UAVs Relay in Emergency Communications with Strict Requirements on Quality of Information

Ameer Shakayb Arsalaan*†, Mah-Rukh Fida‡, Hung Nguyen§
*Teletraffic Research Centre, University of Adelaide, Australia
†Department of Computer Science, University of Swabi, Pakistan
‡School of Computing and Engineering, University of Gloucestershire, United Kingdom
§School of Computer and Mathematical Sciences, University of Adelaide, Australia

Abstract—Mobile Ad-hoc Networks (MANETs) are increasingly being used to provide communications in emergencies. Maintaining the required quality of service of information including its timely delivery, in MANETs, is extremely challenging under dynamic conditions such as a changing bushfire scenario. Nodes get destroyed or their link qualities degrade leading to sub-optimal paths, reduced network performance, and disconnected end nodes. To remedy partitioning in ground-based MANETs and to improve their communication performance, Unmanned Aerial Vehicles (UAVs) have emerged as a promising relay solution. This paper presents U-QoIT, a UAV deployment algorithm for connecting the disconnected end nodes in a dynamic bushfire scenario. It aims to meet the Quality of Information requirements, with a minimal number of UAVs deployed. U-QoIT takes the UAV deployment problem as a combinatorial search problem and reduces the deployment complexity by applying a reduction by minimization approach. This is done to deploy fewer UAVs without compromising connectivity. Extensive simulation results depict that U-QoIT significantly improves user usability by 48% compared to the best-performing baseline approach, of Dynamic Service Area (DSA) based UAV deployment, at the expense of a negligible increase in computational time. It achieves a 38% drop in packet latency and retains its out-performance in a highly mobile network. In a sparse network, it improves user usability by 106%. Furthermore, it needs fewer UAVs, compared to the baseline UAV deployment methods, without compromising its performance gains.

Index Terms—MANET, Bushfire, Quality of Information, User-usability, UAV.

1 Introduction

The risk of bushfires has surged due to climate change, notably in Australia, where frequent fires underscore the need for a robust communication network for emergency services. Beyond human loss, bushfires devastate wildlife, destroy habitats, and reduce biodiversity [1], with over 800 million animals dying in the first three months of the 2019-2020, in New South Wales fires alone [2]. Effective communication is vital for rescue operations to mitigate such damage [3], [4].

Mobile Ad Hoc Networks (MANETs) have been widely used for emergency services due to their wireless connectivity and flexibility [5]. However, bushfires pose communication challenges due to reduced Signal-to-Noise Ratio (SNR) from smoke, heat, and ash. Unmanned Aerial Vehicles (UAVs) can extend network coverage in these scenarios, offering a solution by maintaining line-of-sight communication with ground nodes [6], [7].

This study aims to ensure effective MANET communication during bushfires, focusing on scenarios where query nodes require timely, accurate, and reliable sensor information, with strict Quality of Information (QoI) [8], [9], [10] requirements. The goal is to optimize user-usability[1] by establishing and recovering paths from selected information sources. This is achieved using the Quality of Information with Threshold (QoIT) [11], which selects the most suitable sources and routes to meet the communication needs of query nodes. To maintain connectivity, the study proposes a UAV deployment strategy, U-QoIT, designed to enhance user-usability with minimal UAV deployment.

The main contributions of this study are as follows:

1) The UAV placement problem is formulated to maintain QoIT communications in bushfire scenarios, utilizing realistic models for radio propagation, bushfires, and UAVs. The optimal placement of a minimal number of UAVs is demonstrated to be a combinatorial search problem solvable by the reduction by minimization method.
2) A realistic bushfire communication simulation platform is developed, incorporating radio, bushfire, QoIT, and UAV models. The proposed solution is implemented on this testbed, with the code and data made available to support further research in this critical area.[2]
3) U-QoIT is shown to achieve significant improvements over alternative approaches with only a negligible increase in computational time. In a dense static network, U-QoIT reduces packet latency and enhances user usability by 55% and 122%, respectively, compared to a MANET without UAVs [12]. With a small number of UAVs, U-QoIT outperforms the second-best UAV-based MANET approach, DSA [13], by reducing latency by

1. User-usability of a query node is achieved when all its requirements related to the quality of sought information, are fulfilled.

2. For simulation code of U-QoIT implementation, please see the repository git@github.com:AmeerShakaybArsalaan87/U-QoIT.git

38% and improving user usability by up to 48%. These performance gains are maintained even with a substantial reduction in deployed UAVs compared to baseline methods, with user usability in sparse networks increasing by 106% to 200% compared to baseline UAV deployment methods.

The paper is organized as follows. In Section 2, the QoIT and fire spread models are introduced, and a toy example is used to illustrate the problem statement of the research study. In Section 3, the motivation for the study is described by extending the example to include network partitions caused by bushfire spread. Section 4 presents the `1-Path Problem`, which is composed of procedures and an algorithm for deploying UAV(s) on a single broken source-to-query path. The U-QoIT algorithm is presented in Section 5, which identifies potential positions for UAV deployment to inter-connect the maximum number of broken/unestablished source-to-query paths. Section 6 defines the simulation and experimental test bed, and Section 7 discusses the results of the simulation. In Section 8, an overview of related work is provided, with concluding remarks given in Section 9.

2 Background

This paper utilizes the Quality of Information with Thresholds (QoIT) algorithm [11] as the foundational method for selecting source nodes and determining downstream paths to query nodes in a MANET designed to respond to bushfire emergencies. This section provides an overview of (i) QoIT-based path selection, (ii) the fire spread model underlying the bushfire model, and (iii) the problem statement, which outlines network partitioning resulting from the dynamic nature of bushfires.

2.1 QoIT-based Path selection

The QoIT algorithm selects a source node by carefully evaluating detailed information about each source's capabilities and its downstream path to the query node, aiming to meet the specific information quality requirements of the query node's application. This evaluation considers various network-level metrics, such as available bandwidth, path integrity, latency or hop count, and information utility, with performance thresholds established for each metric.

The selection process begins with identifying the relevant network metrics and their threshold values to satisfy the specific requirements of the query node's application. Once these metrics and thresholds are set, the QoIT algorithm proceeds with the following steps:

1) The algorithm evaluates the current network metrics of connected source nodes and their downstream paths toward the query node.
2) For each source-to-query downstream path, QoIT calculates a goodness measure for each network metric, indicating how well it meets the query node's requirements.
3) QoIT then uses these goodness measures to compute quality metric scores related to the completeness, timeliness, accuracy, and reliability of information. Based on these scores, QoIT assesses how well a source and its downstream path can satisfy the quality needs of the query node.
4) In the final step, QoIT selects the source nodes that maximize user usability across all query nodes by ensuring their quality needs are fully met. User usability is defined as the number of querying nodes whose quality requirements are completely fulfilled. For a more detailed description of the QoIT source selection process, readers can refer to [11].

2.2 Model of Fire Spread

To understand performance of MANETs under bushfires, it is important to have accurate models of wildfires and their spreads. Advances in computational power have led to the accurate modelling of the behaviour of wildfires and to simulate their spread across various landscapes. For field based fire spread model in dry eucalypt forest with a shrubby under-storey, Cheney et al. [14] develop an empirical model that predicts the fire spread rate and flame height. The model determines average spread rate by the mean values of environment and fuel variables. Equation 1 provides the potential rate of fire spread based on the proposed model. Here R_{ss} is the predicted quasi-steady rate of spread (mh^{-1}), $7.3 \leq U_{10} \leq 25.9\ km\,h^{-1}$ indicates the mean wind speed at $10\,m$ in open – a standard exposure for forecast wind measurements in Australia; F_j represents fuel attributes, where $j = 1, \ldots, n$, including live or dead mass, compactness, height, horizontal and vertical continuity, size, mineral and heat content of the fuel; M_f depicts dead fine fuel moisture content (%), θ specifies the slope of the fire field terrain in degrees; and $\Phi_i(.)$ denotes the function of wind speed, fuel attributes, moisture content and slope, for $i = 1, \ldots, 4$, respectively.

$$R_{ss} = \Phi_1(U_{10})\Phi_2(F_1,\ F_2,\ \ldots F_n)\Phi_3(M_f)\Phi_4(\theta) \qquad (1)$$

Fuel moisture function $\Phi_3(M_f)$ depends on the relationship of fuel moisture content and spread rate of Jarrah forest fuels [15]. The relationship is developed from data of dry summer burning conditions with surface fuel moisture content, $5.6\% \leq M_{fs} \leq 9.6\%$. For M_{fs} range, the wind effect in [15] is ruled out and $\Phi_3(M_f)$ is formulated as in Equation 2

$$\Phi_3(M_f) = 18.35 M_{fs}^{-1.495}. \qquad (2)$$

With increasing slope the flames contact with the fuel gets closer. This preheats the fuel bed that efficiently ignites the fuel and increases the fire spread rate. The exponential function used by [16] and formulated by [17] is adopted as it is consistent with slope functions applied to other fire behaviour models [18]. The slope function, $\Phi_4(\theta)$, for $\theta \geq 0$ is given in Equation 3

$$\Phi_4(\theta) = exp(0.069\theta). \qquad (3)$$

The wind function $\Phi_1(U_{10})$ and fuel function $\Phi_2(F_1,\ F_2,\ \ldots F_n)$ in Equation 1 is modelled through R_A, in Equation 4. It describes the fire spread prediction system for the wind speed and fuel hazard score variables

$$R_A = R_t + 1.5308(U_{10} - U_t)^{b_1} \cdot FHS_s^{\,b_2} \cdot (FHS_{ns} \cdot H_{ns})^{b_3} \cdot B_1, \quad (4)$$

where $R_t = 30\,m\,h^{-1}$ is the threshold spread rate at the threshold wind speed $U_t = 5\,km\,h^{-1}$ at 7% M_{fs} and $\theta = 0$, $1.3 \leq FHS_s \leq 3.9$ is the surface fuel hazard score, $0.2 \leq FHS_{ns} \leq 3.8$ is the near-surface fuel hazard score, $5 \leq H_{ns} \leq 30\,cm$ is the near-surface fuel height, $b_1 = 0.8576$, $b_2 = 0.9301$, $b_3 = 0.6366$ are the regression constants regarding R_A. The logarithm of $R_A - R_t$ was regressed on $ln(U_{10} - U_t)$, and various forms of (FHS_s, FHS_{ns}, H_{ns}) to obtain the best fitting model. Use of logarithmic regression to estimate fire spread is convenient but leads to bias in the results when transformed back to the original variables. To correct the result, $B_1 = 1.03$, the ratio of the observed mean to the mean of the predicted values [14] is multiplied to it. The final model is generated by incorporating R_A, $\Phi_3(M_f)$ and $\Phi_4(\theta)$ into Equation 1, to predict the potential fire spread rate (R_{ss}).

Though flame height ($1.5 \leq H_f \leq 75\ m$) corresponds with observed fire spread rate R_o ($m\,h^{-1}$), predictions are improved by integrating elevated fuel height (H_e). The model for H_f, presented in Equation 5, is fitted with the linear mixed model to incorporate head fire spread rate, R_A ($m\,h^{-1}$), that changes with slope and is obtained from Equation 4. Here $H_e \in 1m - 55m$ is elevated fuel height, $b_1 = 0.723$, $b_2 = 0.0064$ are the regression constants regarding H_f, and $B_2 = 1.07$ is the bias used for results correction. Please see [14] for more details.

$$H_f = 0.0193 \cdot R_A^{b_1} \cdot exp(b_2 \cdot H_e) \cdot B_2 \tag{5}$$

In our exploration of a MANET designed for rescue and emergency operations within the context of an Australian bush fire, we employ the fire model of Equation 1 to illustrate the dynamics of fire spread, denoted as R_{ss}. Additionally, we utilize Equation 5 from the same reference to characterize flame height (H_f) in our experimental study. The propagation of fire generates smoke and ash, leading to the scattering and absorption of radio signals. As emphasized by S. S. Faria et al. in [19], wildfires can induce significant signal attenuation. In this study, we therefore adopt the perspective that nodes with radio ranges overlapping the fire's range are unable to facilitate communication.

2.3 Problem Statement

The QoIT [11] aims to choose (among the available options) an optimal source node (S) for transmitting information to a query node (i.e., destination (D)) in a challenged ad hoc network. However, the chosen source-to-query (S-D) path may not meet the required values for network layer metrics, such as the path may offer lower information utility (IU), smaller bandwidth (BW), lesser path-integrity (PI), or higher hop-count (HC). Such deficiencies in the network layer metrics can ultimately affect the quality of information, including its *accuracy*, *timeliness*, *reliability*, and *completeness*. Consequently, the number of users whose application requirements are met, decreases.

Query nodes can have different requirements depending on the services they request. For instance, an on-demand video application can tolerate some delay but cannot compromise on path integrity and utility of the information, and have certain bandwidth requirements for video resolution. An example network scenario of ground nodes (g) is shown in Figure 1 where a source node S transmits the requested on-demand video information to the query node D. The S-D path comprises several links, each labeled with its available network metric values $[g_{BW}, g_{IU}, g_{PI}]$, indicating the state of its three network metrics, namely, bandwidth, information utility, and path integrity. Additionally, HC measures the number of links on the S-D path.

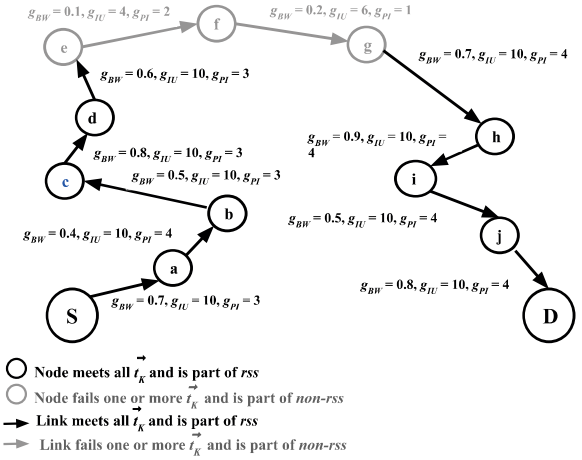

Fig. 1: Network metric values along the links (between ground (g) nodes) on path S-D.

The IU score of an S-D path is determined by the g_{IU} values of the links along the path. It represents the objective value of the data transmitted from the source node to the query node, taking into account factors such as accuracy, relevance, and timeliness of the information provided. It is the measure of how useful or valuable the information is to a query node. In the simulation, the IU score is assumed to range from 0 to 10, as in [20], with 0 representing the lowest and 10 representing the highest information utility score. The PI score, on the other hand, indicates the level of security on the S-D path for data transmission. A set of four keys with varying lengths is assumed, specifically 32-bit, 64-bit, 128-bit, and 256-bit traffic encryption [20], represented by key-IDs 1, 2, 3, and 4, respectively. Each node is randomly assigned one of the given key lengths. The link integrity, g_{PI}, is derived as the minimum strength of the keys held by the adjacent nodes of a link. Path integrity is then determined by the minimum link integrity along the path.

To ensure that the quality requirements for the on-demand video application are met, a threshold vector ($\vec{t}$) of minimum required values for network metrics is used, defined as $\vec{t}$ = [$\vec{t}_{BW}$, $\vec{t}_{HC}$, $\vec{t}_{IU}$, $\vec{t}_{PI}$] = [0.4 $Mbps$, 5 $hops$, 10, 3], as discussed in [11]. The percentage (p) of network metric requirements met by the path S-D (shown in Figure 1) for the given communication requirements are $p_{BW} = 25\%$, $p_{IU} = 40\%$, $p_{PI} = 33.33\%$, and $p_{HC} = 45.5\%$.

For each network metric k = [BW, IU, PI,HC], the ratio of available (a_k) to required ($\vec{t}_k$) network metric values is calculated using the QoIT algorithm [11] to obtain p_k. If any of these ratios are less than 1, it has a negative impact on the quality layer metrics score, as it fails to meet the user's required level of performance and leads to lower user

usability. In short, if at least one link of the path fails to meet one or more network metric thresholds, the user usability drops. For example, in Figure 1, not only does the path S-D exceed t_{HC} by 6 hops, but the links e-f and f-g also fail to meet t_{BW}, t_{IU}, and t_{PI}.

The sub-paths S-a-b-c-d and g-h-i-j-D within the route are considered as `rss` (requirements satisfying sub-paths), while the sub-path e-f-g is considered as `non-rss`. In each S-D pair, an `rss` consists of two sub-paths, one starting from node S and the other ending at node D. In Figure 1, the `non-rss` sub-path e-f-g needs to be replaced with better links to achieve higher user usability.

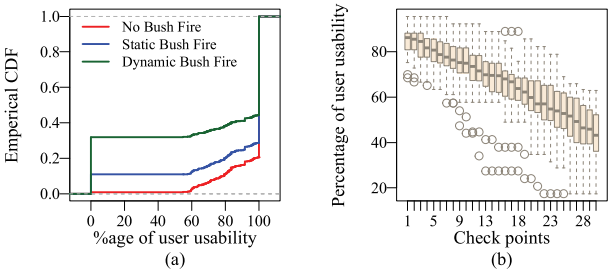

Fig. 3: Performance degradation in MANET with dynamic bush fire scenario.

3 Motivation

The task of meeting user requirements becomes even more challenging in the presence of node mobility and dynamic bushfires. Node mobility causes link breakage, whereas spreading bushfire weakens radio signals and damages nodes within its proximity, as shown in Figure 2 (a). These factors impact communication, reduce user usability, and increase the gap between `rss` sub-paths. For instance, in Figure 2 (a), the requirement failures on links e-f and f-g lead to two `rss` sub-paths: S-a-b-c-d and g-h-i-j-D with $\{p_{BW}, p_{IU}, p_{PI}, p_{HC}\} = \{25\%, 40\%, 33.33\%, 45.45\%\}$. Due to the bushfire's effects on nodes d and g, the `rss` sub-paths shrink to S-a-b-c and h-i-j-D. Additionally, nodes mobility cause links to break for nodes b, h, and i, further reducing the `rss` sub-paths to S-a and j-D.

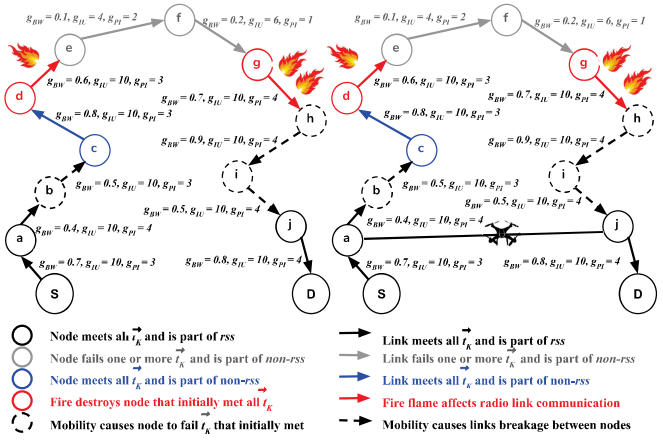

Fig. 2: Fire and node mobility breaks the path and makes it sub-optimal and with (a) depicting rss and $non\text{-}rss$ sub-paths, and (b) re-establishing the paths between rss sub-patch with a UAV.

Figure 3 demonstrates the impact of *no*, *static*, and *dynamic* bushfire scenarios on user usability in a network comprising of 100 static nodes containing 6 source and 20 query (destination) nodes. The simulation environment of [20] is employed in 10 km^2 area and 30 simulations are performed for each scenario. At the start of each simulation, the network paths are established using the network metric values of links chosen from a set of ranges given in [20], whereas the spatial layout of nodes and fires (ignited at 15-25 locations) are chosen randomly. To imitate continuity or variation in topography, environment and fuel bed information, the variable values of the fire spread behavior model (discussed in subsection 2.2) vary with fire locations. Fire spreads from the ignition point in wind direction at the rate of 0.04608 - 12.1824 kmh^{-1}, calculated using the minimum and maximum values of the influencing variables. The fire spread beyond the simulation area is considered to have vanished. Each simulation run executes until all data packets are delivered or the simulation time expires. The time duration for a single simulation scenario is 100 seconds.

In a dynamic bushfire scenario, the paths from the source to the query nodes are evaluated at regular intervals of approximately 3 seconds, known as "checkpoints". If a node or link comes into close contact with the fire, it is considered damaged and immediately excluded from the network for the rest of the simulation. To simplify the process, all query nodes have the same network requirements, represented by the values i.e., $\{t_{BW}, t_{IU}, t_{PI}, t_{HC}\} = \{0.4, 10, 3, 5\}$ respectively. QoIT algorithm [11] is used to determine the source and its optimal path to a given query node. This algorithm considers various network metrics along each potential path and selects the one that best satisfies the query node's requirements. The *user usability* of a query node is determined by comparing the network metric values along its selected path to the corresponding threshold values for each parameter. If all threshold requirements are met along the downstream path from the source to the query node, the node is considered usable.

Figure 3 (a) shows 100% user usability for 80%, 71%, and 55% of the query nodes in case of *no*, *static*, and *dynamic* bushfire scenarios respectively.Additionally, Figure 3 (a) also depicts that among all simulations and iterations of *dynamic* bushfire scenario, 32% of the query nodes have no path towards any of the source nodes. Moreover, Figure 3 (b) indicates that in subsequent iterations of the *dynamic* bushfire scenario, the mean user usability drops from 85% in the first checkpoint to approximately 45% in the last checkpoint. The decrease is due to the loss of S-D paths caused by the dynamic nature of the bushfire.

Broken links and dead nodes reduce overall usability and necessitate mechanisms for re-establishing network paths. UAVs are hypothesized to be a solution for minimizing network partitioning, thereby enhancing usability. UAVs have been widely used to increase coverage by connecting segments of vehicular networks [21], to assist in disaster recovery [22] and for content distribution networks [23]. The altitude of UAVs can be adjusted to avoid obstacles and to enhance the likelihood of establishing line-of-sight

(LoS) communication links to ground nodes. To reduce network partitions, UAVs are either statically or dynamically deployed [24], [6]. Studies like UAV-NetRest [6] propose minimal deployment of UAVs to be cost-efficient.

Under the above-discussed challenges and opportunities, the objective of this research study is two-fold i.e., to (1) maximize the user usability with UAV(s) deployment, and (2) minimize the cost in terms of the number of deployed UAVs, with negligible impact on the computational time of deployment positions of UAV(s).

4 UAV Deployment in a 1-Path Problem

A simplistic case study is conducted as a simulation to understand the impact of UAV (u) deployment in a MANET, focusing on the available network metric values attributed to each UAV. The assumption is made that $u_{IU} = 10$ and $u_{PI} = 4$, where the maximum values of both metrics indicate the highest achievable information accuracy [20]. The bandwidth metric u_{BW} is considered to be above 10 Mbps [25], based on the assumption of the IEEE 802.11n standard, as 2.4 GHz of radio frequency is used for both ground-based and aerial nodes. The hop-count u_{HC} is adjustable to user needs and represents the maximum number of UAVs allowed to establish the S-D path by connecting the rss sub-paths. The hop count $u_{HC} \leq \overrightarrow{t}_{HC} - rss_{HC}$ i.e., u_{HC} value varies for each S-D pair depending upon the values of $\overrightarrow{t}_{HC}$ and rss_{HC}. For example, to connect sub-paths S-a and f-D (in Figure 1) via UAV(s), $u_{HC} \leq 2$ is needed since $\overrightarrow{t}_{HC} = 5$ and $rss_{HC} = 3$.

To maximize user usability, this research study addresses two important questions i.e.,:

- How to optimally place UAVs within a given upper bound on their number?
- Considering the battery-powered nature of these devices, can the longevity of a connected network be extended by deploying UAVs with minimal simultaneous usage?

To achieve high user-usability with a set of UAVs, the process starts with `1-Path Problem` (1-PP). This problem deals with situations where a single S-D path (selected with QoIT [11] method) either fails to meet the communication requirements of the query node or gets disconnected due to the dynamic nature of the network and its spatial environment. As a solution to the given scenario, 1-PP replaces the *non-rss* sub-path with UAV(s) to re-establish the S-D path such that all user requirements are met. As discussed earlier, the *non-rss* sub-path may consist of a single or multiple link(s)/node(s) that either cannot perform data transmission or fail to satisfy communication requirements.

To illustrate the concept, consider the situation where source node S transmits the requested information to its query node D via the path S-a-b-c-d-e-f-g-h-i-j-D as shown in Figure 1. However, Figure 2 (a) shows that along with sub-optimal links and relay nodes, the path gets disconnected due to the loss of intermediate nodes/links caused by bushfire and/or mobility. In such a situation, UAV(s) can be deployed to restore the S-D path, as shown in Figure 2 (b). The deployment strategy of UAV(s) for a single S-D path is defined by the `1-Path Problem`, which is explained in detail in the following subsections and illustrated by the flow diagram in Figure 4.

Symbol	Meaning
u, g, S, D	UAV, $g$round, Source, and Destination (or query) nodes respectively
a	Adjacent node of node u or g
$\overrightarrow{t}_{BW}, \overrightarrow{t}_{IU}, \overrightarrow{t}_{PI}, \overrightarrow{t}_{HC}$	$[BW, IU, PI, HC]$ $t$hreshold values w.r.t each each query node
u_{BW}, u_{IU}, u_{PI}	$[BW, IU, PI]$ values on links between g-u, u-u, and u-g nodes
g_{BW}, g_{IU}, g_{PI}	$[BW, IU, PI]$ values on links between g-g nodes
u_i^x, u_i^y, u_i^z	The x,y,z coordinates of i^{th} UAV node, where $i \geq 1$
a_j^x, a_j^y, a_j^z	The x,y,z coordinates of j^{th} $a$djacent node, where $j = i + 1$
h_{ff}	Fire $f$lame $h$eight
$RR(n_a, n_b)$	Returns `TRUE` if n_a and n_b nodes are in radio range
$LoS(n_a, n_b)$	Returns `TRUE` if n_a and n_b nodes are in line of sight
P_{SD}	S-D Path without UAV(s)
P_{SUD}	UAV(s) based S-D Path
rss	Set of $r$equirement $s$atisfying $s$ub-paths, where $rss \subseteq P_{SD}$
rss_S, rss_D	an rss that starts/ends with S/D, where $rss = \{rss_S, rss_D\}$
L	Last node of rss_S
F	First node of rss_D
N_u	Maximum Number of UAV(s) to be placed between L and F
Set_{CN}	Set of $c$onnecting $n$odes including L, N_u, and F
rss_{HC}, u_{HC}	Hop Count of rss sub-paths and to be deployed UAVs on S-D path

TABLE 1: Symbols and terminologies used in S-D 1-Path Problem.

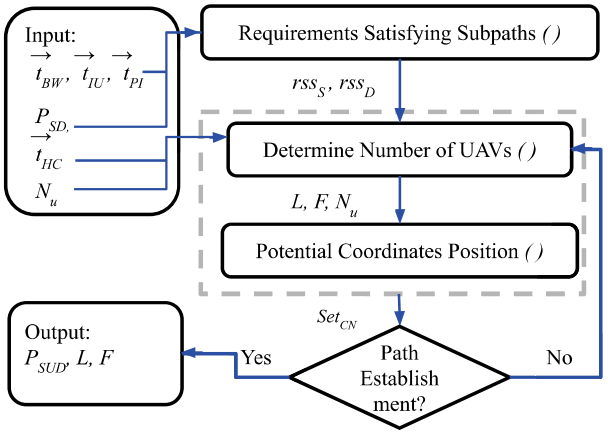

Fig. 4: Flow Diagram of UAV assignment along a single S-D path using 1-PP.

4.1 Requirements Satisfying Subpaths (rss)

The procedure rss generates two disjoint sub-paths, denoted as (rss_S and rss_D), for a single S-D path P_{SD}. Here rss_S initiates from the source node, while rss_D concludes at the query node. The node(s) and link(s) associated with each sub-path fulfill all the network metric requirements (represented by $\overrightarrow{t}$) defined by the user. In the worst-case scenario, $rss_S = \{S\}$ and $rss_D = \{D\}$. The number of links within these sub-paths influences the physical reachability

distance between S and D nodes. It is assumed that network information is consistently known at all times.

Note that the lengths of rss_S and rss_D may exceed one only if, before partitioning, there existed a path P_{SD} between nodes S and D. Let u represent the last node in rss_S, and v be the next connecting node in the original path P_{SD}. Node v is incorporated into rss_S only if nodes u and v remain within each other's radio range, and node v, as well as the $link_{(u,v)}$, still satisfy all the specified $\overrightarrow{t}$ requirements. Similarly, if v is the initial node in rss_D, then its adjacent node u from the previous path is added to rss_D only if nodes v and u are still within each other's radio range, and u and the $link_{(u,v)}$ still meet all $\overrightarrow{t}$ conditions. In the case of Figure 2 (a), the function `RequirementSatisfyingSubPaths()` yields $rss_S = \{S, a\}$ and $rss_D = \{j, D\}$.

```
1: procedure RequirementsSatisfyingSubpaths(P_SD, t⃗)
2:     i_u ← 1                                  ▷ location indicator in a path
3:     i_v ← 1                                  ▷ location indicator in a path
4:     l_PSD ← |P_SD|                           ▷ last index l of P_SD
5:     rss_S = P_SD[i_u]                        ▷ adds S to start of sub path rss_S
6:     rss_D = P_SD[l_PSD]                      ▷ adds D to end of sub path rss_D
7:     i_nxt ← i_u + 1                          ▷ index of next node in P_SD
8:     i_prv ← l_PSD − 1                        ▷ index of second last node in P_SD
9:     while i_nxt <= (l_PSD − 1) do            ▷ loop to populate rss_S
10:        u ← rss_S[i_u]                       ▷ sending node in a link uv
11:        v ← P_SD[i_nxt]                      ▷ receiving node in a link uv
12:        if RR(u,v) AND ({v, link_(u,v)} meets t⃗) then
13:            rss_S ← rss_S ∪ v                ▷ adds v to rss_S
14:            i_u ← i_u + 1
15:            i_nxt ← i_nxt + 1
16:        else
17:            break
18:        end if
19:    end while
20:    while i_prv >= i_nxt do                  ▷ loop to populate rss_D
21:        v ← rss_D[i_v]
22:        u ← P_SD[i_prv]
23:        if RR(u,v) AND ({u, link_(u,v)} meets t⃗) then
24:            rss_D ← u ∪ rss_D                ▷ adds u to rss_D
25:            i_prv ← i_prv − 1
26:        else
27:            break
28:        end if
29:    end while
30:    return (rss_S, rss_D)
31: end procedure
```

4.2 Determine Number of UAVs (N_u)

This process determines the necessary number of UAVs (N_u) to restore the source-to-destination path (P_{SD}). Initially, N_u is set to 0. When `PathEstablishment()`, outlined in Algorithm 1 for the `1-Path Problem` (as described in subsection 4.4), returns `False`, this procedure is invoked, leading to an increment in N_u. The procedure then calculates the anticipated UAV-supported source-to-destination path (P_{SUD}), defined as $P_{SUD} = \{rss_S \cup N_u\ UAV(s) \cup rss_D$ (`line 3 in pseuo code`). If the number of hops in P_{SUD} exceeds the specified $\overrightarrow{t}_{HC}$, the procedure removes the trailing relay node, $L = rss_S[|rss_S|]$ when $|rss_S| > 1$. Alternatively, if $|rss_D| > 1$, the procedure eliminates the leading relay node, $F = rss_D[1]$, from rss_D (`lines 5 to 10 in pseuo code`). This iterative process continues until $(|P_{SUD}|-1) \leq \overrightarrow{t}_{HC}$ or both $|rss_S|$ and $|rss_D|$ equal one.

The procedure is iteratively called from Algorithm 1 until the adjacent nodes of L, N_u UAV(s), and F are within each other's radio ranges or if the number of N_u exceeds the available number of UAVs A_{uav}. In the latter case, no UAVs will be assigned to the broken path. It is important to note that in this model, we assume the presence of a central entity in the network that receives broadcasted messages containing location updates, mobility speed, and direction from reachable live nodes. The central entity, through historical data analysis, can predict the locations of nodes that have recently become non-reachable, as discussed in [26]. However, this study does not specifically address the topic of location prediction for MANET nodes and therefore does not delve further into that aspect. Nodes that fail to broadcast location messages within a specified time period are considered inactive.

```
1: procedure DetermineNumberofUAVs(rss_S, rss_D, t⃗_HC, N_u)
2:     N_u ← N_u + 1                            ▷ identifier of number of UAVs
3:     P_SUD ← rss_S ∪ N_u UAVs ∪ rss_D
4:     while (|P_SUD| − 1) > t⃗_HC do
5:         if |rss_S| > 1 then
6:             L ← rss_S[|rss_S|]
7:             rss_S ← rss_S \ L                ▷ remove last node from rss_S
8:         else if |rss_D| > 1 then
9:             F ← rss_D[1]
10:            rss_D ← rss_D \ F                ▷ remove first node from rss_D
11:        else
12:            break
13:        end if
14:        P_SUD ← rss_S ∪ N_u UAV(s) ∪ rss_D    ▷ update P_SUD
15:    end while
16:    return (N_u, rss_S, rss_D)
17: end procedure
```

4.3 Potential Coordinates Position (P)

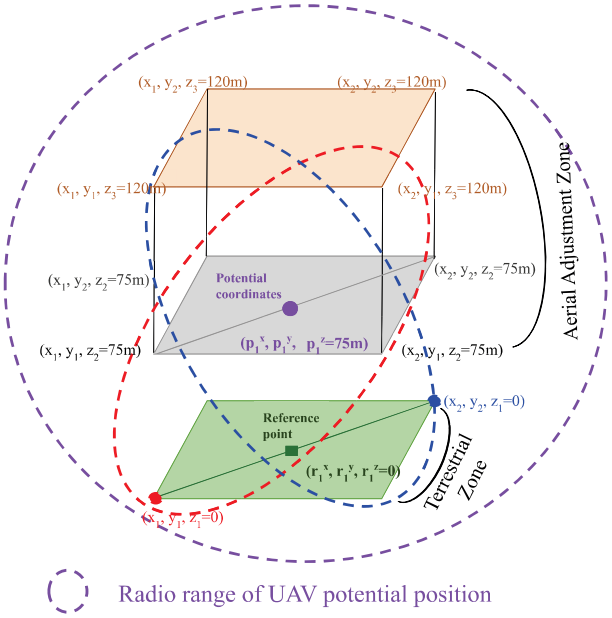

Fig. 5: Initial potential 3D coordinates (p^x, p^y, p^z) of a single UAV with its aerial adjustment zone.

The model assumes that eucalyptus trees and bushes have an altitude range of $z = 5-50m$, while other forest-related obstacles such as buildings, fire lookout towers, and electric poles/wires have an altitude range of $z = 10-20m$. The maximum fire flame height considered by the model is $h_{ff} \leq 70m$. The UAVs in this setting operate at an

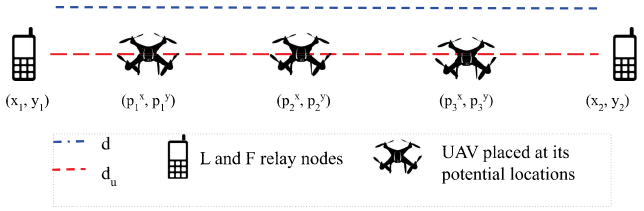

Fig. 6: 2D potential positions of three UAV(s) between L and F relay nodes.

altitude range of $75m \leq z \leq 120m$ as in Figure 5, where the minimum altitude of $z = 75m$ is used to avoid fire flames and collision with obstacles within bushfire field, while the maximum altitude of $z = 120m$ is adopted by civil aviation authorities [27] to provide a buffer zone between UAVs and manned air-crafts operating at $z = 150m$. This altitude range also maximizes the area coverage during emergency situations [28] and enhances the establishment of Line of Sight (LoS) between ground and aerial nodes.

The procedure returns a set P consisting of N_u elements, denoted as p_i where $1 \leq i \leq N_u$. Each p_i represents a potential 3D coordinate position (p_i^x, p_i^y, p_i^z) for a UAV denoted as u_i (i.e., $u_i \in N_u$). The (p_i^x, p_i^y) coordinates lie between $L \in rss_S$ and $F \in rss_D$, such that it establishes P_{SUD}. The potential height of each UAV node, represented by p_i^z, is randomly chosen between 75 meters and 120 meters to avoid any potential obstacles including flames or trees.

This study defines the shortest path between ground nodes L and F to be a straight aerial route. Therefore, the potential positions of the UAV(s), are initialized along the line $\overline{LF}$. The procedure first computes the Euclidean distance d of the line segment $\overline{LF}$. The distance d is then divided into N_u+1 parts of equal distances d_u. This allows for the equidistant placement of all N_u UAV(s) along the line segment $\overline{LF}$. Figure 6 illustrates the placement of three UAVs at equal distances between their corresponding ground nodes L and F.

```
1: procedure PotentialCoordinatePositions(L, F, N_u)
2:     (x_l, y_l, 0) ← location(L)        ▷ z_l = 0, since L is a ground node
3:     (x_f, y_f, 0) ← location(F)        ▷ z_f = 0, since F is a ground node
4:     d = sqrt((x_f − x_l)^2 + (y_f − y_l)^2)        ▷ Euclidean distance between L and F
5:     P = ∅ ▷ set of 3D Potential coordinates of the UAV(s)
6:     d_u = d/(N_u + 1)                  ▷ splits d into N_u + 1 parts
7:     for i ← 1 to N_u do
8:         p_i^x = x_l − (i * d_u * (x_l − x_f))/d      ▷ x-coord of i^th potential position
9:         p_i^y = y_l − (i * d_u * (y_l − y_f))/d      ▷ y-coord of i^th potential position
10:        p_i^z = rand[75m, 120m]         ▷ z-coord of i^th potential position
11:        P = P ∪ p_i                     ▷ p_i ∈ P
12:    end for
13:    return (P)
14: end procedure
```

4.4 Path Establishment

Let Set_{CN} = {L, N_u UAV(s), F} be the set of nodes that need to be connected in order to re-establish a route between

```
1: procedure PathEstablishment(Set_CN)
2:     j = 2
3:     while j < |Set_CN| do   ▷ |Set_CN|: number of elements in Set_CN
4:         i = j − 1
5:         u_i ← Set_CN[j]                ▷ u_i: i^th UAV node
6:         a_l ← Set_CN[j − 1]            ▷ left adjacent node a_l of u_i
7:         a_r ← Set_CN[j + 1]            ▷ right adjacent node a_r of u_i
8:         (u_i^x, u_i^y, u_i^z) ← location(u_i)
9:         (a_l^x, a_l^y, a_l^z) ← location(a_l)        ▷ a_l = L (if i = 1) or a_l = u_{i−1} (if i > 1)
10:        (a_r^x, a_r^y, a_r^z) ← location(a_r)   ▷ a_r = F (if i = |Set_CN|) or a_r = u_{i+1} (if i < |Set_CN|)
11:        if RR(a_l, u_i) AND RR(u_i, a_r) then
12:            while LoS(a_l, u_i) || LoS(u_i, a_r) do
13:                Adjust(u_i^x, u_i^y, u_i^z) such that, (a^x ≤ u_i^x ≤ a_r^x, a^y ≤ u_i^y ≤ a_r^y, 75m ≤ u_i^z ≤ 120m) and (RR(a, u_i) & RR(u_i, a_r))
14:            end while   ▷ assuming LoS is found after finite runs
15:        else
16:            return (false)   ▷ means u ∈ N_u UAV(s) are unable to connect L and F
17:        end if
18:    end while
19:    return (true)
20: end procedure
```

rss_S and rss_D. Although the initial potential positions for N_u UAV(s) are determined, there is no guarantee that they will be within radio range or in the line of sight (LoS) of neighboring nodes along the route. To address this, the procedure checks for radio range overlap and line of sight (LoS) between adjacent nodes in Set_{CN} and makes necessary adjustments to the potential locations, if required.

In emergency networks, UAVs typically operate in the 2.4 GHz band [29] with a radio range of $1km$ or more. The experimental test bed assumes the use of micro UAVs weighing less than 2 kg [30], which are lightweight and omnidirectional with a communication range of up to $1km$ and can be placed to the height of 200 meters. For the ad hoc network's ground nodes, the communication range of 100 meters is considered, with IEEE 802.11b standard [31] at 2.4 GHz. However, due to the increased number of obstacles at higher altitudes, the UAV(s) require LoS communication. In the challenging bushfire scenario, the risk of losing line of sight increases as the flames can produce plasma that interferes with the surrounding magnetic field and degrades the signal strength. To establish a line-of-sight link, the procedure checks for any overlap between the straight line (that connects the coordinate locations) of adjacent nodes and the current fire locations, their expected spread rates, and any other obstacles.

In above discussed situation, the placement of N_u UAVs at their initial potential locations leads to one of the following three cases:

1) If the radio ranges of adjacent nodes in Set_{CN} overlap and there is LoS between each pair of adjacent nodes, then neither the procedure `DetermineNumberofUAVs()` is re-called for additional UAV assignment nor adjustments of potential locations are needed.
2) If the radio ranges of adjacent nodes in Set_{CN} overlap with each other, but one or more UAVs have no LoS to at least one of their adjacent nodes due to obstacles, then the UAVs are relocated. The relocation process is done incrementally, one unit at a time horizontally or vertically while still maintaining radio range overlap

with adjacent nodes and keeping their heights between $75m$ and $120m$, as shown in Figure 5. The process continues until a finite number of iterations or until LoS between adjacent nodes is achieved.

3) If at most one node in Set_{CN} does not have radio range overlap or has no LoS with its adjacent node, then the `DetermineNumberofUAVs()` procedure is called to assign an additional UAV. The process continues until both radio range overlap and LoS exist between all adjacent nodes in $SetCN$, or if N_u has to increase beyond the available number of UAVs ($Auav$) in the network. In the latter case, no path can be established, and therefore no UAVs can be deployed between the pair (L, F).

Algorithm 1 1 Path Problem

Input: P_{SD} , $\overrightarrow{t}$, A_{uav} ▷ Broken path, set of thresholds and available UAVs
Output: P_{SUD}, L, F
$N_u \leftarrow 0$
$(rss_S, rss_D) \leftarrow$ `RequirementsSatisfyingSubpaths(`P_{SD}, $\overrightarrow{t}$`)`
$(N_u, rss_S, rss_D) \leftarrow$`DeterminingNumberofUAVs(`$rss_S$, rss_D, $\overrightarrow{t}_{HC}$, N_u`)` ▷ $\overrightarrow{t}_{HC} \in \overrightarrow{t}$
$L \leftarrow rss_S[length(rss_S)]$
$F \leftarrow rss_D[1]$
$potential\_locations \leftarrow$`PotentialCoordinatePositions(`$L$, F, N_u`)`
$Set_{CN} \leftarrow L$
Let UAV_{SD} be the list of N_u UAVs
$n \leftarrow 1$
while $n \leq N_u$ **do**
  $location(UAV_{SD}[n]) \leftarrow potential\_locations[n]$
  $Set_{CN} \leftarrow Set_{CN} \cup UAV_{SD}[n]$
  $n = n + 1$
end while
$Set_{CN} \leftarrow Set_{CN} \cup F$
$Flag \leftarrow PathEstablishment(Set_{CN})$
if $!Flag$ **then**
  if $(Nu + 1) < A_{uav}$ **then**
    Go to Step 3
  else
    $P_{SUD} \leftarrow \emptyset$
  end if
else
  $P_{SUD} \leftarrow rss_S \cup UAV_{SD} \cup rss_D$
end if

The procedures mentioned above are invoked by the algorithm `1 Path Problem`, which aims to reconnect a sub-optimal or broken path P_{SD} by deploying N_u UAV(s) at their potential positions, given that $N_u \leq A_{uav}$. It is assumed that each UAV can retain source information utility (IU), possess sufficient bandwidth (BW) to fulfill the requirements of the query nodes and act as a reliable relay with a high path integrity (PI) value.

The `1-Path Problem` algorithm iteratively increases the number of required UAVs (N_u) in polynomial time, with a worst-case time complexity of O(A_{uav}). The algorithm achieves the optimal number of deployed UAVs by either returning the potential positions of UAVs that can rectify a broken or sub-optimal path P_{SD} or notifying of the absence of a solution for the available A_{uav} UAVs.

5 UAV Placement under QoIT (U-QoIT)

The goal of the U-QoIT is to maximize the number of query nodes receiving their desired services within a set number of available UAVs. This is accomplished by optimizing paths that fall short of QoIT standards and establishing new paths using UAV(s) within a 3D space. Unlike Algorithm 1, U-QoIT deploys UAV(s) between pairs of source and query nodes that either fail to meet QoIT requirements or, in the worst case, lose connectivity due to dynamic bushfires.

The execution of U-QoIT is assumed to take place at a central entity, as outlined in subsection subsection 4.2. This central entity may function as a coordinator or a base station with a back-haul connection to the MANET. Alongside mobility speed, direction, and location, the accessible nodes broadcast their IU and link characteristics, including BW and PI, to the central entity at a suitable interval for updating.

The proposed solution, U-QoIT (described in Algorithm 2), starts with the identification of all those query nodes whose paths toward their source nodes are either broken, non-existent or do not fulfill the desired quality requirements for the information transfer. Such paths are indicated by input $\hat{P}_{SD}$, with a set of requirement thresholds of the corresponding query nodes represented as $\hat{T}$. `Step 4-12`, of Algorithm 2 applies:

1) `1-Path Problem` i.e., Algorithm 1 to each of the path in $\hat{P}_{SD}$ and ends up in $\hat{L} = \{L_1, L_2, ...L_m\}$ and $\hat{F} = \{F_1, F_2, ...F_m\}$, where m denotes all the query nodes with unfulfilled requirements, that need fewer than A_{uav} UAV(s). Note, the expected allocation of UAV(s) for a path are not subtracted from A_{uav}, after a call to `1-Path Problem`. This is because `1-Path Problem` only assesses the feasibility of whether a path can be connected with available UAV resources or not, and if yes then what can be the potential locations.
2) There is at least one path made up of relay UAV(s), with the capability to connect each corresponding pair of (L_i, F_i) nodes, where $i \in \{1, ..., m\}$.

Algorithm 2 U-QoIT

Input: $\hat{P}_{SD}$, $\hat{T}$, A_{uav} ▷ All un-established and partitioned paths and information thresholds from the corresponding destination nodes, and available number of UAV(s)
Output: $\hat{U}$ ▷ Deployed UAV(s) to re-establish paths
$\hat{L} \leftarrow \emptyset$
$\hat{F} \leftarrow \emptyset$
$\hat{P}_{SUD} \leftarrow \emptyset$
for each $P_{S_iD_i} \in \hat{P}_{SD}$ **do** ▷ Apply 1-Path Problem on each un-established and broken path
  Extract T_i from $\hat{T}$ ▷ Threshold requirements for destination D_i
  $(P_{S_iU_iD_i}, L_i, F_i) = 1PathProblem(P_{S_iD_i}, T_i, A_{uav})$
  if $P_{S_iU_iD_i} != \emptyset$ **then** ▷ If a path can be established within A_{uav} UAV(s)
    $\hat{P}_{SUD} = \hat{P}_{SUD} \cup P_{S_iU_iD_i}$
    $\hat{L} = \hat{L} \cup L_i$
    $\hat{F} = \hat{F} \cup F_i$
  end if
end for
$U \leftarrow \forall$ UAV(s) $\in \hat{P}_{SUD}$ ▷ Note U and UAV(s) are just abstraction of UAV(s) with their potential locations.
$S_{LUF} \leftarrow \hat{L} \cup \hat{F} \cup U$
$AdjMatrix[|S_{LUF}|][|S_{LUF}|] = 0$ ▷ Initialize 2D Adjacency Matrix
for each $n_1 \in S_{LUF}$ **do**
  for each $n_2 \in S_{LUF}$ **do**
    if $RR(n_1, n_2)$ & $LoS(n_1, n_2)$ **then**
      $k_1 = index(n_1, S_{LUF})$ ▷ Index of n_1 in S_{LUF} list
      $k_2 = index(n_2, S_{LUF})$
      $AdjMatrix[k_1][k_2] = 1$
    end if
  end for
end for
$\hat{P}_{LUF} \leftarrow \emptyset$
$i \leftarrow 1$

27: **while** $i \leq length(\hat{L})$ **do**
28: $L_i = \hat{L}[i]$
29: $F_i = \hat{F}[i]$
30: $\hat{P}_{LUF} \leftarrow \hat{P}_{LUF} \cup \forall \overleftrightarrow{L_i F_i}$ ▷ Apply BFS on $AdjMatrix$ to extract all paths without cycles between L_i and F_i and insert them in $\hat{P}_{LUF}$ list
31: $i = i + 1$
32: **end while**
33: **for** each $u_l \in U$ **do** ▷ Reduce a UAV if paths exist between all corresponding L and F pairs
34: $P'_{LUF} \leftarrow \hat{P}_{LUF} \setminus$ (paths containing u_l)
35: $remove \leftarrow 1$
36: **for** each $L_i \in \hat{L}$ **do**
37: **if** $\overleftrightarrow{L_i F_i} \notin P'_{LUF}$ **then**
38: $remove \leftarrow 0$
39: **break**
40: **end if**
41: **end for**
42: **if** $remove = 1$ **then**
43: $\hat{P}_{LUF} \leftarrow \hat{P}_{LUF} \setminus$ (paths containing u_l)
44: $U = U \setminus u_l$
45: **end if**
46: **end for**
47: **while** $|U| < A_{uav}$ **do** ▷ This while loop is optional and can be used to place remaining A_{uav}, at the largest partition in the network
48: $U = U \cup u_l$ ▷ Add an additional UAV abstraction u_l
49: (u_l^x, u_l^y) ← mid-point of largest partition in the network
50: u_l^z = rand(75 m, 120 m)
51: **end while**
52: **while** $|U| > A_{uav}$ **do** ▷ If remaining UAV(s) from 33-46 are still more than A_{uav}
53: $u_l \leftarrow min_{centrality}(\forall uav \in U)$ ▷ Remove the UAV with lowest centrality
54: $U = U \setminus u_l$
55: **end while**
56: $\hat{U}$ ← Set of deployed UAV(s) at the potential locations of U

5.1 Deployment of Available UAV(s)

The aim is to select potential positions for A_{uav} from the list of all possible positions, so that the placement of UAV(s) at the chosen subset maximizes the interconnection of (L, F) pairs. The problem is typically approached as a combinatorial search problem, which requires evaluating a large number of potential combinations. For example, to find the best combinations of r potential locations among the total number of n potential locations, the number of combinations to be evaluated are $\frac{(n)!}{r!(n-r)!}$, with a time complexity of $O(n^r)$. To address this computational challenge, a novel approach is used that initially assumes to have deployed UAV(s) at all potential locations and then iteratively removes one potential location at a time, starting with the one that has the least impact on the overall network connectivity. This iterative elimination approach reduces the time complexity to $O(n-r)$.

The `step 13` of the Algorithm 2 denotes an abstraction U of the total number of UAV(s) with their potential locations, where U is derived from running `1PathProblem()` for un-established paths, broken paths and for down-link paths that fail to meet quality requirements of the query node(s). To determine the number of (L, F) pairs that can be connected by deploying U UAVs, `steps 14-24` construct an adjacency matrix for the set $S_{LUF} = \hat{L}, \hat{F}, U$. Subsequently, `steps 25-32` use breadth-first search to identify all non-cyclic or simple paths connecting each corresponding (L, F) pair. Since interest is in finding all simple paths between an (L, F) pair, an uninformed search method is employed. Depth-first search is avoided to prevent high time complexity.

To minimize the search space of potential positions of the UAV(s), `steps 33-46` employ reduction by elimination. This is accomplished by omitting an abstraction of UAV $u_l \in U$ in such a way that at least a single communication path still remains between each (L, F) pair. The process is repeated for all UAV abstractions in U. Note that the removal of a particular abstraction u_l reduces the list of potential deployment locations for UAVs.

With the execution of `steps 33-46`, if the number of leftover UAV abstractions $|U|$ turn out to be less than A_{uav}, then an optional set of steps i.e., `steps 47-51` can be utilized to deploy the remaining UAVs at the mid-points of the largest partitions in the network. However, `steps 47-51` are only recommended if minimal deployment is not an objective. On the contrary, if $|U| A_{uav}$ then `steps 52-55` removes the abstraction u_l that has the lowest centrality among all remaining abstractions U. Centrality refers to the number of times an abstraction u_l appears on the paths of all the pairs (L, F). By removing the abstraction with the lowest centrality, we ensure that only a minimum number of (L, F) pairs are left disconnected. In the final `step 56`, the remaining set of abstractions U with $|U| \leq A_{uav}$ is converted into a physical deployment of UAVs, denoted by $\hat{U}$.

5.2 QoIT based Source Selection

Before running U-QoIT, the Quality of Information with Thresholds (QoIT) [11] is applied to select the best source nodes and their down-link paths to each query node. This selection is made in a way that a maximum number of quality requirements for each query node can be satisfied. When Algorithm 2 i.e., `U-QoIT()` is called, it deploys the set of UAVs $\hat{U}$ in the network. After UAVs deployment, the QoIT scheme is applied again to the entire network to ensure that the optimal source nodes and down-link paths with respect to quality requirements are selected for the maximum number of query nodes.

5.3 Time Complexity of U-QoIT

Let n be the number of ground nodes in the network, k be the number of query nodes i.e., $k = |P_{SD}| = |\hat{L}| = |\hat{F}|$, and m be the number of UAVs A_{uav}. The main time-consuming modules of Algorithm 2 are (i) `1-Path Problem`, (ii) `AdjMatrix`, and (iii) breadth-first-search `(BFS)`. The time complexity of `1-Path Problem` is $O(km)$, and that of `AdjMatrix` is $O(2k+m)^2 \equiv O(k+m)^2$. The time complexity of `BFS` for a single pair (L_i, F_i) is $O(k+m+e)$, where e is the number of direct links between nodes. The total time complexity of `BFS` for all (L, F) pairs is $O(k*(k+m+e))$.

Therefore, the overall time complexity of Algorithm 2 is $O(km + (k+m)^2 + (k(k+m+e)) \equiv O(k+m)^2$, which makes it quadratic. However, since $k << n$ and $m << n$, the experimental result in Figure 7(c) shows that when 21% of the ground nodes are query nodes, the computational time of U-QoIT is only a few milliseconds more than when no UAVs are deployed.

6 Simulation Model and Experimental Setup

To evaluate the performance of U-QoIT in a realistic MANET setting with a complex and challenging environment, the simulation environment developed in a prior study using Java APIs [20] is being enhanced. This simulation environment incorporated both the QoIT approach [11] for the source selection process, as well as a dynamic bushfire scenario outlined in subsection 6.2. Following subsections discusses several key features of the simulated networks and bushfire model.

6.1 Underlying MANET

The following discussion outlines the design and functionality of our simulation test-bed across various network layers.

6.1.1 Physical Layer

Our simulated network uses the standard log-distance path loss model under dense foliage to calculate the radio attenuation and data rate at the physical layer. More complex models can be used if different attenuation effects are deemed necessary e.g., by adding noise or terrain effects. Network connectivity on a link between two nodes exists if its capacity is greater than $0\,Mbps$. SNR generated using Equation 6, is used to calculate the data rate of a link according to Shannon's channel capacity theorem.

$$SNR = T_p \text{ - } 32.45 \text{ - } 4 * 10log10(f * d) \text{ - } r \text{ - } n_{cutoff} \quad (6)$$

where, T_p is the radio transmission power with simulation using $20\,dBm$ for ground nodes and up to $30\,dBm$ for UAVs. The variable f is the carrier frequency within $2.4\,GHz$ band, d represents Euclidean distance between the two nodes in km, r represents random fading ranging from 0 - $1\,dB$ and n_{cutoff} is the Nyquist noise cutoff level with -180 dBm.

Data Rates (DR) for SNR thresholds values									
SNR(dB)	0.0	2.0	5.0	9.0	11.0	15.0	18.0	20.0	25.0
DR(Mbps)	0.0	7.2	14.4	21.7	28.9	43.3	57.8	65.0	72.2

TABLE 2: SNR threshold values and respective data rates used by the model.

Based on computed SNR values in Equation 6, the data rate of a given link is determined using Table 2. For this study, a 2.4 GHz radio equipped with an omnidirectional antenna for the UAV(s) is assumed. According to Khan et al. [32] the communication technologies available to UAV in 2.4GHz can have up to 250 meters range, while [33] reported it to be between 430 to 950 meters for different propagation models in 802.11n/g and up to 6.7km with 802.11b. It is assumed that a UAV may provide approximately $1\,km$ in an unobstructed environment. However, in the presence of obstacles such as trees, bushes, and bushfires, the effective range of the radio is significantly reduced. Moreover, this experimental study assumes all radios of ground and UAV nodes to be transmitting on the same frequency band such that interference occurs if nodes within range are transmitting simultaneously. In simulation maximum transmission power in not used to mitigate interference. Unless mentioned otherwise, the radio range for the UAV is $0.6\,km$ and for ground nodes, it is $0.1\,km$, in all the simulated experiments.

6.1.2 Data Link Layer

For medium access sharing and control, the simulator deploys Node Activation Multiple Access (NAMA) [34], which is a widely used Time Division Multiple Access (TDMA) MAC protocol in current MANETs [35]. NAMA employs a slot reservation scheme for bandwidth allocation, where the bandwidth reservation is accomplished immediately after source selection. The contention resolution approach of NAMA enables collision-free scheduling, that utilizes two-hop neighborhood information to dynamically determine, at each node, whether it can transmit in a given time slot. The distributed scheduling of NAMA uses a hash function to generate a pseudo-random number for each node, which determines the priority among contending two-hop neighbors. NAMA ensures fairer allocation of resources, particularly in congested networks.

6.1.3 Network Layer

Our simulation field imitates a $10\,km^2$ area with four types of test-bed nodes including (a) source nodes (b) query nodes (c) regular nodes, and (d) UAVs, as described in Table 3. The query, regular, and UAV nodes can act as intermediaries between any two end nodes. All source nodes are assumed to have the same highest reliability or PI value of 4 but with varying IU values between 3 to 10. An IU value of 10 contributes the most to the quality features of an application service[3], such as *Accuracy* and *Completeness*. The impact of a source node's IU value propagates through its down-link path to a query node, whereas the PI and BW of a path are limited by the lowest values of the links along the down-link path. Additionally, the Hop Count (HC) of a path indicates the number of links it contains.

Node type	Relay?	Path Integrity (PI)	Information Utility (IU)	Bandwidth (BW)
Source	No	4	3-10	Depends upon data rate and number of concurrent users
Query	Yes	1-4	Same as the source of the path, where this node acts as a relay	
Regular	Yes	1-4		
UAV	Yes	4		

TABLE 3: Network metric values of different node types in experimented simulations.

To establish routes between end nodes, the source-to-query path with the highest QoIT score [11] is selected, where QoIT score is directly dependant upon the quality metrics that are computed from network metrics such as $\{PI, IU, BW, HC\}$. A QoIT score of 1 means that the path from the source node to the query node meets all the quality needs of the query node, and hence achieves the highest *user usability*. If there are multiple paths from one or more source nodes to a query node that can achieve high *user usability*, then Dijkstra's shortest path algorithm in conjunction with QoIT to select the path with the fewest hops, or shortest path in terms of the number of links, among the feasible options.

3. Interested readers are referred to [11]

At the onset of the simulation, the number of data packets to be transmitted to each query node is selected randomly from a range of 10 to 15. If the selected source node or any intermediate node on the path becomes disconnected prior to the successful delivery of all packets, the query node must obtain the remaining data packets from a newly selected source node and/or an alternative re-established path.

6.1.4 Application Layer

In the simulated network, all traffic flows from source nodes to query nodes. Following QoIT principles, the focus is on meeting users' needs, so no specific application data type is modeled. Instead, differences in the desired features of queried data are synthesized by defining minimum requirements for network metrics. It is assumed that all source nodes possess similar information but with varying IU, as depicted in Table 3.

6.2 Modeling of Dynamic Bushfire

At the start of a simulation run fire is ignited at 15-25 random locations in simulation area, that gradually spreads with simulation time. To imitate continuity or variation in topographic, environmental and fuel bed information, the variable values of fire spread behaviour model change with fire locations. Ranges of fire spread influencing variables are given in subsection 2.2, where the minimum and maximum values of the influencing variables are used to calculate the range of fire spread rate R_p. Fire spreads (using Equation 1) from the ignition point or line of fire in wind direction, at the rate of 0.0128 - 3.384 ms^{-1}.

In the simulation, changes in the scenario and bushfire spread occur with each simulation time unit. Any fire spread beyond the simulation area is considered extinguished. Network paths are initially established at the start of the simulation. Ground nodes that come into close contact with the fire line or fall within the fire spread range are considered dead and are excluded from network statistics for the remainder of the simulation. The death of nodes triggers the re-establishment of new network paths. If a dead node was a selected source or an intermediate node, QoIT determines the best available source or path at that time, potentially resulting in a source handover.

6.3 Test Scenarios and Baseline Methods

The simulation test conducted in this study consists of six different scenarios, presented in Table 4. Unless otherwise stated, each scenario has a dense deployment of 100 static ground nodes with 6 source nodes, 20 query nodes, and 74 relay nodes. Some of the scenarios comprise different types of simulations, each with 30 simulation runs[4]. A simulation run continues until all packets reach their destined query nodes, or the simulation time of 100 time units elapses. At the beginning of each new simulation run, the topology of the ground nodes is randomly generated. The scenarios include:

4. A simulation *type* refers to a specific network configuration, while simulation *runs* denote the number of times a particular network configuration executes, with only changes to the topology and requirements of the querying nodes in each new iteration.

Mobility Speed (km/hr)	Ground Nodes			Available UAVs	Runs
	Source Nodes	Query Nodes	Regular Nodes		
Scenario 1 - Performance Gain/Loss					
Static (0)	6	20	74	4	30
Scenario 2 - Increasing number of available UAVs					
Static (0)	6	20	74	{4, 8, 12, 16, 20}	30 ea.
Scenario 3 - Increasing mobility Speed					
{Static: 0, Walking: 3, Running: 10, Cycling: 20, Driving: 50}	6	20	74	8	30 ea.
Scenario 4 - Sparse network					
Static (0)	6	6	28	4	30
Static (0)	6	20	24	4	30
Scenario 5 - Aiming for minimal UAV deployment					
Static (0)	6	20	74	{4, 8}	30 ea.
Scenario 6 - Varying coverage range of UAVs					
Same as Scenario 1, but with UAV radio range: {0.4,0.6,0.8,1.0,1.2} km					

TABLE 4: Test scenarios, executed for each of the baseline approaches and U-QoIT.

1) *Scenario 1- Performance Gain/Loss*: This scenario involves 4 UAVs and aims to evaluate the trade-off between various network features, including the number of quality metrics met, the achieved QoIT score, and the computational time of both the baseline and U-QoIT methods.
2) *Scenario 2- Increasing Number of Available UAVs*: This scenario consists of five different simulation types, with 4, 8, 12, 16, and 20 A_{uav}, respectively. It gives an insight into the user usability, median packet latency and delivery ratio differences between U-QoIT and baseline methods across low, medium, and high A_{uav}.
3) *Scenario 3- Increasing Mobility Speeds*: The scenario keeps A_{uav} constant at 8 and introduces four different mobility speeds for ground nodes other than *static* case. Each simulation type with mobility employs pseudo-random movement for ground nodes. The mobility speeds are based on the average movement speed of device owners, with walkers at 3 km/hr, runners at 10 km/hr, cycle riders at 20 km/hr, and residential car drivers at 50 km/hr [36], [37]. This scenario investigates the consistency of U-QoIT's performance as the mobility of ground nodes increases, and determines the frequency at which the network needs an update in a similar test environment.
4) *Scenario 4 - Sparse Network*: The objective of this scenario is to assess the performance of U-QoIT in a lower-density network with the same area of 10 km^2 as before. Two simulation types are considered, one with 40 ground nodes and another with 50 ground nodes. Both scenario types have 6 source nodes, but different numbers of query and relay nodes, with 6 query and 28 relays in the first type, and 20 query and 24 relays in the second type. A_{uav} is kept constant at 4 for this scenario.
5) *Scenario 5- Minimal UAV Deployment*: This scenario in-

vestigates whether U-QoIT can minimize the number of required UAVs without compromising the performance quality. Minimal deployment of UAVs is desirable both in terms of cost and least cumulative energy consumption by UAVs.

6) *Scenario 6 - Varying coverage range of UAVs*: Previous works [32], [33] showed that UAVs can differ in their radio ranges. On top of it, varying noise and obstacles in an environment impact the coverage quality of UAVs to different extents. With the simulation area kept static at 10 km^2, the investigation focuses on whether increasing the coverage range leads to the convergence of baseline UAV deployment methods with the performance of U-QoIT.
Experiments comprise of five different simulation types. The first type starts with a low coverage range of 0.4 km for the 4 deployed UAVs. In the last simulation type, UAV ranges are set to 1.2 km. Note, 4 UAVs with 1.2 km coverage ranges serve a substantial part of the simulation area.

The performance gain of U-QoIT in each of the above scenarios is assessed by comparing it against the following four baseline approaches:

- *No UAV*: In this approach, UAVs are not deployed for path re-establishment in the event of network path failures caused by factors such as node mobility or dynamic bushfires.
- *Random UAV Deployment (U-Random)*: This approach randomly deploys A_{uav} UAV(s) in the simulation area, with a time complexity of $O(r)$, where $r = A_{uav}$.
- *Coverage Gap-based UAV (U-Gap)*: A logical approach to reduce partitions in an ad hoc network is to deploy UAVs strategically to bridge the connectivity gaps between the network nodes. A recent study in [6] proposes UAV-NetRest that deploys UAVs by first finding the convex hull of partitions and then applying Minimum Spanning Tree (MST) to find the least distant boundary pair of ground-nodes to optimized UAV placement position. A simple variant of UAV-NetRest named as "UAV-Gap" is implemented, where all such A_{uav} sub-areas are identified that have the largest coverage holes within the simulation area. A_{uav} UAV(s) are then placed at the centroid of the identified sub-areas. The complexity of U-Gap is $O(m^2)$, where m represents the number of ground nodes.
- *Dynamic Service Area based deployment (DSA)*: Recently Shih-Fan C. et al [13] proposed dividing the network into service regions by using KMeans and Voronoi Tessellation. The UAVs are deployed at centroids of the service regions to optimize the quality of service requirements of the nodes within the corresponding service regions. The area of a service region is dynamic as it changes with the dynamics of the ground nodes. DSA has a complexity of $O(r * m * t)$ due to the use of KMeans clustering, where $r = A_{uav}$, m represents the total number of ground nodes and t denotes the number of iterations in KMeans clustering. The number of iterations is lower bounded by the convergence of centroids in KMeans. Here the lower complexity terms of QoS optimization part in DSA are ignored.

In case of loss of a link in the paths from source to query end-nodes, the A_{uav} UAVs are redeployed (possibly at different locations) using the same approach as the initial deployment. Note that the loss of a link can result either from the mobility of the ground nodes or the proliferation of dynamic bushfires.

7 Results and Discussion

The performance of U-QoIT is evaluated in comparison to the four baseline approaches by presenting simulation results based on the parameters outlined in Table 5.

	Performance Parameter	Description
1.	*Fraction of quality metrics met*	Each query node has different requirements for the four quality metrics i.e., accuracy, reliability, completeness, and timelines, where each of these is dependent upon the network metrics namely PI, IU, BW, and HC. The parameter indicates the fraction of quality metrics requirements, of a query node, being met for the information flow on the source-query path.
2.	*QoIT-score*	QoIT-score assesses the weighted extent to which each quality need is fulfilled. A QoIT-score of 1 (calculated based on the achieved priority scores of the quality metrics met) indicates that the source meets all the quality requirements [11].
3.	*Percentage of user-usability*	The percentage of querying nodes whose all quality metrics requirements got fulfilled.
4.	*Computational time*	It represents the time, in milliseconds, taken by each scheme in deploying UAVs.
5.	*Packet delivery ratio*	It shows the delivered packets out of total transmitted data packets from source to query nodes in a simulation run. The ratio is less than 1 for a simulation run, if the simulation run finishes but there is at least a single un-delivered packet.
6.	*Median packet latency*	Median amount of simulation time taken by all the data packets, in a simulation run, to arrive at their query nodes.

TABLE 5: Performance parameters.

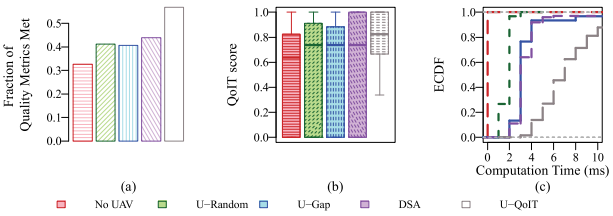

Fig. 7: Gain and loss of U-QoIT, compared to the baseline approaches using experiments from *Scenario 1*.

The trade-off analysis of U-QoIT in network performance is illustrated by three parameters in Figure 7. With a 0.6 km UAV coverage range and only four A_{uav} (Figure 7 (a)), U-QoIT fulfills individual quality metric requirements of all query nodes by up to 57%, surpassing the second best method, DSA, by 29.5%. The high QoIT-score of U-QoIT indicates superior performance by interconnecting downlink paths from source nodes, ensuring the meeting of either all or high-priority quality metrics. Figure 7 (b) shows that U-QoIT achieves a first quantile QoIT-score of 0.66, contrasting with 0.00 for U-Gap, U-Random, and DSA. A high QoIT-score is desirable, considering the varying priorities assigned by query nodes to quality metrics based on their application services. Figure 7 (c) depicts that the computational cost of U-QoIT is marginally higher, with a

median increase of up to 4 ms compared to baseline methods, which is insignificant for non-real-time applications. In summary, U-QoIT provides a superior trade-off in network performance compared to other UAV deployment strategies.

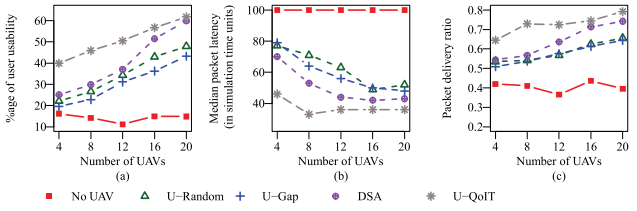

Fig. 8: Impact on percentage of user-usability, median packet latency, and packet delivery ratio, using experiments from *Scenario 2*.

The impact of an increasing number of A_{uav} on user usability is demonstrated by Figure 8 (a), revealing an expected correlation between a higher number of UAVs and improved simulation area coverage, consequently enhancing user usability. For example, employing U-QoIT results in a user-usability percentage rise from 40% with 4 UAVs to 62% with 20 UAVs, aligning with expectations. With fewer UAVs U-QoIT surpasses the baseline methods by at least 48% in user usability. This difference retains, except for DSA which gradually converges with U-QoIT's performance. The use of K-Means in DSA ensures extensive coverage of ground nodes with high A_{uav}, leading to a performance trend consistent with U-QoIT across parameters such as *median packet latency* and *packet delivery ratio* (as shown in Figure 8 (b) and (c)). It's important to recognize that achieving 100% user usability is challenging due to ongoing bushfires in the simulation area, posing a continuous threat to source and query nodes.

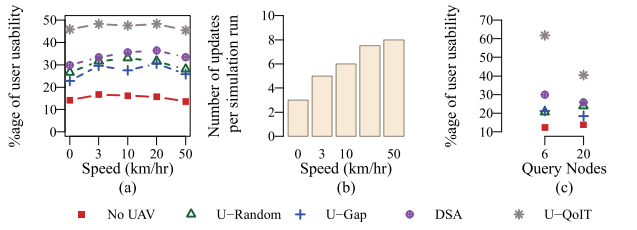

Fig. 9: Percentage of (a) user-usability and (b) number of UAV placement updates in different mobility scenarios, using experiments from *Scenario 3*. Part (c) depicts the percentage of user-usability, in a sparse network, using experiments from *Scenario 4*.

To investigate the robustness of U-QoIT with increasing speed of the ground nodes, experiments depicting owners of the devices to be static, walking, running, cycling, and driving are conducted. Figure 9 shows results from the deployment of eight UAV(s). It is observed that the percentage of user-usability remains similar across different speeds, for all the approaches. Interestingly, U-QoIT outperforms the second best baseline method of DSA, in all mobility cases, by 33-53% higher user-usability. The computation of source-query down-link paths and accordingly UAV placement in the network gets updated, when either a single relay node (i.e., a node currently part of an active path) dies out due to fire or a single active path is lost due to the mobility of its relay or end-node(s). This latter case leads to more frequent updates in high mobility scenarios (as is depicted by Figure 9 (b)).

In Figure 9 (c), the performance of U-QoIT is demonstrated in a sparse network, with two sets of results involving 6 and 20 query nodes among 40 and 50 ground nodes, respectively. Despite having only 4 UAVs and a reduced number of ground-based relay nodes, U-QoIT maintains a high level of user usability. In sparse networks, U-QoIT demonstrates a substantial improvement in user usability, showing a 200% enhancement compared to U-Gap and U-Random, and a 106% improvement compared to DSA when there are six query nodes. Even in the case of 20 query nodes, U-QoIT exhibits a minimum out-performance of 58%. This suggests that U-QoIT is well-suited for deployment in sparse networks.

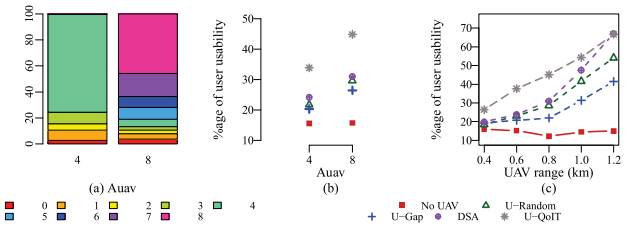

Fig. 10: Minimal deployment of UAV(s) by U-QoIT when A_{uav} is (a) 4 and 8, and their resultant (b) percentage of user-usability compared to baseline approaches that use all A_{uav}, using experiments from *Scenario 5*. Part (c) depicts the percentage of user usability with increasing UAV ranges from *Scenario 6*.

UAVs, being battery-powered, require substantial energy for sustained flight and hovering. Various research studies, such as [38], [39], aim to reduce UAV energy consumption to prolong their operational duration. To achieve prolonged coverage for MANET, one strategy is to delay deploying additional UAVs until necessary. Preserving a UAV contributes to the network's longevity by substituting an out-of-battery UAV when needed.

As per scenario 5, U-QoIT and baseline methods are run for $A_{uav} = \{4, 8\}$, respectively. In this scenario the `steps 47 - 51` in Algorithm 2 are skipped for U-QoIT, so that to evaluate its effectiveness for minimal UAV deployment. The results of the scenario are presented in Figure 10 (a, b). Remarkably, U-QoIT maintains its higher user-usability. It achieves more than 36% improvement in user-usability compared to the baseline method by utilizing less than four UAVs 22% of the time, with $A_{uav} = 4$. The efficiency is even more pronounced with $A_{uav} = 8$, where less than half of A_{uav} are used in 20% of the runs. U-QoIT still enhances user-usability by 44.5% (refer to Figure 10 (b)), in contrast to DSA which deploys all eight UAVs in every simulation run.

In the exploration of U-QoIT's performance within a limited simulation area of 10 km^2 and increasing radio ranges of UAVs, experiments involving four UAVs were conducted. Figure 10 (c) depicts improvements in user usability for all the UAV-based methods. Recall findings from Figure 8 (a), where DSA converged in performance with U-QoIT with $A_{uav} = 20$. DSA, utilizing KMeans, covers nearly the entire simulation area with UAVs of 1.2 km radio range. This, once again leads to convergence of DSA with U-QoIT. We,

however, hypothesize that such a scenario is improbable in real-world MANET fields due to challenges like wider area extent, interference, signal propagation issues, and constraints on UAV resources.

Takeaways: The simulations indicate that, with a marginal increase of 4ms in computational time, U-QoIT enhances user usability by 33-53% across diverse mobility scenarios compared to the best-performing baseline method. In sparse networks, it demonstrates a remarkable improvement of 106-200% and consistently outperforms baseline methods even with fewer deployed UAVs. U-QoIT ensures superior user usability with both increasing numbers of UAVs and expanding UAV radio ranges. In the latter two cases, DSA converges with U-QoIT's performance only when UAVs provide coverage for the entire simulation area.

8 Related work

UAVs giving life to broken/absent end-to-end paths: UAVs have proved their importance in a wide range of applications. These include but are not limited to aerial photography and videography [40], crop health monitoring [41], search and rescue operations [42], environmental monitoring [43], infrastructure inspection [44], intelligence gathering, and package delivery in hard-to-reach areas [45]. This study utilizes UAV(s) to connect information sources to query nodes in an ad hoc network built for rescue teams, early responders, and emergency teams in a bushfire scenario. The objective is twofold: (i) to ensure that information delivery meets the required quality for all query nodes, and (ii) to minimize the number of UAVs required (out of A_{uav}), by optimally positioning them in 3D space.

In terms of UAV placement strategies, an early and related work performed by K. Chandrashekar et al. [46], suggests adding an aerial layer of UAV(s) to MANET. It proposes that all UAVs should not only be connected with each other but also to at least a single connected cluster of ground-based nodes of MANET. The UAV(s) update their locations with changes in the movements of the ground nodes. Similarly, to provide a continuously connected mobile backbone network to ground nodes, H. Wang et al. [47] proposed placements of UAV(s) using a quadratic unconstrained binary optimization (QUBO) framework and solve these QUBO models with a Tabu search heuristic with preprocessing. Yet another work, conducted by Robert A.H et al. [48] used particle swarm optimization for placement of UAV(s) in a static mobile ad hoc network. The goal was to position UAV(s) so as to have a minimum average number of neighbors of the network nodes and with minimum required transmission power. This is to enhance the capacity of the network with minimal interference and power consumption. A recent work [49] proposes the placement of UAV(s) at positions that have high centrality and ranking in terms of interconnection among ground-based MANET nodes. Along with these, multiple studies propose UAV placement strategies to maximize the interconnectivity among all ground-based nodes such as [7], [50], [6], [13], [51].

Current UAV deployment investigations typically pursue one or more of the following objectives: (a) enhancing overall coverage for all ground nodes, (b) establishing connections between ground nodes and infrastructural backbones, and (c) delivering coverage with minimal power consumption. In contrast, U-QoIT adopts a different approach by deploying UAVs to interconnect existing query nodes with optimal source nodes. Its primary focus is on linking broken or requirement-failing source-to-query node paths, aiming to maximize user usability while minimizing the deployment of UAVs.

Modeling Bushfire: In this subsection, a brief overview of related work of bushfire models, specifically those used to model Australian bushfires, is provided.

Australian fire behavior models are fuel type specific [52] with various models developed for diverse vegetation types to predict the spread rate, direction, size, height, and intensity of a propagating fire. Surveys of such models for Australian fuel types are presented in [53], [54]. Fire-spread rate prediction models can be categorized as semi-empirical, physical, and empirical [55]. Semi-empirical models integrate empirical and physical approaches and are classified into quasi-empirical or quasi-physical models, based on their closeness to an integrated method [53]. Physical models deal with the physical and chemical processes related to fuel combustion [53] and understanding the interaction of system components, however, they fail to satisfy user needs due to high computation time that precludes real-time forecasts [56], [57]. Empirical formulations [58], [59], [60] use statistical analysis to illustrate observed fire behavior and are composed of analytical functions that relate dependent variables (e.g., rate of spread) to independent variables (e.g., wind speed, fuel moisture, slope steepness, etc.) for a given fuel type [52], however, their scope is limited to the fuel complexes used in model development. Modeling approaches have unique properties for various purposes, but only the empirical and quasi-empirical models produce working models suitable for operational use due to their computational simplicity and ease of use [59]. Field-based estimation of the effects of various fuel attributes on the fire-spread rate is complex, usually hindered by the variation among fuel properties or by the correlation between fuel characteristics and fire environment [14]. Empirical-based fire-fuel relationships can reduce such limitations, by controlling the impacting variables and identifying and quantifying their effects [61].

The choice of model to be used for a particular fuel bed becomes hard with multiple fire spread models available. This study, however, assumes a dry eucalypt forest with a shrubby under-story. For a field-based fire spread model in a dry eucalypt forest with a shrubby under-story, an empirical model is developed [14], discussed in detail in subsection 2.2, to predict fire spread rate and flame height beyond the range of experimental data. The model uses inputs of fuel moisture content, wind speed, fuel height, numerical fuel hazard score, and positive slope values. The model was developed using Project Vesta experimental fires data [62] and quantitative measurements of fuel structure and tested against fire spread observations assembled from other experimental fires and from well-documented wildfires in dry eucalypt forests in southern Australia.

9 Conclusion

While MANETs are commonly employed in emergency response systems, increased dynamism in the network field leads to a higher frequency of network partitions. This escalation in partitions occurs when nodes and links are affected by the dynamic characteristics of the network field, as observed in dynamic bushfire scenarios. This paper focuses on this case study and introduces a minimum UAV deployment strategy, U-QoIT, to establish and re-establish missing/disconnected paths between information sources and their query nodes.

U-QoIT transforms the UAV deployment challenge into a polynomial-time solvable combinatorial search problem. Despite a slight increase in computational time, U-QoIT efficiently meets data transfer requirements for up to 48% more query nodes compared to its counterparts and over 100% more than a MANET without UAV deployment. Notably, U-QoIT maintains its superiority in terms of user usability percentage, packet delivery ratio, and data arrival latency in scenarios with varying densities, mobility speeds, and UAV radio ranges. The minimal simultaneous UAV utilization makes U-QoIT a promising candidate, especially when a rescue network needs connectivity for an extended period.

This study presumes that the positions of ground nodes are known for UAV deployment. Future work will assess the effectiveness of U-QoIT and its adaptation in situations where part of the network status is unknown due to partitioning or inaccessibility.

References

[1] Udaya Dampage, Lumini Bandaranayake, Ridma Wanasinghe, Kishanga Kottahachchi, and Bathiya Jayasanka. Forest fire detection system using wireless sensor networks and machine learning. *Scientific reports*, 12(1):46, 2022.

[2] University of Sydney. More than one billion animals killed in australian bushfires. https://www.sydney.edu.au/news-opinion/news/2020/01/08/australian-bushfires-more-than-one-billion-animals-impacted.html, accessed 08.01.2023.

[3] Sandra Mueller. How smart technology can minimize the threat of wildfires. https://www.momenta.one/insights/value-vector-how-smart-technology-can-minimize-the-threat-of-wildfires, accessed 10.01.2022.

[4] Ajmal Khan, Adnan Munir, Zeeshan Kaleem, Farman Ullah, Muhammad Bilal, Lewis Nkenyereye, Shahen Shah, Long D. Nguyen, S. M. Riazul Islam, and Kyung-Sup Kwak. Rdsp: Rapidly deployable wireless ad hoc system for post-disaster management. *Sensors*, 20(2), 2020.

[5] Ankur O Bang and Prabhakar L Ramteke. Manet: History, challenges and applications. *International Journal of Application or Innovation in Engineering & Management (IJAIEM)*, 2(9):249–251, 2013.

[6] Aditi Zear and Virender Ranga. Uavs assisted network partition detection and connectivity restoration in wireless sensor and actor networks. *Ad Hoc Networks*, 130:102823, 2022.

[7] Zhu Han, A. Lee Swindlehurst, and K.J. Ray Liu. Optimization of manet connectivity via smart deployment/movement of unmanned air vehicles. *IEEE Transactions on Vehicular Technology*, 58(7):3533–3546, 2009.

[8] Ameer Shakayb Arsalaan, Hung Nguyen, and Andrew Coyle. Evaluating the performance of qoi algorithms in realistic manets. In *2018 28th International Telecommunication Networks and Applications Conference (ITNAC)*, pages 1–6, 2018.

[9] Kevin S. Chan, Kelvin Marcus, Lisa M. Scott, and Rommie L. Hardy. Quality of information approach to improving source selection in tactical networks. In *18th International Conference on Information Fusion, FUSION 2015, Washington, DC, USA, July 6-9, 2015*, pages 566–573. IEEE, 2015.

[10] Mini Mathew and Ning Weng. Quality of information and energy efficiency optimization for sensor networks via adaptive sensing and transmitting. *IEEE Sensors Journal*, 14(2):341–348, 2014.

[11] Ameer Shakayb Arsalaan, Hung Nguyen, Andrew Coyle, and Mahrukh Fida. Quality of information with minimum requirements for emergency communications. *Ad Hoc Networks*, 111, 2021.

[12] Muath Obaidat, M Ali, Ihsan Shahwan, Mohammad S Obaidat, and Suhaib Obeidat. Qos-aware multipath communications over manets. *Journal of Networks*, 8(1):26, 2013.

[13] Shih-Fan Chou, Chen-Yu Yu, and Sok-Ian Sou. Efficient multi-uav-aided communication service deployment in disaster-resilient wireless networks. In *2023 IEEE Vehicular Networking Conference (VNC)*, pages 1–8, 2023.

[14] N. P. Cheney, J. S. Gould, W. L. McCaw, and W. R. Anderson. Predicting fire behaviour in dry eucalypt forest in southern australia. *Forest Ecology and Management*, 280:120–131, 2012.

[15] N. D. Burrows. Fire behaviour in jarrah forest fuels: 2. field experiments. *CALM Science*, 3:57–84, 1999.

[16] A. G. McArthur. *Control burning in eucalypt forests*. Commonwealth of Australia Forestry and Timber Bureau, Canberra, ACT, 1962.

[17] I. R. Noble, A. M. Gill, and G. A. V. Bary. Mcarthur's fire-danger meters expressed as equations. *Australian Journal of Ecology*, 5:201–203, 1980.

[18] C. E. V. Wagner. Effect of slope on rate of spread. *Canadian Department of Forestry, Bi-monthly Research Notes*, 33:7–8, 1977.

[19] Stefânia S. Faria, Nuno Leonor, Carlos A. Fernandes, João Felício, Carlos Salema, and Rafael F. S. Caldeirinha. Radiowave propagation modelling in the presence of wildfires: Initial results. In *2020 14th European Conference on Antennas and Propagation (EuCAP)*, pages 1–5, 2020.

[20] Ameer Shakayb Arsalaan, Hung Nguyen, and Mahrukh Fida. Impact of bushfire dynamics on the performance of manets. In *2021 16th Annual Conference on Wireless On-demand Network Systems and Services Conference (WONS)*, pages 1–4, 2021.

[21] Yi Zhou, Nan Cheng, Ning Lu, and Xuemin Sherman Shen. Multi-uav-aided networks: Aerial-ground cooperative vehicular networking architecture. *IEEE Vehicular Technology Magazine*, 10(4):36–44, 2015.

[22] Yuntao Wang, Zhou Su, Qichao Xu, Ruidong Li, and Tom H. Luan. Lifesaving with rescuechain: Energy-efficient and partition-tolerant blockchain based secure information sharing for uav-aided disaster rescue. In *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, pages 1–10, 2021.

[23] Zhou Su, Minghui Dai, Qichao Xu, Ruidong Li, and Hongbing Zhang. Uav enabled content distribution for internet of connected vehicles in 5g heterogeneous networks. *IEEE Transactions on Intelligent Transportation Systems*, 22(8):5091–5102, 2021.

[24] Erdem Koyuncu, Maryam Shabanighazikelayeh, and Hulya Seferoglu. Deployment and trajectory optimization of uavs: A quantization theory approach. *IEEE Transactions on Wireless Communications*, 17(12):8531–8546, 2018.

[25] Hanif Ullah, Mamun Abu-Tair, Sally McClean, Patrick Nixon, Gerard Parr, and Chunbo Luo. An unmanned aerial vehicle based wireless network for bridging communication. In *2017 14th International Symposium on Pervasive Systems, Algorithms and Networks & 2017 11th International Conference on Frontier of Computer Science and Technology & 2017 Third International Symposium of Creative Computing (ISPAN-FCST-ISCC)*, pages 179–184. IEEE, 2017.

[26] U Palani, KC Suresh, and Alamelu Nachiappan. Mobility prediction in mobile ad hoc networks using eye of coverage approach. *Cluster Computing*, 22(Suppl 6):14991–14998, 2019.

[27] Australian Government Civil Aviation Safety Authority. Drone Safety Rules. https://www.casa.gov.au/drones/drone-rules/drone-safety-rules, accessed 10.07.2022.

[28] Hafiz Suliman Munawar, Ahmed W.A. Hammad, and S. Travis Waller. Disaster region coverage using drones: Maximum area coverage and minimum resource utilisation. *Drones*, 6(4):96–123, 2022.

[29] Milan Erdelj, Michał Król, and Enrico Natalizio. Wireless sensor networks and multi-uav systems for natural disaster management. *Computer Networks*, 124:72–86, 2017.

[30] Justyna Jeziorska. Uas for wetland mapping and hydrological modeling. *Remote Sensing*, 11:1997, 08 2019.

[31] Arta Cika. *On the Modeling of Mobile Ad Hoc Networks*. PhD thesis, Oxford, United Kingdom, 2020.

[32] Muhammad Asghar Khan, Ijaz Mansoor Qureshi, and Fahimullah Khanzada. A hybrid communication scheme for efficient and low-cost deployment of future flying ad-hoc network (fanet). *Drones*, 3(1), 2019.

[33] Antonio Guillen-Perez, Ramon Sanchez-Iborra, Maria-Dolores Cano, Juan Carlos Sanchez-Aarnoutse, and Joan Garcia-Haro. Wifi networks on drones. In *2016 ITU Kaleidoscope: ICTs for a Sustainable World (ITU WT)*, pages 1–8, 2016.

[34] L. Bao and J. J. Garcia-Luna-Aceves. A new approach to channel access scheduling for Ad Hoc networks. *in Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom '01), ACM*, (1):210–221, 2001.

[35] Yao Liu, Hongjing Zhou, and Jiawei Huang. OCA-MAC: A Cooperative TDMA-Based MAC Protocol for Vehicular Ad Hoc Networks. *Sensors*, 19(12):2691, 2019.

[36] Speed, Think Metric. https://thinkmetric.org.uk/basics/speed/, accessed 03-02-2023.

[37] NAverage. Average running speed. https://www.onaverage.co.uk/speed-averages/average-running-speed, accessed 03-02-2023.

[38] Hasini Viranga Abeywickrama, Beeshanga Abewardana Jayawickrama, Ying He, and Eryk Dutkiewicz. Comprehensive energy consumption model for unmanned aerial vehicles, based on empirical studies of battery performance. *IEEE Access*, 6:58383–58394, 2018.

[39] Mehmet Ariman, Mertkan Akkoç, Tolga Sari, Muhammed Raşit Erol, Gökhan Seçinti, and Berk Canberk. Energy-efficient rl-based aerial network deployment testbed for disaster areas. *Journal of Communications and Networks*, 2023.

[40] Peter Cleveland. *Drama drones: An investigation into integrating drones into real world filmmaking in New Zealand*. PhD thesis, Auckland University of Technology, 2017.

[41] Mohammad Ammad Uddin, Ali Mansour, Denis Le Jeune, Mohammad Ayaz, and El-Hadi M Aggoune. Uav-assisted dynamic clustering of wireless sensor networks for crop health monitoring. *Sensors*, 18(2):555, 2018.

[42] Ebtehal Turki Alotaibi, Shahad Saleh Alqefari, and Anis Koubaa. Lsar: Multi-uav collaboration for search and rescue missions. *IEEE Access*, 7:55817–55832, 2019.

[43] Alessio Fascista. Toward integrated large-scale environmental monitoring using wsn/uav/crowdsensing: a review of applications, signal processing, and future perspectives. *Sensors*, 22(5):1824, 2022.

[44] Brodie Chan, Hong Guan, Jun Jo, and Michael Blumenstein. Towards uav-based bridge inspection systems: A review and an application perspective. *Structural Monitoring and Maintenance*, 2(3):283–300, 2015.

[45] Christopher S. Tang and Lucas P. Veelenturf. The strategic role of logistics in the industry 4.0 era. *Transportation Research Part E: Logistics and Transportation Review*, 129:1–11, 2019.

[46] K. Chandrashekar, M.R. Dekhordi, and J.S. Baras. Providing full connectivity in large ad-hoc networks by dynamic placement of aerial platforms. In *IEEE MILCOM 2004. Military Communications Conference, 2004.*, volume 3, pages 1429–1436 Vol. 3, 2004.

[47] Haibo Wang, Da Huo, and Bahram Alidaee. Position unmanned aerial vehicles in the mobile ad hoc network. *Journal of Intelligent & Robotic Systems*, 74:455–464, 2014.

[48] Robert A. Hunjet, Andrew Coyle, and Matthew Sorell. Enhancing mobile adhoc networks through node placement and topology control. In Rodrigo C. de Lamare, Paul D. Mitchell, Martin Haardt, Yuriy V. Zakharov, and Alister G. Burr, editors, *Proceedings of the 2010 7th International Symposium on Wireless Communication Systems, ISWCS 2010, 19-22 September 2010, University of York, York, UK*, pages 536–540. IEEE, 2010.

[49] Ekaterina V Gromova, Sergei Kireev, Alina Lazareva, Anna Kirpichnikova, and Dmitry Gromov. Manet performance optimization using network-based criteria and unmanned aerial vehicles. *Journal of Sensor and Actuator Networks*, 10(1):8, 2021.

[50] Mansi Peer, Vivek Ashok Bohara, and Anand Srivastava. Multi-uav placement strategy for disaster-resilient communication network. In *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*, pages 1–7, 2020.

[51] Na Lin, Yuheng Liu, Liang Zhao, Dapeng Oliver Wu, and Yifan Wang. An adaptive uav deployment scheme for emergency networking. *IEEE Transactions on Wireless Communications*, 21(4):2383–2398, 2021.

[52] M. G. Cruz, J. S. Gould, M. E. Alexander, W. L. Mccaw, and S. Matthews. *A guide to rate of fire spread models for Australian vegetation*. Australasian Fire and Emergency Service Authorities Council Ltd and Commonwealth Scientific and Industrial Research Organisation, 2015.

[53] A. L. Sullivan. Wildland surface fire spread modelling, 1990–2007. 1: Physical and quasi-physical models. *International Journal of Wildland Fire*, 18:349–368, 2009.

[54] A. L. Sullivan, W. L. McCaw, M. G. Cruz, S. Matthews, and P. F. Ellis. *Fuel, fire weather and fire behaviour in Australian ecosystems*. In: R.A. Bradstock, A.M. Gill and R.D. Williams (eds.), Flammable Australia: Fire Regimes, Biodiversity and Ecosystems in a Changing World, CSIRO Publishing, Collingwood, Vic, 2012.

[55] E. Pastor, L. Zárate, E. Planas, and J. Arnaldos. Mathematical models and calculation systems for the study of wildland fire behaviour. *Progress in Energy and Combustion Science*, 29:139–153, 2003.

[56] M. G. Cruz, M. E. Alexander, and A. Sullivan. Mantras of wildland fire behaviour modelling: Facts or fallacies? *International Journal of Wildland Fire*, 26:973–981, 2017.

[57] W. Mell, M. A. Jenkins, J. Gould, and P. Cheney. A physics-based approach to modelling grassland fires. *International Journal of Wildland Fire*, 16:1–22, 2007.

[58] R. C. Rothermel and H.E. Anderson. Fire spread characteristics determined in the laboratory. *U.S. Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station. 34 p*, 1966.

[59] A. L. Sullivan. Wildland surface fire spread modelling, 1990–2007. 2: Empirical and quasi-empirical models. *International Journal of Wildland Fire*, 18:369–386, 2009.

[60] D. Ward. A controlled experiment to study factors influencing fire rate of spread in pinus pinaster litter. *Forest Notes*, 9:39–47, 1971.

[61] C. G. Rossa and P. Fernandes. Empirical modeling of fire spread rate in no-wind and no-slope conditions. *Forest Science*, 64:358–370, 2018.

[62] W. L. McCaw, J. S. Gould, N. P. Cheney, P. F. M. Ellis, and W. R. Anderson. Changes in behaviour of fire in dry eucalypt forest as fuel increases with age. *Forestry Ecology and Management*, 271:170–181, 2012.

Ameer Shakayb Arsalaan is a Lecturer in Department of Computer Science at the University of Swabi and a PhD scholar at the Teletraffic Research Centre, University of Adelaide. Specializing in wireless networks, particularly MANETs, IoT, and QoI-based data transmissions, his work focuses on advanced routing protocols and adaptive algorithms to optimize network performance and security in decentralized systems. Recognized for his contributions to emergency response and autonomous vehicle networks, Arsalaan has published extensively and collaborated with industry and academic partners to bridge research with real-world applications in wireless communication.

Mah-Rukh Fida holds a PhD in Measurements and Analytics in Mobile Broadband (MBB) Computing from the University of Edinburgh, UK and Post Doc. in performance modeling in MBB networks from SimulaMet, Oslo Norway. She is currently a Senior Lecturer in the School of Business, Computing, and Social Sciences at the University of Gloucestershire, UK. In addition to her existing work on telemetry and performance evaluation within programmable networks, her research interests include applied data analytics and AI-based modeling, particularly in the fields of the Internet of Things and secure communications.

Hung X. Nguyen is an associate professor in the School of Computer and Mathematical Sciences and the leader of the Information warfare and advanced cyber theme within the Defence trailblazer, the University of Adelaide. He leads a research group on Cyber-AI, applying new advances in AI to solve problems in network fragility and security. His research focus is on developing autonomous and provable cyber defensive solutions for next generation networking and AI technologies. This means building and configuring systems that are secure by design and training trustworthy AI agents to help defend networked systems. By employing tools from graph theory, game theory and AI/ML we are able to develop practical solutions that help human operators deal with the complexity, fast-paced and deceptive nature of the cyber-physical environments.