures and wireless technologies. It has been concluded that multi-layer UAV ad-hoc network is the most suitable architecture for networking a group of heterogeneous UAVs, while 802.15.1 is the strongest candidate in short-range wireless technologies. Some of the attractive reasons behind choosing 802.15.1 are as follows: it operates in the unlicensed spectrum; it has reasonable data rates and transmission coverage; and it can be easily integrated with small UAVs. On the other hand, the transmission speed of 802.11 is higher than that of 802.15.1. For this reason, we propose a hybrid wireless communication scheme, which can utilize the strong features of each of the two options: i.e., the high-speed data transmission rate of 802.11 and the low-power consumption of 802.15.1.

The remainder of the paper is organized as follows: Section 2 presents various communication architectures for networking of UAVs with a discussion on the pros and cons; Section 3 elaborates existing communication technologies to address the communication needs of FANET; Section 4 details resource management and energy efficiency; Section 5 describes the proposed hybrid communication scheme; Section 6 presents our Optimized Network Engineering Tool (OPNET) simulation setup and performance metrics; Section 7 contains results and analysis; Section 8 concludes the work.

2. Networking Architectures

In FANET, UAVs maintain two basic functions to perform various tasks i.e., relaying UAV-to-UAV (U2U) communications and gathering network information. Besides, UAVs also perform two additional tasks: they work as remote access units to extend the coverage of infrastructures, and they can develop a dynamic network with heterogeneous resources. According to the roles performed by UAVs in a FANET architecture, two networking modes must be enabled: UAV-to-UAV (U2U) and UAV-to-Infrastructure (U2I). For this purpose, one of the UAVs is referred to as the “backbone UAV,” as it plays the role of a gateway in the FANET architecture. The backbone UAV collects data from the member UAVs (through U2U) and then relays the aggregated data to the ground station (GS) using U2I communication.

Li et al. [5] presented two types of architectures for connecting multiple UAVs, i.e., centralized and decentralized. Centralized architecture involves the ground station as a central node, and all the UAVs are directly linked to it, as illustrated in Figure 2a. Thus, data transmission between any two UAVs needs to be sent via the ground station (GS). Contrarily, UAVs can communicate directly or indirectly in the decentralized architecture without relying on a GS. The authors further introduced three types of decentralized architectures: UAV ad-hoc network, multi-group UAV ad-hoc network, and multi-layer UAV ad-hoc network. Similarly, in [6], the authors explored such architectures for the deployment of FANET. The three architectures are discussed in the following paragraphs.

2.1. UAV Ad-Hoc Network

In a UAV ad-hoc network, each UAV contributes to the data forwarding process for all the other UAV of the network, as shown in Figure 2b. The backbone UAV serves as a gateway between the GS and the member UAV in this specific architecture. The backbone UAV is normally equipped with two radios: low power short-range and high power long-range. Low power short-range radio is used for communication between the UAVs, and high power long-range radio is required to communicate with the GS. Since only one backbone UAV is connected with the GS in the UAV ad-hoc networking architecture, the coverage area by the network is substantially extended. Furthermore, as the distance between the UAV is relatively small, the transceiver mounted on the UAV can be cost-effective and lightweight, which makes it extremely attractive for small-sized UAVs. However, to make the network more persistent, the characteristics such as mobility patterns, speeds, altitudes, and heading directions of all the connected UAVs need to be similar. Therefore, such networking architecture is best suited for surveillance, monitoring operation, and other mission(s) where a small number of homogeneous UAVs are required for deployment.

2.2. Multi-Group UAV Ad-Hoc Network

A multi-group UAV ad-hoc network is, primarily, an integration of a UAV ad-hoc network and a centralized network architecture, as depicted in Figure 2c. In this architecture, UAVs within a group form an ad-hoc network, and the backbone UAV of each group is further connected to the GS in a centralized manner. Intra-group communication is performed within a UAV ad-hoc network without involving the GS, while inter-group communication is conducted via the backbone UAV. This architecture is best suited for missions where a large number of heterogeneous UAVs need to be deployed. However, this networking architecture lacks robustness due to its semi-centralized nature.

2.3. Multi-Layer UAV Ad-Hoc Network

Another architecture in the form of networking multiple groups of heterogeneous UAVs is the multi-layer UAV ad-hoc network, as illustrated in Figure 2d. In this architecture, the networking between member UAVs within a specific group form an ad-hoc UAV network, which corresponds to the lower layer of the network. The backbone UAVs of all groups are connected to each other and come under the upper layer. However, only one backbone UAV from a group is further connected to the GS. In addition, only the GS holds the information that is routed to it so as to reduce communication and the computation load on the GS. Therefore, this architecture is suitable for missions that involve a large number of heterogeneous UAVs.

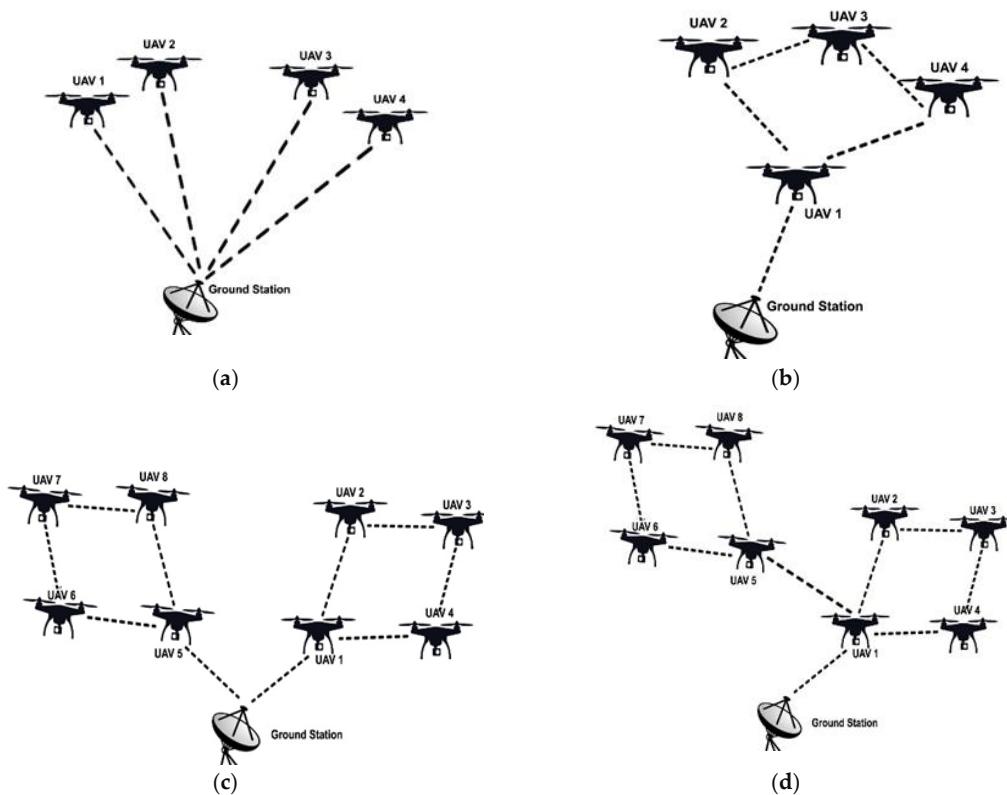


Figure 2. Communication architectures. (a) Centralized architecture; (b) unmanned aerial vehicle (UAV) ad-hoc network; (c) multi-group UAV ad-hoc network; (d) multi-layer UAV ad-hoc network.

In summary, a decentralized communication architecture is suitable for connecting multiple UAVs in FANET. With the help of its multi-hop schema, this communication architecture can provide extended coverage for data transmission. Moreover, the UAV ad-hoc network architecture is suitable for UAVs that are homogeneous and less in number. On the other hand, multi-group and multi-layer architectures can be deployed in situations where a large number of heterogeneous UAVs are required. In addition, a multi-layer UAV ad-hoc network may be preferred to a multi-group counterpart due

to its fully decentralized nature. A multi-layer ad-hoc UAV networking architecture, furthermore, is more flexible in providing “on-the fly” communication network, chiefly due to its robustness against a single point of failure.

3. Wireless Communication Technologies

In the previous section, we focused on different networking architectures between UAVs and GS. According to the architectures presented in Figure 2, many wireless communication technologies are potential choices to provide reliable and flexible communication links for the fast deployment of FANET. A list of likely candidates that can be selected for the different links of UAV-to-UAV (U2U) and UAV-to-Infrastructure (U2I) communications is tabulated in Table 1. The table shows their main characteristics such as mobility, data rates, communication range, latency, and network topology. The choice of suitable technology depends on the nature of the application and the type of mission. However, demand for the control traffic of FANET is low and can be supported by all wireless technologies. Wireless technologies can be divided into two main categories: short-range and long-range communication technologies. Short-range communication technologies, such as Wi-Fi, ZigBee, and Bluetooth, are used for short distance communication, while the long-range communication technologies, such as cellular, WiMAX, and satellite, can be employed for larger areas. Both of the technologies are being further explored in the following paragraphs.

3.1. Short-Range Communication Technologies

Short-range communication technologies are not only concerned with providing wireless access in the instant vicinity, but, in a broader perspective, they also offer off-the-shelf, light-weight, and cost-effective communication links due to the spectrum free bands. Short-range communication technologies offer a transfer of information from millimeters to a few hundreds of meters. The maximum indoor/outdoor distance for some of the short-range communication technologies are shown in Figure 3. Some of the major short-range communication technologies are being investigated as follows:

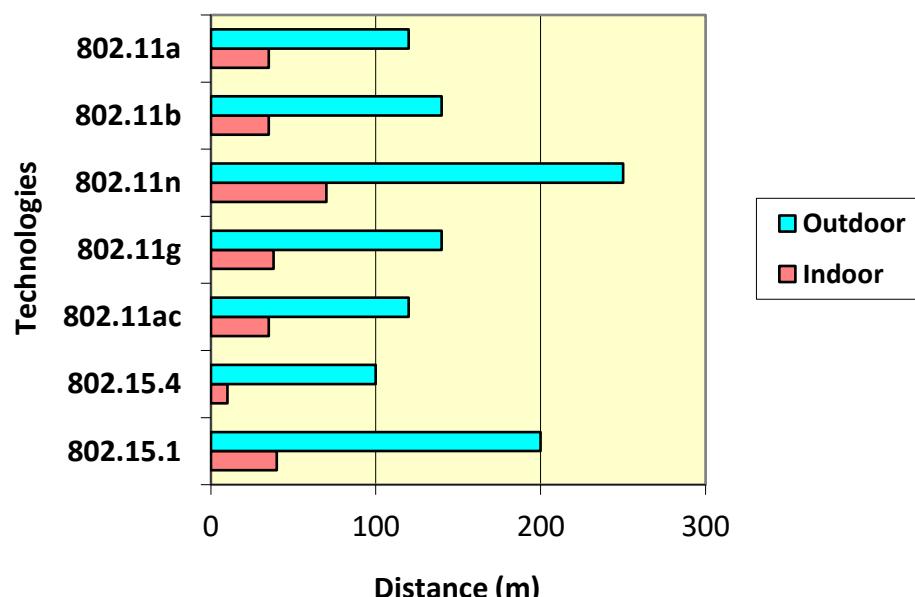


Figure 3. Outdoor/indoor distances for some of the short-range wireless technologies.

Table 1. Comparison between different communication technologies available for UAVs.

Communication Technology	IEEE Standard	Frequency/Medium	Spectrum Type	Device Mobility	Theoretical Data Rate	Range Indoor-Outdoor	Network Typology	Latency	Advantages	Limitations
Wi-Fi [7,8]	802.11	2.4 GHz IR	Unlicensed	Yes	Up to 2 Mbps	20 m–100 m	Ad-hoc, star, mesh, hybrid	<5 ms	High speed and cheap	Limited range
	802.11a	5 GHz	Unlicensed	Yes	Up to 54 Mbps	35 m–120 m	Ad-hoc, star, mesh, hybrid			
	802.11b	2.4 GHz	Unlicensed	Yes	Up to 11 Mbps	35 m–140 m	Ad-hoc, star, mesh, hybrid			
	802.11n	2.4/5 GHz	Unlicensed	Yes	Up to 600 Mbps	70 m–250 m	Ad-hoc, star, mesh, hybrid			
	802.11g	2.4 GHz	Unlicensed	Yes	Up to 54 Mbps	38 m–140 m	Ad-hoc, star, mesh, hybrid			
	802.11ac	5 GHz	Unlicensed	Yes	Up to 3466 Mbps	35 m–120 m	Ad-hoc, star, mesh, hybrid			
Bluetooth 5 [9–12]	802.15.1	2.4 GHz	Unlicensed	Yes	Up to 2 Mbps	40 m–200 m	Ad-hoc, piconet	3 ms	Energy-efficient	Low data rate
ZigBee [13–15]	802.15.4	2.4 GHz	Unlicensed	Yes	250 Kbps	10 m–100 m	Ad-hoc, star, mesh, tree, cluster	15 ms	Low cost	Low data rate
WiMAX [16–18]	802.16a	2 to 11 GHz	Licensed	Yes	Up to 75 Mbps	Up to 48 km	Wide-area wireless backhaul	30 ms	High throughput	Interference issues
LTE [19–22]	LTE	Up to 20 MHz	Licensed	Yes	Up to 300 Mbps	Up to 100 km	Flat, IP based	5 ms	High bandwidth	Expensive
5G [23–29]	5G (eMBB)	28 GHz	Licensed	Yes	Up to 20 Gbps	Wide Area	IP based	1 ms	High data rate	Expensive
Satellite [30,31]	Satellite	Up to 40 GHz	Licensed	Yes	Up to 1 Gbps	World Wide	-	550 ms	Wide coverage	High delay and high cost

3.1.1. Wi-Fi (IEEE 802.11)

Wi-Fi, or Wireless Fidelity, is a short-range communication technology that consists of a set of standards for designing WLAN (Wireless Local Area Network) in the following radio bands: 2.4 GHz, 3.6 GHz, 5 GHz, and 60 GHz. To provide the throughput required for large-sized data transmissions like videos and images, variants of IEEE 802.11a/b/g/n/ac, can be the top choice for many FANET applications. The transmission range of a traditional Wi-Fi system is around 100 m. However, the transmission range can be extended up to several kilometers with the help of ad-hoc networking between the UAV. In a typical 802.11 network, clients, on the basis of announcement by access points (APs), discover and associate with wireless local area networks (WLANs) [7]. In this manner, a device behaves either as a client or as an AP. Moreover, these roles are assigned dynamically and could even be executed simultaneously by the same device. In [8], the authors investigated the performance measurement in terms of throughput, RSSI, and distance of a wireless link from UAV to GS. It is also suggested that applying an 802.11a wireless link for UAV and GS is useful for a UAV-based networking.

3.1.2. Bluetooth (IEEE 802.15.1)

Bluetooth operates in the unlicensed 2.4 GHz frequency band with a communication range of 10 to 200 m. Bluetooth technology can be found in multiple versions with a data rate ranging from 1 to 3 Mbps. However, the maximum data rate can be reached up to 24 Mbps. The Bluetooth Special Interest Group (SIG) proposed Bluetooth Low Energy (BLE) in the Bluetooth 4.0 specifications. Bluetooth 5 [9] is the latest version of the Bluetooth core specifications. Bluetooth 5 mainly focuses on improvement in speed, transmission range, energy efficiency, and co-existence with other short-range technologies. Additionally, Bluetooth 5 is also able to broadcast richer data, which extends far beyond the location information and includes multimedia and URL's files. Considering the significant improvements, it seems like Bluetooth 5 is, in a true sense, a potential candidate for the low-cost and low-power deployment of future FANET. Asghar et al. [10] proposed a hybrid scheme using 802.15.1 for FANET. In the proposed scheme, data transmission between the member UAV was done by 802.15.1. In [11], the authors described the implementation of a wireless network platform using Bluetooth technology on a round robin scheduling algorithm. The established Bluetooth-based UAV network includes one master UAV station and up to seven slave UAV stations. The results authenticate that the proposed platform provides reliable communication with low-computational power. Hoffmann et al. [12] conducted a testbed on the basis of the same technology for multiple UAV. The testbed surfaces the way for practical employments of UAV network using Bluetooth technology.

3.1.3. ZigBee (IEEE 802.15.4)

ZigBee technology is normally used in the low data rate applications that require high battery life and secure networking. It covers a distance ranging from 10 to 100 m. Compared to Bluetooth and Wi-Fi, it is less expensive and simpler. It operates in the 2.4 GHz frequency range with a data rate of 250 kbps. It has 16 channels and each requires 5 MHz of bandwidth. In [13], the authors presented indoor localization of a quadcopter using ZigBee. The results authenticate that the localization system using ZigBee is viable, effective, and easy deployable. Jiang et al. [14] analyzed UAV landings, where ZigBee was tested for communication and position estimation. The study demonstrates, as an outcome, a significant decrease in error and an accurate estimation of the position. Zafar et al. [15] proposed a hybrid scheme where ZigBee was used for intra-cluster communication. The simulation results reveal that ZigBee can be a potential candidate for low-data-rate FANET applications.

3.2. Long-Range Communication Technologies

Long-range communication technologies can be used as a backhaul between two sights to deliver data communication services over a large distance. These technologies may also be useful

for aerial vehicles to enable them communicate directly with each other (through U2U) and with the fixed infrastructure (using U2I). In the following subsection, capabilities of the existing short-range communication technologies will be examined.

3.2.1. WiMAX (IEEE 802.16)

WiMAX is a technology standard that aims to deliver broadband access over long distances in a variety of ways, ranging from point-to-point links to full mobile cellular type access. This technology is designed to accommodate both fixed and mobile broadband applications. It supports a data rate of up to 75 Mbps for the fixed applications (20 to 30 Mbps per subscriber) and for mobile applications, the data rate stretches to 30 Mbps (3 to 5 Mbps per subscriber) [16]. WiMAX is developed to provide high-quality voice and video streaming while maintaining the desired quality of service (QoS). With reference to UAV, WiMAX is considered the most appropriate technology for UAV-based rescue systems in hostile environments [17]. In [18], the authors provided methodology for network planning in terms of the number and position of UAVs. Simulation results demonstrate the likelihood of calculating the dimensions, i.e., the position and altitude, of each UAV and guarantee a certain QoS using WiMAX.

3.2.2. Long-Term Evolution (LTE)

LTE offers secure wireless connectivity, mobility, and a high data rate, which can considerably enhance control and safety beyond visual line-of-sight (LOS) use cases. LTE is optimized for IP with scalable bandwidths of 20 MHz, 15 MHz, 10 MHz, and less than 5 MHz. It supports both frequency division duplexing (FDD) (paired) and time division duplexing (TDD) (unpaired) spectrums. The optimum cell size is 5 km, even though it can attain a reasonable performance within 30 km and provide an acceptable performance up to 100 km. Not surprisingly, utilizing an established LTE network for UAVs have seen a surge of activities in the last three years. In [19], low altitude UAVs are connected through LTE. The work points out that the existing mobile LTE network can boost network performance in UAV-based disaster response scenarios. Qazi et al. [20] demonstrated the effectiveness of UAV-based real-time video streaming and surveillance over the 4G-LTE system. The study investigates the performance of the network in relation to the physical aspects of wireless propagation using Network Simulator-3. Similarly, in [21], the authors considered two scenarios in which UAVs either work as a base station transmitting in the downlink or as user equipment transmitting in the uplink using LTE network. The work spotlights that the present LTE network does require important modifications for a smooth integration of LTE-enabled UAVs. Nguyen et al. [22] also investigated the performance of aerial radio connectivity in a rural LTE network. The results show that some practical and relatively low complex mitigation schemes of interference have good potential for deployment in a rural LTE network.

3.2.3. Fifth Generation (5G)

Following 2G (GSM), 3G (UMTS), and 4G (LTE/WiMAX), fifth generation, or 5G, is the latest generation of cellular mobile communication. Its prominent features include a high data rate, reduced latency, energy saving, enhanced system capacity, and ubiquitous connectivity. The International Telecommunication Union (ITU) foresees the launch of 5G mobile networks by 2020. Such systems will have, per user, a speed of 100 GB/s with a capacity stretching as much as 1000 times [23]. Owing to such features, 5G technology is poised to play a critical role in UAV communication systems and, thus, pave ways for novel applications. For instance, in the case of UAVs in a 5G environment, less dynamic network segments of the FANET architecture could be interconnected with the core network. This will ease the provision of services such as surveillance multimedia streaming. The requirement of backhauling for seamless connectivity, however, still eclipses the inherent clarity of UAVs in a 5G network [24–26]. Actions such as connecting UAVs with a base band unit (BBU) and involving a macro-cell base station (MBS) do ensure fulfillment of the backhauling requirement.

Further, the connectivity does vary depending on the factors related to service provision and network planning. In [27], the authors proposed a multi-layer hierarchical architecture with distributed features that smoothly enables the integration of UAVs with next-generation wireless communication networks. Moreover, authors in [28] provided an overview on some of the recent research endeavors in UAV communication systems that addresses 5G techniques from the perspectives of the physical layer, the network layer, and joint communication, computing, and caching. The case where UAVs carry base stations (BSs) for providing 5G network connectivity in rural settings is explored in [29].

3.2.4. Satellite Communication (SATCOM)

SATCOM is used for sending electromagnetic signals from ground stations to space stations, or satellites, and vice versa. In SATCOM, various frequency bands are used by different satellites. C-Bands, which are still tied up with some systems, utilize an uplink band of 6 GHz and a downlink band of 4 GHz. X-Bands, on the other hand, which are normally used by the military and governmental systems, use 8 GHz for uplink and 7 GHz for downlink. The so-called Ku-Bands, operates on 14 GHz for uplink and 11–12 GHz for downlink. Moreover, these bands are also becoming saturated; therefore, Ka-Bands are evolving to be implemented with the time. Ka-Bands work on a 30 GHz uplink and a 20 GHz downlink. In [30], the authors studied key specifications such as satellite downlink, UAV uplink transmission power, and the image transmission rate. It was specified that satellite relay achieves a larger overlay range and provides a large-coverage image transmission with a high image-quality. Additionally, in [31], the authors highlighted the main problems behind applying SATCOM for live image and video transmissions with the help of micro and mini UAS. The two major issues are inadequate bandwidth and a high cost of data transmission.

From the above discussion, it can be concluded that short-range communication technologies like Bluetooth, ZigBee, or Wi-Fi may be considered for medium-range FANET applications based on the range and throughput requirements. However, if the coverage area is large and these short-range technologies are not able to support the required throughput demands, long-range communication technologies such as WiMAX, LTE, 5G, and SATCOM may be far more appropriate.

In summary, based on the transmission characteristics, we consider short-range communication technologies for our proposed scheme. There are valid reasons for choosing these technologies: they operate in the unlicensed spectrum; they do not require strict LOS; and they offer a reasonable data rate and coverage. Moreover, they can be easily integrated with small UAV. Moreover, due to the significant improvements in speed, power consumption, capacity, and coverage, Bluetooth 5 is the best option amongst the aforementioned short-range wireless technologies. However, in order to utilize the best features, Wi-Fi is integrated with Bluetooth 5 in our proposed approach.

4. Resource Management and Energy Efficiency

Limited on-board energy is one of the major limitations that hinders the potency of a UAV system. However, consideration of two important factors, i.e., energy-efficient deployment and energy-efficient operations, can pave ways for resolution. Energy-efficient deployment means interruption-less utilization of power reserves. Energy-efficient operations stipulate measures aimed at achieving the objectives with minimal on-board power. One way to achieve energy-aware deployment is to harness inter-UAV cooperation such that, at any given time, only one UAV can abandon its mission to top up its power reserves [32]. In the literature, the topic of energy efficiency and resource management has been consulted from various angles. For instance, in [33], the authors propose an analytical framework aimed at finding the optimal trajectory, which can minimize the energy consumption of a fixed-wing UAV. Energy efficiency is addressed by a scheduling framework for cooperative UAVs communication in [34]. In [35], the authors studied the energy efficiency of drones in target tracking scenarios by adjusting the number of active drones. Besides, a drone-assisted strategy is proposed in [36], which entails harnessing LoRaWAN architecture and the smart deployment of drones for energy conservation. In [37], an energy-efficient system is proposed for the drones to collect

sensor data. The issues of data dissemination and energy conservation are addressed by a firefly optimization algorithm. Supported by acceptable modelling, simulation, and numerical results, a self-adaptive energy-efficient operation in UAV-assisted public safety networks is suggested in [38]. The authors in [39] investigated, for small UAVs, energy harvesting from vibrations and solar sources. Optimization of transmission time in a user–UAV communication set-up is proposed in [40]. This resulted in the maximization of the user’s minimum throughput. Utilization of an antenna array on UAVs is investigated in [41] in order to improve SNR and reduce transmission power. A flying access point equipped with an energy harvesting mechanism is investigated in [42] with the objective of optimal resource allocation. UAVs servicing ground users face bandwidth and flight-time optimization problems, which is addressed in [43]. A resource allocation framework is proposed in [44] that enables cache-enabled UAVs to effectively service users over licensed and unlicensed bands.

Undoubtedly, the area of energy efficiency and resource management demand insightful consideration since it affects the performance of FANET systems in terms of throughput and end-to-end delay.

5. The Proposed Hybrid Scheme

In the proposed hybrid scheme, two key short-range communication technologies, Wi-Fi (802.11) and Bluetooth (802.15.1), are interconnected. The major reason behind designing such a scheme is to utilize the strong features of both of the technologies so as to attain features such as low-cost, low-power, high range, and high speed. We consider multiple representative classes of UAV: Bluetooth slave (BS), Bluetooth master (BM), and Gateway (GW). At first, with the help of beacon signals, discovery function is performed to establish a connection between the associated UAV. Once the connection is established, data are transferred among the BM-UAV and BS-UAV using 802.15.1, while the data communication between the GW-UAV and GS is established through 802.11. Further details of the proposed scheme are given in the following subsections.

5.1. UAV Station Model

All UAVs carry a global positioning system (GPS) and an inertial measurement unit (IMU) to uninterruptedly monitor the position and location of neighboring UAVs in the proposed scheme. The position and location of UAV are then updated in the medium access control (MAC) table.

5.2. Mobility Model

Mobility models are commonly used to analyze newly designed schemes or protocols in both cellular and ad-hoc networks. In FANET, the choice of suitable mobility model is essential to obtain results with maximum precision and conformity due to the agile motion of UAVs. In this context, mobility models are classified into five classes: purely randomized, time-dependent, path planned, group, and hybrid [45]. In order to patrol a specific region, there are many situations where a group of UAVs move together following a common point in the FANET system. For this reason, we used reference point group mobility (RPGM) [46] to simulate a group of UAVs in the proposed scheme. In the RPGM model, UAVs are grouped to achieve a collective task by moving together around a logical reference point, i.e., a group leader. The behavior of the reference point’s motion defines the entire group’s motion including location, altitude, speed, and direction. Usually, UAVs are uniformly distributed within the geographic scope of a group. The group leader follows a random waypoint (RWP) mobility model, and other UAVs will be moving around the center with their own mobility pattern (see Figure 4b) [47]. This mobility model has numerous variants, such as column (CLMN) [47], nomadic community (NC) [47], and purse (PRS).

In the proposed scheme, BM-UAV of each group is located at the reference point. The altitude, speed, and direction of BS-UAV depend on such a reference point. Moreover, the location of BS-UAVs is updated according to the BM-UAV.

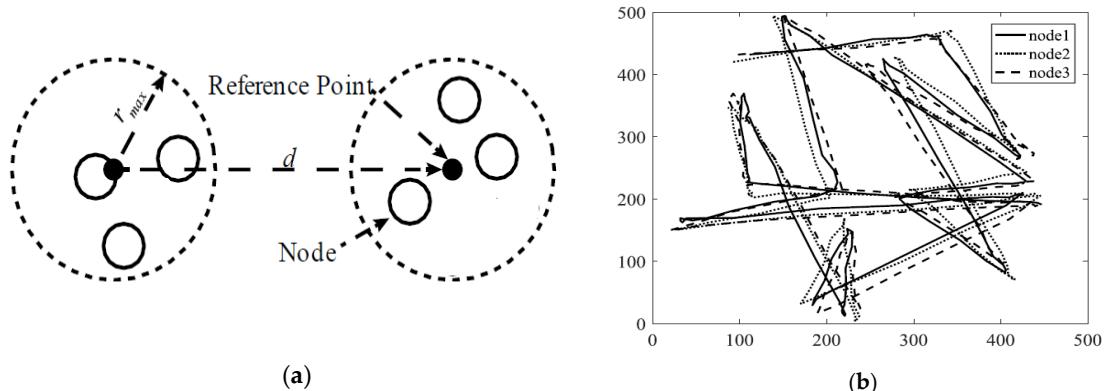


Figure 4. Mobility models: (a) nomadic community (NC) with three nodes, where the nodes have a maximum distance r_{max} to move away from the reference point and the reference point moves through the simulation area a random distance d following a random waypoint (RWP) model; (b) mobility pattern of one group (i.e., three mobile nodes) using reference point group mobility (RPGM).

5.3. Propagation Model

In our proposed scheme, the FANET system consists of n UAVs, where $n \geq 2$, and each UAV is equipped with both the radios i.e., 802.15.4 and 802.11 with dual-band capability. Two techniques, path loss and fast fading, can be used to model a wireless link between two UAVs. Attenuation of the radiated power affects path loss. On the other hand, fading results due to the multipath effect of low altitude UAV propagation. Path loss can be expressed mathematically as follows between the transmitter and receiver [48]:

$$P_l(dB) = \begin{cases} 20 \log_{10} \left(\frac{4\pi d}{\lambda} \right) & \text{if } d \leq d_0 \\ 20 \log_{10} \left(\frac{4\pi d_0}{\lambda} \right) + 40 \log_{10} \left(\frac{d}{d_0} \right) & \text{if } d > d_0 \end{cases} \quad (1)$$

where λ is the wavelength of the transmitted signal and can be represented as the ratio of the speed of light c to the carrier signal f : $\lambda = c/f$, while d is the distance between transmitting and receiving UAVs, and d_0 , i.e., 8 m, is the length of the LOS. However, we employ a free space propagation model in our proposed scheme. The reason for selecting this model is that we assume a clear LOS between sender and receiver UAVs. The well-known Friis equation [49] is given, as it can express this model mathematically:

$$P_r = \frac{P_t G_t(\theta_t, \phi_t) G_r(\theta_r, \phi_r) \lambda^2}{(4\pi d)^2} \quad (2)$$

$$P_r = P_t G_t(\theta_t, \phi_t) G_r(\theta_r, \phi_r) \left(\frac{\lambda}{4\pi d} \right)^2 \quad (3)$$

where the alphabet P represents power and G denotes gain; the subscripts t and r stand for “transmitted” and “received,” respectively. Moreover, the symbols θ_t and ϕ_t are used for elevation angle (range: 0–180°) and azimuth angle (range: 0–360°), respectively. Further, λ is the wavelength of the transmitted signal, and d equates to the distance between the transmitting and receiving antenna. We set the receiving antenna gain $G_r(\theta_r, \phi_r) = 1$ in Equation (3), and we can write received power as

$$P_r = P_t G_t(\theta_t, \phi_t) \left(\frac{\lambda}{4\pi d} \right)^2. \quad (4)$$

Taking logarithm on both sides of Equation (4), we have

$$10 \log_{10} (P_r) = 10 \log_{10}(P_t) + 10 \log_{10}(G_t(\theta_t, \phi_t)) - 20 \log_{10} \left(\frac{4\pi d}{\lambda} \right) \quad (5)$$

$$P_r (dB_w) = P_t (dB_w) + G_t (dB_i) - P_l (dB) \quad (6)$$

where $P_l (dB)$ represents the free-space path loss and is given by

$$P_l (dB) = 20 \log_{10} \left(\frac{4\pi d}{\lambda} \right).$$

5.4. Application Scenarios and Architecture

In this subsection, we describe the possibility of integrating our proposed scheme with the FANET application scenario. In order to deploy FANET in different application scenarios, the UAV can be equipped with cameras, data storage, sensors, an IMU, a GPS unit, and other embedded devices such as a flight controller, on-board processing units, and short-ranged radio transceivers (i.e., Bluetooth and Wi-Fi). In surveillance applications, such as patrolling a specific region, UAVs play an important role in minimizing human involvement. These missions involve information collection from battlefield to earthquake regions. Surveillance tasks may include collecting images and video streaming from the mission area. In our proposed system, the mission area is divided into zones, where a multi-layer UAV ad-hoc architecture is deployed as shown in Figure 5. In this architecture, multiple groups of heterogeneous UAVs are placed in different zones. The lower layer is used for communication between the BM-UAV and GW-UAV, while the upper layer is concerned with communication between the BM-UAV and BS-UAV. The GW-UAV will be further connected with the ground station. The ground station only holds the information, which is exclusively routed to it, thus reducing the computation costs on the ground station.

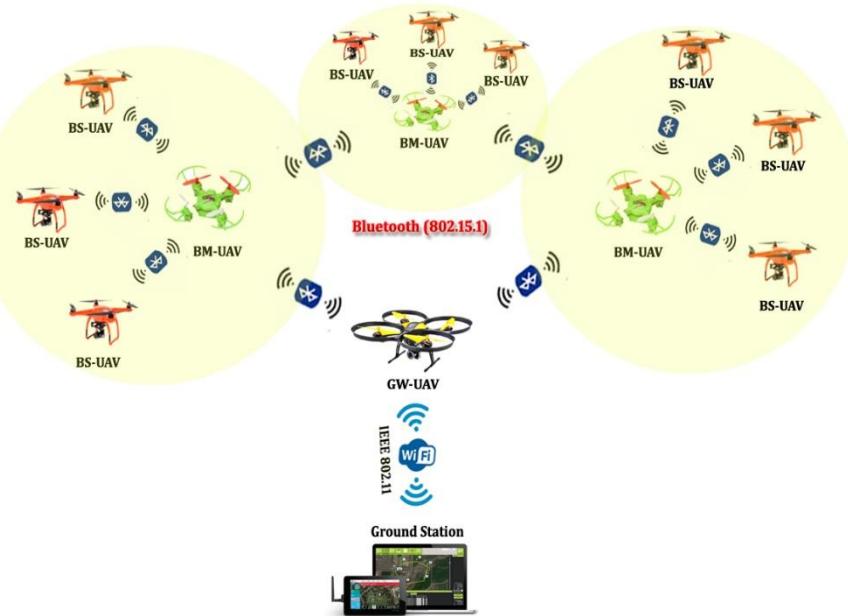


Figure 5. Sample topology for multi-layer FANET.

6. Simulation Setup and Performance Metrics

6.1. Simulation Setup

The proposed scheme is designed and implemented by the OPNET. The node model of the MANET node is modified for GW-UAV in the node editor, as shown in Figure 6. To register the MAC addresses, the Bluetooth connection is established through the address resolution process (ARP) model in the GW-UAV. Figure 7 shows the node model for both BM-UAV and BS-UAV. Even though the node models of BS-UAV and BM-UAV seem similar, they act differently. Their working is implemented in the function block of the process model editor. The BM-UAV module process schedules asynchronous

connectionless (ACL) and synchronous-connection oriented (SCO) transmissions to the BS-UAV. BS-UAV transmit data only if it receives a packet from its BM-UAV.

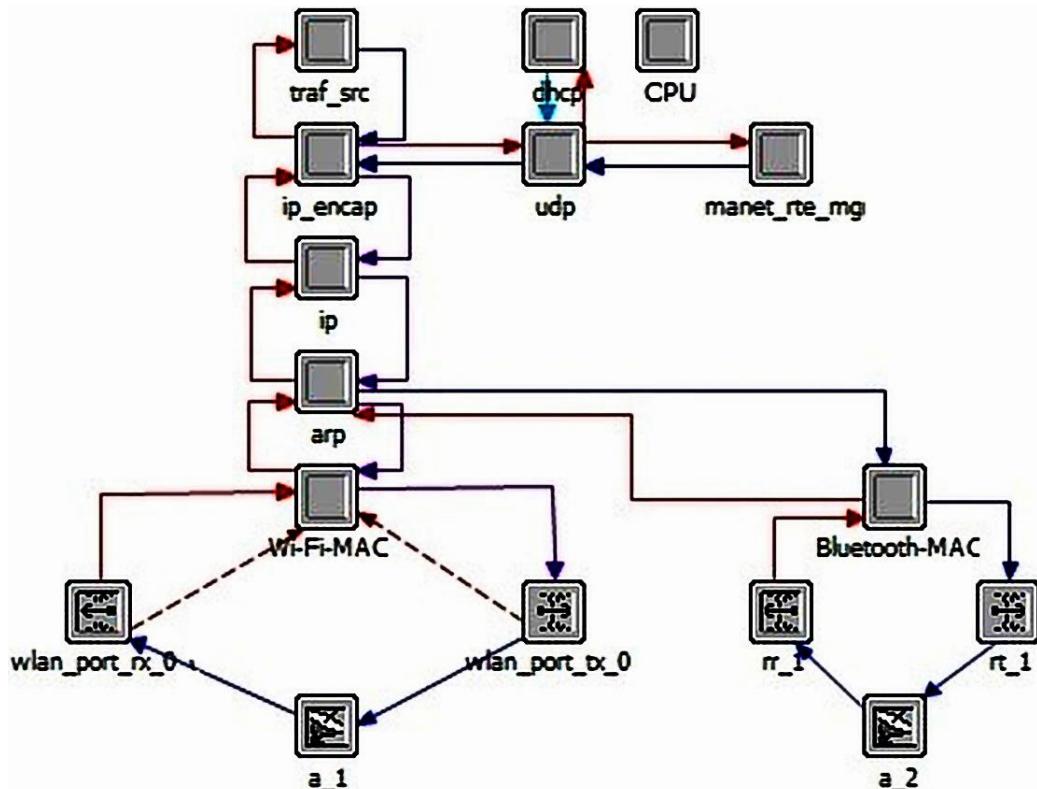


Figure 6. Node model for Gateway-UAV (GW-UAV).

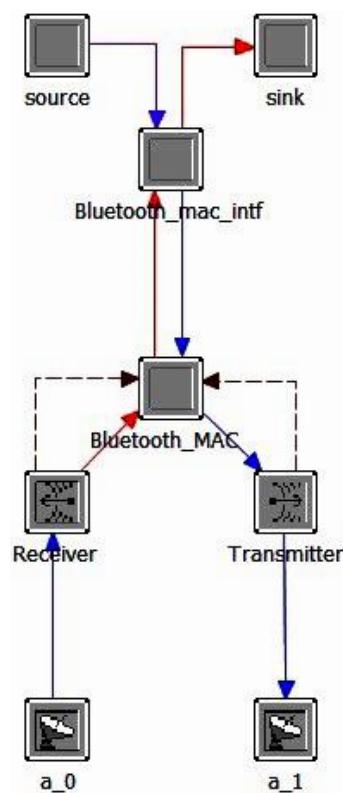


Figure 7. Node model for BM-UAV and BS-UAV.

6.2. Performance Metrics

6.2.1. Throughput (bits/s)

Throughput is the amount of successful transmissions over a communication link in a particular unit of time. Higher throughput is the requirement and characteristic of any network. In the proposed scheme, average throughput is measured across the network. A throughput variation graph using the OPNET is also presented in order to demonstrate the consistency of the proposed approach. Mathematically, throughput can be determined using the following equation:

$$\text{Throughput} = \frac{N \times S \times 8}{T} \quad (7)$$

where N is assumed to be the number of successful packets transferred, S is the size of the packet, and T is the time duration.

6.2.2. Delay (s)

Delay refers to the amount of time a signal takes across the network from a source UAV to a destination UAV. Delay is a key factor for measuring the performance of a communication network. The model aims to limit the delay by facilitating direct communication between the UAV and with the ground station. End-to-end delays are comprised of processing, queuing, and transmission delay of the link in a network. Mathematically, the average end-to-end delay can be shown with the help of Equation (8):

$$D_{\text{end-to-end}} = \sum_{t=1}^N (T_t + R_t + B_t + P_{rt}) \quad (8)$$

where T_t is Transmission time, R_t is retransmission time, B_t is buffer time, and P_{rt} is processing time.

7. Results and Analysis

Figure 8a,b and Figure 9a,b illustrate the performance of FANET in terms of throughput and end-to-end delay under RPGM and RWP mobility models, respectively. In the graphs, the x-axis indicates the simulation time, in seconds, and the y-axis specifies the throughput and delay, in bits per second. The proposed scheme was tested on an area of 1×1 km with 42 UAVs. Seven zones were considered with each comprised of 6 UAVs. Five among the six UAVs were equipped with 802.15.1, while one UAV carried both 802.11 and 802.15.1 radios. Simulation were carried out for two different scenarios in order to emulate a group of UAVs under both RPGM and RWP. Table 2 depicts the remaining parameters along with respective values. From our simulation results, it is evident that the desired QoS requirements are met by the proposed scheme. That is, it provides significant performance improvements compared to IEEE 802.15.1 in terms of throughput and end-to-end delay (Figure 8a,b). At the beginning stage of the connection, there is no great difference between Bluetooth and the proposed scheme. The data transmission speed of Wi-Fi is relatively low due to the long connection establishment time of Wi-Fi technology. As a result, the proposed hybrid overcomes the shortcomings of a long initial connection setup time of Wi-Fi as well as a low transmission speed of Bluetooth. Moreover, a slight decline was observed in network performance, particularly in reference to throughput and end-to-end delay under RWP mobility model (Figure 9a,b). It can be observed from Figure 8a,b that RPGM provides higher throughput and minimum end-to-end delay for both IEEE 802.15.1 and the proposed hybrid scheme compared to × RWP mobility model (Figure 9a,b). This is because the movement of UAVs under the RPGM model is governed by defining a specific trajectory around a reference point, i.e., the group leader. Thus, the network is in a fully connected state most of the time.

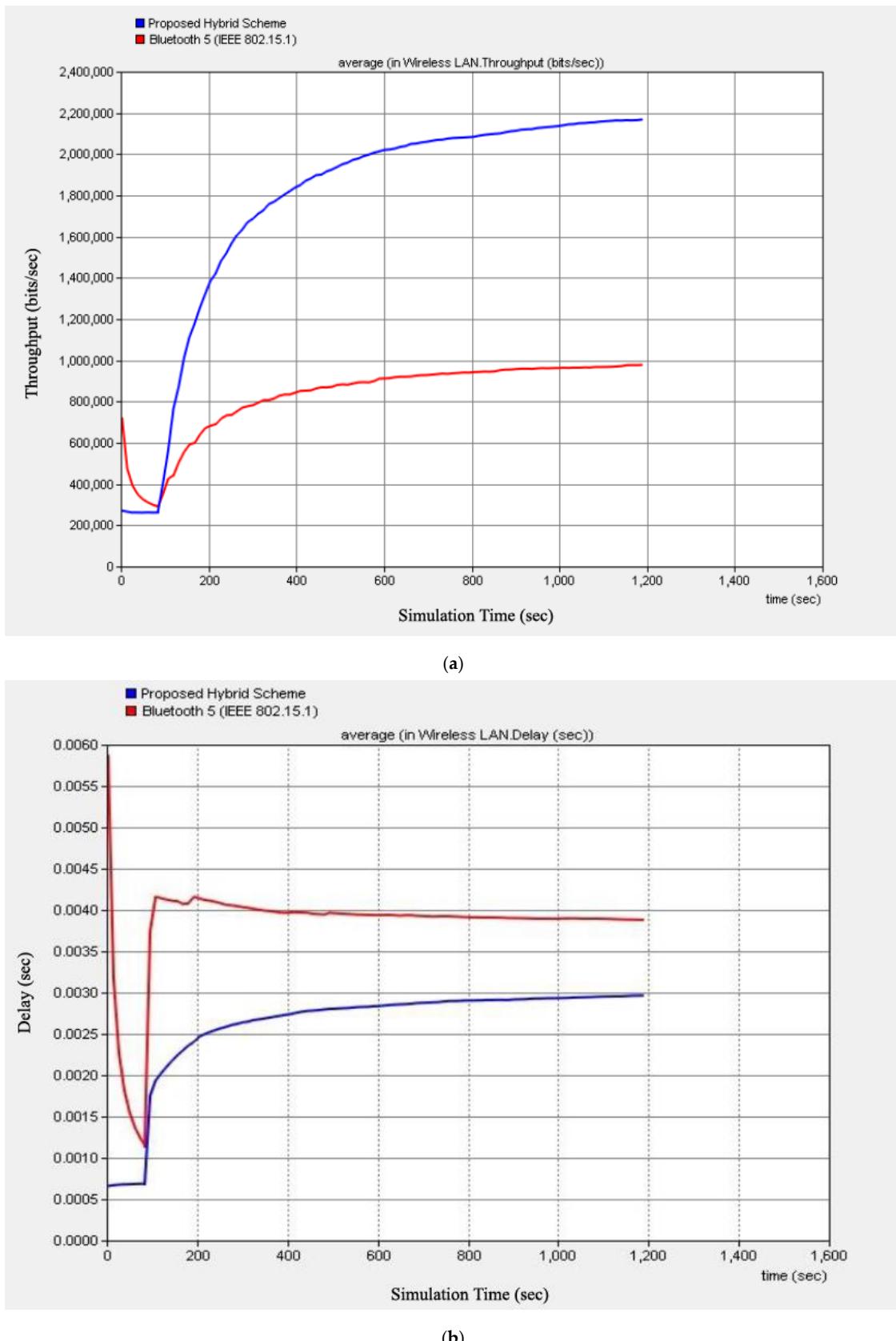


Figure 8. (a) Throughput vs. simulation time under RPGM. (b) End-to-end delay vs. simulation time under RPGM.

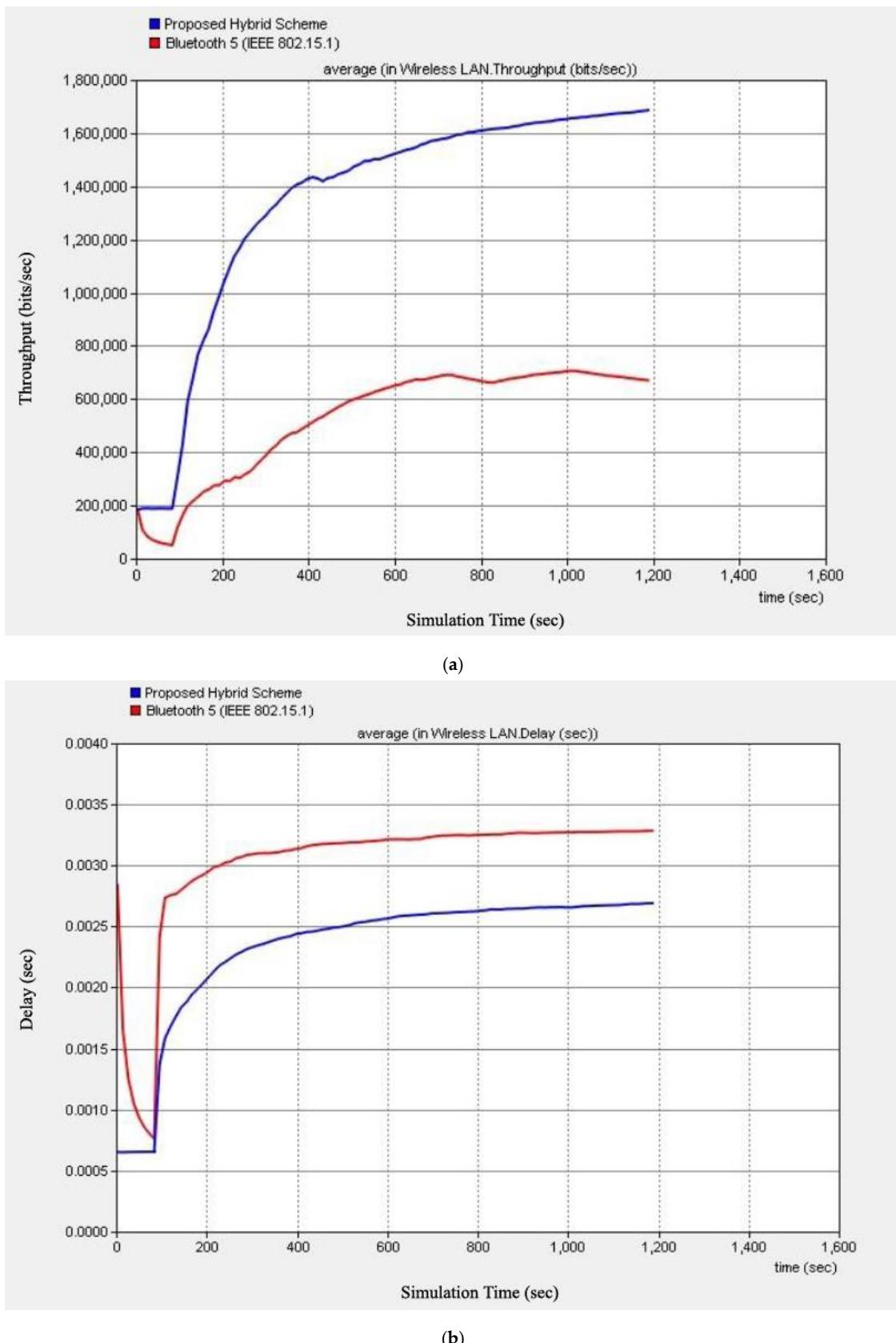


Figure 9. (a) Throughput vs. simulation time under RWP. (b) End-to-end delay vs. simulation time under RPW.

Table 2. Simulation parameters.

Parameter	Value
Area Dimensions	1 km × 1 km
Altitude of UAVs	25 m
Number of UAVs	42
Directional Gain	10 dBi
Frequency	2.4 GHz
Data Rates	2 Mbps, 11 Mbps
Packet Interval (s)	Exponential (1)
Packet Size(byte)	1024
Simulation Time	1600 s
Node Type	Mobile
Mobility Model	RPGM, RWP
Speed of UAVs	15 m/s
Transmission Power	−97 dBm

8. Conclusions and Future Remarks

In order to pave ways for smart services, FANET needs to promise uninterrupted wireless connectivity in a ubiquitous fashion. An apparent outcome is the emergence of new possibilities in the form of rapid, efficient, and low-cost deployment of future flying ad-hoc networks. In this context, we explored different architectures and wireless technologies that can be used in the communication links between UAV and the ground station. We also proposed a hybrid wireless communication scheme in order to employ the low-power consumption feature of Bluetooth (802.15.1) and the high data transmission rate capability of the Wi-Fi technology (802.11). The proposed scheme was designed, implemented, and simulated with the OPNET. The simulation results, in terms of throughput and delay, demonstrated the viability of our scheme for FANET. Our research contribution is hoped to enable researchers and engineers to explore more avenues for coming forth with improved UAV wireless communication systems. For instance, the proposed hybrid scheme does not cater to the backhaul link for the FANET system. Existing FANET system(s) mainly rely on simple point-to-point communication over the unlicensed band (e.g., ISM 2.4 GHz). Such an unlicensed band is of a low data rate and operates over a limited range. The number of UAVs and their associated applications are poised to grow in the near future. This further stresses the need to develop new hybrid schemes such as WiMAX, LTE, and 5G at the backhaul to enable enhanced UAV–ground communications.

Author Contributions: Conceptualization, M.A.K.; Methodology and Implementation, M.A.K. and I.M.Q.; Simulation, M.A.K.; Validation, M.A.K.; Data Curation, I.M.Q.; Writing—Original Draft Preparation, F.K. and M.A.K.; Writing—Review & Editing, F.K. and M.A.K.; Supervision, I.M.Q.

Funding: This research received no external funding.

Acknowledgments: A part of this paper was published in Proceedings of the 2nd International Conference on Future Networks and Distributed Systems (ICFNDS '18), Amman, Jordan, 26–27 June 2018.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Zeng, Y.; Zhang, R.; Teng, J.L. Wireless communications with unmanned aerial vehicles: Opportunities and challenges. *IEEE Commun. Mag.* **2016**, *54*, 36–42. [[CrossRef](#)]
- Hayat, S.; Yanmaz, E.; Muzaffar, R. Survey on Unmanned Aerial Vehicle Networks for Civil Applications: A Communications Viewpoint. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 2624–2661. [[CrossRef](#)]
- Sahingoz, O.K. Networking models in flying Ad-hoc networks (FANETs): Concepts and challenges. *J. Intell. Robot. Syst.* **2014**, *74*, 513–527. [[CrossRef](#)]
- Sharma, V.; Kumar, R. Cooperative frameworks and network models for flying ad hoc networks: A survey. *Concurr. Comput. Pract. Exp.* **2017**, *29*, e3931. [[CrossRef](#)]

5. Li, J.; Zhou, Y.; Lamont, L. Communication architectures and protocols for networking unmanned aerial vehicles. In Proceedings of the 2013 IEEE Globecom Workshops (GC Wkshps), Atlanta, GA, USA, 9–13 December 2013; pp. 1415–1420.
6. Khan, M.A.; Qureshi, I.M.; Safi, A.; Khan, I.U. Flying Ad-Hoc Networks (FANETs): A Review of Communication architectures, and Routing protocols. In Proceedings of the 2017 First International Conference on Latest Trends in Electrical Engineering and Computing Technologies (INTELLECT), Karachi, Pakistan, 15–16 November 2017; pp. 692–699.
7. Joh, H.; Yang, I.; Ryoo, I. The internet of everything based on energy efficient P2P transmission technology with Bluetooth low energy. *Peer-to-Peer Netw. Appl.* **2015**, *9*, 520–528. [[CrossRef](#)]
8. Cheng, C.-M.; Hsiao, P.-H.; Kung, H.T.; Vlah, D. Performance measurement of 802.11a wireless links from UAV to ground nodes with various antenna orientations. In Proceedings of the 15th International Conference on Computer Communications and Networks, Arlington, VA, USA, 9–11 October 2006; pp. 303–308.
9. Bluetooth Core Specification, Bluetooth Special Interest Group (SIG). 2016. Available online: <https://www.bluetooth.com/specifications/bluetooth-core-specification> (accessed on 6 October 2018).
10. Khan, M.A.; Khan, I.U.; Qureshi, I.M.; Alam, M.K.; Shah, S.B.; Shafiq, M. Deployment of reliable, simple, and cost-effective medium access control protocols for multi-layer flying ad-hoc networks. In Proceedings of the 2nd International Conference on Future Networks and Distributed Systems (ICFNDS '18), Amman, Jordan, 26–27 June 2018; p. 49.
11. Afonso, J.A.; Coelho, E.T.; Carvalhal, P.; Ferreira, M.J.; Santos, C.; Silva, L.F.; Almeida, H. Distributed sensing and actuation over Bluetooth for unmanned air vehicles. In Proceedings of the International Conference on Robotics and Automation, Orlando, FL, USA, 15–19 May 2006.
12. Hoffmann, G.; Rajnarayan, D.G.; Waslander, S.L.; Dostal, D.; Jang, J.S.; Tomlin, C.J. The Stanford testbed of autonomous rotorcraft for multi agent control (STARMAC). In Proceedings of the 23rd Digital Avionics Systems Conference (IEEE Cat. No.04CH37576), Salt Lake City, UT, USA, 28 October 2004.
13. Yut, L.; Fei, Q.; Geng, Q. Combining Zigbee and inertial sensors for quadrotor UAV indoor localization. In Proceedings of the 2013 10th IEEE International Conference on Control and Automation (ICCA), Hangzhou, China, 12–14 June 2013; pp. 1912–1916.
14. Jiang, Y.; Cao, J.; Du, Y. Unmanned air vehicle landing based on Zigbee and vision guidance. In Proceedings of the 2006 6th World Congress on Intelligent Control and Automation, Dalian, China, 21–23 June 2006; pp. 10310–10314.
15. Zafar, W.; Khan, B.M. A reliable, delay bounded and less complex communication protocol for multicluster FANETs. *Digit. Commun. Netw.* **2017**, *3*, 30–38. [[CrossRef](#)]
16. Banerji, S.; Chowdhury, R.S. Wi-Fi & WiMAX: A comparative study. *Indian J. Eng.* **2013**, *2*.
17. Rahman, M.A. Enabling drone communications with WiMAX Technology. In Proceedings of the IISA 2014, The 5th International Conference on Information, Intelligence, Systems and Applications, Chania, Greece, 7–9 July 2014; pp. 323–328.
18. Dalmasso, I.; Galletti, I.; Giuliano, R.; Mazzenga, F. WiMAX networks for emergency management based on UAVs. In Proceedings of the 2012 IEEE First AECC European Conference on Satellite Telecommunications (ESTEL), Rome, Italy, 2–5 October 2012; pp. 1–6.
19. Lin, X.; Yajnanarayana, V.; Muruganathan, S.; Gao, S.; Asplund, H.; Maattanen, H.; Bergstrom, M.; Euler, S.; Wang, Y. The Sky Is Not the Limit: LTE for Unmanned Aerial Vehicles. *IEEE Commun. Mag.* **2018**, *56*, 204–221. [[CrossRef](#)]
20. Qazi, S.; Siddiqui, A.S.; Wagan, A.I. UAV based real time video surveillance over 4G LTE. In Proceedings of the 2015 International Conference on Open Source Systems & Technologies (ICOSSST), Lahore, Pakistan, 17–19 December 2015; pp. 141–145.
21. Van der Bergh, B.; Chiumento, A.; Pollin, S. LTE in the sky: Trading off propagation benefits with interference costs for aerial nodes. *IEEE Commun. Mag.* **2016**, *54*, 44–50. [[CrossRef](#)]
22. Nguyen, H.C.; Amorim, R.; Wigard, J.; Kovács, I.Z.; Sørensen, T.B.; Mogensen, P.E. How to Ensure Reliable Connectivity for Aerial Vehicles Over Cellular Networks. *IEEE Access* **2018**, *6*, 12304–12317. [[CrossRef](#)]
23. Jiang, D.; Liu, G. *An Overview of 5G Requirements—5G Mobile Communications*; Springer: Cham, Switzerland, 2017; pp. 3–26. [[CrossRef](#)]
24. Bor-Yaliniz, I.; Yanikomeroglu, H. The new frontier in ran heterogeneity: Multi-tier drone-cells. *IEEE Commun. Mag.* **2016**, *54*, 48–55. [[CrossRef](#)]

25. Dong, Y.; Hassan, M.; Cheng, J.; Hossain, M.; Leung, V. An edge computing empowered radio access network with uav-mounted fso fronthaul and backhaul: Key challenges and approaches. *arXiv*, 2018; arXiv:1803.06381. [CrossRef]
26. Sharma, V.; Song, F.; You, I.; Chao, H.-C. Efficient management and fast handovers in software defined wireless networks using uavs. *IEEE Netw.* **2017**, *31*, 78–85. [CrossRef]
27. Huo, Y.; Dong, X.; Lu, T.; Xu, W.; Yuen, M. Distributed and multilayer UAV network for the next-generation wireless communication. *arXiv*, 2018; arXiv:1805.01534.
28. Li, B.; Fei, Z.; Zhang, Y. UAV communications for 5G and beyond: Recent advances and future trends. *IEEE Internet Things J.* **2018**, in press. [CrossRef]
29. Amorosi, L.; Chiaraviglio, L.; D’Andreagiovanni, F.; Blefari-Melazzi, N. Energy-efficient mission planning of UAVs for 5G coverage in rural zones. In Proceedings of the 2018 IEEE International Conference on Environmental Engineering (EE), Milan, Italy, 12–14 March 2018; pp. 1–9.
30. Ma, D.; Yang, S. UAV image transmission system based on satellite relay. In Proceedings of the ICMMT 4th International Conference on, Proceedings Microwave and Millimeter Wave Technology, Nanjing, China, 18–21 August 2004; pp. 874–878.
31. Skinnemoen, H. UAV & satellite communications live mission critical visual data. In Proceedings of the 2014 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology, Yogyakarta, Indonesia, 13–14 November 2014; pp. 12–19.
32. Naqvi, S.A.R.; Hassan, S.A.; Pervaiz, H.; Ni, Q. Drone-Aided Communication as a Key Enabler for 5G and Resilient Public Safety Networks. *IEEE Commun. Mag.* **2018**, *56*, 36–42. [CrossRef]
33. Zeng, Y.; Zhang, R. Energy-efficient UAV communication with trajectory optimization. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 3747–3760. [CrossRef]
34. Tran, T.X.; Hajisami, A.; Pompili, D. Cooperative Hierarchical Caching in 5G Cloud Radio Access Networks. *IEEE Netw.* **2017**, *31*, 35–41. [CrossRef]
35. Zorbas, D.; Razafindralambo, T.; Guerriero, F. Energy efficient mobile target tracking using flying drones. *Procedia Comput. Sci.* **2013**, *19*, 80–87. [CrossRef]
36. Sharma, V.; You, I.; Pau, G.; Collotta, M.; Lim, J.D.; Kim, J.N. LoRaWAN-Based Energy-Efficient Surveillance by Drones for Intelligent Transportation Systems. *Energies* **2018**, *11*, 573. [CrossRef]
37. Sharma, V.; You, I.; Kumar, R. Energy efficient data dissemination in multi-UAV coordinated wireless sensor networks. *Mob. Inf. Syst.* **2016**, *2016*, 8475820. [CrossRef]
38. Sikeridis, D.; Tsiropoulos, E.E.; Devetsikiotis, M.; Papavassiliou, S. Self-Adaptive Energy Efficient Operation in UAV-Assisted Public Safety Networks. In Proceedings of the 19th IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC 2018), Kalamata, Greece, 25–28 June 2018; pp. 1–5.
39. Anton, S.R.; Inman, D.J. Performance modeling of unmanned aerial vehicles with on-board energy harvesting. In *Active and Passive Smart Structures and Integrated Systems*; International Society for Optics and Photonics: Bellingham, WA, USA, 2011; Volume 7977.
40. Lyu, J.; Zeng, Y.; Zhang, R. Cyclical multiple access in UAV aided communications: A throughput-delay tradeoff. *arXiv*, 2016; arXiv:1608.03180.
41. Sharawi, M.S.; Aloi, D.N.; Rawashdeh, O. Design and implementation of embedded printed antenna arrays in small UAV wing structures. *IEEE Trans. Antennas Propag.* **2010**, *58*, 2531–2538. [CrossRef]
42. Ceran, E.T.; Erkilic, T.; Uysal-Biyikoglu, E.; Girici, T.; Leblebicioglu, K. Optimal energy allocation policies for a high altitude flying wireless access point. *Trans. Emerg. Telecommun. Technol.* **2017**, *28*, e3034. [CrossRef]
43. Mozaffari, M.; Saad, W.; Bennis, M.; Debbah, M. Wireless Communication using Unmanned Aerial Vehicles (UAVs): Optimal Transport Theory for Hover Time Optimization. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 8052–8066. [CrossRef]
44. Chen, M.; Saad, W.; Yin, C. Liquid state machine learning for resource allocation in a network of cache-enabled LTE-U UAVs. In Proceedings of the GLOBECOM 2017—2017 IEEE Global Communications Conference, Singapore, 4–8 December 2017; pp. 1–6.
45. Bujari, A.; Calafate, C.T.; Cano, J.C.; Manzoni, P.; Palazzi, C.E.; Ronzani, D. Flying ad-hoc network application scenarios and mobility models. *Int. J. Distrib. Sens. Netw.* **2017**, *13*, 1–17. [CrossRef]

46. Hong, X.; Gerla, M.; Pei, G.; Chiang, C.-C. A group mobility model for ad hoc wireless networks. In Proceedings of the 2nd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems, Seattle, WA, USA, 20 August 1999; pp. 53–60.
47. Guillen-Perez, A.; Cano, M.D. Flying Ad Hoc Networks: A New Domain for Network Communications. *Sensors* **2018**, *18*, 3571. [CrossRef] [PubMed]
48. Shin, S.Y.; Choi, S.; Park, H.S.; Kwon, W.H. Lecture notes in computer science: Packet error rate analysis of IEEE 802.15.4 under IEEE 802.11b interference. In Proceedings of the 3rd International Conference on Wired/Wireless Internet Communications (WWIC '05), Malaga, Spain, 11–13 May 2005; pp. 279–288.
49. Mylin, A.K. A Communication Link Reliability Study for Small Unmanned Aerial Vehicles. Master's Thesis, University of Kentucky, Lexington, KY, USA, 2007.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Application of Bee Colony Algorithm for FANET Routing

Alexey V. Leonov¹, *Student Member, IEEE*

¹Omsk State Technical University, Omsk, Russia

Abstract – FANET are wireless ad hoc networks on unmanned aerial vehicles, and are characterized by high nodes mobility, dynamically changing topology and movement in 3D-space. FANET routing is an extremely complicated problem. The article describes the bee algorithm and the routing process based on the mentioned algorithm in ad hoc networks. The classification of FANET routing methods is given. The overview of the routing protocols based on the bee colony algorithms is provided. Owing to the experimental analysis, bio-inspired algorithms based on the bee colony were proved to show good results, having better efficiency than traditional FANET routing algorithms in most cases.

Index Terms – FANET, Routing Protocols, Network Simulation, Bee Colony Algorithms, BeeAdHoc.

I. INTRODUCTION

IN 1989 GERARDO BENI and Jing Wang introduced the term «swarm intelligence» (SI). This term was used for the cellular robotic system and meant collective behavior of the decentralized, self-organized system [1]. Later SI became a recognized optimization algorithm in artificial intelligence theory. SI is a multi-agents system of intellectual optimization with self-organized behavior. Collective system is capable of solving complex dynamic problems of collaborative work that cannot be performed by a single element of the system under diverse circumstances without an external management, control or coordination.

To realize SI systems, the so-called swarm algorithms are used. The algorithms are based on modeling social behavior of fish or birds in flocks or insects in swarms. Swarm algorithms are used successfully in complex optimization problem solution. Such problem solution aims to find the optimal solution for the target function (price, precision, time, distance, etc.) optimization (the discovery of the maximum or minimum) in a discrete set of the possible solutions [2].

At the present time various algorithms based on the swarm intelligence have been developed and are successfully used: the ant colony optimization, bee colony algorithm and particle swarm optimization algorithm, among others [3].

The paper examines the algorithms based on the bee algorithm realized for solving various graph NP-complete problems, as well as routing problems in Mobile Ad Hoc Networks (MANETs), Vehicular Ad Hoc Networks (VANETs). Experimental investigations have been carried out to examine the possibility of the algorithms application to Flying Ad Hoc Networks (FANETs) [4].

II. BEE ALGORITHM

The algorithm is the sub-class of a bio-inspired algorithms and models bee behavior in natural habitat.

In wildlife the bee hive operating principle is based on a clear distribution of responsibilities among the bees. All bees of the hive can be divided into three groups [5] [6]:

1. employed bees;
2. onlookers;
3. scouts.

Employed bees search for a nectar source and provide information on the quality of the explored sites to the other bees (onlookers) (Fig.1). Onlookers stay in the hive and receive the information on the study subject only from the employed bees, using special “waggle dance”. The moves of a bee dance show to the other bees where exactly the food source is located and the amount of the pollen or nectar in it. The waggle dance is based on the Sun location (Fig. 2). Scouts, in their turn, look for new nectar sources in random [7] [8].

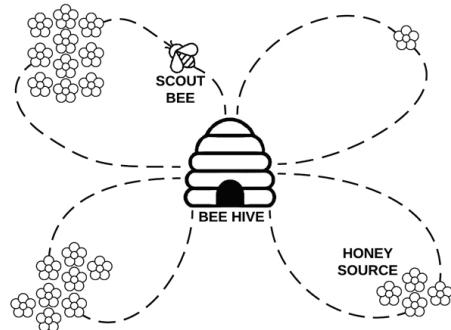


Fig. 1. The diagram of the bee colony behavior in wildlife.

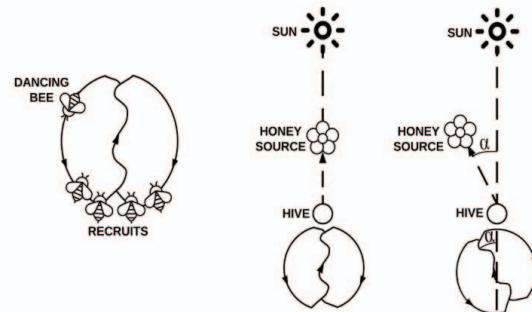


Fig. 2. The diagram of the bee waggle dance.

Initially scouts provide numerous promising sites (sources), then forager-bees (employed bees) explore the source neighborhoods. The aim of the bee colony is the search for the source with the maximal nectar amount, i.e. the target function of the optimization problem.

Every solution in the algorithm under consideration is represented as a point (site) in the search space. The determined amount of the nectar represents the target function (TF) value at this point (Fig. 3).

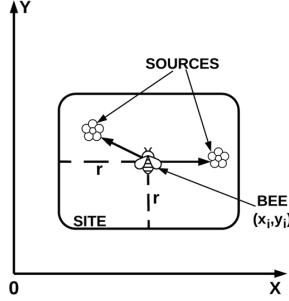


Fig. 3. The model of determining the site for nectar search.

Suppose the search space is two-dimensional space, where x_i, y_i are the bee coordinates (function parameter), r is the radius (size) of the site, n is the number of function parameters, in this particular case it is the variant with two parameters x and y . As seen in Fig. 2, the bee can move to other nectar sources only within the site limited by the radius. The nectar amount is in inverse proportion to the function value at coordinates x and y . In other words, the agent is looking for another solution not much different from the previous one [9].

To develop the behavioral model of the bee colony self-organization is to develop the methods and techniques:

- formation of scout-agents swarm and forager-agents swarm;
- search for perspective sites by scouts;
- choice of the basic sites among the perspective ones to explore their neighborhoods;
- information transfer among the scouts and the foragers;
- choice of the basic sites by the foragers;
- choice of the sites in the basic sites neighborhood by the foragers;
- general structure of the optimization process [10].

III. SEARCH PROCEDURES ORGANIZATION

Bee algorithm modeling is performed as following:

1. The basic parameters of bee algorithm are set: L is the maximum number of the iterations; n_r is the initial number of the scout-agents; n_δ is the number of the basic sites; λ is the threshold value of the neighborhood size; n_f is the initial number of the forager-agents; $n_{\delta l}$ is the number of the basic sites formed from the best sites $a_s^*(l)$, found by the swarm at the l -th iteration; n_{rl} is the number of the scout-agents randomly choosing new sites; $n_{\delta 2}$ is the number of the basic sites formed from the best new sites found by the scout-agents at the l -th iteration.
2. At the first iteration ($l=1$) scout agents in the number of n_r spread randomly in search space. This operation includes random generation of a set of different sites $R = \{r_s | s=1,2,\dots,n_r\}$. The target function $F_s(l)$ value is calculated for each site. The number n_δ of the best solutions $X^\delta = X_s$ is chosen. A set of basic sites $R^\delta(l) = \{r_s^\delta(l) | s=1,2,\dots,n_\delta\}$, corresponding to the set X^δ is formed.
3. Random generation of the different lists $E(l) = \{E_s(l) | s=1,2,\dots,n_r\}$, having corresponding set of sites $A(l) = \{a_s(l) | s=1,2,\dots,n_r\}$.
4. Internally stable set $X_s(l)$ with remainder $E_{os}(l)$ is formed and the target function $F_s(l)$ value is calculated for each list $E_s(l)$.
5. A set of basic solutions $X^\delta(l) \subset X(l)$ with the best target functions $F_s(l)$ and the corresponding set of the basic sites $A^\delta(l) \subset A(l)$. $|A^\delta(l)| = |X^\delta(l)| = n_\delta$. $z = 1$ (z is the sequence number of the forager-agent) are formed.
6. Choice with probability $P(a_s^\delta) = \frac{F_s^\delta}{\sum_s F_s^\delta}$ of the basic site $a_s^\delta(l) \in A^\delta(l)$.
7. Probabilistic choice of the site $a_z(l)$, situated in the neighborhood of the basic site $a_s^\delta(l)$, with the corresponding solution $X_z(l)$.
8. If the site $a_z(l)$ coincides with the previously chosen site, then the transition to step 9 is performed, otherwise, proceed to step 10.
9. The site $a_z(l)$ is included in the set of the sites $O_s(l)$ chosen by the forager-agents.
10. The calculation of the target function $F_z(l)$ value for the solution $X_z(l)$.
11. If $z < n_f$, then $z = z + 1$ and the transition to step 7 is performed, otherwise ,proceed to step 13.
12. Formation of the domain $D_s(l) = O_s(l) \cup a_s^\delta(l)$ for each basic site $a_s^\delta(l)$.
13. In each domain $D_s(l)$, the best site $a_s^*(l)$ with the best solution $X_s^*(l)$ is chosen.
14. The best solution $X_s^*(l)$ is chosen among $X^*(l)$
15. If $X_s^*(l)$ is better than $X^*(l-1)$, it is kept, otherwise $X^*(l) = X^*(l-1)$.
16. If $l < L$, then $l = l + 1$ and the transition to step 18 is performed, otherwise, proceed to step 21.
17. The first part $X^{\delta l}(l)$ receives $n_{\delta l}$ of the best sites from the sites $X_s^*(l-1)$, found by the agents in each of the domains $D_s(l-1)$, formed at the previous iteration.

18. Random generation of the set of different lists

$$E(l) = \{E_s(l) | s = 1, 2, \dots, n_r\} \text{ having the corresponding set of the sites } A(l) = \{a_s(l) | s = 1, 2, \dots, n_r\}. |E(l)| = n_r.$$

19. From the set $A(l)$ of new sites, found by the scout-agents at the 1st iteration $n_{\dot{a}1} + n_{\dot{a}2} = n_{\dot{a}}$, the best sites are included into the set $A^{\delta^2}(l)n_{\delta^2}$.

20. Forming of the set of the basic sites

$$A^\delta(l) = A^{\delta^1}(l) \cup A^{\delta^2}(l). \text{ Transition to the step 6.}$$

21. The end of the algorithm. Solution $X^*(l)$ is the best solution found by the agents swarm.

22. The operation speed of the algorithm depends on the lifetime of the colony l (iteration number), the number of the sites c and the number of the agents m , and is determined as $O(l * c^2 * m)$ [11].

IV. BEE ALGORITHM OPERATION

Algorithm operation depends on the following basic parameters [7]:

1. total number of the scouts (N);
2. total number of the sites (m);
3. total number of the elite sites (\dot{a});
4. number of the scouts at the elite sites (n);
5. number of the bees (l) at the rest ($m - \dot{a}$) of the sites;
6. initial size of the sites, i.e. the size of the sites with the neighborhoods (S);
7. maximum number of the iterations (I).

Let us describe a modified bee algorithm (Artificial bee colony (ABC) algorithm) [12]. The flow-chart of the algorithm is given in Fig. 4 [11].

1. According to the partition problem statement and the given data, the population of N bees is formed.
2. Departure of the onlookers. Determination of the nectar source v_i location. The initial position x_i is randomly set for each bee on the surface (in neighborhood) of m sites.
3. TF evaluation for the bees in the population. The choice of the nectar source by the onlooker-bee with a definite probability depending on its quality. Each bee is assigned with the best (elite) site that the bee has been visiting since the first iteration and the target function fit_i value at this site. Sites e with higher TF values (elite sites) are chosen to find the solution in their neighborhoods. For each onlooker, their number determined as $\frac{1}{2} m$, the search site is chosen by the expression:

$$v_{ij} = x_{ij} + \phi_{ij} * (x_{ij} - x_{kj}), \quad (1)$$

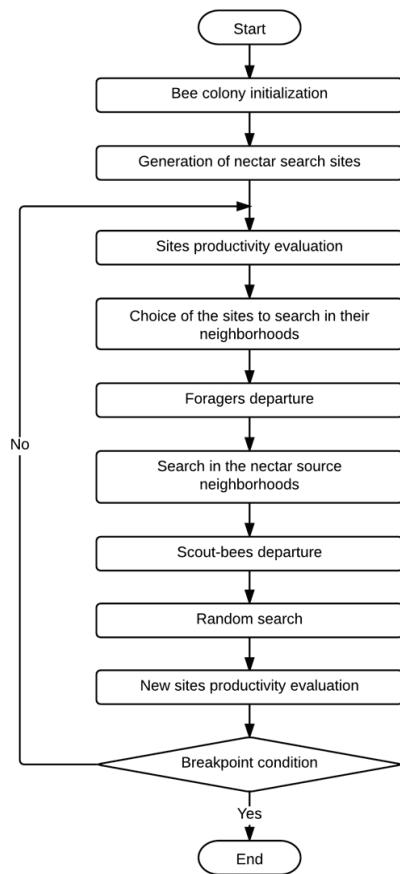


Fig. 4. The flow chart of ABC algorithm.

where ϕ_j is a random number from the range $[-1; 1]$, k in the subscript of the solution randomly chosen from the colony ($k = \text{int}(rand * m) + 1$), $j = 1 \dots, D$, D is the problem dimension. After the site v_i was determined, the resulting solution is compared to x_i , and the employed bee proceeds to the better (elite) site of the two. Now, to find the solution in the elite sites neighborhoods, the solutions are chosen with probability calculated with the expression:

$$p_i = \frac{fit_i}{\sum_{j=1}^S fit_i}, \quad (2)$$

- where fit_i is the target function x_i value. Moreover, more detailed explorations are undertaken at these sites, i.e. more bees than to each of $m - \dot{a}$ sites are sent.
4. TF values are estimated, and the best bees are chosen according to TF values of the explored sites.
 5. If the solution for the explored site does not improve in the course of several iterations, then the transition to step 6 is performed, otherwise, proceed to step 3.
 6. Scouts proceed to search randomly and evaluate TF.

Onlookers of the site become scouts, and the choice of the solutions is performed randomly by the formula:

$$x_{ij} = x_j^{\min} + (x_j^{\max} - x_j^{\min}) * \text{rand} . \quad (3)$$

7. The new population of bees is formed, and it includes the bees with the best TF values of the elite sites along with the bees having random TF values.
8. The solution search space is set randomly. It provides an opportunity to construct a hybrid structure of the solution search, based on the combination of genetic and evolutionary search and SI algorithm.
9. TF evaluation for the bees in the population. The choice of the nectar source by the onlooker with a definite probability depending on its quality. Each bee is assigned with the best (elite) site that the bee has been visiting since the first iteration and the target function fit_i value at this site. Sites e with higher TF values (elite sites) are chosen to find the solution in their neighborhoods.
10. Verification of the algorithm stopping criterion being met.
11. The end of the algorithm.

V. FLYING AD HOC NETWORK

Lately the so-called Flying Ad Hoc Network (FANET) has become currently important, representing a special type of peer-to-peer ad hoc network based on unmanned aerial vehicles (UAV). Potential of using UAV as the foundation for creating network infrastructure is considered to be an attractive approach for an effective expansion of communicative capabilities of the networks. According to the data provided by International Institute for Strategic Studies (IISS) about 56 types of UAVs were used in 11 countries in 2012 [13]. UAV based network is characterized by the versatility, flexibility and relative small operating costs [14]. Such network can be applied in various branches of agriculture (Fig. 5). The comparison of peer-to-peer ad hoc networks MANET (Mobile Ad Hoc Network), VANET (Vehicular Ad Hoc Network) and FANET is given in Table 1 [15].

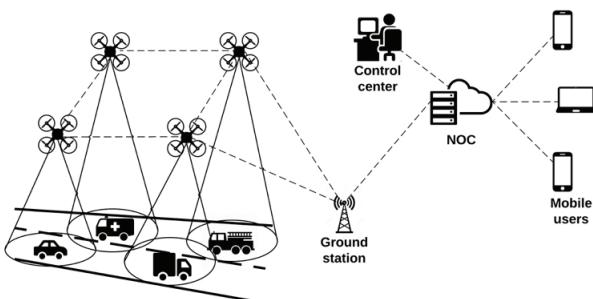


Fig. 5. Example use of FANET in traffic monitoring

TABLE I
COMPARATIVE DESCRIPTION OF DIFFERENT AD HOC NETWORKS

	MANET	VANET	FANET
Description	Mobile wireless nodes connect with other nodes within communication range in an ad-hoc manner (No centralized infra-structure required)	Ad-hoc networks in which vehicles are the mobile nodes. Communication is among vehicles and between vehicles and road side units	Ad-hoc or infrastructure based networks of airborne nodes. Communication among UAVs and with the control station
Mobility	Slow. Typical speeds 2 m/sec. Random movement. Varying density, higher at some popular places	High-speed, typically 20-30 m/s on highways, 6-10 m/s in urban areas. Predictable, limited by road layout, traffic and traffic rules	Speeds from 0 to typically as high as 100 m/s. Movement could be in 2 or 3 dimensions, usually controlled according to mission.
Topology	Random, ad-hoc	Star with roadside infrastructure and ad-hoc among vehicles	Star with control center, ad-hoc/mesh among UAVs.
Topology Changes	Dynamic - nodes join and leave unpredictably. Network prone to partitioning.	More dynamic than MANETs. Movement linear. Partitioning common.	Stationary, slow or fast. May be flown in controlled swarms. Network prone to partitioning
Energy Constraints	Most nodes are battery powered so energy needs to be conserved.	Devices may be car battery powered or own battery powered.	Small UAVs are energy constrained. Batteries affect weight and flying time
Typical use cases in public and civil domains	<ul style="list-style-type: none"> Information distribution (emergencies, advertising, shopping, events) Internet hot spots 	<ul style="list-style-type: none"> Traffic & weather info, emergency warnings, location based services Infotainment 	<ul style="list-style-type: none"> Rescue operations Agriculture-crop survey Wildlife search Oil rig surveillance

VI. ROUTING METHODS AND ALGORITHMS IN FANET

In the present time organization techniques of FANET are still being developed, while new routing algorithms and methods appear and old ones are improved.

The choice of the routing protocol is a complex task and depends on the application requirements: the number of the net users, their mobility level, the necessity to support multimedia data transmission, quality indicators QoS, etc.

The foundation of any routing protocol is the algorithms.

The routing algorithms are applied to find the best route (usually the shortest one) for data transmission from the source node to the destination node. Various criteria (metrics) are used in choosing the optimal routes. Metrics may include such information as the route length (steps number), reliability, delay, bandwidth, load, communication cost, power consumption, etc.

Moreover the routing methods must meet the following requirements [16]:

- compliance with FANET characteristics;
- self-organization of network nodes;
- decentralized operation;
- rapid convergence and absence of route looping;
- minimal load of the network by the service information;
- on-demand route generation (network «silent» mode);
- providing several routes for data delivery to the addressee;
- providing routes of the specified quality (based on productivity, delay, etc.);
- capability of simplex channel;
- data collection and aggregation;
- power consumption minimization for the network nodes;
- routing process safety, etc.

VII. THE CLASSIFICATION OF FANET DYNAMIC ROUTING PROTOCOLS

To choose the routing protocol in accordance with the problem under consideration is the key aspect, moreover the protocol shall be capable of providing effective network operation under the specified conditions.

The total number of the routing algorithms is large enough due to the diversity of the requirements to the information exchange, characteristics of the transmitted data and functionality of hardware and software. Each protocol offers advantages and disadvantages.

The protocols can be divided into five main categories (Fig. 6).

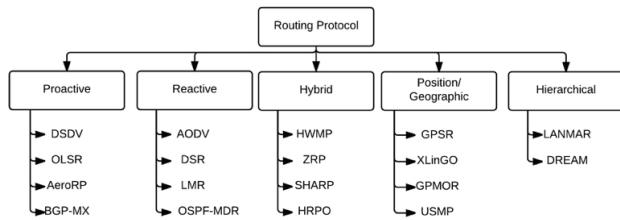


Fig. 6. Classification of the routing protocols in FANET

Proactive, or table-driven, protocols operate periodically. If the network topology is changed, service message broadcast is initiated to inform of these changes.

In reactive or on-demand protocols the routes are generated only when necessary, i.e. data needs to be transmitted.

Hybrid protocol combine the techniques of proactive and reactive protocols. As a rule, they divide the network into a set of sub-networks with proactive protocol, while

the interaction between them is conducted with reactive methods.

As a separate group of FANET routing protocols, one can distinguish geo-routing protocols, making use of the data on the network users location (GLONASS/GPS). The main advantages of geo-routing protocols is the absence of the necessity to store the routing information in transit nodes and the possibility of the routes optimization due to the available information on the nodes location.

Another solution for FANET routing is hierarchical protocols, developed to solve the problems of the network scalability. To solve this problem, a hierarchical control should be introduced, i.e. the network is to be divided into separate sectors (clusters) with the main network nodes, gateway nodes and internal nodes.

VIII. ROUTING IN AD HOC NETWORK BASED ON BEE COLONY ALGORITHM

When observing bee colonies, the scientists noticed some interesting features proved to be useful for solving routing problems in wireless ad hoc networks [17] [18].

Let us consider the routing process in the ad hoc network based on the bee algorithm.

Just like in any other swarm algorithm, in bee algorithms everything starts with the reception of the information to be transmitted to the other node.

The first stage is called “scouting” and divided into forward and backward scouting. Forward scouts inspect the network for the destination node. They carry four categories of information: scout ID (Identification Number), source node ID, minimal residual energy (initially equal to infinity) and number of hops (equal to zero). The scouts are sent to all source node neighborhoods and so on. Arriving at each intermediate node, the counter of hops number is increased by 1. Energy efficiency of the data transmission depends on this counter. Once the destination node is found, the backward scout returns to the source node, and multiple channels are created between the source and destination node.

At every visited node the algorithm checks the scouts characteristics, and, based on this, draws a conclusion on the route efficiency. The procedure is rather complicated and undergoes several verification steps. For instance, if the node receives another forward scout, it checks if the scout with the same ID has arrived before, etc.

Upon completion of the scouting procedure and the route determination, the number of forager-bees necessary for the route is calculated with the help of “dance formula”. The stage of “resource foraging” starts. The foragers transfer the data just like the real bees transport the food.

Data transmission process is rather complicated as well. The data is transmitted from the source to the destination node, the number of forager-bees varying due to the network situation. The probability distribution table is used for probabilistic calculation. The main difference from the ant colony optimization algorithm is that the intermediate nodes make no decisions on routing, all decisions come from the source node [19].

IX. ROUTING IN AD HOC NETWORK BASED ON BEE COLONY ALGORITHM

Bee optimization algorithms for routing in telecommunication networks differ from the most traditional routing algorithms. This section presents the comparative analysis of bio-inspired adaptive routing algorithms for peer-to-peer networks based on the bees behavior in wildlife.

Table 2 contain the applications of the bee colony algorithms [20].

Based on the information from the tables, one may conclude that the algorithms simulating the bees behavior in wildlife are effective for such types of peer-to-peer networks as VANET, WSN, and MANET.

X. SIMULATION SCENARIOS

The comparison of the routing protocols is hindered by the fact that data transmission process in Ad Hoc networks is affected by a large number of various factors, most of them being random and difficult to be mathematically analyzed.

Simulation modeling means are used for a proper functional check of Ad Hoc network. The appearance of the software for simulation modeling allowed to undertake necessary experiments and investigations, improve developed models, protocols, stacks, and routes, and optimize them without real network deployment.

Nowadays to solve simulation modeling problems there is a wide range of software, from function libraries for standard compilers to specialized programming languages. From the simulation means for the network modeling, developed so far, one may distinguish ns-2, ns-3, OPNET, OMNET++, QualNet/GloMoSiM, and JiST/SWANS.

For this paper the investigation of the routing protocol BeeAdHoc was carried out in the network simulator ns-2. For modeling version 2.35 was used, and protocol BeeAdHoc was pre-installed as well. The comparison was undertaken on the protocols, which models were realized in ns-2: AODV, DSDV, and DSR. Input data for the simulation modeling of all protocols were the same (Table 3).

XI. PERFORMANCE PARAMETERS

To analyze the performance of the routing protocols, the following metrics were used:

- End-to-End delay is the delay between the first sent byte and the last received byte. It includes transmission delay, process queue delay and propagation delay.
- Throughput characterizes the maximal possible rate of successful packet delivery along the communication channel.
- Routing overhead is the overhead of a route discovery and the construction of a routing table.

XII. ROUTING IN AD HOC NETWORK BASED ON BEE COLONY ALGORITHM

Protocol BeeAdHoc with an increased number of nodes was assumed to exhibit better performance parameters

TABLE II
PROTOCOLS FOR AD HOC NETWORKS INSPIRED BY BEE BEHAVIOR

Algorithm	Application	Type of network
BeeAdHoc (2005) [21]	Energy efficient routing	MANET
BeeSec (2007) [22]	Security framework for routing	MANET
BeeAIS (2007) [23]	AIS security for routing	MANET
BeeAIS-DC (2008) [24]	DC inspired AIS security for routing	MANET
BeeIP (2010) [25]	Routing	MANET
iBeeAIS (2011) [26]	Hybrid AIS security for routing	MANET
BeeAdHocAutoConf (2012) [27]	Energy efficient routing	MANET
Bee-Ant Colony Optimized Routing (BACOR) (2012) [28]	Energy efficient routing	MANET
Clustered Artificial Bee Colony (2012) [29]	Clustered routing	MANET
Predictive Energy Efficient Bee Routing Algorithm (PEEBR) [30]	Energy efficient routing	MANET
Independent Zone Routing Protocol (IZRP) (2012) [31]	Hybrid routing	MANET
Bee-MANET (2014) [32]	High-throughput routing	MANET
BeeAdHoc ServiceDiscovery (2014) [33]	Energy efficient and scalable routing	MANET
QoS Bee Routing (QBR) (2014) [34]	QoS routing	MANET
FBeeAdHoc (2015) [35]	Secure routing	MANET
QoS Bee Vanet (2011) [36]	QoS-multicast routing	VANET
Bee Life Algorithm (2013) [37]	QoS-multicast routing	VANET
Hybrid Bee Swarm Routing (HyBR) (2013) [38]	Hybrid routing	VANET
Multicast Quality Of Service Swarm Bee Routing (MQBV) (2013) [39]	QoS-multicast routing	VANET
Bee Optimized Fuzzy Geographical Routing (2014) [40]	Geographical routing	VANET
BeeSensor (2007) [41]	Energy efficient and scalable routing	WSN
Artificial Bee Colony (2012) [42]	Dynamic deployment	WSN
Improved version of Cluster based Wireless sensor network routings using Artificial bee colony algorithm considering Quality of service (ICWAQ) (2012) [43]	Energy efficient and QoS routing	WSN
Bee-Sensor-C (2015) [43]	Energy-Efficient and Scalable Multipath Routing	WSN

TABLE III
SIMULATION PARAMETERS

Parameter	Value
Simulated area	1500 m x 1500 m
Mobility model	Random waypoint
UAV number	10, 20, 30, 40, 50
Simulation runs	10
Simulation duration	20 sec
Nodes velocity	20, 40, 60, 80, 100 m/s
Traffic type	CBR
MAC layer protocol	802.11
Signal Propagation Model	Friis
Data links antenna	Omni
Antenna coverage range	150, 200, 250, 300, 350 m
Transport Protocol	UDP
Packet size	512 Kbytes

compared to protocols AODV, DSDV and DSR. Routing overhead and end-to-end delay were expected to increase, while throughput remains unchanged.

During the experiment the number of the network nodes varied in the range of 10 to 50.

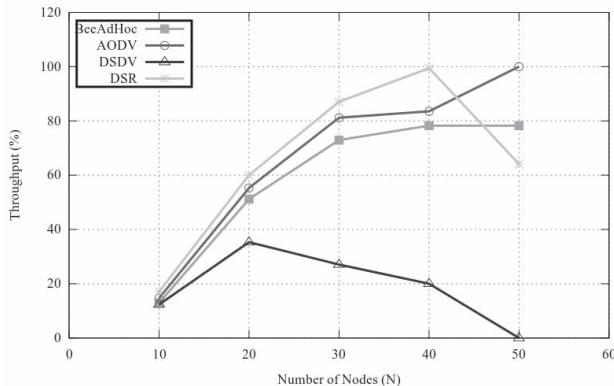


Fig. 7. Number of Nodes Vs Throughput

As seen in Fig. 7, the increase of the nodes amount leads to the increase in the throughput of the protocols BeeAdHoc, DSR, and AODV, however DSDV has the worst values, its throughput decreasing in proportion with the nodes amount increase.

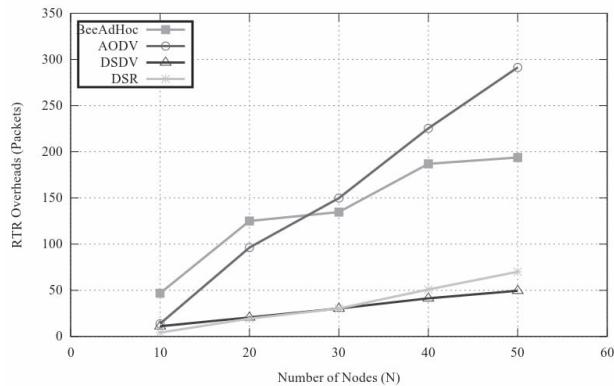


Fig. 8. Number of Nodes Vs Routing Overhead

In Fig. 8 one can see that the nodes amount increase results in routing overhead increase for all protocols under investigation. Moreover, DSR and DSDV show the best figures, and a BeeAdHoc exceeds AODV on the average.

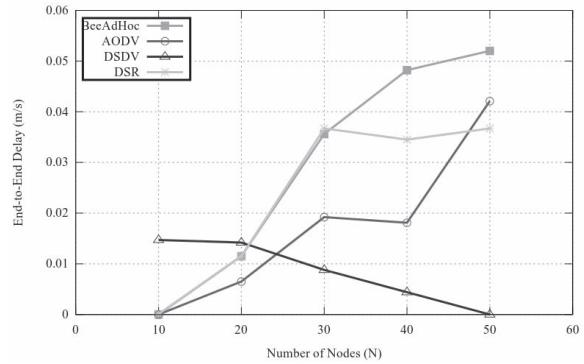


Fig. 9. Number of Nodes Vs End-to-End Delay

Fig. 9 shows the algorithm behavior pattern with regard to the ratio of the end-to-end delay to the nodes amount. BeeAdHoc is slightly inferior to the protocols AODV and DSR. DSDV has the best figures since it is a proactive/table-driven protocol and its routing tables store all node routes.

The diagrams showing the algorithms adaptivity to the topology and nodes mobility changes are given below.

The algorithms demonstrate different behavior in that respect, and DSDV has unstable behavior. BeeAdHoc is the most adaptive algorithm when it comes to the topology changes, in general the rate changes affects its throughput just slightly. With the rate increase, the algorithms BeeAdHoc and AODV quickly adjust to the changes and exhibit the best throughput values.

As one can see from Fig. 10., DSDV and DSR are under small load with respect to routing overhead after the nodes mobility has increased, as shown in Fig. 11. At the same time, BeeAdHoc is under a heavy load. AODV has an opposite behavior compared to BeeAdHoc, due to AODV using the routing table to store several routes to the destination node and the possibility of finding alternative routes without the launch of a new route discovery process.

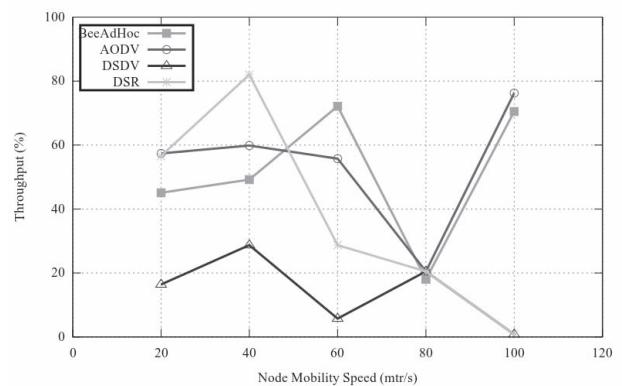


Fig. 10. Node Mobility Vs Throughput

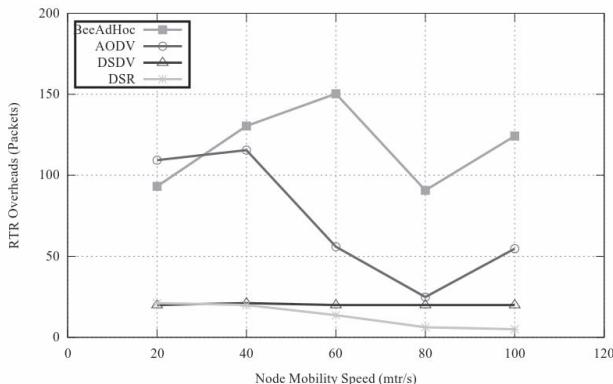


Fig. 11. Node Mobility Vs Routing Overhead

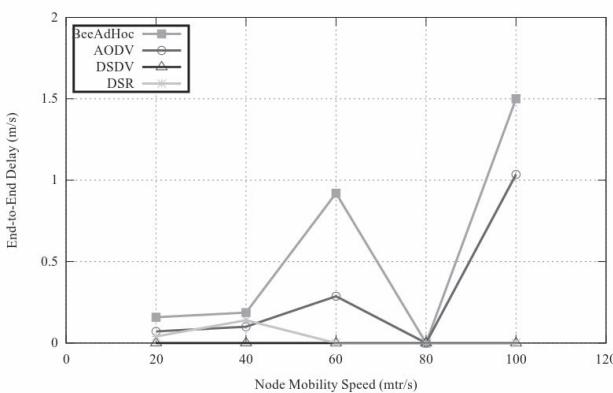


Fig. 12. Node Mobility Vs Routing Overhead

BeeAdHoc has long delays compared to the protocols (see Fig. 12) because of the necessity to find a new route to the destination node, that gived, in its turn, an advantage over the other protocols by the throughput.

Based on the results obtained from the experimental research one may conclude that the algorithm BeeAdHoc provides better throughput than DSDV and DSR, but has longer delay. On the average, the efficiency of BeeAdHoc is comparable to the investigated protocols DSR, DSDV and AODV. BeeAdHoc may be used for packet routing in FANET.

XIII. CONCLUSION AND FUTURE WORK

FANET routing is a complex problem. Bio-inspired algorithms based on the bee colony exhibit good results, having better efficiency than traditional routing algorithms in most cases.

In the preparation of the article, various approaches to the routing problem solution based on the bee algorithm were analyzed.

Routing protocols based on the bee colony methods are proved to be effective compared to the other algorithms. The promising direction of the future research in the studied field seems to be the development and realization of the hybrid algorithm based on the bee and ant colony algorithms. In this

case the disadvantages of one algorithm are compensated by the advantages of the other, resulting in a better convergence rate, this allowing lower requirements to the nodes hardware.

REFERENCES

- [1] G. Beni and J. Wang, "Swarm intelligence in cellular robotic systems," in *Robots and Biological Systems: Towards a New Bionics*, Springer, 1993, pp. 703–712.
- [2] V. V. Baranjuk and O. S. Smirnova, "Roevoj intellekt kak odna iz chastej ontologicheskoy modeli bionicheskikh tehnologij," *Int. J. Open Inf. Technol.*, vol. 3, no. 12, pp. 13–17, 2015. (in Russian).
- [3] A. A. Zajcev, V. V. Kurejchik, and A. A. Polupanova, "Obzor jevolucionnykh metodov optimizacii na osnove roevogo intellekta," *Izvestija JuFU Tehnicheskie Nauki*, vol. 12, no. 113, pp. 7–12, 2010. (in Russian).
- [4] I. Bekmezci, O. K. Sahingoz, and S. Temel, "Flying Ad-Hoc Networks (FANETs): A survey," *Ad Hoc Netw.*, vol. 11, no. 3, pp. 1254–1270, May 2013.
- [5] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Technical report-tr06, Erciyes university, engineering faculty, computer engineering department, 2005.
- [6] D. T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, and M. Zaidi, "The Bees Algorithm—A Novel Tool for Complex Optimisation," in *Intelligent Production Machines and Systems-2nd I* PROMS Virtual International Conference 3-14 July 2006*, 2006, pp. 454–461.
- [7] V. V. Kurejchik and E. E. Polupanova, "Jevolucionnaja optimizacija na osnove algoritma kolonii pcholj," *Izvestija JuFU Tehnicheskie Nauki*, vol. 12, no. 101, pp. 41–46, 2009. (in Russian).
- [8] S. K. Dhurandher, S. Misra, P. Pruthi, S. Singhal, S. Aggarwal, and I. Woungang, "Using bee algorithm for peer-to-peer file searching in mobile ad hoc networks," *J. Netw. Comput. Appl.*, vol. 34, no. 5, pp. 1498–1508, Sep. 2011.
- [9] V. M. Kurejchik and A. A. Kazharov, "Ispol'zovanie roevogo intellekta v reshenii NP-trudnyh zadach," *Izvestija JuFU Tehnicheskie Nauki*, vol. 7, no. 120, pp. 30–36, 2011. (in Russian).
- [10] Je. V. Kuliev and A. A. Lezhebokov, "O gibridnom algoritme razmeshheniya komponentov SBIS," *Izvestija JuFU Tehnicheskie Nauki*, vol. 11, no. 136, pp. 188–192, 2012. (in Russian).
- [11] V. V. Kurejchik and M. A. Zhilenkov, "Pchelinijj algoritm dlja reshenija optimizacionnyh zadach s javno vyrazhennoj celevoj funkciej," *Informatika Vychislitel'naja Tekhnika I Inzhenernoe Obrazovanie*, vol. 1, no. 21, pp. 1–8, 2015. (in Russian).
- [12] D. Karaboga and B. Basturk, "Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems," in *Foundations of Fuzzy Logic and Soft Computing*, Springer, 2007, pp. 789–798.
- [13] "Drones by country: who has all the UAVs? Visualised | News | The Guardian." [Online]. Available: <http://www.theguardian.com/news/datablog/2012/aug/03/drone-stocks-by-country>. [Accessed: 20-Mar-2016].
- [14] S. Wei, L. Ge, W. Yu, G. Chen, K. Pham, E. Blasch, D. Shen, and C. Lu, "Simulation study of unmanned aerial vehicle communication networks addressing bandwidth disruptions," 2014, pp. 90850O-1–90850O-10.
- [15] L. Gupta, R. Jain, and G. Vaszkun, "Survey of Important Issues in UAV Communication Networks," *IEEE Commun. Surv. Tutor.*, pp. 1–32, 2015.
- [16] M. H. Tareque, M. S. Hossain, and M. Atiquzzaman, "On the Routing in Flying Ad hoc Networks," in *Computer Science and Information Systems (FedCSIS)*, Lodz, 2015, pp. 1–9.
- [17] J. S. Baras and H. Mehta, "A probabilistic emergent routing algorithm for mobile ad hoc networks," in *WiOpt '03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, 2003, p. 10–pages.
- [18] G. Di Caro, F. Ducatelle, and L. M. Gambardella, "Swarm intelligence for routing in mobile ad hoc networks.," in *SIS*, 2005, pp. 76–83.
- [19] P. N. Boronin and A. E. Kucherjavij, "Internet veshshej kak novaja konsepcija razvitiija setej svjazi," *Informacionnye Tehnologii I Telekommunikaciij*, vol. 3, no. 7, pp. 7–30, 2014. (in Russian).
- [20] M. KUCHAKI RAFSANJANI and H. FATEMIDOKHT, "A Survey on Algorithms Based on Bee Swarms for Ad Hoc Networks.," *Walailak J. Sci. Technol.*, vol. 12, no. 1, pp. 21–26, 2015.
- [21] H. F. Wedde, M. Farooq, T. Pannenbaecker, B. Vogel, C. Mueller, J. Meth, and R. Jeruschkat, "BeeAdHoc: an energy efficient routing algorithm for mobile ad hoc networks inspired by bee behavior," in *Proceedings of the 7th annual conference on Genetic and evolutionary computation*, 2005, pp. 153–160.
- [22] N. Mazhar and M. Farooq, "Vulnerability analysis and security framework (BeeSec) for nature inspired MANET routing protocols," in *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, 2007, pp. 102–109.

- [23] N. Mazhar and M. Farooq, “BeeAIS: Artificial immune system security for nature inspired, MANET routing protocol, BeeAdHoc,” in *Artificial Immune Systems*, Springer, 2007, pp. 370–381.
- [24] N. Mazhar and M. Farooq, “A sense of danger: dendritic cells inspired artificial immune system for manet security,” in *Proceedings of the 10th annual conference on Genetic and evolutionary computation*, 2008, pp. 63–70.
- [25] A. Giagkos and M. S. Wilson, “BeeIP: Bee-Inspired Protocol for Routing in Mobile Ad-Hoc Networks,” in *SAB*, 2010, pp. 263–272.
- [26] N. Mazhar and M. Farooq, “A hybrid artificial immune system (AIS) model for power aware secure Mobile Ad Hoc Networks (MANETs) routing protocols,” *Appl. Soft Comput.*, vol. 11, no. 8, pp. 5695–5714, Dec. 2011.
- [27] F. De Santis, “An efficient bee-inspired auto-configuration algorithm for mobile ad hoc networks,” *Int. J. Comput. Appl.*, vol. 57, no. 17, pp. 9–14, 2012.
- [28] S. K. Suguna and S. U. Maheswari, “Comparative Analysis of Bee-Ant Colony Optimized Routing (BACOR) with Existing Routing Protocols for Scalable Mobile Ad Hoc Networks (MANETs) based on Pause Time,” *Int. J. Comput. Sci. Netw. Secur. IJCSNS*, vol. 12, no. 4, pp. 9–14, 2012.
- [29] K. G. Santhiya and N. Arumugam, “A Novel Adaptive Bio-Inspired Clustered Routing for MANET,” *Procedia Eng.*, vol. 30, pp. 711–717, 2012.
- [30] I. M. Fahmy, H. A. Hefny, and L. Nassef, “PEEBR: Predictive Energy Efficient Bee Routing algorithm for Ad-hoc wireless mobile networks,” in *Informatics and Systems (INFOS), 2012 8th International Conference on*, 2012, p. NW-18–NW-24.
- [31] T. T. Le, D. D. Tan, and D.-S. Kim, “Zone routing determination for IZRP based on bee-inspired algorithm,” in *Information Fusion (FUSION), 2012 15th International Conference on*, 2012, pp. 750–756.
- [32] Z. Albayrak and A. Zengin, “Bee-MANET: A New Swarm-based Routing Protocol for Wireless Ad Hoc Networks,” *Electron. Electr. Eng.*, vol. 20, no. 3, pp. 91–97, Mar. 2014.
- [33] F. de Santis and D. Malandrino, “QoS-Based Web Service Discovery in Mobile Ad Hoc Networks Using Swarm Strategies,” *J. Comput. Netw. Commun.*, vol. 2014, pp. 1–13, 2014.
- [34] J. Li, Y. Li, Y. Wang, and H. Wang, “QoS Bee Routing Protocol for Ad Hoc Networks,” *J. Softw.*, vol. 9, no. 4, pp. 1019–1027, Apr. 2014.
- [35] M. Kuchaki Rafsanjani and H. Fatemidokht, “FBeeAdHoc: A secure routing protocol for BeeAdHoc based on fuzzy logic in MANETs,” *AEU - Int. J. Electron. Commun.*, vol. 69, no. 11, pp. 1613–1621, Nov. 2015.
- [36] S. Bitam and A. Mellouk, “QoS swarm bee routing protocol for vehicular ad hoc networks,” in *Communications (ICC), 2011 IEEE International Conference on*, 2011, pp. 1–5.
- [37] S. Bitam and A. Mellouk, “Bee life-based multi constraints multicast routing optimization for vehicular ad hoc networks,” *J. Netw. Comput. Appl.*, vol. 36, no. 3, pp. 981–991, May 2013.
- [38] S. Bitam, A. Mellouk, and S. Zeadally, “HyBR: A Hybrid Bio-inspired Bee swarm Routing protocol for safety applications in Vehicular Ad hoc NETworks (VANETs),” *J. Syst. Archit.*, vol. 59, no. 10, pp. 953–967, Nov. 2013.
- [39] S. Bitam, A. Mellouk, and S. Fowler, “MQBV: multicast quality of service swarm bee routing for vehicular ad hoc networks: Wireless MQBV for VANET,” *Wirel. Commun. Mob. Comput.*, vol. 15, no. 9, pp. 1391–1404, Jun. 2015.
- [40] P. Saravanan and T. Arunkumar, “Bee Optimized Fuzzy Geographical Routing Protocol for VANET,” *World Acad. Sci. Eng. Technol. Int. J. Comput. Electr. Autom. Control Inf. Eng.*, vol. 8, no. 12, pp. 2133–2139, 2015.
- [41] M. Saleem and M. Farooq, “Beesensor: a bee-inspired power aware routing protocol for wireless sensor networks,” in *Applications of Evolutionary Computing*, Springer, 2007, pp. 81–90.
- [42] M. Farooq, “Bee-inspired routing protocols for mobile ad hoc and sensor networks,” in *Bee-Inspired Protocol Engineering*, Springer, 2009, pp. 235–270.
- [43] D. Karaboga, S. Okdem, and C. Ozturk, “Cluster based wireless sensor network routing using artificial bee colony algorithm,” *Wirel. Netw.*, vol. 18, no. 7, pp. 847–860, Oct. 2012.



Alexey V. Leonov is a Fellow of IEEE, a Fellow of ACM. He is a Ph.D. student of the Department of Communications and Information Security at Omsk State Technical University (OmSTU). He graduated from the Power Engineering Institute of OmSTU in 2009 (summa cum laude). His interests are various aspects of digital communications, Information Systems, Swarm Intelligence, and Ad Hoc Networks. His research activities are also focused on Flying Ad Hoc Networks and self-organizing networks of Unmanned Aerial Vehicles (UAVs).