Obstacle-Aware and Energy-Efficient Multi-Drone Coordination and Networking for Disaster Response

Chengyi Qu*, Rounak Singh*, Alicia Esquivel Morel*, Francesco Betti Sorbelli†, Prasad Calyam‡, Sajal K. Das§
* ‡ Department of ECE, University of Missouri - Columbia, USA
† Department of Computer Science and Mathematics, University of Perugia, Italy
§ Department of Computer Science, Missouri University of Science & Technology, USA
Email: *{cqy78, rsft6, ace6qv}@mail.missouri.edu, †francesco.bettisorbelli@unipg.it, ‡calyamp@missouri.edu, §sdas@mst.edu

***Abstract*—Unmanned aerial vehicles or drones provide new capabilities for disaster response management (DRM). In a DRM scenario, multiple heterogeneous drones collaboratively work together forming a flying ad-hoc network (FANET) instantiated by a ground control station. However, FANET air-to-air and air-to-ground links that serve critical application expectations can be impacted by: (i) environmental obstacles, and (ii) limited battery capacities. In this paper, we present a novel obstacle-aware and energy-efficient multi-drone coordination and networking scheme that features a Reinforcement Learning (RL) based location prediction algorithm coupled with a packet forwarding algorithm for drone-to-ground network establishment. We specifically present two novel drone location-based solutions (i.e., heuristic greedy, and learning-based) in our packet forwarding approach to support heterogeneous drone operation as per application requirements. These requirements involve improving connectivity (i.e., optimize packet delivery ratio and end-to-end delay) despite environmental obstacles, and improving efficiency (i.e., by lower energy use and time consumption) despite energy constraints. We evaluate our scheme by comparing it with state-of-the-art networking algorithms in a trace-based DRM FANET simulation testbed. Results show that our strategy overcomes obstacles and can achieve between 81-90% of network connectivity performance observed under no obstacle conditions. With obstacles, our scheme improves network connectivity performance by 14-38% while also providing 23-54% of energy savings.**

***Index Terms*—drone mobility management, disaster response, reliable network construction, learning-based scheduling**

I. Introduction

Recently, unmanned aerial vehicles (UAVs) or drones have gained attention from both government and industry communities for a plethora of civil applications such as smart agriculture [1], traffic management [2], parcel delivery [3], and disaster response [4]. In these application scenarios, multiple drones with specific capabilities can execute complex and critical tasks, such as: spreading pesticides over crops, regulating the traffic in smart cities, delivering small parcels to customers, and saving lives in search and rescue operations. Either homogeneous or heterogeneous drones are connected together forming a flying ad-hoc network (FANET) instantiated by a ground control station (GCS), which orchestrates the whole fleet of drones by sending commands to drones, and getting fresh information to decision makers on the ground.

This material is based upon work supported by the National Science Foundation under Award Number: CNS-1647182. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

In a FANET, *drones* are *flying nodes* of a network in which the GCS is in the main stationary node. Drones can typically fly almost everywhere and execute challenging tasks in a cooperative fashion [5] in order to e.g., deliver first-aid and relief goods to first responders, capture images and videos above affected areas of interest, and update maps of disaster scenes. However, drones are energy-constrained vehicles and can only operate, on average, for no more than a few tens of minutes. Depending on the geometry, nodes belonging to a FANET can form different topologies, such as star, mesh, or cluster-based. Accordingly, the connectivity of a FANET is guaranteed via a multi-hop paradigm, i.e., any two nodes can directly or indirectly communicate. In fact, a *source node* can establish a link with a *destination node* with the help of *intermediate nodes* that act as relay nodes.

A major challenge in the mobility management of drones occurs due to possible obstacles (e.g., trees, buildings) that can obstruct the radio signals and impact FANET operation/efficiency in terms of connectivity, throughput, and latency. Obstacles influence the air-to-air (A2A) and the air-to-ground (A2G) network communications between drones and the GCS, respectively [6]. In a DRM scenario, it is common to have a heterogeneous drone setup to handle different tasks. Drones spend energy not only to fly, but also to complete their assigned tasks and use residual energy for network establishment by acting as intermediate nodes to forward packets. To the best of our knowledge, there is a dearth of works that address networking problems of using heterogeneous drones that cooperatively work in a DRM scene with obstacle awareness. Particularly, there is a pent-up need for works on improving drones' *geographical obstacles avoidance* and *efficient energy usage*, while maintaining desired FANET connection performance in DRM scenarios.

In this paper, we present a novel obstacle-aware and energy-efficient multi-drone coordination and networking scheme that features a location-aided prediction algorithm coupled with a packet forwarding algorithm for drone-to-ground network establishment. Our novelty is in the approach of using Reinforcement Learning (RL) to estimate future drones' trajectories based on their coordination status and their on-board sensors information. Specifically, once the intermediate drone accurately predicts the position of the destination drone, then a list of preliminary decisions on where to forward packets is made. We consider various drone mobility models such as

Gaussian Markov Model (GMM), Mission-Based Plan Model (MBPM), and Random Way Point Model (RWPM) within our prediction technique to address the DRM application requirements. Our packet forwarding algorithm features two drone location-based solutions i.e., *heuristic greedy* and *learning-based* that can support heterogeneous drone operation requirements under DRM scenarios. The operation requirements involve improving A2A and A2G network connectivity (i.e., optimized packet delivery ratio and reduced end-to-end delay) despite environmental obstacles, and improving efficiency (i.e., by lowering energy use and time consumption) despite battery capacity limitations in re-establishing network connectivity.

We evaluate our proposed scheme by comparing it with state-of-the-art multi-drone networking algorithms in a trace-based DRM FANET simulation testbed [7]. We analyze the performance by evaluating network (i.e., packet delivery ratio, end-to-end delay) and energy (i.e., energy usage proportion for communication) metrics in several experimental settings in terms of different transmission range, amount of drones, and obstacle density. Lastly, we present results to show how our strategy overcomes obstacles and can achieve better network connectivity performance observed under obstacle-free conditions. Similarly, we present results that show how our scheme outperforms state-of-the-art algorithms in the presence of obstacles in terms of improving network connectivity performance while also providing energy savings.

The rest of the paper is organized as follows: Section II presents related work. Section III provides an overview of our DRM coordination and networking problem along with the solution approach. Section IV introduces our location prediction model. Section V details our heuristic-based and learning-based algorithms. Section VI presents our performance evaluation results. Section VII concludes the paper.

II. Related Work

Handling of DRM involves, e.g., search and rescue operations that must be managed as promptly and efficiently as possible to save human lives. In this regard, the major problem is the lack of technologies that can provide the necessary situational awareness for the incident commanders making decisions to deploy first responder resources [8]. Drones, compared with ground-based solutions such as robots or cars, provide unique advantages such as the ability to observe devastated areas from the sky, flying above possible ruins and avalanches. Additionally, drones can provide monitoring and logistic services to address handling of DRM scenarios in the absence of traditional communication infrastructure [9], [10]. Our proposed networking scheme can be used in works such as [11], [12] and help with achieving drones' monitoring function in order to provide situational awareness for rapid and effective decision making to handle DRM. Although reliable communication architectures have been studied in [13], [14], [15] in the context of drones, they do not address the underlying multi-drone coordination and networking aspects that are necessary to deploy DRM applications.

To achieve reliable communication between drones and GCS in a DRM, a suitable packet forwarding algorithm needs to be employed. In [16], authors proposed a strategy that salvages packets in the presence of void nodes, providing a low-complexity and low-overhead recovery for Greedy Geographic Forwarding (GGF) failure. In a DRM, it is crucial to consider a location-based packet forwarding protocol that can be used in ad hoc network deployments. The geographic/position-based routing protocol, Greedy Perimeter Stateless Routing (GPSR) protocol was proposed in [17], which utilizes GPS information to assist the packet forwarding procedure. Our work is highly related to studies in [18], [19], wherein authors described how a packet forwarding strategy is used inside these protocols for large density drones network deployments, through experiments in multi-drone settings. Based on results in prior works, the packet forwarding algorithm described in GPSR outperforms other state-of-the-art packet forwarding strategies described in either proactive or reactive-based protocols.

Our approach is motivated by the work in [20], in which a suitable packet forwarding algorithm for DRM operations is studied. They proposed a Location-Aided Delay-Tolerant Routing protocol (LADTR), a study that employs location-aided forwarding combined with a Store-Carry-Forward (SCF) method. The goal of the proposed strategy in this routing protocol is to guarantee the connection rate between drone nodes and enable a high packet delivery ratio.

In contrast to these prior works, our proposed approach considers environmental awareness in terms of energy consumption as well as the presence of obstacles in drone path computations at the GCS for A2A and A2G links. To enhance our packet forwarding algorithm by considering both energy efficiency and obstacles awareness, we build upon the recent prior work in [21] as well. In this work, authors studied energy consumption issues for mobile devices management in the context of a mobile edge computing paradigm; note that drone use cases were not addressed in this prior work. Their goal was to consider human mobility while handling user requirements of energy conservation over low-latency or vice versa in visual edge-based application data processing. Specifically, they presented the SPIDER algorithm, which was built upon recent advances in the geographic routing area [22]. This study presented a novel AI-augmented geographic routing approach (AGRA) that uses physical obstacle information obtained from satellite imagery by applying deep learning at a network-edge site. The SPIDER algorithm is shown to perform better as a packet forwarding strategy than other stateless geographic packet forwarding solutions, as well as, stateful reactive mesh routing in terms of packet delivery success ratio and path stretch. Our work builds on the SPIDER algorithm, and is suitable for high-density drones network deployments and considers environmental awareness in terms of energy consumption and obstacle avoidance in DRM scenarios.

III. DRM Coordination and Networking

In this section, we present the multi-drone coordination and networking problem for DRM, describing the essential system components, related requirements, and inherent assumptions. Following this, we present our solution overview.

A. Multi-Drone Networking System

We consider a DRM scenario that involves multiple critical tasks (e.g., search and rescue after an earthquake, providing relief goods to people, or simply monitoring people in a crowd) being executed by heterogeneous drones on a given FANET topology. Figure 1 illustrates the system setup including a GCS and three different types of drones i.e., *delivery drones*, *monitoring drones*, and *map drones*. The delivery drones carry first aid and relief goods to people with necessities, the monitoring drones have embedded cameras used for searching for objects and finding missing people, and finally the map drones are in charge of keeping the rescue area map up-to-date. The GCS sends requests to drones for executing specific tasks on certain locations (e.g., video recording above ruins). Furthermore, the drones will send back to the GCS the retrieved situational awareness information.

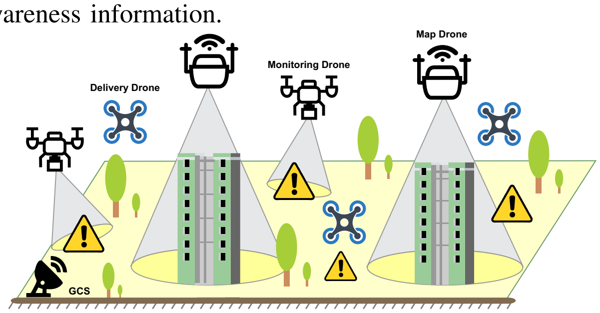

Fig. 1. Coordination of drones in the proposed DRM scenario. Delivery drones carry first aid and relief goods, monitoring drones search for missing people or objects, and map drones keep the rescue area map up-to-date.

B. System Requirements and Assumptions

The fundamental requirement that needs to be guaranteed is the establishment of reliable communication links between the drones and GCS. In general, the communication radius is not sufficiently large for allowing single-hop links, due to the fact that the topology can be arbitrary, and also the presence of obstacles can make the links unstable or even blocked. Hence, communication links should be built according to a multi-hop paradigm. In our DRM scenario, high-throughput links are required only when data transmission is in process. To save energy, delivery drones could individually deliver goods without further notice to the GCS [23], monitoring drones can perform pre-processing functions on-board without guidance from the GCS, and map drones can cache the image/video data inside the embedded storage until the queue is full.

In our scenario, the GCS is not an energy-constrained device and also its computational performance outperforms that of the drones. We assume our system to be comprised of heterogeneous drones with knowledge of their global positions because they are enabled with GPS capability. Delivery drones have pre-computed routes for carrying goods to locations and move with respect to the MBPM [24]. They have sufficient energy for executing delivery tasks and flying safely back to the ground. They can act as forwarding drones for generating connection links only if they have residual energy. Moreover, they can lose the connectivity with GCS due to network bandwidth and flying range constraints. Monitoring drones initially move according to the RWPM [25] flying through many pre-determined points of interest (PoIs). Once they discover objects or people on the ground, they change their mobility model following the GMM when executing tasks on specified target areas, as explored in [26]. Map drones fly at higher altitudes and over longer distances than the other ones, having more computation resources for executing pre-stage map generation and image mosaicing tasks [27]. However, it is not required by them to constantly transmit back the captured data to the GCS, because severe and adverse weather conditions or other unexpected events may result in task failure. Thus, we propose a periodic communication scheme between map drones and the GCS for transferring the stored data on drones to the GCS according to a suitable queuing mechanism for the data (see details in Section VI-A). Similarly, since the purpose of map drones is to cover the whole area in a short amount of time, a GMM or a MBPM is used to achieve the related task. We assume the delivery and monitoring drones' height to be 100m, while the same for map drones is 200m. In addition, we consider the average obstacles' height to be 100m.

C. Obstacle-Awareness in Communication

In our proposed scenario, monitoring drones and map drones have to request re-establishment of the packet forwarding path in order to maintain the connection to the GCS. Once this request is initiated, a packet forwarding procedure is performed. Let us consider a node n that forwards a packet p towards a destination d for re-establishing A2A and A2G communication links. The node n has to decide which neighbor must receive the packet p to progress towards d. Such a decision needs to also balance between the neighbor's residual energy and the total throughput of p, and should minimize the following:

$$f(n,d,\theta) = \theta \cdot \tau(n,d) + (1-\theta) \cdot \epsilon \tag{1}$$

where, $\tau(n,d)$ is the normalized updated shortest path approximation time [22] with respect to the obstacle blockage, ϵ is the average residual energy at node n, and $\theta \in [0,1]$ is the balancing parameter. The purpose is to find the best θ value which gives the minimum energy consumption from node n, and keeps the network connection continuous and stable by calculating the path. Once n is aware of its propagation Fresnel zone radius and the i^{th} obstacle's center C_i, it computes $\tau(n,d)$ as follows:

$$\tau(n,d) = \sum_{i=1}^{M} \frac{O_i - 1/\sqrt[\delta]{\|n-d\|}}{\|n-C_i\|^{\delta}} \tag{2}$$

where $\|\cdot\|$ represents the Euclidean distance, δ is the attenuation order of obstacles' potential field [22], M is the number of obstacles, and O_i is the intensity of i^{th} obstacle induced by the destination node d, calculated as follows:

$$O_i = \frac{F_i^{\delta}}{\delta(\|d-C_i\| + F_i)^2} \tag{3}$$

where F_i represents the i^{th} Fresnel zone in 3D. We suppose that two devices can communicate if the blockage, due to the presence of obstacles, is up to 20% of the Fresnel zone [28], otherwise, the communication is obstructed.

D. Solution Overview

As shown in Figure 2, our multi-drone coordination and network strategy consists of two main stages i.e., *data collection* (shown left) and *data transmission* (shown right). The DRM application starts with either one map drone (e.g., generating and storing data) or monitoring drone (e.g., flying for PoIs) acting as a target drone. When it requires to communicate with the GCS, a request for establishing a connection is created. At this point, if the GCS is in range, the target drone will finish the packet forwarding and the link is established; otherwise, a list of candidate drones in the neighborhood is generated. From this list, the target drone can use a simple packet forwarding algorithm (SPF algorithm, Section V-A) to establish links. Alternatively, a more accurate packet forwarding algorithm (HGPF algorithm, Section V-B) is used to optimally balance the connection link quality and energy consumption. Finally, if there is no restricted task execution time requirement on the target drone, an integrated RL based packet forwarding algorithm (IRLPF algorithm, Section V-C) is applied. In this case, the system not only achieves the balance on network connection quality and energy consumption, but also schedules task execution such that the optional forwarding drone for the neighbor drone candidate list will appear in the right position when the target drone makes a request.

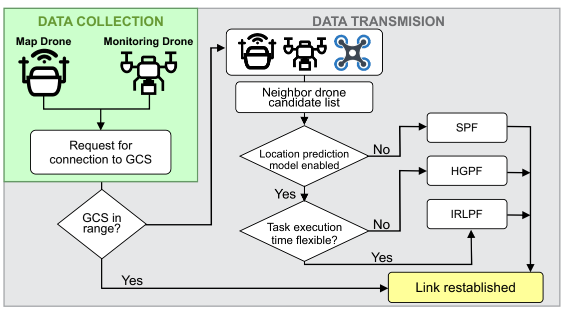

Fig. 2. Obstacle-aware and energy-efficient multi-drone coordination and networking solution steps overview.

IV. Approach for Location Prediction

As discussed earlier, we assume that all the drones are connected forming a FANET. By this, they communicate the mapping and monitoring information over the same network to the delivery drones in order to carry out a delivery task. Consequently, the network topology of the multi-drone system keeps on changing based on the mobility of the drones.

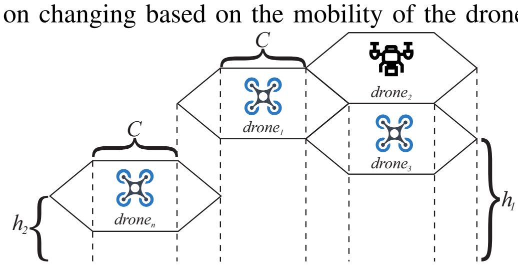

Fig. 3. Multi-drone distribution in the environment state space.

A. The Position Estimation of Drones

The position estimation of the drones must be performed in very short intervals of time using the new coordinates being updated rapidly within the FANET. Each drone in the FANET is considered to have a GPS module and an Inertial Measurement Unit to record its current location. This information is broadcast to the FANET so that the other drones in the vicinity are recognized for packet or information transfers when needed. In prior works such as [29], the Extended Kalman Filter was considered as the best algorithm choice to predict the location coordinates of a drone. However, the location information of the drones can be misleading if the sensors malfunction or the accuracy of the information gets compromised. Therefore, we have developed a novel multi-agent deep RL approach that predicts the drones' coordinates in the FANET for effective communication and packet transfer.

We allot a hexagonal area as in [30] of side C for each drone as shown in Figure 3, in order to maneuver inside them and change their heading to any direction to efficiently cover a surveillance area and discover PoIs. For initiating a packet transfer between the drones in the environment, we consider the location of them as $P_n = (x_n, y_n, h_n)$, where x_n, y_n are the location coordinates, and h_n denotes the altitude. Moreover, D_{ij} is the distance among any two drones $i \neq j$.

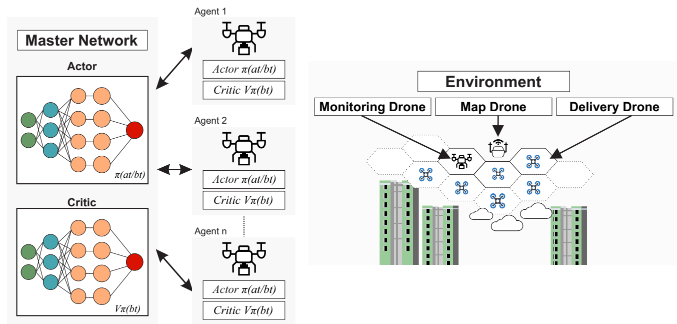

Fig. 4. Location prediction using multi-agent A3C network.

Due to the potential uncertainties in the environment that affect the drones' localization and orientation, the location prediction of the drones can be formulated as a Partially-Observable Markov Decision Process (POMDP) defined by the tuple $[S, A, P, Z, O, R]$, where S, A, P, Z, O, and R denote the states, actions, probability of transition, probability distribution function for observing states, observations, and rewards, respectively. The environment states are defined as $s_t = (P_i, \phi_i, P_j, \phi_j, D_{ij})$ where P_i and P_j are the positions, while ϕ_i and ϕ_j are the headings of the packet transferring and receiving drones. The actions performed by the drones (agents) are defined as a set of 7 flight operations a_t, i.e., hover, forward, backward, up, down, yaw-left, and yaw-right. Moreover, the rewards R_t are defined as follows: $+100$: successful packet transfer; $+10$: every drone action; -50: going out of Hexagonal cell; and -100: collision with obstacles/drones;

Let T be the episode in which the agent performs action in the state space. The agent does not track the exact states $s_t \in S$, but uses the observations $o \in O$ in any given episode T. Therefore, it has to rely on the history of actions and observations $S_t = (a_t, o_t; a_{t-1}, o_{t-1}; \ldots; a_0, o_0)$ to perform intelligent actions that allow higher rewards. However, this history S_t exponentially grows with every action taken and every state observed. The agent rather chooses to utilize the belief states b which are single valued and represent the probability distribution $Z = p(o_t|s_t, a_t)$ over all possible states s_t in a given episode. These belief states are a sufficient measure of history, and given the current belief state b_t, the

POMDP aims to find an optimal policy π^* that maximizes a value function V^π while following a sequence of actions and observations. The value function V^π is thus given by:

$$V^\pi(b) = \mathbb{E}[\sum_{i=0}^{\infty} \gamma^{i-t} R_i | b, \pi]. \quad (4)$$

Our approach, aims to solve the POMDP problem using a multi-agent Asynchronous Advantage Actor Critic (A3C) Network such that the best possible actions are chosen in given states, and the cumulative reward G_t (i.e., the accumulated discounted return) gets maximized, as follows:

$$G_t = \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t R(b_t, a_t)] \quad (5)$$

where γ is the discount factor. Finally, the action value function is given by:

$$Q_\pi(b_t, a_t) = \mathbb{E}[\sum_{n=0}^{\infty} \gamma^n R(b_t, a_t) | b_t, a_t] \quad (6)$$

B. A3C based solution

Compared to other deep RL approaches, an A3C network allows multiple agents to interact in a parallel fashion with the environment in order to generate individual policies that are the outputs. Each episode in the A3C network stochastically progresses, and each corresponding action is probabilistically sampled. The *actor* and related *critic* networks within the A3C network are deep neural networks (DNNs) and have target networks. In the following, we detail the proposed A3C network functioning. A master network has an actor network that generates a policy $\pi(a_t|b_t;\theta_a) = P(a_t|b_t;\theta_a)$ and a critic network that generates the state value function to test the expected return under belief state $V_\pi(b_t;\theta_v)$, where, b_t is the belief state of an episode, θ_a denotes weights of actor network, and θ_v denotes the weight of critic network. The weights in these cases are updated using back-propagation. Each copy of the Master Network is sent to the agents as shown in Figure 4. The actor network is trained with the loss function:

$$L(\theta_a) = \frac{1}{N_{batch}} \sum_{i=1}^{n} [-Q(b_t(i), \pi_b(i); \theta_a)]^2 \quad (7)$$

where, N_{batch} is the batch size and the critic network is trained with the loss function given by:

$$L(\theta_v) = \frac{1}{N_{batch}} \sum_{i=1}^{n} [y_t(i) - Q(b_t(i), a_t(i); \theta_v)]^2 \quad (8)$$

where $y_t(i) = R_t + \gamma \cdot Q^{'}(b_{t+1}, \pi^{'}(b_{t+1}; \theta_a^{'}) | \theta_v^{'})$ and $\pi^{'}(b|\theta_a^{'})$ and $Q^{'}(b, a|\theta_v^{'})$ are target networks of actor and critic networks. Due to the fact that gradient methods are used to optimize the network weights, there are chances of high variance occurrence in the critic network. Therefore, we employ an advantage function $\Omega(b_t, a_t) = Q(b, a) - V^\pi(b_t, \theta_v)$ to overcome this high variance problem. Upon solving the POMDP using the proposed A3C network, we get a policy $\pi : b \rightarrow a$ that maps the actions to belief states b. The optimal policy obtained from the actor and critic network operations is given by:

$$\pi^*(b) = \arg\max_b V^\pi(b) \quad (9)$$

V. Packet Forwarding Solution Approach

In this section, we initially describe a simple technique called Simple Packet Forwarding (SPF), and two novel algorithms called Heuristic Greedy Packet Forwarding (HGPF) and Integrated RL-based Packet Forwarding (IRLPF).

A. The SPF Algorithm

To address the trade-off in energy-efficiency versus continuous connection, there is a need for flexible packet forwarding algorithms among the drones and GCS. A choice among *stateful* and *stateless* packet forwarding design has to be carefully determined based on the cost of maintaining forwarding paths or infinite loops in paths [21]. Consequently, we rely on a stateless instead of stateful design, and we consider on-demand communications when requested. By design of SPF, if any drone is within the transmission range of a target drone, then it will be selected as a candidate forwarding drone for forwarding packets. If there are no candidate drones, the target drone will hover and wait in the current position, thus pausing the task. As a second attempt, if no forwarding drone candidate is within the transmission range, the target drone forwards the data to the farthest drone in range. Note that any drone can act as an intermediate or forwarding drone. The data will be transferred through multi-hop drones to the GCS. Since SPF does not use any predicted drones' positions, it can be implemented in any drone.

TABLE I
SPF algorithm performance comparison

	Throughput (Mbps)				Energy (J)	
	GPSR	AODV	HWMP	Our	GPSR	Our
low	2.48±1.7	1.94±1.8	0.84±0.2	2.18±0.6	522±46	519±42
	2.35±0.9	0.87±0.3	0.77±0.2	2.26±0.6	515±13	506±18
high	2.06±1.8	1.26±1.5	0.77±0.3	2.05±1.1	497±47	498±40
	1.95±0.8	0.62±0.4	0.72±0.2	2.32±0.5	499±16	495±40

Despite its simplicity, it outperforms in terms of energy consumption and network throughput, other existing stateless algorithms which do not rely on predictions. As a baseline, we have chosen Greedy Perimeter Stateless Routing (GPSR) [31], Ad-Hoc On Demand Distance Vector (AODV) [32], and the Hybrid Wireless Mesh Network (HWMP) [33]. Table I shows the average throughput and energy consumption on target drone for link establishment with different density of obstacles (10%: low, 60%: high) and number of drones (10, 30) with the simulation settings described in Section VI. Since each protocol follows different path forwarding strategies (e.g., AODV and HWMP do not consider energy constraints), it is hard to compare side-by-side performance with each of these approaches. Hence, in this comparison, some part of the results are omitted. However, if we do not consider predicting the future positions of drones, both SPF and GPSR are Pareto-optimal, i.e., they have no alternative solutions. In order to have improvements, more features need to be considered with additional information as detailed in the following algorithms.

B. The HGPF Algorithm

As we can observe from SPF, if the prediction of the future position of the drones is not used, the target drone may have a lesser chance to find the potential candidate forwarding drone when it flies beyond the GCS's transmission range. Consequently, in HGPF design we use the RL-based location prediction A3C method to enhance the packet forwarding performance in the presence of environmental obstacles, while using the residual energy from the forwarding drone candidates. Based on the drone position prediction information, we can obtain the local relative distance between the drones in advance. Thus, to increase the accuracy of the drone path computation procedure, we use HGPF that proactively performs a greedy calculation of the local-optimal path solutions.

To simplify our algorithm and save time in real-world experiments, we periodically run the RL location prediction every 10 seconds according to state-of-the-art drone position model evaluation metrics [34]. For every 10 seconds, we found that the corresponding time is within acceptable range (2.20 ± 0.05 s), and the estimation error of the model is relatively small (0.43 meters). Therefore, for each time slot, we can estimate the $f(n, d, \theta)$ results given $\theta \in \{0, 0.2, 0.5, 0.8, 1\}$ values. Corresponding to these results, we select the minimum value f with the related θ value to guide the next 10 seconds flight. Thus, by such a greedy finding of the discrete local minimum value, we enhance the overall performance in terms of the total residual energy and overall throughput.

The main purpose of HGPF is to utilize the drone position model for predicting the relative position of drones. The time period t in the algorithm is by default set to 10 seconds. For each time period, all θ values are calculated for each alternate neighbor node. Following this, we calculate the performance gain on the drone (i.e., throughput gain by processed energy) of each θ, and perform a greedy select of the highest performance gain. In addition, we will use this θ until the next time period t, when HGPF is invoked again. Consequently, HGPF will always select the local optimal choice, which however may not be the overall best for the entire system.

C. The IRLPF Algorithm

Since HGPF provides the local optimal choice by time period, it may lack information on the global optimal performance gain over the entire system. However, during DRM, the position of the geographical obstacle may change frequently. In this case, it is not reasonable to obtain predictions of exact positions of obstacles that may block the connection. Thus, we propose an alternative way to indirectly provide forwarding drone candidates to target drone positions by scheduling the take-off time for different tasks. For example, once a monitoring drone takes off and starts the task, it will require a connection when PoIs are in sight. If we plan to take-off a delivery drone in the monitoring drone's transmission range, the monitoring drone will have a higher chance to search and find the delivery drone and select it as a forwarding drone. To achieve this, we abstract the multi-drone multi-task packet forwarding procedure as a Markov-decision process (MDP), which can provide a mechanism to evaluate and learn the potential take-off time and the θ value by using rewards. Thus, the optimization problem can be redefined as: *find the optimal task schedule which minimizes the global performance gain for the packet forwarding link generation*. This finite-time MDP problem can be expressed as $M = (S, A, P, R, T)$ where S is the state space, A is the action space, P is the probability function that indicates the probability of action $a \in A$ in state $s \in S$ at time $t \geq 0$ will lead to state $s' \in S$ at time $t + 1$, R is a reward function, and T represents the last acceptable time when the drone has to take-off and process the task.

To ensure that the near optimal θ can be chosen at every step, we consider the dynamic decision-making problem to be used for downstream tasks at time t. The situation will turn to an energy-oriented case, when action goes to -1. At the same time, the situation will turn to a throughput-oriented case, when action goes to 1 in terms of the chosen step size δ value. In the case where the δ value is small, the system will take more resources (i.e., energy and time) to calculate the optimal value, and each action change may only produce small reward gains. Thus, we require the δ value selected to be at most 0.5, and greater than 0.15 to prevent the void and eliminate redundant calculations.

Specifically, we obtain the state s in the MDP problem using information about the actions a with a given time period t. The state s thus contains the stored previous prediction f results, and the remaining query budget on both energy-oriented as well as throughput-oriented cases. The state will also record the time when the potential forwarding drone takes-off and starts the task. Thus, the state s for a given time period t can be formalized as follows:

$$s^t = [\theta_t, f_{previous}, f_{(\theta-\delta, \theta+\delta)}, T - t] \tag{10}$$

The reward function is given to minimize the cost of drone-based flight energy ϵ and obstacle-awareness recovery time τ given in Equation (1). Consequently, we considered the reward function to be the same as in Equation (1), with respect to the energy consumption on a single drone and considering theoretical guarantees on packet delivery in obstacle situations. Thus, we can ultimately define the reward function as follows:

$$R^t(s^t, a^t) = -\alpha \cdot \underbrace{\epsilon(f^t, \hat{f}^t)}_{\text{residual energy}} - \beta \cdot \underbrace{\tau(cost(a^t))}_{\text{obstacle}} \tag{11}$$

Having defined the obstacle-aware drone delivery scenario as an MDP, we can evaluate the overall performance by minimizing the expected total reward the system achieves. In this context, we can state: Given the choice of optimized θ values for highest gains of f, a finite horizon of T time slot, and an MDP problem M, find the optimal routing policy $\pi : s \to a$ that maximizes the expected cumulative reward R:

$$\pi \in \arg\max \mathbb{E}\Big(\sum_T R(s, a)\Big) \tag{12}$$

Although we framed this problem as an MDP, it is not easy to apply conventional techniques such as dynamic programming for solving the problem. This is because, many of the aspects of this problem are hard to analytically characterize, especially the dynamics of the sensory input stream (e.g., GPS, obstacle, energy). This motivates our integration of a soft optimal solution that uses a model-free RL technique. The reason to use such a technique is as follows: it is capable of learning optimal discrete policies based solely on the features included in the state, and avoids the need to predict the future states (as considered in HGPF). Specifically, we use the state-of-the-art Q-learning algorithm [35], which is easy to deploy, efficient to evaluate in terms of dynamics, and is amenable to effectively perform optimization-based action selection.

VI. Performance Evaluation

In this section, we first discuss our experiments setup, and provide details of environment settings, evaluation metrics, and relevant parameters used in the experiments. Following this, we discuss the baseline approaches used for comparison. Finally, we present the detailed performance results and discuss the benefits of our approach.

A. Experiment Setup

To evaluate our packet forwarding algorithms, we initialized a simulated urban environment using 3D building models in the ns-3 simulator based on a urban road map. The simulation script randomly generates task locations (e.g., delivery points, rescue areas) in the range of the simulated DRM. The simulator scripts can change the density of the buildings to simulate various DRM scenarios, i.e., from urban areas to metropolitan areas, with different building obstacle densities. Table II shows the basic setting of the simulations, including various application and network settings. The flying altitudes vary from 100m for delivery drones and monitoring drones, and 200m for map drones. To run the experiments in a reproducible and reliable testbed, we leverage a trace-based drone-edge simulation platform on top of ns-3 [7]. This platform integrates simulation of both drones and networks for DRM scenarios, and provides flexibility in adding plugins, e.g., changing the mobility model of the drones, adding multi-sensor simulations, and applying realistic map interfaces. For each setting, we run the experiments in: (i) 5 different transmission settings ranging from 50-250m, (ii) various number of drones ranging from 10-30, and (iii) various density percentages of obstacles in DRM ranging from 0-60%. Settings are summarized in Table II.

TABLE II
Environment Settings used in the Simulation Experiments

Application Settings		**Network Settings**	
Number of drones:	10-30	Transport protocol:	RUDP
Disaster area:	10-15 miles	Application Bit rate:	6 Mbps
Obstacle size:	60*30*95 m	Tx power:	32-48 dBm
Transmission range:	50-250 m	Tx/Rx gain:	3 dB
Simulation time:	1000-3000 s	WIFI protocol	802.11 n/ac
Avg. drone speed:	10 - 35 mph	Modulation:	OFDM
Prop. Model:	TWO RAY	Data rate:	65 Mbps

To evaluate the performance of HGPF and IRLPF algorithms, we compared them with state-of-the-art energy-aware or location-aware packet forwarding algorithms. In this context, we used the following performance metrics in our simulation environment:

Packet Delivery Ratio: Packet delivery ratio is defined by the ratio of the packets that are successfully delivered to the target location (e.g., GCS). A higher packet delivery ratio means a more reliable network connection.

End-to-end Delay: The packet forwarding end-to-end delay is calculated as the sum of the link re-establishment duration when a drone carries its generated data, and the data transmission time to the GCS. A lower end-to-end delay implies better performance on network links.

Energy Usage Proportion: For the energy measurement, we check the distribution of the average residual energy percentage used for the networking procedure of all drones in the DRM scenario. Given that the total energy usage varies for each kind of drone, we calculate the energy usage proportion in order to normalize the metrics. A lower energy usage proportion means the drone could use more energy for its own task (e.g., monitoring, search and rescue, mapping or delivery).

B. Baseline Solution Approaches

To design a better packet forwarding approach for the multi-drone coordination problem, we compare both HGPF and IRLPF algorithms with several existing state-of-the-art algorithms. All experiments use the same settings previously detailed. We used three different baselines (BLs) approaches detailed below based on their different perspectives:

BL 1 – GPSR [31]: Given how we considered GPSR with SPF in Section V, we employ GPSR as a baseline for comparing our location-aided packet forwarding approaches. Although GPSR does not use any predictions based on mobility, it provides location awareness information for packet forwarding that can be used as a reference from the location awareness perspective. GPSR is considered as a widely used baseline method in multi-drone communication system design [36].

BL 2 – SPIDER [21]: This algorithm solves the packet forwarding problem by considering edge devices' energy consumption as well as presence of obstacles, in a manner that is very close to our system formulation. However, in terms of obstacle awareness, our approach considers three dimensional obstacles with consideration of the Fresnel Zone on communication blockage, which is not involved in the design of the SPIDER algorithm.

BL 3 – LADTR [20]: It consists of a location-aided drone packet forwarding algorithm design. One difference between LADTR and our location prediction model is that LADTR only considers GMM, while ours considers various drone mobility models and uses RL to enhance the location prediction precision. In addition, LADTR does not consider obstacles that may block the communication, whereas we consider obstacles in our DRM packet forwarding solutions.

C. Performance Results

In this section, we evaluate the performances under two metrics, i.e., network quality and energy usage.

Network Quality: Figure 5 shows the results on the network performance for all 5 packet forwarding algorithms.

Considering various transmission ranges of the nodes (Figures 5a and 5d), we can see that all the algorithms perform better in larger range cases. This is because a large range provides more contact opportunities for creating transmission links, which will in turn increase the packet delivery ratio and decrease the delay. HGPF and IRLPF outperform GPSR and SPIDER under all transmission settings. LADTR performs better than other state-of-the-art strategies due to its use of ferrying drones that use up all of their energy on networking procedures instead of focusing on their own tasks. On the contrary, IRLPF which only uses residual energy on transmission, achieves $\approx 90\%$ of the performance of LADTR. This result shows that the performance of IRLPF is comparable to the LADTR, which although it has a better performance, has the problem of energy wastage in drone ferrying.

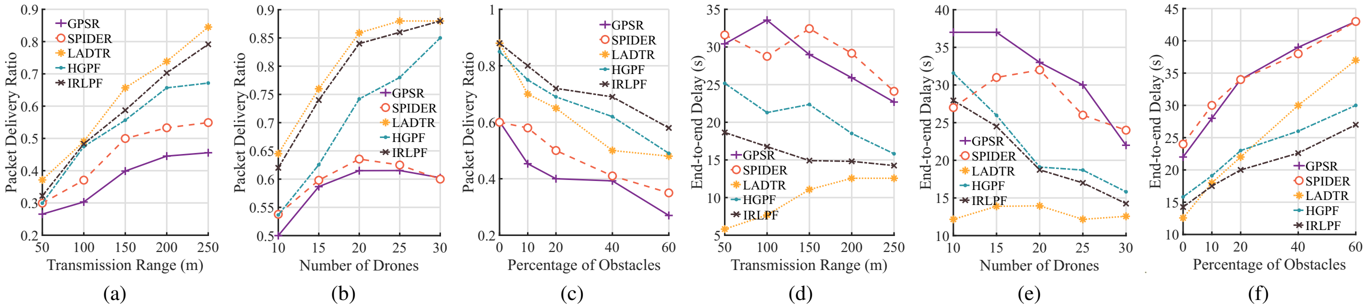

Fig. 5. Performance metrics on packet delivery ratio (a,b,c) and average end-to-end delay (d,e,f) in terms of varying: 1) transmission range, 2) amount of drones, and 3) percentage of obstacles in a realistic DRM scenario.

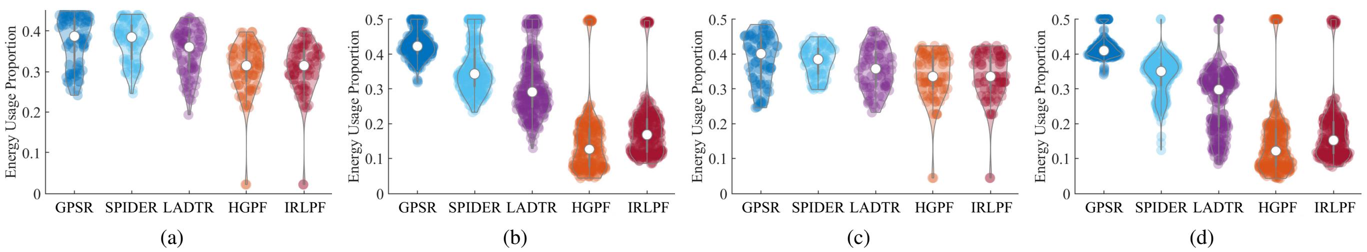

Fig. 6. Energy usage proportion for communication in 40% of obstacles with four different experiments with varying transmission range and drone amounts for the cases: a) 50m, 10 drones; b) 50m, 30 drones; c) 250m, 10 drones, and d) 250m, 30 drones

From Figure 5b it is possible to observe that when the number of drones increases, SPIDER and GPSR do not provide better results due to the unpredictable drone positions and unplanned forwarding node selection. By combining Figures 5d and 5e together, we can state that LADTR's results do not provide lower end-to-end delay when the transmission range increases or when more drones are added. This is because LADTR follows a stateful setting rather than a stateless one, which is used in other approaches. Thereby, the forwarding node for transmission will not change once selected, and the delay for LADTR remains unchanged once a network connection is established.

Both the above experiments do not consider any obstacles in the DRM application scenario. However, if we consider obstacle blockage of the connections, both HGPF and IRLPF outperform the other baseline approaches. Figures 5f and 5c depict the network performance with the increase in the percentage of obstacles in the DRM scenario. It is obvious that the network quality will decrease with more obstacles for all approaches. Given that we considered algorithms that can deal with obstacles, other solutions without obstacle awareness (e.g., LADTR and GPSR) experience poor performance than HGPF or IRLPF. Although LADTR can achieve similar end-to-end delay under low obstacle settings, this condition is not always promising in complex metropolitan DRM scenarios with high building obstacle densities.

To conclude, without considering obstacles, LADTR could achieve better network performance due to the stateful connection settings, and our solutions can achieve $\approx$ 81-90% of the LADTR performance. However, LADTR is not robust considering that obstacle blockage is common in DRM scenarios, and in such situations, an average of $\approx$ 14-38% promotion is obtained by using our baseline approaches.

Energy Usage: Violin distribution diagrams shown in Figure 6 depict results from four experiments with varying transmission ranges (50m, 250m), 40% of obstacle settings, and varying number of drones (10 drones, 30 drones). We can observe from all four experiments that both HGPF and IRLPF save $\approx$ 23-54% of energy. This is because: (i) establishment of network connectivity is not performed if there are no transmission tasks, and (ii) only residual energy is used to establish network links for generating forwarding drone candidates. Moreover, as we can observe from Figures 6b and 6d, HGPF saves more energy in larger drone number situations. Due to the fact that IRLPF presents more forwarding drone candidates by intelligently scheduling the task execution in presence of more and more drones, an increased energy consumption occurs while simultaneously providing guarantees on better network connection quality in comparison with HGPF.

Discussion of Results: Herein, we discuss the choice for the most appropriate packet forwarding algorithm in realistic DRM scenarios. First, if drones have no restrictions on take-off time, IRLPF is the most suitable because it will find the optimal task schedule. Consequently, it provides better network performance compared to HGPF, while also saving energy. Second, if drones need to be scheduled in advance for accomplishing specific tasks in accordance with a restricted time table while saving energy, HGPF is preferred. This is because HGPF can provide acceptable network quality performance with considerable energy savings. Third, as the number of drones increases, both HGPF or IRLPF are acceptable in terms of conserving energy usage, irrespective of the number of drones. Consequently, they can provide a pertinent solution for DRM application operations involving large number of drones in the presence of obstacles. Lastly, in DRM scenarios where there are no obstacle blockages and energy consumption is not an issue, LADTR will be the best one in terms of network performance. However, the links generated by LADTR are unstable and unreliable when they are blocked due to obstacles. Thus, in cases where the environmental condition of a DRM scenario is unknown, HGPF or IRLPF algorithm are more pertinent for DRM application operations.

VII. Conclusion

In this paper, we proposed a novel *obstacle-aware packet forwarding scheme* for multi-drone cooperation in various applications related to disaster response management (DRM). Our contributions advance current knowledge on the design and development of location-aided packet forwarding for FANETs with consideration of geographical obstacles blockage, and energy efficiency for coordinated drone flights in DRM application scenarios. Specifically, we devised an accurate drone location position prediction method using the deep RL to suit multi-drone DRM application scenarios. We also proposed two different algorithms, i.e., HGPF and IRLPF algorithms that utilized the drone location predictions for improving the network connectivity, while also providing better energy efficiency compared to state-of-the-art approaches such as GPSR, SPIDER, and LADTR. Specifically, our proposed schemes outperformed the state-of-the-art approaches in terms of network connectivity performance (e.g., packet delivery ratio, end-to-end delay) and energy usage (i.e., energy usage proportion for communication) in experiments with a trace-based drone-edge simulation platform featuring different transmission ranges, amount of drones, and obstacle density.

Future work can consider dynamics of wind conditions to be integrated via a wind model into the obstacles formulation. Our solution can also be extended to DRM applications that require sustained video throughput in A2G links, while also avoiding disruptions in A2A and A2G connectivity.

References

[1] P. M *et al.*, "Unmanned aerial vehicles in smart agriculture: Applications, requirements, and challenges," *IEEE Sensors Journal*, 2020.

[2] R. Rumba and A. Nikitenko, "The wild west of drones: a review on autonomous-uav traffic-management," in *2020 Int. Conf. on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2020, pp. 1317–1322.

[3] F. B. Sorbelli, F. Corò, S. K. Das, and C. M. Pinotti, "Energy-constrained delivery of goods with drones under varying wind conditions," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[4] M. T. Rashid, D. Zhang, and D. Wang, "Demo abstract: Social media-driven uav sensing frameworks in disaster response applications," in *INFOCOM Workshops*. IEEE, 2020, pp. 1340–1341.

[5] A. Khochare, Y. Simmhan, F. B. Sorbelli, and S. K. Das, "Heuristic algorithms for co-scheduling of edge analytics and routes for uav fleet missions," in *IEEE INFOCOM*, 2021.

[6] F. Betti Sorbelli, C. M. Pinotti, S. Silvestri, and S. K. Das, "Measurement errors in range-based localization algorithms for *UAVs*: Analysis and experimentation," *IEEE Trans. on Mobile Computing*, 2021.

[7] C. Qu, A. E. Morel, D. Dahlquist, and P. Calyam, "Dronenet-sim: a learning-based trace simulation framework for control networking in drone video analytics," in *6th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, 2020, pp. 1–6.

[8] M. Erdelj, E. Natalizio, K. R. Chowdhury, and I. F. Akyildiz, "Help from the sky: Leveraging uavs for disaster management," *IEEE Pervasive Computing*, vol. 16, no. 1, pp. 24–32, 2017.

[9] V. Mayor, R. Estepa, A. Estepa, and G. Madinabeitia, "Deploying a reliable uav-aided communication service in disaster areas," *Wireless Communications and Mobile Computing*, vol. 2019, 2019.

[10] P. Rotter and W. Chmiel, "A framework for a hierarchical model of cooperation between unmanned airplanes," *European Journal of Remote Sensing*, vol. 51, no. 1, pp. 276–284, 2018.

[11] B. Shishkov, S. Hristozov, and A. Verbraeck, "Improving resilience using drones for effective monitoring after disruptive events," in *9th Int. Conf. on Telecommunications and Remote Sensing*, 2020, pp. 38–43.

[12] Y. Gao, Z. Lyu, H. Assilzadeh, and Y. Jiang, "Small and low-cost navigation system for uav-based emergency disaster response applications," in *4th Joint Int. Symposium on Deformation Monitoring*, 2019, pp. 15–17.

[13] J. Xu, K. Ota, and M. Dong, "Big data on the fly: Uav-mounted mobile edge computing for disaster management," *Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2620–2630, 2020.

[14] K. G. Panda, S. Das, D. Sen, and W. Arif, "Design and deployment of uav-aided post-disaster emergency network," *IEEE Access*, vol. 7, pp. 102 985–102 999, 2019.

[15] W. Ejaz, M. A. Azam, S. Saadat, F. Iqbal, and A. Hanan, "Unmanned aerial vehicles enabled iot platform for disaster management," *Energies*, vol. 12, no. 14, p. 2706, 2019.

[16] M. Biomo *et al.*, "Routing in unmanned aerial ad hoc networks: A recovery strategy for greedy geographic forwarding failure," in *2014 Wireless Comm. and Networking Conf.* IEEE, 2014, pp. 2236–2241.

[17] B. Karp and H.-T. Kung, "Gpsr: Greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th annual Int. Conf. on Mobile computing and networking*, 2000, pp. 243–254.

[18] M. Hyland, B. E. Mullins, R. O. Baldwin, and M. A. Temple, "Simulation-based performance evaluation of mobile ad hoc routing protocols in a swarm of unmanned aerial vehicles," in *AINAW'07*, vol. 2. IEEE, 2007, pp. 249–256.

[19] R. Shirani, M. St-Hilaire, T. Kunz, Y. Zhou, J. Li, and L. Lamont, "The performance of greedy geographic forwarding in unmanned aeronautical ad-hoc networks," in *2011 Ninth Annual Communication Networks and Services Research Conf.* IEEE, 2011, pp. 161–166.

[20] T. Akram, M. Awais, R. Naqvi, A. Ahmed, and M. Naeem, "Multicriteria uav base stations placement for disaster management," *IEEE Systems Journal*, vol. 14, no. 3, pp. 3475–3482, 2020.

[21] H. Trinh, P. Calyam, D. Chemodanov, S. Yao, Q. Lei, F. Gao, and K. Palaniappan, "Energy-aware mobile edge computing and routing for low-latency visual data processing," *IEEE Transactions on Multimedia*, vol. 20, no. 10, pp. 2562–2577, 2018.

[22] D. Chemodanov, F. Esposito, A. Sukhov, P. Calyam, H. Trinh, and Z. Oraibi, "Agra: Ai-augmented geographic routing approach for iot-based incident-supporting applications," *Future Generation Computer Systems*, vol. 92, pp. 1051–1065, 2019.

[23] R. Kellermann, T. Biehle, and L. Fischer, "Drones for parcel and passenger transportation: A literature review," *Transportation Research Interdisciplinary Perspectives*, vol. 4, p. 100088, 2020.

[24] M. B. Yassein, N. Alhuda *et al.*, "Flying ad-hoc networks: Routing protocols, mobility models, issues," *Int. Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 7, no. 6, 2016.

[25] A. Chriki, H. Touati, H. Snoussi, and F. Kamoun, "Fanet: Communication, mobility models and security issues," *Computer Networks*, vol. 163, p. 106877, 2019.

[26] A. Bujari, C. T. Calafate, J.-C. Cano *et al.*, "Flying ad-hoc network application scenarios and mobility models," *Int. Journal of Distributed Sensor Networks*, vol. 13, no. 10, p. 0, 2017.

[27] H. Trinh, J. Li, S. Miyazawa, J. Moreno, and S. Pankanti, "Efficient uav video event summarization," in *Proceedings of the 21st Int. Conf. on Pattern Recognition (ICPR2012)*. IEEE, 2012, pp. 2226–2229.

[28] "Fresnel zone," Jan 2021. [Online]. Available: https://en.wikipedia.org/wiki/Fresnel_zone

[29] C. Luo, S. I. McClean, G. Parr *et al.*, "Uav position estimation and collision avoidance using the extended kalman filter," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 6, pp. 2749–2762, 2013.

[30] P. Sujit and D. Ghose, "Search using multiple uavs with flight time constraints," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 2, pp. 491–509, 2004.

[31] C. Lochert, H. Hartenstein, J. Tian *et al.*, "A routing strategy for vehicular ad hoc networks in city environments," in *IV2003 Intelligent Vehicles Symposium*. IEEE, 2003, pp. 156–161.

[32] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings WMCSA'99. Second IEEE Workshop on Mobile Computing Systems and Applications*. IEEE, 1999, pp. 90–100.

[33] G. R. Hiertz, D. Denteneer, S. Max, R. Taori, J. Cardona, L. Berlemann, and B. Walke, "Ieee 802.11 s: the wlan mesh standard," *IEEE Wireless Communications*, vol. 17, no. 1, pp. 104–111, 2010.

[34] J. Peng, P. Zhang, L. Zheng, and J. Tan, "Uav positioning based on multi-sensor fusion," *IEEE Access*, vol. 8, pp. 34 455–34 467, 2020.

[35] D. Tamagawa, E. Taniguchi, and T. Yamada, "Evaluating city logistics measures using a multi-agent model," *Procedia-Social and Behavioral Sciences*, vol. 2, no. 3, pp. 6002–6012, 2010.

[36] A. Rovira-Sugranes, F. Afghah, J. Chakareski, and A. Razi, "A review of ai-enabled routing protocols for uav networks: Trends, challenges, and future outlook," *arXiv preprint arXiv:2104.01283*, 2021.