UAV Charging Station Placement in Opportunistic Networks

Salih Safa Bacanli [1], Enas Elgeldawi [2], Begümhan Turgut [3] and Damla Turgut [1,*]

1   Department of Computer Science, University of Central Florida, Orlando, FL 32816, USA
2   Department of Computer Science, Faculty of Science, Minia University, Minia 61519, Egypt
3   Department of Computer Science, Rutgers University, Piscataway, NJ 08854, USA
*   Correspondence: damla.turgut@ucf.edu

Abstract: Unmanned aerial vehicles (UAVs) are now extensively used in a wide variety of applications, including a key role within opportunistic wireless networks. These types of opportunistic networks are considered well suited for infrastructure-less areas, or urban areas with overloaded cellular networks. For these networks, UAVs are envisioned to complement and support opportunistic network performance; however, the short battery life of commercial UAVs and their need for frequent charging can limit their utility. This paper addresses the challenge of charging station placement in a UAV-aided opportunistic network. We implemented three clustering approaches, namely, K-means, Density-Based Spatial Clustering of Applications with Noise (DBSCAN), and random clustering, with each clustering approach being examined in combination with Epidemic, Spray and Wait, and State-Based Campus Routing (SCR) routing protocols. The simulation results show that determining the charging station locations using K-means clustering with three clusters showed lower message delay and higher success rate than deciding the charging station location either randomly or using DBSCAN regardless of the routing strategy employed between nodes.

Keywords: UAV; opportunistic networks; charging stations; clustering; DBSCAN; K-means

1. Introduction

Unmanned aerial vehicles (UAV)s, also known as drones, are aircraft that can fly autonomously or as interconnected swarms to perform various tasks, such as data collection and delivery. Although UAVs have been used by the military for many years to fulfill a multitude of mission profiles across surveillance, rescue, border observation, firefighting, etc., they are now commercially available, with prices ranging from hundreds of dollars for small short-range drones to professional sensor-powered drones that cost thousands of dollars. As UAVs have gained popularity, their use has spread to a wide variety of civilian applications, such as traffic monitoring [1,2], public safety [3], emergency communications [4], and many others [5–8]. Recently, companies such as Amazon, UPS, and Google are adding drones to their vehicle fleet to make delivery services to customers much faster [9,10]. We expect that this broad appeal, derived from UAVs characteristics of mobility and flexibility, will increase our contact with UAVs in our daily lives.

One challenge to the increased civilian usage of UAVs is that of limited battery life. Civilian UAVs are generally smaller than military variants with lower charge capacities, restricting cruise times to approximately 30 min and, thus, resulting in more frequent recharge cycles. To address this challenge, the application scenario we will consider is a UAV-aided opportunistic network. In a UAV-aided opportunistic network, the UAVs cruise around an area and communicate with both mobile nodes on the ground and other UAVs in a peer-to-peer manner, as shown in Figure 1. As the routes between the mobile nodes are created dynamically via the passing UAVs overhead, messages created by the mobile nodes are distributed by the UAVs using the store-carry-and-forward paradigm. In this model each node will store a received message and forward it along to newly discovered nodes who continue the pattern until the ultimate recipient is encountered. The integration

of UAVs in this type of network can provide various benefits, however, the UAVs' limited on-board battery life restricts their flight time. An efficient charging strategy can help achieve maximum productivity of the UAVs.
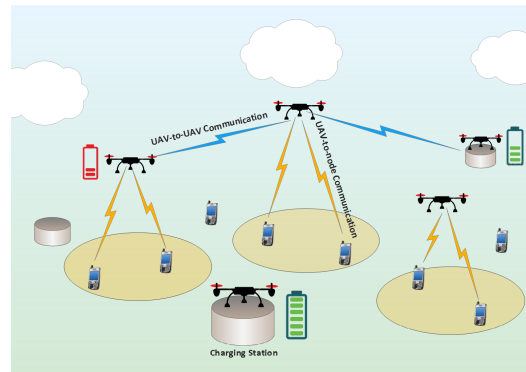

Figure 1. UAV-aided opportunistic network architecture.

One such challenge that UAVs have been applied to is map scanning. Although few studies used one UAV for scanning the entire map [11–13], scanning the whole map with a single UAV may not always be possible or practical, as the flight time is relatively short for commercially available UAVs. To resolve this issue, a scanning method to cluster the map into small sections should be developed along with an approach to maximize the flight time of the UAVs through the strategic placement of charging stations. The locations of these stations are critically important as the UAVs need to keep reasonable proximity to a charging station so they can return whenever they drain their batteries. Charging stations that are too close together may affect the UAVs' ability to scan different parts on the map while stations that are too far away from each other may cause fragmented areas to appear on the map.

Another challenge that arises, due to the nature of the opportunistic network environment, is the selection of an appropriate packet routing strategy. Routing is the decision-making process in which a given node selects one or more of its outgoing links to forward a packet on its journey to the final destination. The routing mechanism is responsible for choosing appropriate paths to transfer data based on network requirements and restrictions. In addition to providing a route to the destination, many of the existing routing protocols also aim to optimize various metrics, such as minimizing energy consumption, congestion, storage, maintenance, etc. [14–17]. As the topology of opportunistic networks is dynamic lacking fixed end-to-end transmission routes between nodes, traditional topology-based routing algorithms cannot be directly applied. Thus, implementing a suitable routing algorithm remains as challenging problem in opportunistic network research.

In this paper, we investigate the problem of UAV charging station placement as batteries limit the flight time of the UAVs. The contributions of this work are summarized as follows:

- We define an application scenario where commercially available UAVs are leveraged in an opportunistic environment.
- We introduce a cluster-based solution for UAVs' limited-lifetime battery problem. Through clustering, load balancing is achieved, which, in turn, has an impact on UAV throughput.
- For our application scenario, we have tested three methods (random placement, with K-means clustering and DBSCAN) to decide the position of the charging stations. We also examined Epidemic, Spray and Wait, SCR packet routing strategies for each charging station placement method. The charging station placement techniques are evaluated by performance metrics of different routing strategies on the application scenario.

The novelty of the paper is about examining the challenge and providing a systematic solution to the problem of the allocation of the charging stations. Although K-means and

DBSCAN are well-established clustering algorithms, they were not previously applied in the context of charging station placement in opportunistic networks to the best of our knowledge. We believe that the simulation study results are extensive and more realistic in nature as we used two real-world datasets.

The rest of the paper is organized as follows. Related work is given in Section 2. A mathematical representation of the model is explained in Section 3. Section 4 gives a detailed description of the application scenario. Our proposed approach is presented in Section 5. Section 6 explains the clustering approaches while the routing protocols are explained in Section 7. The simulation study along with the performance evaluation results are discussed in Section 8. Finally, our conclusions are given in Section 9.

2. Related Work

The work we present in this paper is related to three main ongoing research areas, UAV application, routing in opportunistic networks, and charging station research. Each related area is discussed in this section.

2.1. UAV Applications

UAVs have gained popularity due to their their wide spectrum of applications and services, both in military and civilian domains, with new UAV applications being addressed in literature every day.

Noor et al. [18] and Mohsan et al. [19] have presented extensive reviews on UAVs applications, the expected challenges, and open topics for future research. Raoult et al. [20] surveyed the use of UAVs in studying marine animal life, their behavior, movement patterns, and health. Amaro, Ángeles, and Juárez [21] provided a theoretical framework for single drone use in animal monitoring application scenarios. The animal monitoring scenario covers the case where there are static sensor locations and random animal movements in the environment with the UAV collecting the information. Arvanitaki and Pappas [13] studied the UAV-based data collection system from stable sensors with a discrete-time Markov chain approach. Similarly, Xu et al. [21] experimented on the scenario where a UAV uses a hidden Markov model to guess the next location for the cluster head while monitoring the animal movements. ASTRO+, an aerial-ground pollution monitoring platform, has been built by Boubrima and Knightly [22] to collect aerial pollution measurements. The paper showed that the dynamic airflow caused by the movements of the drones may affect the temperature and humidity levels of the ambient air, which, in turn, could affect the quality of the sensor's measurement. They also designed an algorithm that modifies the routes of the drones while taking into consideration the aerial measurement quality. Bartolini et al. [23] developed Druber, a delivery service system where the service fleet is comprised of drones that belong to multiple private owners. The system allows customers to request parcel deliveries where a drone picks up a parcel and delivers it via multiple coordinated drones that share resources with intermediate battery replacement spots. PENGUIN, a UAV-based solution for identifying and classifying aquatic plastic pollutants according to specifics of the material is proposed in [24]. It is a two-phase sensing process that first uses computer vision techniques to recognize underwater plastic wreckage. Then, optical sensing is used to classify the type of plastic for each recognized object to one of six Resin Identification Codes (RICs) which is a set of symbols written on plastic products that identify the plastic resin out of which the product is made. Another UAV application in package delivery within opportunistic network environment has been introduced by Bacanli et al. [25] using different machine language techniques. Ribeiro et al. [26] developed an optimization model for belt conveyor inspection used in the mining industry. The goal of this model was to define how many UAVs are suitable for monitoring the system based on mixed-integer linear programming. Ma et al. [27] proposed two opportunistic routing protocols to handle the scenario where a UAV is collecting data from mobile sensors moving with different velocities. Ref. [28] proposed a system in which clusters of small UAVs share their computing resources to help clusters with limited resources finish their

computational tasks in time. Oubbati et al. [29] introduced a routing scheme to investigate the incorporation of UAVs in vehicular ad hoc networks for rescue services using synthetic vehicle mobility data. In our case, the ground users, who represent the mobile nodes, carry the data collection devices. The mentioned work is just a few among many other UAV applications research [30–32] and while there is much existing research leveraging UAVs within opportunistic networks [33**?** –36], none have used a real-world dataset.

2.2. Routing in Opportunistic Networks

Many routing strategies have been developed within opportunistic networks. Some of the most well-known approaches are Epidemic routing [37], Spray and Wait [38], PROPHET [39], and State-based Campus Routing (SCR) [40]. Wang et al. [41] have proposed improvements to the PROPHET. Bacanli et al. [40] have worked on opportunistic network message flooding strategy in a campus environment and used the encounter dataset to create SCR (State-based Campus Routing) without any UAVs. Bacanli and Turgut [11] made a further extension to their work to integrate one or two standard military UAVs to cover the whole environment, the model did not include the use of charging stations. UAV path planning [13,21,42], the positioning of the UAVs [43], providing network coverage [44], and energy consumption of the UAVs [45] represent other areas of UAV research. Vargas et al. have explored the UAV scanning while routing them with grids using fixed and randomly moving mobile targets [46].

2.3. Energy Consumption and Charging Station

Da Silva et al. [47] considered a flying ad hoc network of drones that collects data from mobile sensors and sends the data to a base station. They presented a mixed-integer linear program to find the optimal trajectories for the drones to maintain this network through time while balancing between distance traveled by drones and their energy consumption. Hong et al. [48] presented a temporal routing algorithm to adjust the trajectory of UAVs used to collect data from sensors in a specific service area. The algorithm adapts the UAV's trajectory based on their current power state to preserve energy. Although the authors in [49] made an extensive study to investigate how to balance between maximizing the UAVs battery level and maximizing the battery level across installed in-ground sites to provide the energy required to recharge the UAVs batteries. Galkin and DaSilva [50] suggested three options for UAV battery charging. The first option is to use multiple UAVs to provide continuous coverage to a specified area with each UAV taking its turn to provide the coverage until its battery is drained and then is replaced by another UAV. The second option is to use a single UAV and a docking station with a mechanical actuator to switch the drained battery of a UAV with a new one. The final option utilizes a laser to wirelessly power the UAV. Tseng et al. [45] modeled battery consumption of 3DR Solo and DJI Matrice 100 drones. They tested with a flight plan with static locations for an application case where predefined locations received the payloads [51]. Gong et al. [42] tested a scenario where data is collected from a set of sensors aligned in a straight line to achieve the minimum flight time. This work gives a theoretical basis and formula proof for our application scenario. Michini et al. [52] has worked on a scenario where the battery of the drones is swapped in less than 30 s by the automated battery swapping robots. This work contributes to the feasibility of our application scenario. Won [53] proposed UBAT, which is a heuristic framework based on ant colony optimization in order to solve the charging station deployment problem for UAVs. Unlike our study, it was a traveling salesman problem with ant colony optimization. This work extends the earlier research by Bacanli et al. [54] and our approach differs from the approaches mentioned above by UAVs collecting data from mobile nodes and traveling to stable positions for charging/battery replacement.

3. System Model

We consider a system consisting of a finite set of both mobile nodes and UAVs. All nodes and UAVs can wirelessly communicate and exchange messages when they come within a certain proximity of each other or when they are within the radio range of each other without relying on any fixed infrastructure.

Due to the constantly moving nature of the nodes, the connection among them changes over a time span T. The model can be represented by the graph $G = (N, U, E)$, at a certain time in T, where N indicates the set of nodes, U the set of UAVs, and E the set of edges. An edge e_{ij} exists between i and j, $\forall i \in N|U$ and $\forall j \in N|U$ lies within the radio range for each other, and, hence, can share and exchange data over the provided wireless medium. The system model can, thus, be represented by a seven-tuple graph as follows

$$G = (N, U, E, T, \alpha, \beta, \gamma) \tag{1}$$

where,

$$\alpha : N \times T \to \{0, 1\} \tag{2}$$
$$\beta : U \times T \to \{0, 1\} \tag{3}$$
$$\gamma : E \times T \to \{0, 1\} \tag{4}$$

α in Equation (2) is the node presence function that indicates whether a given node exists at a given time, β in (3) is the function that indicates whether a given UAV exists at a given time, and similarly, γ in (5) is the edge presence function that indicates whether a given edge exists at a given time.

It should be noted that the graph G is not necessarily be connected at every t_i, where $0 < t_i < T$. Nodes could be disabled over the time span T either due to being switched off by the user exiting radio range.

At any given time t, where $0 > t > T$, when a message is created at a source node, an end-to-end path between the source and the destination nodes may not exist in the underlying graph G since nodes and/or drones may be on or off due to their proximity to each other at the current moment.

In our system model, a message sent from a source node u travels in a journey through a sequence of sub-graphs $(g_1, g_2, ..., g_n)$, where $g_i \subset G$ until it reaches a destination node v. The message is sent, carried, and forwarded in a manner determined by different routing schemes (see net section). When two nodes meet, they seize this opportunity to distribute the messages they are holding in order to increase the number of nodes carrying these messages depending on the routing scheme that was selected. Each node determines the messages it is interested in collecting and is willing to serve as their carrier from the current underlying subgraph $g_i \subset G$ to another subgraph $g_j \subset G$.

Given the previous assumptions, the representation of the graph G described in (1), and a set of charging stations $C = (C_1, C_2, ..., C_n)$, find the locations $L = (L_1, L_2, ..., L_n)$ of C that gives the best values of the network metrics being used.

4. Application Scenario

In our application scenario, there are three types of communicating pairs: UAV-to-UAV, node-to-node, and UAV-to-node. The pairs will communicate whenever they encounter each other. The system consists of people moving who carry electronic devices with WiFi connectivity. These people serve as the ground nodes for the opportunistic network with a set of UAVs that support the network in order to achieve high network performance in terms of low message delays and high success rate.

Nodes create text messages every 30 min with a 1-min error rate and distribute them in a hop-by-hop manner. These messages have a three-hour lifetime, starting from the initial creation of the message, after which the nodes stop receiving messages.

Both UAVs and nodes on the ground are using the IEEE 802.11 Wi-Fi protocol. The maximum connectivity distance is 250 m, which is appropriate for the IEEE 802.11 standard.

The UAVs, however, are only responsible for distributing the messages through the network. The charging station is a circle with a one-meter diameter. The UAV stays at the charging station for 30 s to charge its battery. As charging a battery will likely take longer than 30 s, we assume that instead of charging the battery, the drained battery will be replaced with a fully charged one in 30 s.

The UAVs perform random spiral scans on the environment at a 100 m altitude. We assume that the UAV's maximum speed is 20 m per second and 30 min of cruising time with a single charge. The UAV's specifications are appropriate for a commercial drone, for example a DJI Matrice 300 RTK [55]. The UAV scanning strategy is autonomous, so there is no need for the central station. How far UAVs the move away from a charging station is determined by whether their battery time is enough to reach another charging station or not. In this case, it is also possible that the charging stations may also include control stations.

The main application scenario of the system is for disaster cases where the Internet is either not available, or the connection is not stable. The proposed system can broadcast messages through the campus area with the UAVs considering a university landscape where the dataset is collected.

5. Proposed Approach

The simulation starts off with each charging station having at least one UAV ready to take off. The UAV departs the charging station towards a random spot on the map section under study. At the center of the chosen spot, the UAV starts a spiral scan with a random spiral radius. The spiral radius is the maximum distance between the center of the spiral and the end of the last turn (curve) of the spiral. If the maximum spiral radius has been reached or the distance between the UAV and the center of the spiral exceeded the mapped area, the UAV first stops at that point on the air and then proceeds to another random spot to perform another spiral scan.

At any given moment, if the UAV barely has enough battery to reach the closest charging station, the UAV cancels its current cruising trip and immediately hits the nearest charging station. The UAV stays 30 s at the charging station for either battery replacement or charging, after which takes off again for another spiral scanning trip.

The spiral used in our approach is the Archimedes spiral. Archimedes spiral is a special case of the Archimedean spirals, named after the Greek mathematician Archimedes. It is defined as the space made over time by a point moving outward from a fixed central point with a constant speed along a line that rotates with constant angular velocity. An Archimedes spiral can be described with polar coordinates by the equation,

$$r = a\theta \tag{5}$$

where r is the length of the radius from the center of the spiral, a is a constant that affects the distance between each turn of the spiral, changing a controls the distance between loops, and θ is the amount of angular rotation of the spiral. The distance between successive turns of the spiral is constant and equals $2\pi a$.

If θ is measured in radians, then the equivalent parametric equations for the Archimedes spiral are,

$$x(\theta) = a\theta \cos(\theta) + x_{initial} \tag{6}$$
$$y(\theta) = a\theta \sin(\theta) + y_{initial} \tag{7}$$

As UAVs perform spiral scans on the opportunistic network, the spiral pattern is selected as scanning a small area will be possible with a spiral by adjusting d (density) and R (maximum radius) parameters of the spiral which is shown in Figure 2. What is interesting about the Archimedes spiral is that the density of the spiral (d) stays constant unlike other spirals, such as the Fibonacci (Golden) spiral, where density(d) increases in every turn. The density of the spiral is the distance between two consecutive arcs, while

the maximum radius of the spiral is the distance between the center of the spiral and the end of the largest arc. For our application case, the density parameter (*d*) is set to 200. The maximum communication distance between nodes and UAVs is around 223 m. In that case, if the spiral density is 200 m, at least 2 encounters can be completed with a stable node on the ground. The maximum radius of the spiral is set to 400 m.
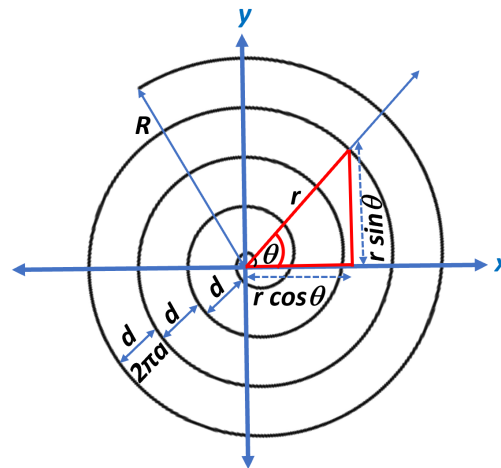

Figure 2. Archimedes spiral with density d and maximum radius as R.

UAVs do not request any location information or private information from either the nodes or the other UAVs. The packet exchange between nodes and UAVs is minimal so that an encounter time between nodes and UAVs can be sufficient enough to obtain an efficient packet exchange. The UAVs have a GPS or a location sensor as they travel around the locations determined dynamically. As a result, the UAVs can be added or removed from the environment for maintenance or different purposes.

The locations of the charging stations are determined according to three different techniques, namely, K-means clustering, density-based spatial clustering of applications with noise (DBSCAN), and at random. In addition to the aforementioned clustering techniques, three routing protocols are used in our study: Epidemic, Spray and Wait, and State-Based Campus Routing (SCR). The next two sections explain the clustering techniques and the routing protocols used.

The charging station locations are determined as follows. The locations of the mobile nodes on the ground in the dataset are given as input to clustering algorithms. The generated cluster centers are then used in the network simulation as charging station locations. For the K-means clustering approach, the number of clusters is also given as input to the K-means clustering algorithm. For our case, we have given 3 as the number of desired clusters. For the DBSCAN algorithm, *maximum distance between clusters* parameter is set to 800 m and *the minimum number of points in a cluster* is defined as 5 points. Those parameters have given output as three clusters for the DBSCAN algorithm. In that manner, we achieve three cluster centers from both clustering algorithms. In addition, 3 random points are created to compare with the clustering algorithms as we found that three clusters showed the best results for message delay and success rate metrics.

6. Clustering Approaches

We used three clustering approaches to find the best spots to place the charging stations, K-means, DBSCAN, and Random clustering. Each clustering approach is explained in this section. The three chosen clustering approaches represent parametric, non-parametric, and random techniques.

6.1. K-Means Clustering

The K-means algorithm is an iterative parametric algorithm that aims to divide a dataset into K distinct non-overlapping clusters where each data point of the dataset should

belong to only one cluster. The algorithm starts by having K randomly chosen centroids for each cluster. Each data point of the dataset is then assigned to the nearest cluster centroid. The centroids are calculated again based on the newly formed clusters, which, in return, are calculated by obtaining the sum of the squared distance between the data points and the cluster's centroid. The process is repeated until the new centroids are no longer changed. As the algorithm proceeds, homogeneous data points come closer to each other and they are contained within the same cluster. The way K-means works is depicted in Algorithm 1.

Algorithm 1 K-means clustering algorithm

Input: $X = \{x_1, x_2, ..., x_n\}$ A set of data Points
        k number of clusters
Output: $C = \{c_1, c_2, .., c_k\}$ Cluster centroids

for $j \leftarrow 1$ **to** k **do**
   Randomly initialize cluster centroids c_i
end
while *C of newly formed clusters do not change* **do**
     for $i \leftarrow 1$ **to** n **do**
         for $j \leftarrow 1$ **to** k **do**
             find closest centroid $c_j \in C$ to x_i;
             Assign x_i to cluster j;
         end
     end
     foreach *Centroid c_j* **do**
         Recalculate c_j with the mean of points in X assigned to cluster j;
     end
end

The locations of the UAV charging stations are determined using the K-means clustering on the dataset by taking the number of clusters to be created as a parameter and returns the cluster center locations of the created clusters. These centers act as the charging locations.

6.2. DBSCAN

The second approach used is DBSCAN, which is a non-parametric clustering algorithm. The idea of DBSCAN is to pack together points with many nearby neighbors and marking those points that lie alone in low-density regions as outliers. DBSCAN is depicted in Algorithm 2.

Algorithm 2 DBSCAN Clustering Algorithm

Input: $X = \{x_1, x_2, ..., x_n\}$ A set of data Points

ϵ distance

$minPts$ minimum number of nodes

Output: $C = \{c_1, c_2, .., c_k\}$ Cluster centroids

foreach *unvisited $x \in X$* **do**

Mark x as visited;

$N \leftarrow FindNeighbours(x, \epsilon)$

if $| N | < minPts$ **then**

Mark x as unvisited;

else

$C \leftarrow \{x\}$;

foreach *point $x' \in N$* **do**

$N \leftarrow N \setminus x'$;

if *x' is not visited* **then**

Mark x as unvisited;

$N' \leftarrow FindNeighbours(x', \epsilon)$;

if $| N' | \geq minPts$ **then**

$N \leftarrow N \cup N'$;

end

end

if *x' is not yet a member of any cluster* **then**

$C \leftarrow C \cup \{x'\}$;

end

end

end

end

In terms of our scenario, DBSCAN takes the maximum distance between clusters and the minimum number of nodes in a cluster as parameters. Based on these parameters, DBSCAN gives different numbers of cluster centers. The minimum number of nodes in a cluster is set to 3 for the DBSCAN cases.

6.3. Random

In this approach, the locations of the charging stations are chosen randomly on the map. We set out how many charging stations are needed, then the locations are placed randomly.

7. Routing Protocols

In this section, a brief description of the three routing protocols used in our framework is given, namely Epidemic, Spray and Wait routing, and SCR.

7.1. Epidemic Routing Protocol

In epidemic routing [37], each host maintains a summary vector (SV) that consists of a unique identifier for every message this host carries. The message either originated at this host or it has been buffered on behalf of other hosts. When two hosts come into communication range of one another, an anti-entropy session is initiated, during which the two hosts exchange their summary vectors to determine which messages the hosts carry, then each host decides which messages it needs by performing a logical AND operation between the negation of its summary vector and the summary vector of the other host. Following this, each host requests copies of those messages that it has not yet seen. Algorithm 3 depicts the Epidemic Routing protocol.

Algorithm 3 Epidemic routing protocol

Input:　Hosts u and v
　　　　boolean *antiEntropySession*
　　　　$SV(u)$ and $SV(v)$
Output: boolean *messagesExchange*

Drop messages with expired lift-time from u and v;
if *antiEntropySession* **then**
　Host u sends $SV(u)$ to v;
　$\overrightarrow{msgs} \leftarrow SV(u) + (\neg SV(v))$;
　Host v requests $\overrightarrow{msgs}$ from u;
　Host u sends $\overrightarrow{msgs}$ to v;
　foreach $m \in \overrightarrow{msgs}$ **do**
　　if *isDestination(v)* **then**
　　　consume(v, m);
　　else
　　　store(v, m);
　　end
　end
　messagesExchange $\leftarrow$ *true*
end

Vahdat and Becker [37] proposed a session strategy that is utilized in Epidemic, Spray and Wait, and SCR. Figure 3 shows an example session between nodes A and B. Considering that node A and B have encountered each other, node A will send its ID, concatenated with the message IDs that node A has in its buffer. In this example case, node A has messages with IDs 5, 14, 28, and 89. Once node B receives this message, node B replies with a message containing the message IDs that node B does not have. Node A sends the requested messages, with IDs 28 and 89, to Node B. Once the message exchange finishes, the same session strategy is repeated for node B and node A. At the end of the next session, the messages between A and B will be synchronized.
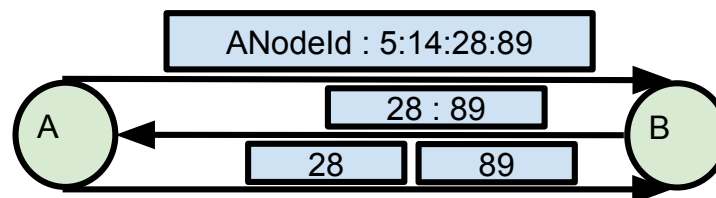

Figure 3. Session example in epidemic routing.

7.2. Spray and Wait Routing Protocol

Spray and Wait (SW) [38] aims to decrease the number of transmissions and limit the number of message copies spans over the network. The algorithm consists of the spray and the wait phases. In the spray phase, each message is associated with a limit L of the number of copies the next node that is encountered is allowed to distribute. This number is decreased by the number of transfers for this packet at each node. When the number of copies reaches one, the node carrying the message enters the wait phase, as it stops distributing any more copies of the packet and waits until it either meets the destination or the lifetime of the messages expires. If the destination is not found in the spraying phase, each of the nodes carrying a copy of the message waits to perform a direct transmission which means it will forward the message only to its destination.

A binary version of SW is also introduced in [38]. In the binary SW, a message initially originating at the source has L copies. Any node that has more than one copy of the message and encounters another node with no copies sends half of its copies to the encountered

node and keeps the other half for itself. When only one copy remains, it switches to direct transmission. The pseudo-code for Spray and Wait is shown in Algorithm 4.

Algorithm 4 Spray and Wait routing protocol

Input: Hosts u and v
         boolean *encounterSession*
         boolean *messagesSpray*

Drop messages with expired lift-time from u and v;
Spray(m, L)
for $i \leftarrow 1$ **to** L **do**
 | **if** *encounterSession*(u, v_i) **then**
 | | Host u sends m to v_i;
 | |
 | **end**
end
return;
Wait(m)
foreach *node u* **do**
 | **foreach** *encounterNode v* **do**
 | | **if** *isDestination*(v) **then**
 | | | forward m to v;
 | | **else**
 | | | Keep m;
 | | **end**
 | **end**
end
return;

7.3. State-Based Campus Routing (SCR) Protocol

In SCR [40], nodes can serve either as a sender or a receiver. When two nodes come within a certain proximity to each other, they open a session. First, the sender node sends the message vector to the receiver node while keeping in mind that each message has an expiration date. The receiver node replies by sending another message vector that holds the message IDs that the receiver node does not have. The sender then sends the required messages whose IDs are listed in the previous transmission. Following this, the two nodes switch sender and receiver roles and open a new session.

In this routing method, each node can be either in an idle or active state. A node is considered to be in an idle state if the node encounters many nodes within a short period of time compared to its past encounter frequencies, at which point we presume that the node is in a standby area, such as a waiting place or a hot spot. If the node is in an active state, it means that the node is either walking or at someplace where encounter frequency is lower. The pseudo-code for an SCR session is shown in Algorithm 5.

Algorithm 5 State-based Campus routing (SCR) protocol

Input: Hosts u and v
           boolean *Session*
           $SV(u)$ and $SV(v)$
Output: boolean *messagesExchange*

Drop messages with expired lift-time from u and v;
if *Session* **then**
    Host u sends $SV(u)$ to v;
    $\overrightarrow{msgs} \leftarrow SV(u) + (\neg SV(v))$;
    Host v requests $\overrightarrow{msgs}$ from u;
    Host u sends $\overrightarrow{msgs}$ to v;
    foreach $m \in \overrightarrow{msgs}$ **do**
       if *isDestination*(v) **then**
          consume(v, m);
       else
          store(v, m);
       end
    end
    Host v sends $SV(v)$ to u;
    $\overrightarrow{msgs} \leftarrow SV(v) + (\neg SV(u))$;
    Host u requests $\overrightarrow{msgs}$ from v;
    Host v sends $\overrightarrow{msgs}$ to u;
    foreach $m \in \overrightarrow{msgs}$ **do**
       if *isDestination*(u) **then**
          consume(u, m);
       else
          store(u, m);
       end
    end
    messagesExchange $\leftarrow$ *true*
end

---

8. Simulation Study

In this section, we explain our simulation environment, the datasets, and the metrics used to evaluate the simulation scenario.

8.1. Simulation Environment

The simulation study for the application scenario was carried out using Java. The in-house simulator by Bacanli and Turgut [11] is further developed for this application. The opportunistic wireless environment includes a 10% error rate for wireless communication.

We assume that the encounter duration between the nodes is long enough for packet exchange. Based on the nature of the opportunistic network, if the receiver does not receive a protocol message, the sender does not send the packet again. The reason being the message may be received from another node again when encountering takes place.

8.2. Dataset

Two datasets have been used in our simulation (Table 1), North Carolina State University (NCSU) and the Korea Advanced Institute of Science and Technology (KAIST) dataset by Rhee et al. [56]. The datasets contain the traces of the nodes every 10 s. The locations in the datasets are relative to an origin location. We have used the data points of all the nodes on the map at 300 s time intervals while applying the clustering techniques. Below is the description of each dataset:

NCSU Dataset: NCSU dataset contains 35 movement traces of students in the NSCU campus for 78,090 s which equates to more than 21 h. The width of the simulation environment is 14,629 m by 9714 m. We have removed the third movement trace. The removed movement trace goes to the corner of the map where no node movement activity takes place after around.

KAIST Dataset: KAIST dataset contains 22 movement traces of students in KAIST campus for 83,970 s which equates to more than 23 h. The width of the simulation environment is 13,226 m by 19,208 m. We have removed the traces that are shorter than 2 h in the original KAIST dataset, leaving 22 movement traces.

Table 1. Dataset properties.

Dataset Property	NCSU	KAIST
Number of nodes	35	22
Length of the map (in meters)	14,628	19,208
Width of the map (in meters)	9713	13,227
Data collection duration (in seconds)	78,090	83,970
Data collection frequency (in seconds)	300	300

8.3. Metrics and Evaluation

Nine combinations are examined, DBSCAN with Epidemic (D_EP), DBSCAN with SCR (D_SCR), DBSCAN with Spray and Wait (D_SW), K-Means with Epidemic (K_EP), K-Mean with SCR (K_SCR), K-Mean with Spray and Wait (K_SW), Random with Epidemic (R_EP), Random with SCR (R_SCR), and Random with Spray and Wait (R_SW).

In our simulation, the number of clusters for K-means and Random clustering has been set to 3, and the maximum distance between clusters for DBSCAN has been set to 900 which generates 3 cluster centers as well. The number 3 has been chosen after comparing the results of several cluster center numbers as in (2, 3, 4, and 5), and the comparison resulted in favor of 3 clusters.

It is important to note here that certain parameters should be analyzed collectively to evaluate a given network. Therefore, we have used four metrics to quantitatively and qualitatively assess the effect of the aforementioned nine techniques on network performance. The metrics used are message delay, success rate, number of tours, and number of messages sent.

8.3.1. Message Delay

Message delay is one of the main metrics used to measure network performance, as it focuses on the time spent in the successful transfer of data packets from one point to another.

Message delay is defined as the time elapsed from creation until receipt of the message. Message delay is calculated for every message created during the simulation. Figures 4 and 5 shows the cumulative distribution of message delay in the NCSU and KAIST datasets, respectively. To better visualize the message delay, we use box plots to present the data.
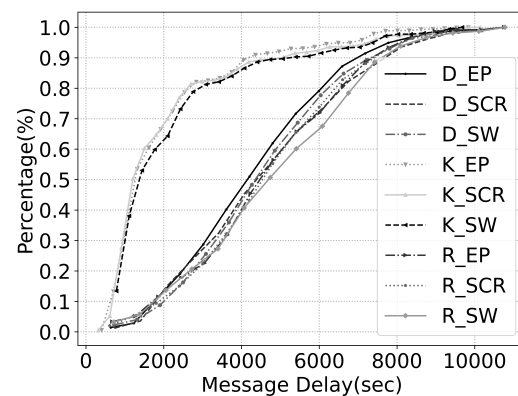
Figure 4. Cumulative distribution function of Message Delay on NCSU Dataset.
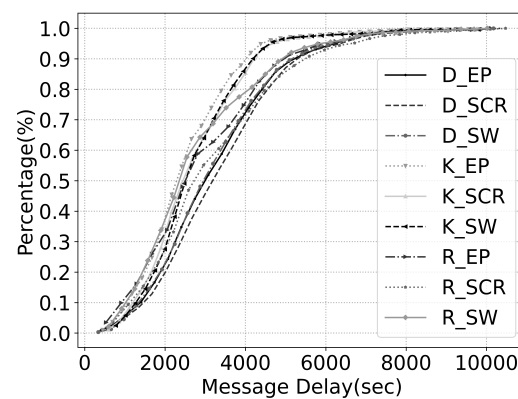

Figure 5. Cumulative distribution function of Message Delay on KAIST Dataset.

Figures 6 and 7 shows the box plots of message delay distribution for the NCSU and KAIST datasets, respectively. In each box-and-whisker plot, the top and the bottom of the box are given by the 75th and 25th percentile, while the median is represented by the mark inside the box. The upper and lower whiskers are the maximum and minimum, respectively, after excluding the outliers. The outliers above the upper whiskers are those exceeding 1.5 of the upper quartile, and those below the minimum are those less than 1.5 of the lower quartile.
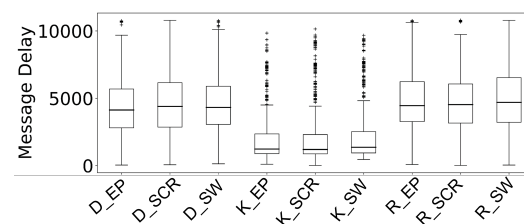
Figure 6. Box plot distribution results of Message Delay on NCSU Dataset.
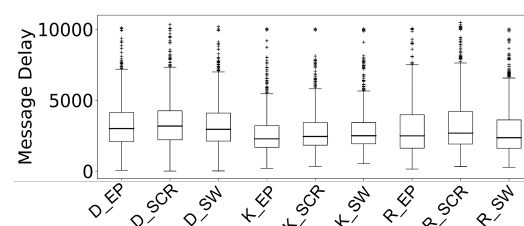
Figure 7. Box plot distribution results of Message Delay on KAIST Dataset.

Regardless of the routing protocol used, the figures show that K-Means clustering (K_EP, K_SCR, K_SW) performs noticeably better than Random (R_EP, R_SCR, R_SW) and DBSCAN (D_EP, D_SCR, D_SW) clustering. When focusing on routing protocols, we see that Epidemic and SCR have very close results as both graph lines almost coincide with each other, yielding better results than Spray and Wait routing.

The best message delay readings can be found when K-Means clustering is combined with Epidemic routing. It is also clear that the behavior of every clustering approach is enhanced when Epidemic or SCR Routing protocols are chosen.

8.3.2. Success Rate

The success rate is defined as the percentage of the nodes that received a specific message. In other words, it is the rate of successful message delivery. The higher the success rate for a message, the higher the percentage would be. The success rate is the distribution rate of the message. If a message stays in the buffer of the node, then the success rate of the message is 0%. On the other hand, if the message is distributed to every node, the success rate is 100% [57].

Figures 8 and 9 shows the complementary cumulative distribution of success rate in the NCSU and KAIST datasets, respectively. Figures 10 and 11 shows the box plots of success rate distribution for three generated cluster centers for the KAIST and NCSU datasets, respectively. Again, the K-Means clustering with Epidemic and SCR schemes appear to achieve the highest success rates among their counterparts. It can be seen that both datasets show consistent behaviors.

It is worth noting that message delay and success rate are not dependent, meaning that a scheme that produces low message delay values does not necessarily guarantee a high success rate. For example, if a message is transmitted to only one node during its lifetime, it will have a low message delay and a low success rate as well.
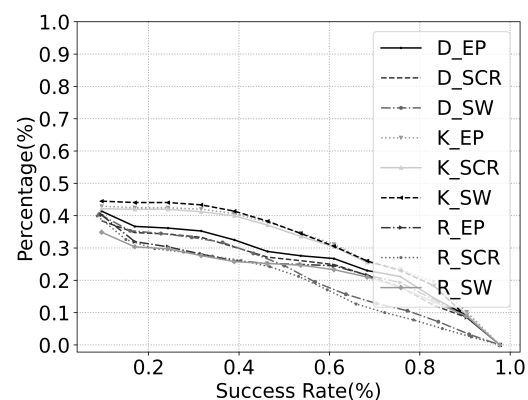

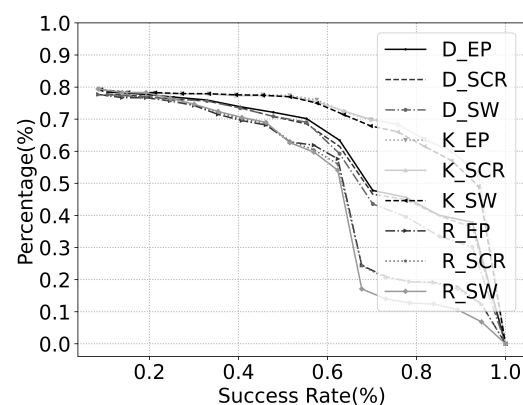**Figure 8.** Distribution results of Success Rate on NCSU Dataset.

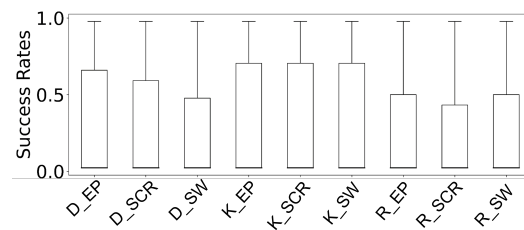**Figure 9.** Distribution results of Success Rate on KAIST Dataset.

Figure 10. Box plot distribution results of Success Rate on NCSU Dataset.
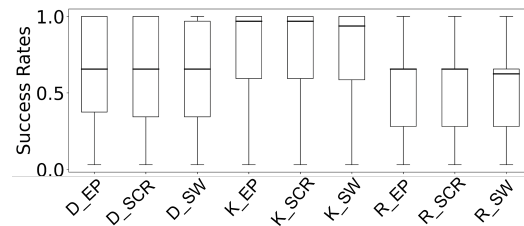

Figure 11. Box plot distribution results of Success Rate on KAIST Dataset.

8.3.3. Number of UAV Tours

Figures 12 and 13 shows the number of tours made by the UAVs when each of the nine schemes is used in the NCSU and KAIST datasets, respectively. A tour begins when a UAV takes off from the charging station until it returns and lands at the charging station again. Meaning that every time a UAV takes off it starts a new tour.

The figures show that the minimum number of UAV tours is obtained when K-Means clustering is used, the three routing protocols (Epidemic, SCR, and Spray and Wait) yield similar performance. Our simulation shows that the total distances traveled by the nine schemes are similar which is around (16,266 km). This means that a UAV travels the same distance no matter which duet of clustering approach and routing algorithm is used, however, it made a fewer number of tours when K-Means is in effect. The lower number of tours means there were fewer numbers of landings and take-offs, which concludes that K-Means Clustering is the most energy-efficient approach as UAV land whenever it needs charging.
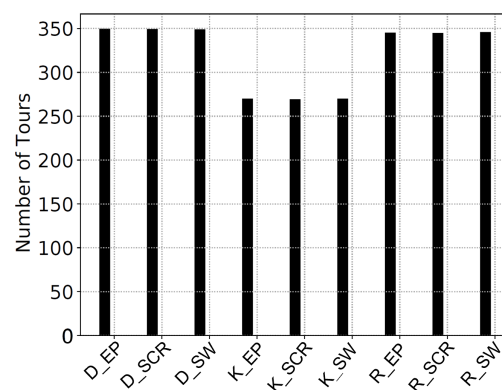

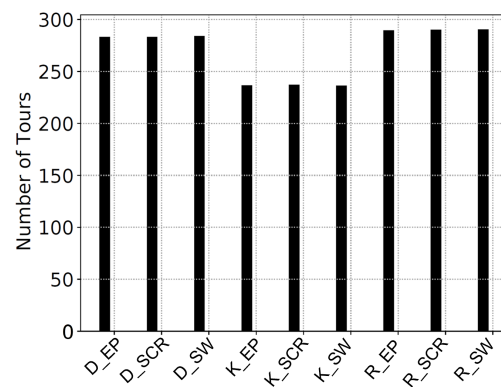
Figure 12. Number of UAV Tours in NCSU Dataset.

Figure 13. Number of UAV Tours in KAIST Dataset.

8.3.4. Number of Messages Sent

Figures 14 and 15 shows the number of messages sent by the nine schemes in the NCSU and KAIST datasets, respectively.

It is interesting to note that when Epidemic routing is used with any clustering algorithm, it gives the highest total number of messages sent, which is an indication that it is a less energy-efficient routing protocol. This observation conceptually matches the fact that Epidemic routing consumes network resources at a high rate.

From the metrics introduced in this section, it can be inferred that SCR manages to balance the load and energy more efficiently than the other protocols.
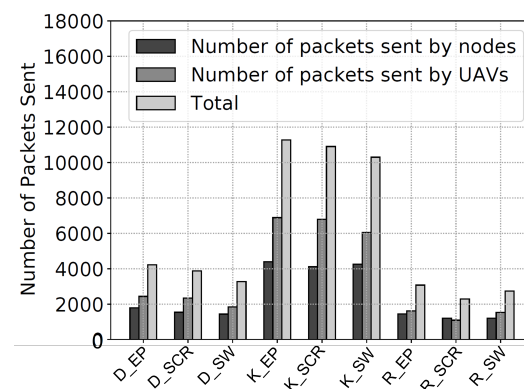

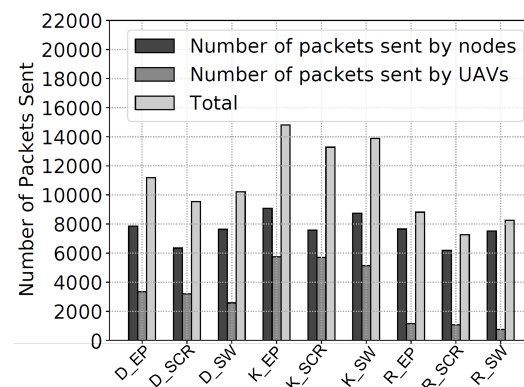**Figure 14.** Number of UAV Tours in NCSU Dataset.

Figure 15. Number of UAV Tours in KAIST Dataset.

8.3.5. Number of Messages Added to Buffer

When an encounter session is initiated, the nodes and UAVs exchange protocol messages, as well as actual messages. The more actual messages exchanged between nodes, the more efficient the communication becomes in an opportunistic network. If the number of messages added to buffers is small, it can be interpreted as the messages are not being distributed efficiently. For both datasets, K-Means shows a higher number of messages added to buffer comparing to DBSCAN and randomly places charging stations. Figures 16 and 17 shows the number of messages added to buffer.
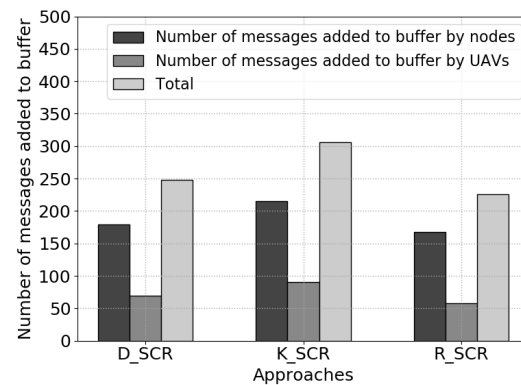

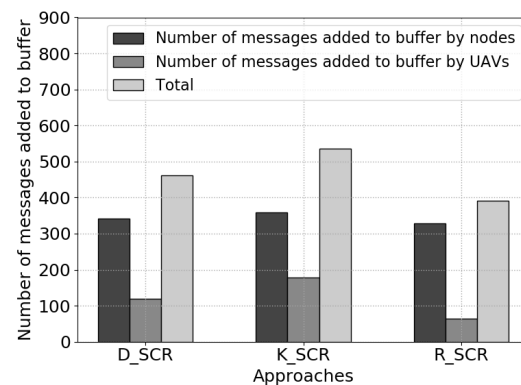**Figure 16.** Number of Messages Added to Buffer in NCSU Dataset.

Figure 17. Number of Messages Added to Buffer in KAIST Dataset.

9. Conclusions

UAVs are becoming more widely used in many military and civilian applications. On the other hand, the limited battery life of UAVs creates challenges that need practical solutions. In this paper, we proposed a framework to solve the problem of charging station placement in a UAV-aided opportunistic network. The framework consists of three main parts which together form the proposed solution: (i) the way UAVs scan the area of interest; (ii) the use of three routing protocols to forward messages during UAV-to-UAV, UAV-to-Node, and Node-to-Node communication; and (iii) the use of clustering approaches to find the best spot to place the charging stations.

Different network performance metrics for all combinations between routing protocols and clustering approaches have been closely investigated to give the best understanding and assessment for the proposed framework.

The simulation study shows that K-Means clustering gives better results when compared to the DBSCAN and Random clustering approaches. Epidemic routing gives the best message delay and success rate readings among the SCR and the Spray and Wait routing protocols; however, it overloads network resources which results in high energy consumption rates. SCR shows almost the same message delay and success rates as Epidemic routing with the best energy consumption performance that makes it competitive

with the Epidemic routing. We conclude that SCR performs the best when it comes to the overall network performance as it succeeded in maintaining a good balance among all the metrics used.

References

1. Liu, B.; Zhang, W.; Chen, W.; Huang, H.; Guo, S. Online Computation Offloading and Traffic Routing for UAV Swarms in Edge-Cloud Computing. *IEEE Trans. Veh. Technol.* **2020**, *69*, 8777–8791. [CrossRef]
2. Elloumi, M.; Dhaou, R.; Escrig, B.; Idoudi, H.; Saidane, L.A. Monitoring road traffic with a UAV-based system. In Proceedings of the 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, Spain, 15–18 April 2018; pp. 1–6. [CrossRef]
3. Shakoor, S.; Kaleem, Z.; Baig, M.I.; Chughtai, O.; Duong, T.; Nguyen, L.D. Role of UAVs in Public Safety Communications: Energy Efficiency Perspective. *IEEE Access* **2019**, *7*, 140665–140679. [CrossRef]
4. Oubbati, O.S.; Lakas, A.; Lorenz, P.; Atiquzzaman, M.; Jamalipour, A. Leveraging Communicating UAVs for Emergency Vehicle Guidance in Urban Areas. *IEEE Trans. Emerg. Top. Comput.* **2019**, *9*, 1070–1082. [CrossRef]
5. Mozaffari, M.; Saad, W.; Bennis, M.; Nam, Y.H.; Debbah, M. A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2334–2360. [CrossRef]
6. Sharma, A.; Vanjani, P.; Paliwal, N.; Basnayaka, C.M.; Jayakody, D.; Wang, H.; Muthuchidambaranathane, P. Communication and networking technologies for UAVs: A survey. *J. Netw. Comput. Appl.* **2020**, *168*, 102739. [CrossRef]
7. Fotouhi, A.; Qiang, H.; Ding, M.; Hassan, M.; Giordano, L.G.; Garcia-Rodriguez, A.; Yuan, J. Survey on UAV Cellular Communications: Practical Aspects, Standardization Advancements, Regulation, and Security Challenges. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 3417–3442. [CrossRef]
8. Liu, Y.; Dai, H.N.; Wang, Q.; Shukla, M.K.; Imran, M. Unmanned aerial vehicle for internet of everything: Opportunities and challenges. *Comput. Commun.* **2020**, *155*, 66–83. [CrossRef]
9. The Federal Aviation Administration—United States Department of Transportation. *Press Release—U.S. Department of Transportation Issues Two Much-Anticipated Drone Rules to Advance Safety and Innovation in the United States*; United States Department of Transportation: Washington, DC, USA, 2020.
10. Amazon.com Incorporation. *Amazon Prime Air*; Amazon.com Incorporation: Bellevue, WA, USA, 2020.
11. Bacanli, S.S.; Turgut, D. Unmanned Aerial Vehicles in Opportunistic Networks. In Proceedings of the IEEE GLOBECOM'19, Waikoloa, HI, USA, 9–13 December 2019.
12. Xu, J.; Solmaz, G.; Rahmatizadeh, R.; Turgut, D.; Bölöni, L. Animal Monitoring with Unmanned Aerial Vehicle-Aided Wireless Sensor Networks. In Proceedings of the IEEE LCN'15, Clearwater Beach, FL, USA, 26–29 October 2015; pp. 125–132.
13. Arvanitaki, A.; Pappas, N. Modeling of a UAV-based data collection system. In Proceedings of the IEEE CAMAD'17, Lund, Sweden, 19–21 June 2017; pp. 1–6.
14. Radwan, A.; Girgis, M.; Mahmoud, T.; Elgeldawi, E. Improving the Efficiency of the Flow Deviation Method for Solving the Optimal Routing Problem in a Packet-Switched Computer Network. *Int. J. Appl. Math.* **2001**, *5*, 171–187.
15. Johnson, D.B. Routing in Ad Hoc Networks of Mobile Hosts. In Proceedings of the 1994 First Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, USA, 8–9 December 1994; pp. 158–163.
16. Radwan, A.; Elgeldawi, E. Solving the Optimal Routing Problem in a Packet-Switched Computer Network Using Decomposition. *Egypt. Int. J.* **2003**, *4*, 1–13.
17. Ennaji, R.; Boulmalf, M. Routing in wireless sensor networks. In Proceedings of the 2009 International Conference on Multimedia Computing and Systems, Ouarzazate, Morocco, 2–4 April 2009, pp. 495–500. [CrossRef]
18. Noor, F.; Khan, M.A.; Al-Zahrani, A.; Ullah, I.; Al-Dhlan, K.A. A Review on Communications Perspective of Flying Ad-Hoc Networks: Key Enabling Wireless Technologies, Applications, Challenges and Open Research Topics. *Drones* **2020**, *4*, 65. [CrossRef]
19. Mohsan, S.A.H.; Khan, M.A.; Noor, F.; Ullah, I.; Alsharif, M.H. Towards the Unmanned Aerial Vehicles (UAVs): A Comprehensive Review. *Drones* **2022**, *6*, 147. [CrossRef]

20. Raoult, V.; Colefax, A.P.; Allan, B.M.; Cagnazzi, D.; Castelblanco-Martínez, N.; Ierodiaconou, D.; Johnston, D.W.; Landeo-Yauri, S.; Lyons, M.; Pirotta, V.; et al. Operational Protocols for the Use of Drones in Marine Animal Research. *Drones* **2020**, *4*, 64. [CrossRef]
21. Vera-Amaro, R.; Rivero-Ángeles, M.E.; Luviano-Juárez, A. Data Collection Schemes for Animal Monitoring Using WSNs-Assisted by UAVs: WSNs-Oriented or UAV-Oriented. *Sensors* **2020**, *20*, 262. [CrossRef]
22. Boubrima, A.; Knightly, E.W. Robust Mission Planning of UAV Networks for Environmental Sensing. In Proceedings of the 6th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications (DroNet-2020), Toronto, ON, Canada, 15 June 2020.
23. Bartolini, N.; Coletta, A.; Maselli, G.; Piva, M. DRUBER: A Trustable Decentralized Drone-Based Delivery System. In Proceedings of the 6th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications, Toronto, ON, Canada, 15 June 2020; Association for Computing Machinery: New York, NY, USA, 2020; DroNet'20. [CrossRef]
24. Flores, H.; Zuniga, A.; Motlagh, N.H.; Liyanage, M.; Passananti, M.; Tarkoma, S.; Youssef, M.; Nurmi, P. PENGUIN: Aquatic Plastic Pollution Sensing Using AUVs. In Proceedings of the 6th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications, Toronto, ON, Canada, 15 June 2020; Association for Computing Machinery: New York, NY, USA, 2020; DroNet'20. [CrossRef]
25. Bacanli, S.; Cimen, F.; Elgeldawi, E.; Turgut, D. Placement of Package Delivery Center for UAVs with Machine Learning. In Proceedings of the IEEE GLOBECOM 2021, Madrid, Spain, 7–11 December 2021.
26. Ribeiro, R.G.; Júnior, J.R.C.; Cota, L.P.; Euzébio, T.A.M.; Guimarães, F.G. Unmanned Aerial Vehicle Location Routing Problem With Charging Stations for Belt Conveyor Inspection System in the Mining Industry. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 4186–4195. [CrossRef]
27. Ma, X.; Chisiu, S.; Kacimi, R.; Dhaou, R. Opportunistic communications in WSN using UAV. In Proceedings of the IEEE CCNC'17, Las Vegas, NV, USA, 8–11 January 2017; pp. 510–515.
28. Valentino, R.; Jung, W.S.; Ko, Y.B. Opportunistic Computational Offloading System for Clusters of Drones. In Proceedings of the International Conference on Advanced Communication Technology, Chuncheon-si, Korea, 11–14 February 2018; pp. 303–306.
29. Sami Oubbati, O.; Atiquzzaman, M.; Ahamed Ahanger, T.; Ibrahim, A. Softwarization of UAV Networks: A Survey of Applications and Future Trends. *IEEE Access* **2020**, *8*, 98073–98125. [CrossRef]
30. Turgut, E.; Cenk Gursoy, M.; Guvenc, I. Energy Harvesting in Unmanned Aerial Vehicle Networks With 3D Antenna Radiation Patterns. *IEEE Trans. Green Commun. Netw.* **2020**, *4*, 1149–1164. [CrossRef]
31. Kim, J.; Kim, S.; Ju, C.; Son, H.I. Unmanned Aerial Vehicles in Agriculture: A Review of Perspective of Platform, Control, and Applications. *IEEE Access* **2019**, *7*, 105100–105115. [CrossRef]
32. Shakhatreh, H.; Sawalmeh, A.H.; Al-Fuqaha, A.; Dou, Z.; Almaita, E.; Khalil, I.; Othman, N.S.; Khreishah, A.; Guizani, M. Unmanned Aerial Vehicles (UAVs): A Survey on Civil Applications and Key Research Challenges. *IEEE Access* **2019**, *7*, 48572–48634. [CrossRef]
33. Ma, X.; Kacimi, R.; Dhaou, R. Fairness-aware UAV-assisted data collection in mobile wireless sensor networks. In Proceedings of the IWCMC, Paphos, Cyprus, 5–9 September 2016; pp. 995–1001.
34. Xu, C.; Zhang, X. A routing algorithm for schismatic communication network based on UAV. In Proceedings of the IEEE ICEIEC, Beijing, China, 15–17 November 2013; pp. 49–52.
35. Liu, D.; Xu, Y.; Wang, J.; Chen, J.; Yao, K.; Wu, Q.; Anpalagan, A. Opportunistic UAV Utilization in Wireless Networks: Motivations, Applications, and Challenges. *IEEE Commun. Mag.* **2020**, *58*, 62–68. [CrossRef]
36. Alzahrani, B.; Oubbati, O.S.; Barnawi, A.; Atiquzzaman, M.; Al-Ghazzawi, D.M. UAV assistance paradigm: State-of-the-art in applications and challenges. *J. Netw. Comput. Appl.* **2020**, *166*, 102706. [CrossRef]
37. Vahdet, A.; Becker, D. *Epidemic Routing for Partially Connected Ad Hoc Networks*; Technical Report, CS-200006; Duke University: Durham, NC, USA, 2000.
38. Spyropoulos, T.; Psounis, K.; Raghavendra, C.S. Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks. In Proceedings of the ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN), Philadelphia, PA, USA, 26 August 2005; pp. 252–259.
39. Lindgren, A.; Doria, A.; Schelén, O. Probabilistic Routing in Intermittently Connected Networks. *SIGMOBILE Mob. Comput. Commun. Rev.* **2003**, *7*, 19–20. [CrossRef]
40. Bacanli, S.S.; Solmaz, G.; Turgut, D. Opportunistic Message Broadcasting in Campus Environments. In Proceedings of the IEEE GLOBECOM'15, San Diego, CA, USA, 6–10 December 2015.
41. Wang, G.; Tao, J.; Zhang, H.; Pan, D. A improved Prophet routing based on congestion level of nodes in DTN. In Proceedings of the IEEE IAEAC, Chongqing, China, 25–26 March 2017; pp. 1666–1669.
42. Gong, J.; Chang, T.; Shen, C.; Chen, X. Flight Time Minimization of UAV for Data Collection Over Wireless Sensor Networks. *IEEE J. Sel. Areas Commun. (JSAC)* **2018**, *36*, 1942–1954. [CrossRef]
43. Akbas, M.; Solmaz, G.; Turgut, D. Molecular Geometry Inspired Positioning for Aerial Ad Hoc Networks. *Comput. Netw.* **2016**, 72–88. [CrossRef]
44. Brust, M.; Akbas, M.; Turgut, D. VBCA: A Virtual Forces Clustering Algorithm for Autonomous Aerial Drone Systems. In Proceedings of the IEEE International Systems Conference (SysCon'16), Orlando, FL, USA, 18–21 April 2016.
45. Tseng, C.; Chau, C.; Elbassioni, K.M.; Khonji, M. Flight Tour Planning with Recharging Optimization for Battery-operated Autonomous Drones. *CoRR* **2017** [CrossRef]

46. Vargas, P.; Kappel, K.; Marins, J.; Cabreira, T.; Ferreira Jr, P. Patrolling Strategy for Multiple UAVs with Recharging Stations in Unknown Environments. In Proceedings of the 2019 Latin American Robotics Symposium (LARS), Rio Grande, Brazil, 22–25 October 2019; pp. 346–351. [CrossRef]
47. Dias Da Silva, I.; Caillouet, C. Optimizing the trajectory of drones: Trade-off between distance and energy. In Proceedings of the IAUV 2020—2nd International Workshop on Internet of Autonomous Unmanned Vehicles, Como, Italy, 22–26 June 2020.
48. Hong, Y.W.P.; Cheng, R.H.; Hsiao, Y.C.; Sheu, J.P. Power-Efficient Trajectory Adjustment and Temporal Routing for Multi-UAV Networks. *IEEE Trans. Green Commun. Netw.* **2020**, *4*, 1106–1119. [CrossRef]
49. Amorosi, L.; Chiaraviglio, L.; Galán-Jiménez, J. Optimal Energy Management of UAV-Based Cellular Networks Powered by Solar Panels and Batteries: Formulation and Solutions. *IEEE Access* **2019**, *7*, 53698–53717.
50. Galkin, B.; Kibilda, J.; DaSilva, L.A. UAVs as Mobile Infrastructure: Addressing Battery Lifetime. *IEEE Commun. Mag.* **2019**, *57*, 132–137. [CrossRef]
51. Yu, K.; Budhiraja, A.K.; Tokekar, P. Algorithms for Routing of Unmanned Aerial Vehicles with Mobile Recharging Stations. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 5720–5725. [CrossRef]
52. Michini, B.; Toksoz, T.; Redding, J.; Michini, M.; How, J.; Vavrina, M.; Vian, J. Automated Battery Swap and Recharge to Enable Persistent UAV Missions. In Proceedings of the Infotech@Aerospace 2011, St. Louis, MO, USA, 29–31 March 2011. [CrossRef]
53. Won, M. UBAT: On Jointly Optimizing UAV Trajectories and Placement of Battery Swap Stations. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020.
54. Bacanli, S.S.; Elgeldawi, E.; Turgut, D. Charging Station Placement in Unmanned Aerial Vehicle Aided Opportunistic Networks. In Proceedings of the ICC 2021—IEEE International Conference on Communications, Virtual, 14–23 June 2021; pp. 1–5. [CrossRef]
55. DJI Matrice 300 RTK. DJI Matrice 300 RTK **2020**.
56. Rhee, I.; Shin, M.; Hong, S.; Lee, K.; Kim, S.J.; Chong, S. On the Levy-Walk Nature of Human Mobility. *IEEE/ACM Trans. Netw.* **2011**, *19*, 630–643. [CrossRef]
57. Bacanli, S.S.; Turgut, D. Energy-efficient unmanned aerial vehicle scanning approach with node clustering in opportunistic networks. *Comput. Commun.* **2020**, *161*, 76–85. [CrossRef]