Global NEST Journal, Vol 26, No 6, 06108

Reliable air quality assessment in dynamic environments using reward-learning routing with flying manet technology

K. Karthick* and Dr.R.Asokan

Assistant Professor, Department of Computer Science and Engineering Kongunadu College of Engineering and Technology Trichy, Tamil Nadu, India

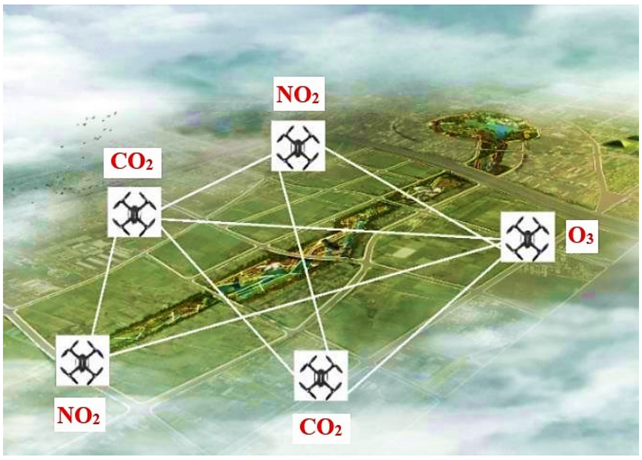
Professor, Department of Electronics and Communication Engineering Kongunadu College of Engineering and Technology Trichy, Tamil Nadu, India

Received: 25/04/2024, Accepted: 28/06/2024, Available online: 28/06/2024

*to whom all correspondence should be addressed: e-mail: karthivel.me@gmail.com

https://doi.org/10.30955/gnj.006108

Graphical abstract



Abstract

Air pollution poses significant threats to ecosystems, atmosphere, and the health of both humans and wildlife. It is primarily caused by industrial and chemical pollutants, leading to deterioration in air, water, and soil quality. As a result, effective air quality monitoring is crucial for environmental management. Flying Mobile Ad hoc Networks (FMANETs) offer a promising solution for intelligent air quality monitoring and assessment by employing unmanned aerial vehicles (UAVs) to measure air pollutants. However, FMANET-based systems face unique challenges, including UAVs' three-dimensional movement, high dynamism, rapid topological changes, resource constraints, and low UAV density. Therefore, routing optimization becomes a fundamental issue in these systems. In this paper, we propose a novel routing method named RFMAN (Reward-learning-based FMANET) for intelligent air quality monitoring systems. RFMAN consists of two main components: route discovery and route maintenance. In the route discovery phase, we introduce a Reward-learning-based mechanism to identify optimal routes, while also implementing a filtering parameter to refine the UAV selection process and reduce the search space. The route maintenance phase of RFMAN focuses on detecting and correcting paths nearing breakdowns, as well as promptly replacing failed paths to ensure continuous monitoring efficiency. The result section pictures that RFMAN outperforms other methods. The training dataset consists of 81,334 samples with the distribution: 57.2% for 1-AQIs, 16.2% for 2-AQIs, 10.7% for 3-AQIs, 7.8% for 4-AQIs, 7.8% for 5-AQIs, and 6.5% for 6-AQIs. The testing dataset consists of 13,432 samples with the distribution: 54.9% for 1-AQIs, 17.8% for 2-AQIs, 12.1% for 3-AQIs, 8.0% for 4-AQIs, 6.7% for 5-AQIs, and 5.6% for 6-AQIs. Overall, RFMAN offers a promising solution for enhancing the effectiveness and reliability of air quality monitoring in FMANET-based systems.

Keywords: Air pollution, FMANETs, UAVs, routing optimization, reward learning, environmental monitoring

1. Introduction

Technology has advanced worldwide due to economic growth, but these developments have also pushed chemical and industry infections that harm the environment and the well-being of people, especially when it comes to agricultural goods, insects, and creatures (Simo *et al*. 2021) (Lambey *et al*. 2021). A variety of air pollutants, such as NO_2, CO, PM, O_3, CO_2 are often measured in order to assess air quality requirements. These contaminants seriously harm people's health. They eventually harm a person's lung function and cause an early death (Singh *et al*. 2021). For this purpose, evaluation of air quality is critical. Environmental groups and governmental agencies have examined this topic. Conventional air quality surveillance techniques rely on stationary observation sites. However, such remedies face several obstacles and constraints since such stations are capable of covering a small region and establishing them is not easy and expensive (Javaheri *et al*. 2018) (Heikalabad *et al*. 2016) (Mesbahi *et al*. 2017). Air quality in urban, rural, and commercial environments is now managed utilizing sophisticated networking techniques including Fying Ad Hoc Networks (FANETs), Vehicular Ad Hoc Networks (VANETs), Internet of Things (IoT), Wireless

Sensor Networks (WSNs) (Yousefpoor *et al*. 2021) (Barati *et al*. 2020). FANETs are particularly significant since they may span a large operating atmosphere and might be installed affordably and effortlessly (Shahzadi *et al*. 2021) (Lakew *et al*. 2020).

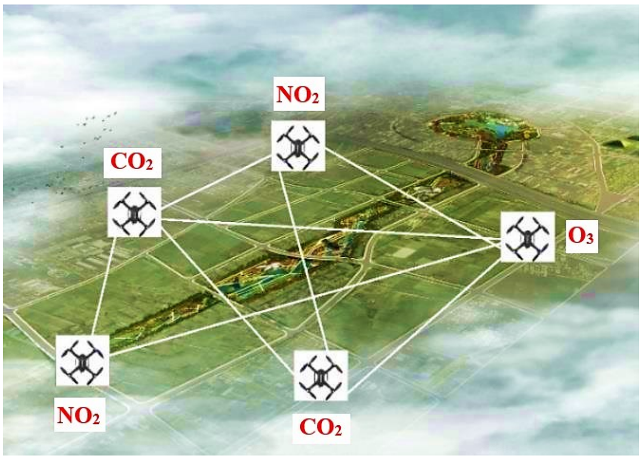


Figure 1. FANETs for measuring air quality

FANET is a loosely structured network of UAVs were seen as a potential method of intelligent air quality surveillance. UAVs travel in three dimensions at a typical speed of thirty to forty-six km/h. UAVs have high computational capacity and energy supplies since they demand to energy to fly, which is significantly greater than the energy required for analyzing data (Nawaz *et al*. 2021). Yet, energy is a significant problem for tiny drones. The FANETs application for AQIs monitoring is shown in Figure 1. Currently, several researchers are looking at using UAVs for detecting air pollution intelligently (Alsamhi *et al*. 2021). For instance, the authors of (Koo *et al*. 2012) have provided a part for the UAV framework that uses real-time air pollution management. (Smidl *et al*. 2013) combines UAVs and wireless stations to continuously track environmental metrics near highways. In (Zang *et al*. 2012), the researchers described trials to test water quality utilizing UAVs in southwest China. The authors of (Burgues *et al*. 2020) stressed the value of using UAVs to keep an eye out for dangerous contaminants like chemical leaks. Furthermore, (A1 tahtawi *et al*. 2019) combined UAVs and IoT sensors to track air quality indicators. Every unmanned aerial vehicle (UAV) in the FANET-based atmospheric surveillance system is outfitted with sensors and specific detectors to gauge air contaminants. With the use of this technology, unmanned aerial vehicles (UAVs) may gather environmental information and transmit to ground station (GS).

FANETs contain unique characteristics such as low densities and quickly alterations to UAV situations, including as deleting and adding UAVs. When information pertaining to the detection of air pollutants is sent to a GSs, certain aspects result in connection failure. In a FANET-based intelligent surveillance of air quality system, the duration necessary to transfer information packets from source to target ought to be quick, with no packet collisions. Implementing an effective routing technique for this intelligent system is difficult because of the unique properties of FANETs (Vijitha *et al*. 2022). Sending and receiving data among two UAVs, referred to as the source and destination UAVs, is known as routing. An suitable routing procedure enables the computerized air quality monitoring system to manage air quality in an area in a consistent and consistent manner. Many studies have developed hybrids, position-based, topology-based, and nature-inspired navigation methods for networks of ad hoc devices (Oubbati *et al*. 2019) (Azevedo *et al*. 2020). The route determination problem in FANETs is unable to sufficiently solved by currently available conventional networking techniques. The routing issue in FANETs has been the subject of multiple investigation efforts lately, including AI and ML techniques (Rezwan *et al*. 2021). Thus, UAVs may autonomously pick up the FANETs transporting network. One effective artificial intelligence (AI) technique that may develop routing strategies in FANETs independently and effectively is RLs. Real-language learning (RL) is popular because it uses the trial-and-error approach. The algorithm is made up of an agent and an environment (Yousefpoor *et al*. 2021). In a reinforcement learning-based forwarded approach, the agent investigates the network's circumstances and takes appropriate actions to determine the optimum routing scheme. To do this, an agent has to determine the best paths among multiple source-destination combinations while keeping into consideration a variety of optimisation parameters.

(Rahmani *et al*. 2021) Typically, this method leverages nodes' local knowledge to reduce energy usage and enhance network connectivity. (Yousefpoor *et al*. 2019) To obtain an efficient routing approach, the whole system must be identified.

In this study, we introduce QFAN, a Q-learning-based forwarding method for the AQI surveillance system utilizing FANETs. QFAN consists of two parts: route discovery and route maintenance.

Route discovery: RFMANs employs a Reward-Learning-based route discovery technique to found path among source and destinations pairs. The RREQ as an agent. Moreover, the network serves as the educational setting. The agent's status describes the node that received the RREQ msg. We provide the filtering parameter based on a number of variables, including the direction of travel, energy remained connection quality, and separation, in order to narrow down the search space throughout the route finding process. The agent may therefore respond optimally (i.e., via the optimum path) in a shorter amount of time. A reward function is also used by QFAN, and it takes into account three factors: latency, hop count, and route ftness.

Route maintenance is divided into two sections. During the first phase, RFMAN identifies and fixes routes that are about to fail. This aids RFMAN in maintaining uninterrupted data delivery. In the second section, QFAN swiftly identifies broken routes and replaces them with other pathways to reduce latency throughout the information transfer processes.

The major novel contributions of the proposed model are,

The novel contribution of RFMAN lies in its incorporation of a reward-learning mechanism into the route discovery phase. This mechanism enables the UAVs to adaptively learn and identify optimal routes for air quality monitoring based on past experiences and environmental conditions, thus enhancing the efficiency of route selection in dynamic FMANET environments.

RFMAN introduces a filtering parameter during route discovery to refine the selection process of UAVs, thereby reducing the search space and improving the overall efficiency of route determination by ensuring that only the most suitable UAVs are chosen for air quality monitoring tasks.

Dynamic route maintenance strategy, which focuses on detecting and correcting paths nearing breakdowns in real-time. By promptly replacing failed paths and adapting to rapid topological changes, RFMAN ensures continuous monitoring efficiency in FMANET-based systems.

The paper is arranged as following: A few FANET routing techniques are explored in "Related works." Reinforcement learning is introduced in "Basic concepts" and is employed in RFMAN design along with proposed model in this section. Finally, "Conclusion" serves as the paper's last section.

2. Related works

The Q-learning-based topology-aware routing (QTAR) system for FANETs has been described by (Arafat *et al*. 2021). In order to construct dependable pathways, QTARs leverages both single-hop and two-hop neighbor information. The researchers suggest this approach speeds up route computation, enhances path discovery, and enhances next-hop node selections. However, this method adds routing cost and system complexity. QTAR selects the next-hop node by taking into account factors including spatial information, latency, speed, and energy. In addition, QTAR offers a method for estimating the Hello frequency of connecting time by computing the link lifespan. In this procedure, Q-Learning parameters includes rate of learning & rewards ratio are flexibly set in response to network circumstances. FANET-acceptable 3D environments are used to run QTAR.

The Q-Learning-based Geographic route strategy (QGeo) has been proposed by (Jung *et al*. 2017). It is a decentralised method that routes traffic based on local knowledge. This approach aims to cut down on routing overhead and processing time. In this procedure, QGeo relied on geographical data as well as node velocity in the framework while overlooking attributes like energy and latency. This problem might lead to the creation of high-delay, low-energy pathways. Based on the neighbors' movement patterns and geographical separation, this approach specifies the reduction value for Q-learning. Nevertheless, FANET is incompatible with this method as it operates in a two-dimensional environment.

The greedy perimeter stateless router (GPSR) system of AODV has been made available by (Karp *et al*. 2000). GPSR uses the single-hop nearby UAVs' position data to guide its routing choices. GPSR reduces routing overhead and speeds route finding. Greedy networking is used to find the next-hop node. Nevertheless, in dynamical contexts, GPSR ignores factors like node acceleration, energy, and latency. It has restrictions as a consequence of FANETs' selection of routes procedure.

The energy-awared fuzzy sequencing technique for FANETs has been provided by (Lee *et al*. 2021). This approach, which is an enhanced version of AODV, extends the lifespan of the entire network by creating reliable pathways and equitably distributing the consumption of energy across locations. The two components of this route technique include route maintenance and networking path finding. In the first section, each node is assigned a score that determines whether or not to rebroadcast the RREQ signal. Distributing RREQs across the network is the responsibility of only high-score nodes. Furthermore, this approach regulates RREQ food in the network and avoids broadcast storms. A fuzzy system is intended to choose pathways with few hops, high ftness, and low latency while making its decision. The second component entails preventing route failure by recognizing and changing pathways near the failure threshold, as well as reconstructing failed routes by quickly detecting and substituting these paths. This technique is used in a 3D with FANETs. Despite the fact that this method requires a lot for connecting overhead.

Q-learning-based multi-purpose router (QMRs) has been proposed by (Liu *et al*. 2020). By reducing latency, finding the most efficient paths, and using less energy overall, QMRs has enhanced the network's exploration and exploitation mechanism. These nodes in QMRs routinely broadcast hello messages to gather data about nearby UAVs in order to accurately choose the next-hopping vertex. Q The learning rates with discount variable, were modified in QMRs based on network circumstances. This problem makes QMR more adaptable to the changing network. The routing procedure is more precise when certain settings are changed. This lowers the number of network disconnections. The QMR suggests a way to stop route failure. Nevertheless, it operates in a 2D were unsuitable for FANETs. Furthermore, QMRs fails to provide a suitable control mechanism to manage network congestion.

(Oubbati *et al*. 2019) have introduced ECaDs, a routing method for FANETs. Its foundation is AODVs. High-energy UAVs may help with route finding in ECaDs. This aids in ECaD's creation of steady routes. When a node is unable to locate high-energy surrounding nodes to serve as the next-hop node, this approach becomes difficult. Unfortunately, this node is unable to establish a path to with a network. Furthermore, while determining the next-hop a single node ECaDs simply takes into account the energy factor. This approach creates stale pathways in the network and spreads use of energy equitably. Furthermore, it employs a predictive technique to stop connection failure. Three dimensions are used to mimic ECaD.

Q-FANET suggested by (daCosta *et al*. 2021), a Q-learning-based router system, for FANETs. Q-FANET helps reduce network latency. It has been influenced by two techniques: Q-Noise+ and QMR. There are two primary components to it: neighbor discovery and routing. During the neighbor detection procedure, UAVs communicate information at regular intervals. In order to enhance QMRs & Q-Noise+, Q-FANETs uses an enhanced Q-learning method known as Q-Learning+ in the second section. Q-FANETs takes channel quality circumstances into account while calculating Q-values, allowing high-quality connections to achieve greater Q-values. However, this protocol does not take energy into account while determining routing pathways. This method works on a two-dimensional space that is not approved for use with FANETs.

The method of AODVs has been presented by (Perkins *et al*. 1999). It is an on-demand technique. This indicates that routing pathways are created only in response to route requests. A lot of routing strategies from the past decade are influenced by AODV. FANETs' high-speed fying nodes and quick connection failure, among other characteristics, make this approach very difficult to implement. Moreover, whenever AODV finds a new path, it takes a while. A maintenance mechanism is introduced by AODV to identify and fix failing routes. AODV takes a substantial amount of bandwidth and latency to create pathways and repair broken routes, particularly in large-scale networks.

(Rahmani *et al*. 2022) have introduced OLSR+, an improved version of the Optimised Link State Routing Protocols (OLSR) designed as FANETs. They proposed a unique approach for analyzing link lifespan by taking into account many scales, including connectivity effectiveness, geographical distance, comparative speed, and the path of fying stations. A fuzzy logic-based method is also being developed by OLSR+ for selecting multipoint relays (MPRs) inside the network. 3 fuzzy inputs are used in this fuzzy structure: the UAV's available energy, its relationship with its lifespan, and its neighborhood level. It gives every UAV a score. MPRs are chosen based on this score. This approach calculates a routing table using 3 factors: the total amount of hops, path energy of path, and lifespan of path. Packet loss is decreased and reliable, energy-efficient pathways are created via OLSR+. Additionally, this technique is used in a 3D area that is suitable for FANET. On the other hand, OLSR+ has a significant routing expense. since of FANETs' unpredictable topology, updating the neighborhood database and routing database is difficult since it incurs significant latency and communications cost.

A FANET-based Q-learning-based geographic routing (QLGR) technique has been developed by (Qiu *et al*. 2021). The routing model of this protocol is designed using a multi-agent-learning method. In this paradigm, each drone is considered an agent, with the whole internet serving as a learning mechanism. While the action sets comprises the neighbors of the current UAV, the state gathering set defines network conditions. Using local data such as connection effectiveness, energy, and queuing time, a distributed network in QLGRs determines the optimum relay UAV for every UAV by calculating its ftness. The UAVs exchange regular hello messages in order to collect this data. Following a decision, the surroundings offers the player two incentives: the local reward (LR), which corresponds to the local gets back of neighboring nodes and is based on their interaction and load capability, and the global reward (GR), which shows the UAVs' returns worldwide after that action has been carried out taken. The packet's present position and its destination are the basis for determining GR. This approach provides a rebroadcast management system for determining the broadcast frequency of greeting messages as well as managing network connection costs. The protocol at issue is not suitable for FANET as it operates in two dimensions. Table 1 presents the key advantages and disadvantages of the relevant studies and contrasts their plans with ours.

2.1. Basic perception

We briefly explain Q-leaning and the systemic framework in this section as we will be using those concepts to create the routing finding process in QFAN.

2.2. Learning based on reinforcement

One effective and reliable tool for artificial intelligence (AI) is reinforcement learning (RL). The two primary components of RL are the environment and the agent. This method involves the agent exploring the surroundings via a variety of activities (Jang *et al*. 2019). The MDPs, a helpful framework for modeling a variety of problems using adaptive programming and reinforcement learning approaches, is used to choose the course of action. It facilitates the agent's random control over processes. Typically, MDP is determined by four factors, denoted by ¬ (S, A, p, r). In order for S and A to denote the fnite status and activity sets, respectively. Additionally, p defines the chance that the current scenario will be replaced by the subsequent state once action a has been performed. Additionally, r is a prize provided by the surroundings after the task has been accomplished. On Figure 2, the agent acts on its most recent phases_t to get the incentive rt and acquires a state with new s_{t+1} from the circumstances at t. The living thing is trying to figure out which policy (π) maximizes the benefit it receives from its surroundings. The predicted discounted total reward has to be raised via max $\sum_{t=0}^{T} \delta r_t\left(s_t, \pi\left(s_t\right)\right)$ in the long run. The discount factor, δ ∈ [0, 1], represents the relevance of the incentive and the agent's effort in learning the environment. δ = 0.7 in the suggested methodology. When the likelihoods of transitions (i.e. p) are established, the Bellman formula known as the Q-function (Eq. 1) is used to conduct the next action at+1 utilizing MDP, due to the postponed reward.

$$Q\left(s_t, a_t\right) = (1-\alpha) Q\left(s_t, a_t\right) + \alpha\left[r + \delta\left(maxQ\left(s_t, a_t\right)\right)\right] \quad (1)$$

The learning rate, denoted by α, has a range of 0 to 1. α (also known as step size) is a Q-learning parameter. It ranges from 0 to 1. The Q-values were not updating if α = 0. Alternatively, if α = 1, the learning procedure is highly quick and the agent concentrates exclusively on the most recent knowledge while researching options. In actuality, researchers often take into account a constant training rate, such α = 0.136. The suggested approach uses α = 0.1.

With the use of Equation 1's Q-function, the Q-Learning offers a model-free and off-policies reinforcement learning approach. Learning optimal behaviors is aided by it (Sutton *et al*. 1998) (Rahmani *et al*. 2021). A table, or Q-Table, is used by Q-learning to store all optimum state-action pairings. This table has one output, known as the Q-value, and two inputs, or the state-action pairs. The goal of Q-Learning is to maximize Q-value. In order to do this, agents use the outside world to choose the optimal course of action. The agent evaluates whether a certain action is suitable given the present situation based on the learning operations. The agent selects a better course of action in the subsequent phase determined by this choice. In each step, Q-values are updated in accordance with Equation (1). Then, the agent takes advantage of the surroundings by acting in a way that maximizes its Q-value. It is said that this approach is "greedy." According to this strategy, the agent will investigate or take advantage of the surroundings based on the probability value. Research Gaps Identified are,

- Many techniques, such as QTARs, QGeo, and Q-FANETs, are predominantly implemented in 2D settings, which may not fully capture the complexities of 3D environments like FANETs. This limitation affects the practical application of these methods in real-world scenarios involving UAVs.
- Techniques like AGeos and QMRs often overlook crucial factors such as the residual energy of nodes and resource parameters, leading to potential instability and inefficient routing. The lack of consideration for energy during the routing process can result in rapid depletion of node resources and reduced network lifespan.
- Several methods, including QLSRs and QFANs, face significant computational difficulties and elevated routing overheads. The complexity in synchronizing agents and the high latency in route discovery processes hinder their efficiency and scalability, especially in dynamic and resource-constrained environments like FANETs.

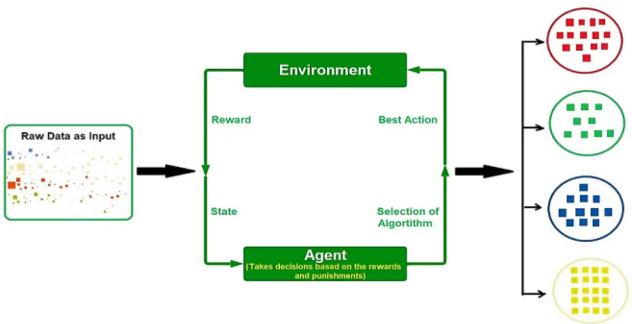


Figure 2. Process of Reinforcement learning

2.3. Framework of network System

The system of networks used in QFAN is shown in this section. This network architecture has many fying nodes and is homogenous. They are arranged in a three-dimensional area and dispersed at random across the structure. These drones have comparable power sources, processing capabilities, and storage capacities. IEEE 802.11n, which has appropriate qualities including fast productivity, a high information rate, acceptance of long-range with strong resistance for disruption, consistent coverage within a region, is the WSN connection in the Mac layer for every UAV (Stallings *et al*. 2012). The fying nodes are constantly moving, and their distance from one another changes quickly. Every UAV is equipped with a unique ID. They have a localization system similar to the GPS installed in them. UAVs are thus always aware of their location and speed. Both UAV-to-UAV and UAV-to-GS connections are used by the QFAN network concept.

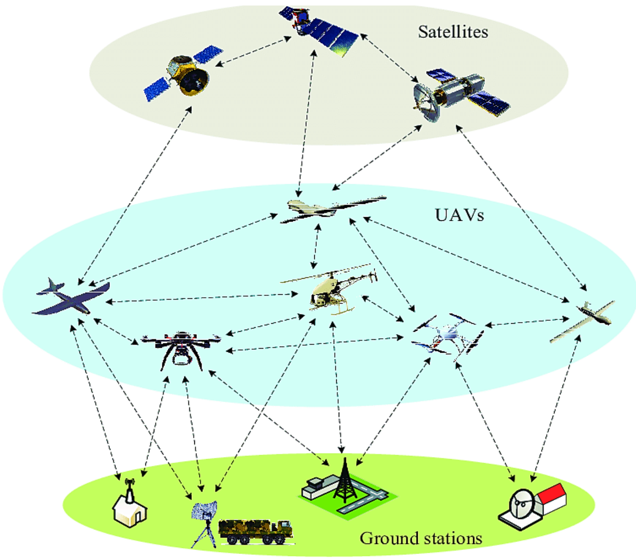


Figure 3. QFANs framework

3. Proposed FMANET based prediction methodology

We introduce QFAN in FANETs, a Q-Learning networking technique, in this section. QFAN enhances the AODVs protocol uses the Q-learning. You should be aware that AODVs is a well-known reactionary technique defined as AODV due to its primary characteristic, which is that it is on-demand. As AODV is helpful and practical, numerous algorithms nowadays, including (Baccour *et al*. 2004), are influenced by it. Unfortunately, AODV cannot be used in FANETs due to its failure to take into account the special characteristics of these networks, such as high-mobility UAVs and regular interruptions. As a result, we work to make this routing protocol better so that it works with FANETs. Different criteria such motion orientation, network reliability, routing delay, energy, and distance between nodes are all taken into account in QFAN in order to generate stable pathways with minimal latency and increase the packet delivery rate (PDR). Using a uniform distribution of energy consumption across network nodes, QFAN aims to increase network lifetime.

There are two sections to QFAN: 1) Finding the route; 2) Maintaining the route.

Table 1. Merits and De- Merits of similar efforts

Techniques	Merits	De-Merits
QTARs	Consider characteristics such as energy, geographic data latency, and navigation by UAVs for finding directions, altering the broadcasting hello interval depending on the way lifespan, modifying the speed of learning and discount percentage dynamically, and applying in a 3D environment.	Excessive overhead as a result of having to know the single-hop & two-hop
AGeos	Discount factor is constantly adjusted by assessing the distance and movements of UAVs.	Ignoring the residual energy of failing nodes and reducing the routing process's latency, creating instability ,using QGeo in a 2D setting, and expanding the Q-table by adding more nodes to the connection.
GPSRs	lowering latency and overhead of routing while building pathways, excellent flexibility	constructing pathways only on the basis of geographical separation, ignoring variables like energy, latency, and connection quality when executing the process, which increases the likelihood of creating instability and low-energy router and is unsuitable for 3D settings like FANETs
Lees	Distribute energy consumption equally in the network, extending networks lifespan, establishing high-energy paths, developing a food controller for path RREQs in the system, anticipating path failure, in 3D environment	Extended time lag in route finding and elevated routing overhead
QMRs	Adaptively modifying Q-Learning settings, establishing an adaptive method to create a trade-off among exploring and profiting, and offering a penalty system for resolving the path hole issue	Not creating a broadcasted control system, not considering resource parameters in router procedure leads to the risk of generating instability paths, and modeling QMR in a two-dimensional setting
ECaDs	deciding on routes while taking the energy of UAVs into account, avoiding low-energy locations in the route building process, and using the routing technique in a three-dimensional space	overflowing routing messages
Q-FANETs	reducing route latency and predicting path failure	Using Q-FANETs in a 2D space, ignoring the factor of energy during routing, the potential for unstable routes to emerge inside the network, and fixed Q-Learning variables
AODVs	Developing an on-demand algorithms and a route preservation concept	Excessive latency and overhead during route creation, the potential for broadcast storms, failing to take the energy parameter into account when determining pathways, and failing to take into account FANET characteristics
OLSR+	Utilizing this routing technique across a three-dimensional space, figuring out linking lifetime depend on conjunction qualities, geographical separation, relatives velocity, and the position of motion of unmanned aerial vehicles; providing a fuzzy logic-based technique for select MPRs that takes into account route energy and direction delaying before making a choice; lowering loss of packets and enhancing a pattern of reliability;	Significant computational difficulty
QLSRs	introducing a multi-agent routing strategy, enhancing the learning algorithm's rate of convergence, offering high flexibility, offering a broadcast management system for greeting messages, and lowering the costs of routing	Significant computational difficulty , challenges synchronizing different agents throughout the training procedure, and using this routing approach in a space with two dimensions
QFANs	Utilizing our scheme into practice in a three-dimensional space, filtering particular states in the restricting search, and speeding up the suggested method's integration; decreasing path creation delay; enhancing adaptability; taking into the variable & UAVs' movement pattern during router process; organizing a preventive; reconstructing failed routes; extending network lifetime;	Elevated routing costs and fixed Q-learning technique factors

3.1. Finding the route

In order to provide a dependable channel for information packet exchange, a source node (U_S) in QFAN looks through its routing record before establishing an encrypted link with a destination node (U_D). If U_S cannot find a valid route among itself and U_D, it employs a Q-Learning-based route process to determine the optimum path ($Best_{Route}$) for transferring information packets. Every fying node (U_i) in this procedure periodically transfers Hello messages with the UAVs that are nearby. This communication includes geographic data (x_i^t, y_i^t, z_i^t), velocity vector ($v_{x,i}^t, v_{y,i}^t, v_{z,i}^t$), delay data, Q-value, and energy (E_i^t) at time t. Ui maintains data on its neighbors in a nearby table. Ui determines the filtering parameter ($S_{filtering}$) for each nearby node, like Uj, using data from this table. This option enables U_i to filter out some of its neighbors who aren't in a suitable state to broadcast RREQ and group the remaining neighbors into a set of licensed neighbours named $Set_{Licensed}$. Ui employs 4 factors (motion direction, remaining energy, connection quality, and distance) to calculate $S_{filtering}$.

Path of motion ($\theta_{i,j}^t$). The angle that U_i has with Uj at time t is calculated by U_i. U_i enhances $S_{filtering}$ according to Uj to obtain increased possibility of being put in $Set_{Licensed}$ while the interaction link among U_i and U_j is valid for a greater duration if the rotation directions of the two networks . But more steady routes are made. Eq (2) determines the angle among U_i and U_j movement vectors.

$$\theta_{i,j}^t = cos^{-1}\left(\frac{v_{xj,}^t v_{x,i}^t + v_{y,j}^t v_{y,i}^t + v_{z,j}^t v_{z,i}^t}{\left|V_i^t\right|\left|V_j^t\right|}\right), 0 \le \theta \le \pi \tag{2}$$

where the velocity matrices of U_i and U_j are, respectively, ($v_{x,i}^t, v_{y,i}^t, v_{z,i}^t$)and($v_{x,j,}^t, v_{y,j,}^t, v_{z,}^t$) Furthermore, the lengths of these matrices are indicated $\left|v_i^t\right|$ & $\left|v_j^t\right|$ By using Eqs. (3) and (4), respectively, they may be derived.

$$\left|v_i^t\right| = \sqrt{\left(v_{x,i}^t\right)^2 + \left(v_{y,i}^t\right)^2 \left(v_{z,i}^t\right)^2} \tag{3}$$

$$\left|v_j^t\right| = \sqrt{\left(v_{x,j}^t\right)^2 + \left(v_{y,j}^t\right)^2 \left(v_{z,j}^t\right)^2} \tag{4}$$

The remaining energy (E_j^t) To enhance the likelihood of being put in $Set_{Licensed}$, U_i boosts Sfiltering in accordance with high-energy adjacent nodes. The goal of selecting $E_i^t$is to equally distribute consumption of energy across the entire system. This value indicates that secure route formation and path discovery are the jobs of high-energy networks.

Quality of link ($Q_{i,j}^t$). Ui improves $S_{filtering}$ matching neighbors with superior connection quality in comparison to other neighbors. This raises the likelihood that they will be assigned to $Set_{Licensed}$. The selection of ($Q_{i,j}^t$) is done in order to provide superior pathways. When there is a poor connection between U_i and U_j the route becomes unstable. This results in path failure. The received signal strength indication (RSSI) is assumed to be the method used in QFAN to assess the connection quality. It can accurately measure connection quality since research indicates that greater RSSI results in an increased PDRs in both the receptor and the transmitters (Lowrance *et al*. 2017). Furthermore, RSSI is stable for a short period of time (around 2 seconds) and has a standard variance about 1 dBm (Vlavianos *et al*. 2008). For packet reception, we consider radio transmitters have RSSI records to determine signal strength.

Distance ($D_{i,j}^t$). In case U_i distance from its neighbor, Uj, is acceptable, U_i will enhance $S_{filtering}$ in line with U_j. However, U_j has a higher likelihood of being in SetLicensed. If U_j is chosen as a relay node, the number of hops in the network increases, which makes it harder for U_i to find the way. Consequently, a sufficient distance indicates that U_i and U_j are not particularly near to one another. Conversely, the appropriate distance indicates that U_i and U_j are not too far apart since these two networks leave their respective communication range very rapidly. As a result, the route these nodes generate will soon become invalid. The appropriate distance, denoted as ($D_{i,j}^t$) lies among D_{Min} and D_{Max}, with $0 \le D_{Min} < D_{Max}$ and $0 < D_{Max} \le R$. The transmission radius (R) of UAVs inside the network is shown. U_i and U_j may create stable routes in this situation. The Euclidian distance, denoted as ($D_{i,j}^t$), among U_i and U_j may be calculated using Eq (5).

$$D_{i,j}^t = \begin{cases} 1 - \frac{\left|\sqrt{\left(x_i^t - x_j^t\right)^2 + \left(y_i^t - y_j^t\right)^2 + \left(z_i^t - z_j^t\right)^2}\right| 0 \le D_{i,j}^t < D_{min}}{D_{min}} \\ 1 - \frac{\left|\sqrt{\left(x_i^t - x_j^t\right)^2 + \left(y_i^t - y_j^t\right)^2 + \left(z_i^t - z_j^t\right)^2} - D_{Max}\right| D_{Max} < D_{i,j}^t \le R}{R - D_{Max}} \end{cases} \tag{5}$$

where the geographic coordinates of U_i and U_j are indicated by the variables x_i^t, y_i^t, z_i^t , and x_j^t, y_j^t, z_j^t respectively. After determining those variables, U_i uses Eq. (6) to determine Sfilteringj, which corresponds to U_j:

$$S_{filtering} = \omega_1\left(1 - \frac{\theta_{i,j}^t}{\pi}\right) + \omega_2\left(\frac{E_j^t - E_{min}}{E_{max} - E_{min}}\right) + \omega_3\left(\frac{Q_{i,j}^t - Q_{min}}{Q_{max} - Q_{min}}\right) + \omega_4\left(D_{i,j}^t\right) \tag{6}$$

where $E_{max} > 0$ is the main energy of UAVs and $E_{min} >= 0$ is the minimum energy (20% of initial energy). Furthermore, the lowest quality of the connection and greatest link quality among two separate nodes are denoted by $Q_{min} \ge 0$ and $Q_{max} > 0$ accordingly. PDR is 0 whenever the RSSI is zero, per [39]. PDR is also almost 99% when RSSI is -87 dBm. These amounts are referred to as Q_{min} and Q_{max}, respectively. ω1 + ω2 + ω3 + ω4 = 1 are the weight coefficients. These coefficients show how each element affects $S_{filtering}$ in Eq (6). It's crucial to remember that all weight coefficients must add up to one. This work takes comparable values for these coefficients into consideration, i.e., ω1 = ω2 = ω3 = ω4 = 1 4. The impact of these coefficients is outside the boundaries of this work (Holicky *et al*. 2013).

Eq (7) has allowed us to standardize $S_{filterin}$. It calculates the difference between $S_{filteringj's}$ average value & SD. A value of $S_{filtering}$ above the average is greater than the amount of $S_{filteringj}$ below the normal range. A population's mean is subtracted from each person's raw rating to get a standardized evaluation, which is subsequently splited by the population's standard variance.

$$S_{standard} = \frac{S_{filtering} - \mu}{\sigma} \tag{7}$$

The variables μ & σ represented as average & standard error of their $S_{filtering}$ of *η neighbor* respectively.

$$\mu = \frac{1}{\eta\, neighbor} \sum_{i=1}^{\eta\, neighbor} S_{filtering} \tag{8}$$

$$\sigma = \sqrt{\frac{1}{\eta\, neighbor} \sum_{i=1}^{\eta\, neighbor} \left(S_{filtering} - \mu\right)^2} \tag{9}$$

where *η neighbor* is the count of U_i neighbors

At this point, Uj is put in $Set_{Licensed}$ if $S_{Standardj} \geq 0$. Every RREQ message functions as an agent in QFAN. QFAN's RREQ format is comparable to that of the AODV protocol. The two extra fields, F_{Route} and T_{Route}, are included anyway. The path's fitness, which is in [0, 1], is represented by F_{Route}. The value of the filtering factor of every node that form a route will be equivalent to its minimum value. This parameter is computed using Eq (10):

$$F_{Routei} = min_{U_{j \in Routei}} \left(S_{filtering}\right) j = 1, \ldots m_i \tag{10}$$

where M_i represents the Uj number in $Route_i$

The amount of time required to send RREQ through U_S to U_D is used to determine the route interruption, or T_{Route}. Eq (11) is used to compute this factor:

$$T_{Route\ i} = \sum_{i=1}^{M_i} t_{i,j} + 1 \tag{11}$$

M_i is the total number of $Route_i$ fying connections. Additionally, the single-hop latency among 2 networks, such as U_i & U_{i+1}, time is indicated by ti, t_{i+1}. This transfer RREQ from U_i and U_{i+1}. As per [29] two parameters were queuing delay ($D_{quei,\ i+1}$) and the medium's accessibility latency $D_{maci,\ j+1}$—are used to determine $t_{i,\ i+1}$. The time required by the medium access technique to send this message or remove the duplicate message is $D_{maci,\ j+1}$. The time it takes for a message to get from the transferred queued is shown by Dquei, i+1. which is dependent on velocity of light and relatively. Consequently, $t_{i,\ i+1}$ is expressed using Eq. (12).

$$t_{i,i+1} = D_{quei,i+1} + D_{maci,j+1} \tag{12}$$

The fying node, or U_i, that has obtained the RREQ msg is represented by the agent's state. The action taken in this procedure denotes a group of licensed nearby nodes (U_{negi} belongs to $Set_{Licensed}$) that are capable of receiving the RREQ msg in its current condition (U_i). The set A = ($U_i \rightarrow U_{neg1}$, $U_i \rightarrow U_{neg2}$, $U_i \rightarrow U_{negk}$) is shown. When the agent (the RREQ msg) is in the state Ui and performs the action $U_i \rightarrow U_j$, the state is changed to U_j. We initialize several Q-Learning parameters, such as the rate of learning (α), discount rate (γ), and Q-value, as begin this path finding procedure. We show you how to get the reward function in the following. It plays a significant role in Q-Learning. Te reward function shows how the world reacts to the action that was taken in state U_t. When converting RREQ between U_S to U_D, the suggested methodology for learning has to optimize the reward values. Three scales are taken into account in QFAN in order to build the incentive system for determining the best way ($Best_{Route}$) to transport information packets from U_S to U_D: routing time (T_{Route}), hop_{count}, and routing fitness. However, the route with the lowest hop count, minimum latency, and maximum fitness is called $Best_{Route}$.

- F_{Route}: When F_{Route} be the identified route is composed of high-energy nodes & linkages. Moreover, there is a suitable distance among these network and they move in similar directions. This guarantees that the path will stay constant for a considerable amount of time
- Hop_{count}: A route has reduced latency while transmitting data packets when its hopcount is low. It improves QFAN's efficiency.
- T_{Route}: It is an important aspect in determining diverse routes in FANETs since the rapid speed of fying vertex generates connection with un-stable pathways. A route may be severed before data is sent to U_D if there is a significant delay in the connection. This causes the network to lose more packets.

Consequently, the reward function is determined using Eq (13):

$$reward\left(U_t, a_t\right) = \begin{cases} R_{max} & U_{t+1}\ is\, destination \\ R_{min} & U_{t+1}\ is\, local\, minimum \\ 1 - \dfrac{T_{route}}{\max\left(T_{route}\right)} + \left(1 - \dfrac{hop_{count}}{N-1}\right) + F_{Route} & otherwise \end{cases} \tag{13}$$

where T_{Route} denotes the path's latency from the U_S to U_{t+1}. The highest delay in the pathways to U_{t+1}that have been found is max (T_{Route}). Additionally, hopcount shows how many hops there are in the journey. In RREQ, this parameter is added. N represents all of the UAVs. In terms of the reward function, the connection among U_t & U_{t+1} receives their value, if U_D is the next-hop node. At a local minimum, state farthest from U_D compared to the current network get the lowest possible reward. The route latency, hop count, and route ftness are used in various modes to assess the reward function.

In QFAN, conditions of the learning algorithm needs to find the optimal route, or $Best_{Route}$. Where this process is run 5 times were their answer stays unchanged, it indicates that the technique were arrived at obtained ideal response ($Best_{Route}$). The innovative route is recorded in United States' routing table. This operation's pseudo-code is expressed in Algorithm 1.

Algorithm 1: Reward learning based route discovery

Inputs: RREP msg,
RREQ msg
U_s, U_d, U_i i=1,2…N N: Nodes of flying
Outcome: B_{Route}
For i= 1 to n
U_i= Hello msg minate from the neighbor UAVs
U_i = record the data in the neighbor table
U_i= Evaluate $S_{filtering}$ in eq 6
U_i=standard $S_{filtering}$ in eq 7
For j=1 to $n_{neighbor}$
If $S_{standard} \geq 0_{then}$
U_i: insert $U_j$in $set_{Licensed}$
End if
End for
If U_s ->U_D transmit the packet
U_s evaluate the router table with an path U_D
U_s ->U_D transmit the data
Else
While condition of convergence is not met
If No data about neighbor
End if
Choose Q from $set_{lincensed}$
Update the Q(value, table)
End while
U_s data packets send to U_D
End if
Endif
End

Route maintenance: with the unique features of FANETs, includes their highly dynamic topology and fast UAVs, the current pathways could become disconnected. The aim of the network restoration procedure is to find a failing path as soon as possible and replacing it with a fresh way to facilitate packet transfers. The generated route has to be adjusted when any of the following scenarios occur in order to prevent it from failing.

Stage 1: When U_i is dying on a route, its energy level is lesser than the threshold $E_{Threshold}$ ($E_{Ui} < E_{Threshold}$). This route should be changed since U_i is unable to complete the data transmission operation.

Stage 2: When a route's buffer capacity reaches the overflow state, it indicates that the amount of traffic on that path exceeds the traffic threshold $Tr_{Threshold}$, or $Tr_{Ui} > Tr_{Threshold}$. As a result, the route has a significant delay while sending data since it is obstructed. This route should thus be changed since it is unable to forward information packets.

Stage 3: The connection quality of the communication link among U_i and U_j is less than the quality threshold $Q_{Threshold}$ (i.e., $Q_{Linki}–j < Q_{Threshold}$), causing the link to withdraw in that route. This route needs to be adjusted as a result.

The reward associated with U_i will be equivalent to R_{min} if any combination of the three scenarios happens. U_i scans its routing database for self-existing routes. U_i gives its previous-hop nodes incentive feedback. The previous-hop network 10, utilizing the Q-Learning approach, chooses a new UAVs to be the next-hop & adjusts the path. Until the new path is found and utilized in its place, the old one is still operational and is utilized to transfer information packets. After creating the new path, packets of information flow via it instead of the previous one. UAVs must regularly adjust the paths that are kept in the routing database for the purpose to identify and eliminate malfunctioning routes.

To do this, U_S regularly transmits routing control data to U_D via the previously determined route. This route is still acceptable if U_D receives this message correctly. U_S receives the ACK message from U_D. Otherwise, if U_D does not get the communication on time, U_S concludes that the path has been severed. The failed UAV's value in rewards will now be equivalent to R_{min}. To enable its previous-hop UAV to correct the flawed path, this particular node sends this information to it. The previous-hop node selects a fresh UAV to be the next stop on the path and builds an alternate path using Q-Learning.

Algorithm 2 : Maintaining the route

Inputs: U_i=i=1,2,…N
Outcome: path of alternation
1.For i= 1 to N
2. If $E_{Ui}< E_{threshold}$ or Q_{link} i-j< $Q_{threshold}$ or $T_{Ui}< T_{threshold}$
3. U_i: set rewards
4. U_i : update q values
5. $U_{Prev\text{-}hop}$: $U_{alternative}$
6. End if
7. End for
8.U_s->send the control msg to U_D
9.If the path were credible
10.U_D: Ack send to U_s
11.else Go to 8
12.U_s:failed to UAV as R_{min}
13.U_s:Updating the Q-table
14. U_s: choose $U_{alternate}$ to next hop
15. end if
16 end

4. Experimental Setup and interpretation

We utilized Keras version 2.4.3 & Tensorflow version 2.3.0 for our evaluation and experimentation. With 178,992 photos in the whole dataset, 80% are used for model training, 15% are used for testing, and 5% are used for validation. Because we have a larger set of information we choose to utilize a lesser proportion for the testing and validation of databases. By using this strategy, we can use the vast quantity of information for training. Hyper-parameter tuning was performed using the grid searching strategy on the validated database. The learning rate, batch size, and optimization employed are the hyper-parameters that have been adjusted. The grid search yielded the ideal settings, which are as follows: an acquisition rate of 0.001, with the size of 64 pictures with the optimizered for Adam.

4.1. Evaluation

Over the last 2 years, our proposed methods included environmental and meteorological information from Skopje, North Macedonia. The information on air pollution is gathered from pulse.eco's (https://pulse.eco) API endpoints. Various detector & contaminants at various time steps are covered by the API. Every hour, the data for every sensor is combined due to the uneven timesteps. Each row includes the kind of pollutant, the quantity detected, the time stamp (date and time), as well as data from the detector. During our training phase, our algorithms have solely labeled the photos using PM2.5 pollution.

Additionally, we utilized images from a fixed lens atop the Vodno peak in North Macedonia, not far from Skopje. While air pollution sensors are measuring the precise quality of the air, the camera periodically snaps photographs of the city center. The photographs were categorized into six groups according to the European Union's AQIs determined by observations of PM2.5 concentration in the air, as shown in Table 2.

A perfect system could provide a reliable estimate of air pollution. since the suggested approaches attempt to do this utilizing camera images rather than actual air pollution monitors, it is improper to describe the problem as a regression problem. Thus began with 6 courses, one for every group represented in the AQIs. The problem was alternatively reformed as a matter of binary classification, with AQIs-1 & AQIs-2 belonging to the "not polluted" category and the remaining AQIs indexes to the "polluted" category.

In some models, we have further included meteorological data to differentiate between atmospheric conditions and pollutants. (https://www.worldweatheronline.com) API endpoints provided the meteorological data. Weather definition, humidity, precipitation, accessibility, pressure, visibility of clouds, heat index, UV index, temperature, speed of the wind, and direction are among the data points. More complex methods based on generative algorithms may be included, however for handling missing things pollutant information, we employed a straightforward technique that takes the most recent observed value into account.

Even though we consolidated all six categories into two generic categories, the raw data remains extremely unbalanced, and our models may become overly biased toward non-polluted photos. In order to do this, we used several database balancing strategies and assessed how well they affected the success of classification.

Table 2. Categories for the AQIs are determined by translating PM2.5 data categories to labels in various classification activities

AQIs	PM2.5	6 Labels	Labels of binary outcome
GOOD	0 to 50	1st-AQIs	Not polluted
MODERATE	51 to 100	2nd-AQIs	
Unhealthy of SG	101 to 150	3rd-AQIs	Polluted
Unhealthy	151 to 200	4th-AQIs	
Very unhealthy	201 to 300	5th-AQIs	
dangerous	301 & above	6th-AQIs	

Table 3 illustrates the dataset's distribution when six classes are used. After dividing the six classes into two, Table 4 shows their data distribution. under order to provide context, Figure 4 & 5 depict several polluted & Not polluted instances under unpredictable weather with various times of the day.

Table 3. The set of data is divided into six classes

Database	1-AQIs	2-AQIs	3-AQIs	4-AQIs	5-AQIs	6-AQIs
Training (%)	81,334	22,634	14,967	12,098	9876	63455
	57.2	16.2%	10.7%	7.8%	7.8%	6.5%
Testing (%)	13432	5432	3023	1989	1665	1234
	54.9%	17.8%	12.1%	8.0%	6.7%	5.6%

Table 4. The set of data is divided into two groups. The "not polluted" category is made up of the "very low" and "low" categories, whereas the "polluted" category is made up of the remainder of the categories

Database	Polluted	Non-polluted	Overall
Training %	101,965	42,243	154,198
	74.4%	29.5%	
Testing %	18,672	7809	26,230
	70.6%	31.6%	
Overall			169,567

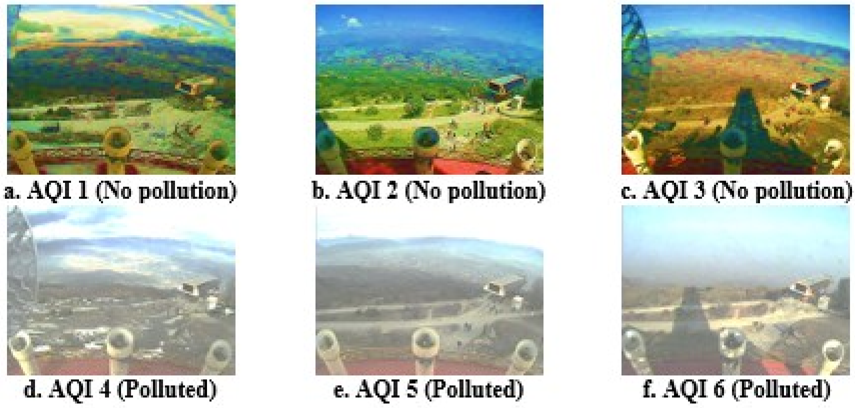


Figure 4. Sample photos from the information taken throughout the entire day in the various sessions

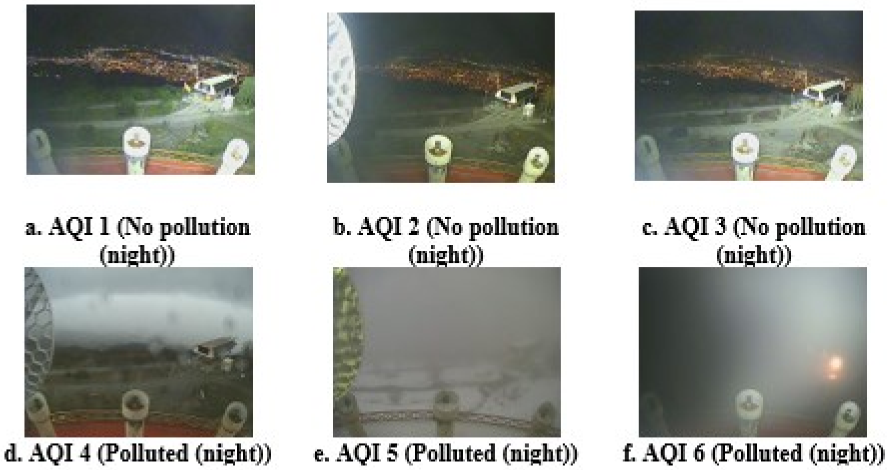


Figure 5. Exemplary photos in the collection from various classes taken at night or in other low-visibility settings

Figure 6 depicts the training and testing precision of various designs as a function of period number. The models' stability is readily apparent, and their effectiveness does not appreciably change over a short period of epochs. The accuracy is around 57%, which is the same as the classes. Accordingly, all photos were trained to be classified "AQIs-1," of their models. One of the primary grounds for assessing the binary classification method—which reduced numerous fields to just two—was that the forecasts are thus useless.

Figure 7 depicts the training and testing precision for the various designs as a function of epoch count and class balance. Based on around 40 epochs of consistent performance, these data verify the stability of the models. Similarly, it is clear that the suggested proposed augmenting of data greatly improved test and training accuracy. It should be noted that while only the training set is balanced in each experiment, the test set is never balanced. That's because, in a production scenario, camera pictures would just be classed instead of undergoing a balancing procedure. We have also implemented a balance strategy by incorporating class weights into the model's training process. With this method, the network may assign distinct learning rate factors to various pictures. Despite its high efficiency in many cases, this approach fared badly in ours, hence we omitted the data.

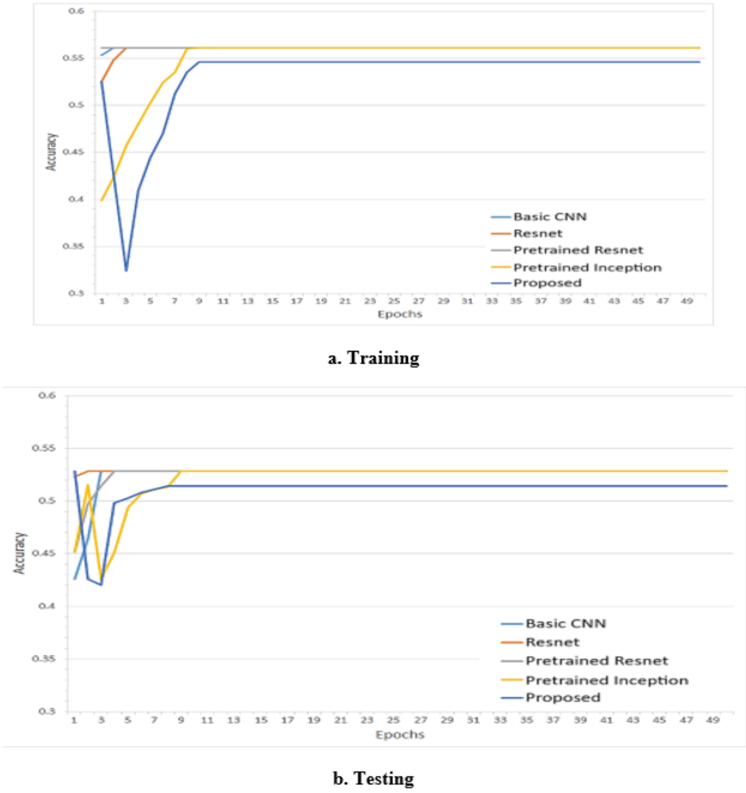


Figure 6. The six-class categorization accuracy of the various architectures

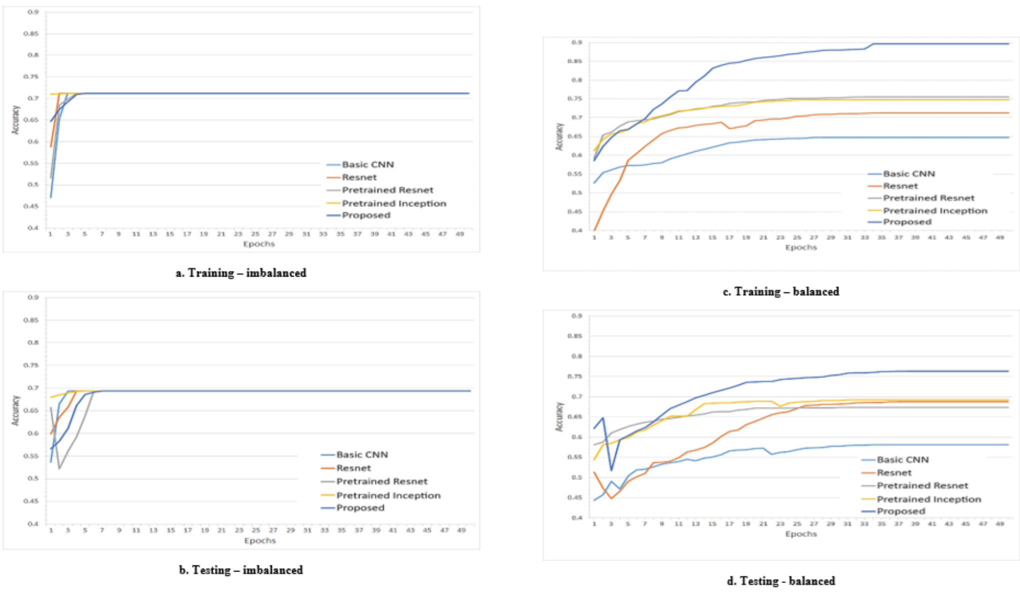


Figure 7. The accuracy of various designs in binary categorization

4.2. Discussion

The experimental results demonstrate that incorporating meteorological data and balancing the dataset significantly enhances the model's classification accuracy. The models showed stability across epochs, maintaining consistent performance, particularly when using data augmentation. Binary classification improved predictive reliability, categorizing images effectively into "polluted" and "not polluted" groups. Despite high computational demands, the proposed approaches successfully leveraged environmental and meteorological information to predict air quality. Future work could focus on refining these methods to further optimize performance and efficiency in real-world applications.

5. Conclusion

In conclusion, this paper introduces RFMAN, an energy-aware Reward-learning-based routing method designed for smart air quality monitoring systems using Flying Ad-hoc Networks (FANETs). RFMAN comprises two main components: route discovery and route maintenance. In the route discovery phase, RFMAN employs a Reward-Learning-based mechanism to select optimal paths between source and destination nodes, enhancing efficiency and response time. Through strategic filtering and neighbor selection, RFMAN effectively narrows down the search space to identify the best path quickly. In the route maintenance phase, RFMAN proactively addresses potential path failures by detecting and correcting paths nearing failure. Additionally, it swiftly identifies and replaces failed paths to ensure uninterrupted monitoring capabilities. Experimental results demonstrate the effectiveness of RFMAN in improving the accuracy and reliability of air quality monitoring systems. While the proposed methods show promise, they exhibit significant computational complexity and high demands on processing power. The imbalanced dataset still poses a challenge, potentially biasing the model towards non-polluted classifications. Additionally, the binary classification approach, although effective, may oversimplify the nuanced variations in air pollution levels.

Future enhancements: As for future work, we plan to explore hybrid models combining Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, aiming to further enhance the performance of our approach. Additionally, we intend to conduct broader experimental analyses with geo-distributed data to validate the scalability and robustness of our methodology.

Annexure

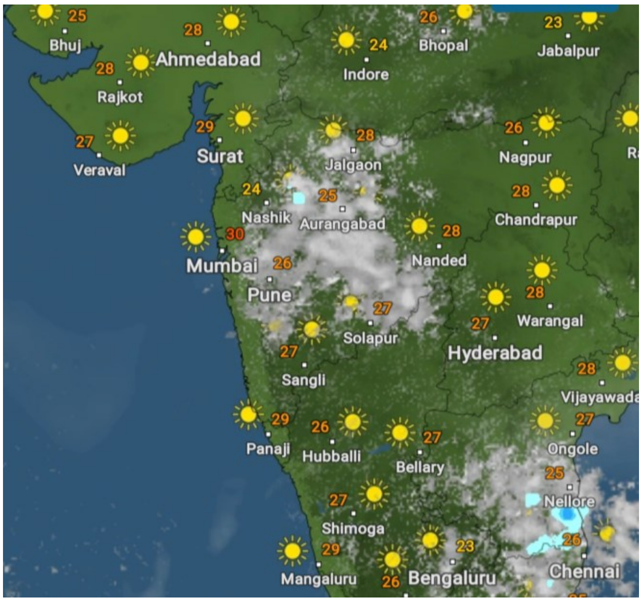


Figure. Location of the camera, sensor, and FMANETs shown on the map

One of the key practical applications of our approach lies in its ability to leverage widely available camera data for air pollution estimation and forecasting, particularly in areas lacking appropriate sensor infrastructure. By correlating image data with sensor data, our methodology offers a versatile solution applicable to various domains where such data fusion can enhance learning tasks. Thereby, our methodology holds promise for applications beyond air quality monitoring, extending to multi-class classification tasks with imbalanced classes and scenarios involving hyperspectral images combined with sensor data. Its adaptability and effectiveness make it a valuable tool for addressing complex environmental monitoring challenges in diverse contexts.

References

Al Tahtawi A.R., Andika E., Yusuf M. and Harjanto W.N. (2019). Design of quadrotor UAV and internet-of-things based air pollution monitoring systems. *International Journal of Information Technology and Electrical Engineering*, **3**(4), 120–127, https://doi.org/10.22146/ijitee.51203.

Alsamhi S.H. (2021). Green internet of things using UAVs in B5G networks: A review of applications and strategies, AdHocNetw,11, 102505. https://doi.org/10.1016/j.adhoc.2021.102505.

Arafat M.Y. and Moh S. (2021). A Q-learning-based topology-aware routing protocol for fying ad hoc networks, *IEEE Internet Tings J*. https://doi.org/10.1109/JIOT.2021.3089759.

Asokan R. and Preethi P. (2021). Deep learning with conceptual view in meta data for content categorization. *In Deep Learning Applications and Intelligent Decision Making in Engineering* 176–191. IGI global.

Azevedo M.I.B., Coutinho C., Toda E.M., Carvalho T.C. and Jailton J. (2020). Wireless communications challenges to fying ad hoc networks (FANET). *Mobile Compute*r. **3**, 25.

Baccour N. (2012) Radio link quality estimation in wireless sensor networks: A survey. *ACM Transactions on Sensor Networks*. **8**(4), 1–33. https:// doi.org/10.1145/2240116.2240123.

Bai D.P. and Preethi P. (2016). Security enhancement of health information exchange based on cloud computing system. *International Journal of Scientific Engineering and Research*, **4**(10), 79–82.

Burgués J. and Marco S. (2020). Environmental chemical sensing using small drones: A review. *Science of the Total Environment*. **748**, 141172, https://doi. org/10.1016/j.scitotenv.2020.141172.

da Costa L.A., L. Kunst R. and de Freitas E.P. (2021). Q-FANET: Improved Q-learning based routing protocol for FANETs. *Computer Networks*. **198**, 108379. https://doi.org/10.1016/j.comnet.2021.108379.

Heikalabad S.R., Navin A.H., Hosseinzadeh M. and Oladghafari T. (2016). Erratum: Midpoint memory: A special memory structure for data-oriented models implementation, *Journal of Circuits, Systems and Computers*, **25**(02), 1692001, https://doi.org/10.1142/S0218126616920018.

Holický M. (2013). Introduction to Probability and Statistics for Engineers https://doi.org/10.1007/978-3-642-38300-7.

Jang B., Kim M., Harerimana G. and Kim J. W. (2019), Q-learning algorithms: A comprehensive classifcation and applications. *IEEE Access* **7**, 133653–133667. https://doi.org/10.1109/ACCESS.2019.2941229.

Javaheri D., Hosseinzadeh M. and Rahmani A. M. (2018). Detection and elimination of spyware and ransomware by intercepting kernellevel system routines, *IEEE Access* **6**, 78321–78332, https://doi.org/10.1109/ACCESS.2018.2884964.

Jung W.S., Yim J. and Ko Y.B. (2017). QGeo: Q-learning-based geographic ad hoc routing protocol for unmanned robotic networks, *IEEE Communications. Letter*. **21**(10), 2258–2261, https://doi.org/10.1109/lcomm.2017.2656879.

Karp B. and Kung H.T. (2000). GPSR: Greedy perimeter stateless routing for wireless networks, In Proceedings of the 6th annual international conference on Mobile computing and networking 243–254, https://doi.org/10.1145/345910.345953.

Koo V. (2012). A new unmanned aerial vehicle synthetic aperture radar for environmental monitoring, *Progress in Electromagnetics Research* **122**, 245–268. https://doi.org/10.2528/PIER11092604.

Lambey V. and Prasad A.D. (2021). A review on air quality measurement using an unmanned aerial vehicle, *Water Air Soil Pollut*ion, **232**(3), 1–32, https://doi.org/10.1007/s11270-020-04973-5.

Lee S.W. (2021). An energy-aware and predictive fuzzy logic-based routing scheme in fying ad hoc networks (fanets). *IEEE Access* **9**, 129977–130005. https://doi.org/10.1109/ACCESS.2021.3111444.

Liu J. (2020). QMR: Q-learning based multi-objective optimization routing protocol for fying ad hoc networks. *Computers. Communucation*. **150**, 304–316, https://doi.org/10.1016/j.comcom.2019.11.011.

Lowrance C.J. and Lauf A.P. (2017). Link quality estimation in ad hoc and mesh networks: A survey and future directions, *Wireless Personal Communications*. **96**(1), 475–508. https://doi.org/10.1007/s11277-017-4180-9.

Mesbahi M.R., Rahmani A.M. and Hosseinzadeh M. (2017). Highly reliable architecture using the 80/20 rule in cloud computing datacenters, *Future Generation Computer Systems*, **77**, 77–86. https://doi.org/10.1016/j.future.2017.06.011.

Oubbati O.S. (2019). ECaD: Energy-efcient routing in fying ad hoc networks. *International Journal of Communication Systems* **32**(18), e4156, https://doi.org/ 10.1002/dac.4156.

Oubbati O.S., Atiquzzaman M., Lorenz P., Tareque M.H. and Hossain M.S. (2019). Routing in fying ad hoc networks: Survey, constraints, and future challenge perspectives, *IEEE Access* **7**, 81057–81105, https://doi.org/10.1109/ACCESS.2019.2923840.

Palanisamy P., Padmanabhan A., Ramasamy A. and Subramaniam S. (2023). Remote patient activity monitoring system by integrating IoT sensors and artificial intelligence techniques. *Sensors*, **23**(13), 5869.

Palanisamy P., Urooj S., Arunachalam R. and Lay-Ekuakille A. (2023). A Novel Prognostic Model Using Chaotic CNN with Hybridized Spoofing for Enhancing Diagnostic Accuracy in Epileptic Seizure Prediction. *Diagnostics*, **13**(21), 3382.

Perkins C.E. and Royer E.M. (1999). Ad-hoc on-demand distance vector routing. In Proceedings WMCSA'99. *Second IEEE Workshop on Mobile Computing Systems and Applications* 90–100. https://doi.org/10.1109/MCSA.1999.749281.

Preethi P. and Asokan R. (2021). Modelling LSUTE: PKE schemes for safeguarding electronic healthcare records over cloud communication environment. *Wireless Personal Communications*, **117**(4), 2695–2711.

Qiu X., Xie Y., Wang Y., Ye L. and Yang Y. (2021). QLGR: A Q-learning-based geographic FANET routing algorithm based on multi-agent reinforcement learning, KSII Trans. *Internet Information Services*. **15**(11), 4244–4274.

Rahmani A.M. (2021) Machine learning (ML) in medicine: Review, applications, and challenges. *Mathematics* **9**(22), 2970. https:// doi.org/10.3390/math9222970.

Rahmani A.M. (2021). An area coverage scheme based on fuzzy logic and shufed frog-leaping algorithm (sfa) in heterogeneous wireless sensor networks, Mathematics **9**(18), 2251. https://doi.org/10.3390/math9182251.

Rahmani A.M. (2022). OLSR+: A new routing method based on fuzzy logic in fying ad-hoc networks (FANETs), *Vehicular communications*. **20**, 100489.

Rezwan S. and Choi W. (2021). A survey on applications of reinforcement learning in fying ad-hoc networks, *Electronics* **10**(4), 449. https:// doi.org/10.3390/electronics10040449.

Simo A., Dzitac S., Dzitac I., Frigura-Iliasa M. and Frigura-Iliasa F.M. (2021). Air quality assessment system based on self-driven drone and LoRaWAN network, *Computer Communications*, **175**, 13–24, https://doi.org/10.1016/j.comcom.2021.04.032.

Singh D., Dahiya M., Kumar R. and Nanda C. (2021). Sensors and systems for air quality assessment monitoring and management: A review. *Journal of Environmental Management*, **289**, 112510. https://doi.org/10.1016/j.jenvman.2021.112510.

Smídl V. and Hofman R. (2012). Tracking of atmospheric release of pollution using unmanned aerial vehicles. *Atmospheric Environment*. **67**, 425–436, https://doi.org/10.1016/j.atmosenv.2012.10.054.

Stallings W. (2004). IEEE 8O2. 11: Wireless LANs from a to n. *IT Profess*ions. **6**(5), 32–37. https://doi.org/10.1109/MITP.2004.62.

Sutton Richard. and Barto. (1998). Andrew. Reinforcement Learning: An Introduction MIT Press.

Vijitha Ananthi J. and Subha Hency Jose P (2022). A review on various routing protocol designing features for fying ad hoc networks, In Mobile Computer, Sustain, Inform, 315–325, https://doi.org/10.1007/978-981-16-1866-6_23.

Vlavianos A., Law L.K., Broustis I., Krishnamurthy S.V. and Faloutsos M. (2008). Assessing link quality in IEEE 802.11 wireless networks: Which is the right metric? In 2008 IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications 1–6, https://doi.org/10.1109/PIMRC.2008.4699837.

Yousefpoor M.S. (2021). Secure data aggregation methods and countermeasures against various attacks in wireless sensor networks: A comprehensive review, *Journal of Network and Computer Applications*, **190**, 10311, https://doi.org/10.1016/j.jnca.2021.103118.

Yousefpoor M.S. and Barati H. (2019), Dynamic key management algorithms in wireless sensor networks: A survey. *Computer Communications*. **134**, 52–69. https://doi.org/10.1016/j.comcom.2018.11.005.

Zang W., Lin J., Wang Y. and Tao H. (2012). Investigating small-scale water pollution with UAV remote sensing technology, In World Automation Congress 2012 1–4.