An Optimized UAV Trajectory Planning for Localization in Disaster Scenarios

Freddy Demiane, Sanaa Sharafeddine,defenses and and Omar Farhat

***Abstract*—Unmanned aerial vehicles (UAVs) are considered one of the most promising emerging technologies to support rescue teams in disaster management and relief operations according to UN and Red Cross reports. In this work, we consider a disaster scene with damaged communication infrastructure and leverage UAVs for efficient and accurate positioning of potential survivors through the seamless collection of the received signal strength indicators (RSSI) of their mobile devices. We assume the scene is divided into multiple regions or cells with varying levels of importance based on the damage degree or the population density for example, and, thus, requiring different localization effort to improve the achieved accuracy. We formulate and solve two complementary subproblems. The first subproblem identifies a minimal number of strategic positions, referred to as waypoints or scanning points, at which the UAV hovers to collect the required number of RSSI signals from all devices within each cell in the disaster scene. Cells assigned higher importance levels call for higher number of RSSI readings from their devices. The waypoints generated from the first subproblem are then input to the second subproblem that constructs an efficient UAV trajectory that traverses all waypoints. By the end of the UAV mission, the collected RSSI measurements are processed to localize the discovered devices while taking into account the wireless channel statistical variability. Simulation results are generated and analyzed to demonstrate the accuracy and effectiveness of the proposed solution approach in localizing an unknown number of mobile devices in disaster scenes with regions of varying importance levels. In addition, an experimental testbed is designed and implemented as a proof of concept to validate the practicality of implementing the proposed localization solution in a realistic setting.**

***Index Terms*—Unmanned Aerial Vehicle (UAV), trajectory planning, localization, testbed implementation.**

I. Introduction

Timely localization of victims in natural or man-made disasters remains crucial for successful search and rescue missions. The safety of first responders is considered top priority in those missions and, thus, accurate localization is extremely critical to reduce risks while navigating damaged structures. Vast range of tools have been developed to support emergency responders and incident commanders, each customized to specific requirements and features [1]. Major effort has been invested in designing localization systems to overcome the limitations of the Global Positioning System (GPS) in indoor environments. Most of these systems are based on fingerprinting techniques that do not require line of sight assumption. Being widely prevalent, the fingerprints of existing WiFi-enabled devices can be readily collected without incurring any additional infrastructural cost. The work in [2] argues that valuable information can be extracted from passive monitoring of WiFi traffic and demonstrated three possible use cases with high accuracy: user localization, user profiling, and device classification. Other technologies have also been investigated in the literature including Bluetooth [3], [4], radio-frequency identification (RFID) [5], [6], acoustic signals [7], light [8], [9], and others [10]. For the case of outdoor localization, GPS systems are accurate and reliable for vast range of applications especially that almost all mobile devices are equipped with GPS receivers. Whether in indoor or outdoor environments, localization becomes a challenging task during emergency situations. Connectivity with users is lost due to potential damage of the communication infrastructure and/or user unconsciousness; in addition, physical structures can be extremely altered rendering any existing fingerprints non-reliable.

Unmanned aerial vehicles (UAVs) have emerged as a promising technology in crisis support ranging from delivering aid [11], to humanitarian data collection, providing intermittent communication [12], [13], [14], and assisting in search-and-rescue missions [15], [16]. In this work, we consider a disaster scene with unknown number of victims who might be lost or trapped under debris. We assume the scene is divided into multiple regions with varying levels of importance. Our objective is to launch a UAV to scan the area and accurately localize potential survivors, in the least possible time, via measuring the received wireless signal strength emitted from the victim's mobile devices, be it Wifi or cellular. To achieve this, we model and solve two complementary sub-problems. The first sub-problem aims at identifying the minimum number of scanning points the UAV should visit to allow a full scan of the whole area while meeting the accuracy levels of the different regions. The second sub-problem aims at constructing the shortest possible UAV trajectory to traverse all the points generated from the first sub-problem. After the UAV traverses its determined route and collects from different positions the RSSI values of the devices residing in the scanned area, localization is then performed. The latter extends on the well-known

All authors are with the Department of Computer Science and Mathematics, Lebanese American University (LAU), Lebanon. Corresponding Author: S. Sharafeddine, Email: sanaa.sharafeddine@lau.du.lb.

trilateration technique to account for statistical variation in the wireless channel model.

A. Related Literature

There exists a rich literature on localization methods and positioning systems designed for various applications including disaster and emergency situations in indoor and outdoor environments. In this section, we focus on recent research efforts that utilize UAVs for improved and efficient localization of devices. A practical UAV-based positioning system is proposed in [17] that does not rely on any fixed infrastructure. The authors use Moore curve and calculate the minimum curve level needed to sense all targets with minimum flight distance. This allows them to identify a hot area that includes potential targets and recursively apply Moore curve for better localization accuracy. GuideLoc proposed in [18] is another UAV-based positioning system with a similar objective. The authors divide the area into multiple unit partitions if the area is larger than the communication range of the UAV. They designate the center of unit partitions to be along the UAV route. At each center, the UAV senses all available devices within this partition and estimates the locations of devices based on RSSI and angle of arrival measurements. This information is then used to build a traveling salesman route to visit all estimated positions in one unit partition before leaving to the center of the next unit partition. At each visited position, the UAV attempts to improve the localization estimate of the corresponding device by flying in the direction of its strongest RSSI and finally sets the device position to the UAV coordinates when the RSSI is at its highest value.
Another recent related work is presented in [19] where a UAV is used to collect WiFi probe requests from known WiFi-enabled mobile devices to identify the geographical zone in which they are located. In the training phase, one UAV flies over the area and gathers WiFi probe requests from each zone. RSSI level and MAC addresses are extracted from the probe request and the actual GPS coordinates of the UAV are associated with the respective reading. After learning, the UAV collects readings and random forest algorithm is utilized to classify the zone in which a given device resides. Experimental tests resulted in around 82% accuracy to identify the right zone. In [20], the authors make use of UAVs to perform secure localization as an alternative to expensive fixed anchors, by taking each anchor as a waypoint for the UAV to perform localization. An algorithm named LocalizerBee is proposed to localize a given number of nodes using a trilateration technique that requires each node to be inside a triangle formed by three scanning points. To do so, a grid of vertices that form isosceles triangles with union that covers the entire area is generated and the travelling salesman problem is then applied to connect the vertices in an efficient way. In [21], the authors suggest using a UAV to provide wireless connectivity to ground users through solving a joint transmit power and trajectory optimization problem, while maximizing the minimum average throughput for a given time period. The aforementioned problem is non-convex and intractable, so it was divided to two convex optimization sub-problems: optimizing the transmit power for a given trajectory and optimizing the trajectory for a given transmit power. A sub-optimal low complexity iterative algorithm was proposed for the joint problem using the solutions of the two previous sub-problems. Simulation results demonstrated the superiority of the proposed solution compared to a benchmark where the UAV flies along a straight line from the starting point to the end point at a uniform speed. In [22], outdoor experiments are conducted to validate the so-called efficient geometry-based localization approach to localize static sensor nodes using a UAV equipped with a global navigation satellite system receiver. Experimental results proved high accuracy precision as compared to standard benchmark techniques.

Moreover, the anticipated prevalent use of UAVs calls for effective surveillance systems to detect unauthorized UAVs and localize them [23]. UAVs can also be leveraged to localize other UAVs. In [24], [25], a swarm of UAVs equipped with basic omni-directional antennas cooperatively localize a moving radio frequency transmitter. The proposed method is based on a predictive approach that computes a new UAV trajectory every time a new estimate of the target location is determined.

B. Novel Contributions

In light of the surveyed literature, we summarize below the major contributions of this paper.

- We address the problem of localizing an unknown number of victims scattered over a disaster scene that includes regions with varying levels of importance. The victims are assumed to hold mobile devices that have lost their wireless connectivity due to destroyed communication infrastructure. A key novelty of our problem formulation is the consideration of disaster scenes with heterogeneous importance levels.
- We propose and develop an efficient localization solution that leverages commodity UAVs and optimizes their flight trajectory to collect seamlessly RSSI measurements from mobile devices scattered arbitrarily over a given disaster scene. Our solution approach decomposes the problem into two subproblems: i. the first determines strategic waypoints at which the UAV hovers and collects the required number of RSSI readings to offer differentiated accuracy levels across the area; ii. the second constructs an efficient path composed of all waypoints generated from the first subproblem. Localization is then performed while accounting for statistical variation in the wireless channel model and by utilizing additional RSSI readings to further enhance accuracy.
- We design and develop an experimental testbed that implements the proposed solution in order to demonstrate its effectiveness under practical settings. These

experimental results complement a wide set of presented simulation results to analyze performance and extract useful insights.

II. System Model

We consider a disaster scene with damaged communication infrastructure where an unknown number of victims are lost or trapped under debris. We assume that most victims carry mobile devices. Figure 1 depicts the system model that shows an example disaster scenario, where the affected area is divided into regions with different levels of damage or importance.

As part of the search and rescue operation, a UAV is launched at a fixed height h to follow an efficient trajectory computed by a centralized server managed by the rescue team in order to scan the disaster area. During its mission, the UAV performs repeated RSSI measurements of the wirelss signals emitted from the victims' mobile devices, and once done delivers all measurements to the server. These readings are then used to identify the location of the victims that we denote as users hereinafter. Whenever the UAV and a user are within the communication range of each other, the UAV can capture all signals transmitted by the user. The radio range to capture the devices' signals at the UAV is denoted as r_s. We define a radius r to be the projection of r_s on the ground, and it can be computed as $r = \sqrt{r_s^2 - h^2}$. Figure 2 depicts the relation among h, r_s, and r.

To compute the trajectory that the UAV follows, the disaster scene is divided into equal sized $n \times m$ square cells of side length ρ, where n is the number of rows and m is the number of columns as demonstrated in Figure 1. Each cell is denoted by (i, j), where i is the row index and j the column index such that $0 \leq i < n$ and $0 \leq j < m$. Cells are given differentiated levels of importance to reflect the corresponding need for localization accuracy and, hence, a higher number of required RSSI readings. We denote by K_{ij} the number of readings required for each cell (i, j). The process of assigning the importance level of cells depends on various metrics such as the damage degree and the population density within each cell. For example, areas that span parks and empty spaces may be assigned less number of readings as compared to areas with building structures that are likely to be more populated and the rescue mission is more challenging. In the case when it is based on the damage degree for example, image processing techniques may be utilized to compare pre- and post-disaster satellite images to assess the different levels of damage within the disaster scene.

Based on the given set of required number of readings over the whole area, we aim at identifying an optimized set of cells with centers that will form the UAV's trajectory waypoints $s \in \mathcal{S}$. Each waypoint s when visited by the UAV allows the scanning of RSSI signals emitted from devices located within the waypoint's cell as well as all neighboring cells that are within the UAV's coverage range. As the UAV follows the trajectory, it collects the required number of RSSI readings for all cells based on the assigned importance levels. Being used for scanning, each waypoint is referred to as a scanning point. The order in which the scanning points are visited is determined by minimizing the total trajectory distance.

The complete localization process is summarized in a step-wise manner below. We note that communication between the server and the UAV is only needed at the start as well as at the end of the UAV mission. The computed trajectory plan is downloaded to the UAV at the beginning of its mission and the collected RSSI readings are uploaded to the server once the UAV mission is completed.

- The map of the disaster scene is input to the server managed by the search and rescue team. The disaster area is divided into cells with different importance levels based on set metrics if available; otherwise, all cells are assigned a uniform level of importance.
- The server computes the UAV trajectory and transfers it to the UAV.
- The UAV is launched and follows the trajectory plan, while hovering over each scanning point for a predefined amount of time and measuring RSSI of all available wireless signals emitted from users in the area.
- The UAV travels back to the server to deliver the collected RSSI readings.
- The server runs the proposed algorithm using the provided input to determine the estimate location of discovered users.

III. Problem Decomposition and Proposed Solution Approach

The addressed problem in this work considers identifying a minimal number of scanning points at which the UAV is expected to hover and collect the required number of RSSI signals dictated by each region of the disaster scene, in addition to constructing an efficient trajectory that traverses all scanning points. This is followed by mapping the RSSI readings to distance estimates while accounting for wireless channel variability, and finally determining the area in which each device may reside. In order to deal with the problem complexity, we decompose it into two subproblems as explained in this section.

A. Scanning Points Identification Subproblem

In this subproblem, we are concerned to find the minimum number of UAV scanning points such that hovering on top of those positions allows the UAV to listen to and collect k_{ij} RSSI readings required by each cell (i, j) such that its residing users can be accurately localized. To do so, the area of interest, after being divided into equal-sized cells, is modeled as a graph $\mathcal{G} = \mathcal{V} \cup \mathcal{E}$ with $\mathcal{V}$ denoting the set of vertices that constitute the centers and $\mathcal{E}$ the set of edges. Using the graph terminology, each vertex v corresponding to cell (i, j) is required to be covered $k_v = k_{ij}$ times. An edge $e \in \mathcal{E}$ exists between two vertices v and u whenever their corresponding cells

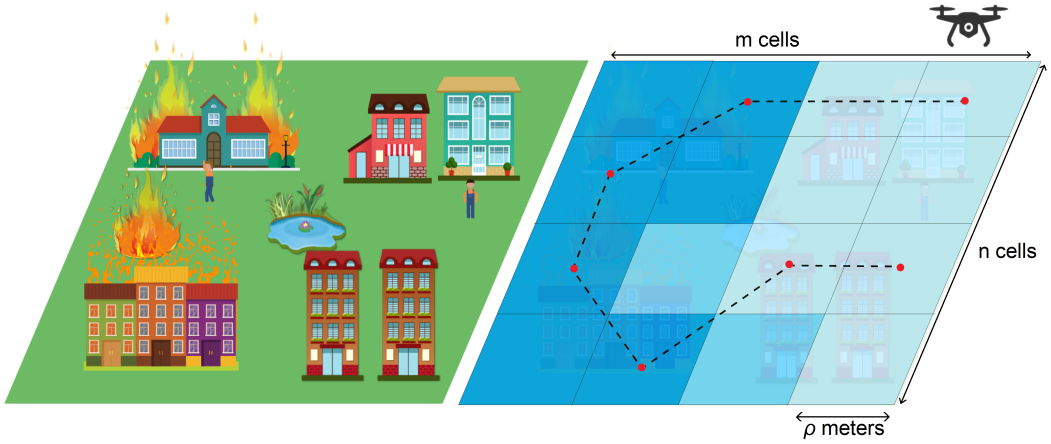

Fig. 1. The image on the left represents an example disaster scene. The figure on the right shows the digitization of the area and the assignment of the number of readings per cell depending on its level of damage or importance. The darker the color, the higher the importance level and the number of required readings. The UAV trajectory is drawn with the red dots representing the UAV's scanning points.

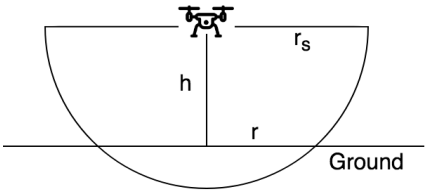

Fig. 2. Projection of the UAV's coverage range on the ground.

are within the coverage range of each other and thus the vertices are considered adjacent when they satisfy the following equation: $\text{dist}(v,u)+\rho\frac{\sqrt{2}}{2} \leq r$, where $\text{dist}(v,u)$ is the Euclidean distance between the two vertices. In other words, when the UAV resides in one cell, it can listen to RSSI signals coming from all users in the other cell. The subproblem is then reduced to determining the minimum subset $\mathcal{S} \in \mathcal{V}$ such that each vertex v in $\mathcal{V}$ is adjacent to k_v vertices in $\mathcal{S}$. Consequently, this subproblem can be reduced to the minimum dominating set problem if we consider one instance of it when the required number of readings is one.

To formulate this subproblem, we define a binary variable d_v that is set to 1 when vertex v is selected to be part of $\mathcal{S}$ and set to 0, otherwise. Thus, the scanning points identification subproblem can be formulated as follows:

$$\text{minimize} \sum_{v \in \mathcal{V}} d_v \tag{1}$$

$$\text{subject to} \sum_{u \in E[v]} d_u \geq k_v \quad \forall v \in \mathcal{V} \tag{2}$$

The constraint in (2) signifies that each vertex v must be scanned k_v times when the UAV visits all vertices $u \in \mathcal{S}$, where $\mathcal{S}$ includes all vertices u that have $d_u = 1$. $E[v]$ in (2) denotes the neighborhood set of vertex v including itself, where the neighborhood of a vertex constitutes all adjacent vertices.

Being an NP-hard problem, we develop a two-step heuristic algorithm based on a greedy approach to solve the dominating set problem as described in [26]. The greedy approach selects the vertex which can dominate the maximum number of uncovered vertices until all vertices are covered. To apply it to our problem that requires every vertex v to be dominated k_v times, the first step selects the vertex with the highest dominance rank based on the adjacent vertices that still have remaining readings. The second step applies a local search that iteratively removes vertices from the dominating set as long as every vertex is still dominated at least k_v times. In each iteration, the vertex which dominates the least number of vertices is considered first. This solution is referred to as SPDS (Scanning Points Dominating Set) and presented in Algorithm 1. SPDS has a worst case time complexity of $O(u^2 deg(u))$, where u is the number of cells and $deg(u)$ represents number of cells the UAV radius r can cover.

A common pitfall of SPDS is that it does not account for the geometrical requirements of trilateration to ensure a more accurate localization. These requirements include having the device to be localized: 1) not co-linear with the scanning points, 2) close to the scanning points, and 3) inside the triangle formed by those points [27]. To address the geometrical requirements of trilateration, we utilize Centroidal Voronoi Tessellation (CVT). The Voronoi tessellation of a given two-dimensional space partitions it into n convex polygons based on distances to n designated points called generators such that each point inside a polygon is closer to its generator. A Voronoi tessellation is considered centroidal when the generators are the centroids of the polygons. After running SPDS, we use the resulting scanning points as initial generators of the CVT that we determine using Lloyd's iterative algorithm as described in [28], which has a time complexity of $O(islog(s))$, where s is the number of scanning points and i is the number of iterations. To avoid situations where CVT results in co-linear scanning points, we scatter the initial generators across the search space and select the vertex v that maximizes $f(v)$ while running SPDS

whenever multiple vertices exhibit the same dominance score. $f(v) = \sum_{u \in \mathcal{S}} \parallel v - u \parallel^2$, where $\mathcal{S}$ is the current set of already chosen vertices that constitute the scanning points. Maximizing the sum of squares of the distances in $f(v)$ avoids having two points close to each other. After forming the CVT, the distance between the centroid and the scanning point that was input as the initial generator is computed. If it is less than a certain threshold, the scanning point is kept in the final solution and the generator is discarded. Otherwise, the centroid replaces the scanning point in the final solution. SPDS is finally re-run to guarantee that the final set $\mathcal{S}$ of scanning points can still ensure k_v readings for all $v \in \mathcal{V}$. This algorithm is presented in Algorithm 2 and is referred to as SPVT (Scanning Points Voronoi Tessellation). SPVT has the same time complexity of SPDS.

B. Trajectory Construction Subproblem

The scanning points subproblem identifies the strategic scanning points that the UAV visits to capture incoming RSSI readings coming from neighboring cells. The trajectory construction subproblem takes the scanning points as an input and generates an efficient UAV trajectory. This subproblem reduces to the Travelling Salesman Problem (TSP) where the weight of an edge that connects two scanning points is the Euclidean distance between the two vertices. In our case, an exact solution isn't required as UAVs have high flight speeds greatly mitigating the time lost navigating the longer path of the approximated solution compared to the exact one. We use the 2-opt heuristic solution of the TSP as described in [29] to efficiently compute the UAV trajectory. The 2-opt local search heuristic has $O(u^2)$ time complexity, where u is the number of scanning points, and can guarantee a 2-optimal solution.

C. Localization with Wireless Channel Variability

As described in [30], many localization techniques can be used in wireless networks like trilateration, multilateration, triangulation and others. The aforementioned techniques are based on GPS, RSSI, AOA (angle of arrival), TOA (time of arrival), or TDOA (time difference of arrival) measurements to perform localization of devices with unknown positions. RSSI-based techniques have been shown to provide an effective tradeoff between accuracy, feasibility and complexity and, thus, are suitable for our proposed solution approach. Once an RSSI reading is captured, it needs to be converted to distance using an appropriate channel model. The channel model proposed in [31] is widely used for aerial platforms; however, it is based on the angle of incidence between each device and the UAV, the fact that makes it not applicable to our problem as the locations of the devices are not known. Consequently, we resort to using the following log-normal shadowing pathloss model as it is capable of modeling wireless environments with acceptable precision [32]:

$$P_r(dB) = P_t(dB) - 10\alpha \log\left(\frac{d}{d_0}\right) + X_\sigma, \qquad (3)$$

Input: $\mathcal{G} = \mathcal{V} \cup \mathcal{E}$, K
Output: $\mathcal{S}$
$\mathcal{G} \leftarrow$ graph containing the vertices and edges
$K \leftarrow$ array holding the number of readings per vertex
$E \leftarrow$ array holding the neighbors of each vertex
$\mathcal{S} \leftarrow \emptyset$
while *NOT* `Constraints_Satisfied(`$\mathcal{G}$, $\mathcal{S}$, K`)` **do**
  $v \leftarrow$ `Best_Vertex(`$\mathcal{G}$, $\mathcal{S}$, K`)`
  $\mathcal{S} \leftarrow \mathcal{S} \cup v$
  `Update_Readings(`$\mathcal{G}$, v , K`)`
end
while *NOT* `Local_Search(`$\mathcal{G}$, $\mathcal{S}$, K`)` **do**
end
return $\mathcal{S}$

Function `Constraints_Satisfied(`$\mathcal{G}$, $\mathcal{S}$, K`)`**:**
  for $v \in \mathcal{V}$ **do**
    if $k[v] > 0$ **then**
      return False
    end
  end
  return True

Function `Best_Vertex(`$\mathcal{G}$, $\mathcal{S}$, K`)`**:**
  $best\_score \leftarrow 0$
  $best\_vertex \leftarrow NULL$
  for $v \in \mathcal{V}$ *and* $v \notin \mathcal{S}$ **do**
    $score = \sum_{u \in E[v]} max(1, k[u])$
    if $score \geq best\_score$ **then**
      $best\_score \leftarrow score$
      $best\_vertex \leftarrow v$
    end
  end
  return v

Function `Update_Readings(`$\mathcal{G}$, v, K`)`**:**
  for $u \in E[v]$ **do**
    $k[u] \leftarrow k[u] - 1$
  end

Function `Local_Search(`$\mathcal{G}$, $\mathcal{S}$, K`)`**:**
  $\mathcal{S} \leftarrow \mathcal{S}$ sorted based on dominance rank
  for $v \in \mathcal{S}$ **do**
    `Update_Readings_Local(`$\mathcal{G}$, v, K`)`
    if `Constraints_Satisfied(`$\mathcal{G}$, $\mathcal{S} - v$, K`)` **then**
      $\mathcal{S} \leftarrow \mathcal{S} - v$
      return True
    end
    else
      `Update_Readings(`$\mathcal{G}$, v, K`)`
    end
  end
  return False

Function `Update_Readings_Local(`$\mathcal{G}$, v, K`)`**:**
  for $u \in E[v]$ **do**
    $k[u] \leftarrow k[u] + 1$
  end

Algorithm 1: SPDS Algorithm

Input: $\mathcal{G} = \mathcal{V} \cup \mathcal{E}$, $\mathcal{S}$
Output: $\mathcal{S}\prime$ - updated set of scanning points
$\mathcal{G} \leftarrow$ graph containing the vertices
$\mathcal{S} \leftarrow$ solution got from the modified SPDS
$CVT \leftarrow$ centroids of the CVT from $\mathcal{S}$
$\mathcal{S}\prime \leftarrow \emptyset$
for $v \in CVT$ **do**
  $s \leftarrow$ vertex in $\mathcal{S}$ closest to v
  $\mathcal{S} \leftarrow \mathcal{S} - s$
  if $\| s - v \| \leq threshold$ **then**
    $\mathcal{S}\prime \leftarrow \mathcal{S}\prime \cup v$
  end
  else
    $\mathcal{S}\prime \leftarrow \mathcal{S}\prime \cup s$
  end
end
while *NOT* `Constraints_Satisfied(`$\mathcal{G}$, $\mathcal{S}\prime$, K`)` **do**
  $v \leftarrow$ `Best_Vertex(`$\mathcal{G}$, K`)`
  $\mathcal{S}\prime \leftarrow \mathcal{S}\prime \cup v$
  `Update_Readings(`$\mathcal{G}$, v , K`)`
end
return $\mathcal{S}\prime$

Algorithm 2: SPVT Algorithm

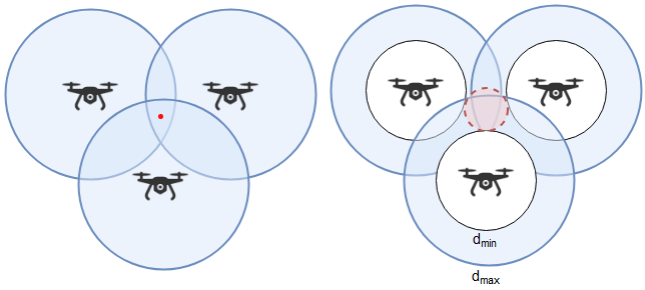

Fig. 3. The left figure depicts trilateration with one fixed X_σ. The right figure shows the case in which X_σ is bounded between two values.

where P_r and P_t denote the received and the transmitted power, respectively, d_0 the reference distance of 1 m, d the distance between the receiver and transmitter, α is the path-loss exponent that depends on the environment and X_σ represents the signal variation caused by shadowing and modeled as a Gaussian random variable with zero mean and standard deviation σ. The distance between the UAV and the device to be localized can then be calculated as follows:

$$d = 10^{\frac{P_t - P_r + X_\sigma}{10\alpha}}. \quad (4)$$

After mapping the received RSSI reading to its corresponding distance, well-known trilateration-based localization techniques can be used. In a two-dimensional space, three distance measurements from three distinct positions are recorded to generate three circles centered at the position where the measurements are taken with radii equal to the respective measurements. Should the distance measurements be accurate, the three circles intersect in one point that constitute the position of the object to be localized. Unfortunately, converting RSSI values to distances does not yield accurate measurements due to the statistical variations in wireless channels. As a result, the circles do not end up intersecting in one point but rather have an intersection area as demonstrated in the left side of Figure 3, and the device's position is then estimated by minimizing the least square error.

Due to variations in different environments, it is not possible in practice to estimate a fixed value for the shadowing component to be factored in the distance calculation in (4). As a result, we address this problem by bounding the shadowing component between two designated values $X_{\sigma\min}$ and $X_{\sigma\max}$ and calculating the corresponding bounding distance values $d_{\min}$ and $d_{\max}$, respectively, to form the radii of two concentric circles centered at the position of the UAV when the corresponding measurement is taken. The user is then expected to reside in the circular ring formed by the area enclosed by the two concentric circles. The UAV then moves and collects measurements from at least two other positions to satisfy the requirement of trilateration. Two concentric circles are generated from each measurement as depicted in the right side of Figure 3 and the user location is then bounded to the area of intersection of all circular rings. The user's location is estimated to be the centroid of the resulting formed area.

TABLE I
TOTAL NUMBER OF VERTICES GENERATED BY SPDS AS COMPARED TO THE OPTIMAL SOLUTION

Number of Readings	Radius in m	Optimal Solution	SPDS Solution	SPDS/Optimal Ratio
3	2	31	36	1.16
3	3	15	17	1.13
3	4	12	12	1.00
4	2	41	47	1.14
4	3	20	22	1.20
4	4	16	16	1.00

IV. Simulation Results and Discussion

A. Scanning Points Identification Subproblem: SPDS vs. Optimal Solution

To evaluate the performance of the proposed SPDS algorithm that generates the dominating set to cover every vertex $v \in \mathcal{V}$ k_v times, we compare it against the optimal solution of the problem defined in (1) – (2) for small-scale scenarios using Matlab's optimization toolbox. The grid size is set to $10m \times 10m$ with a side length $\rho = 1m$. We note that this considered scenario is intentionally limited to a small geographical area so the optimal solution is obtained in reasonable time. More realistic scenarios are considered later in this section to demonstrate the effectiveness of the proposed SPDS approach. Table I compares the results of SPDS versus the optimal solution for a uniform number of readings across all cells. For every k_v, the UAV radius r takes the following values: $2m$, $3m$, and $4m$. Table I shows the number of vertices in the dominating set generated by both the optimal solution and SPDS. As demonstrated, SPDS is capable of generating close-to-optimal results that constitute the initial set of scanning points.

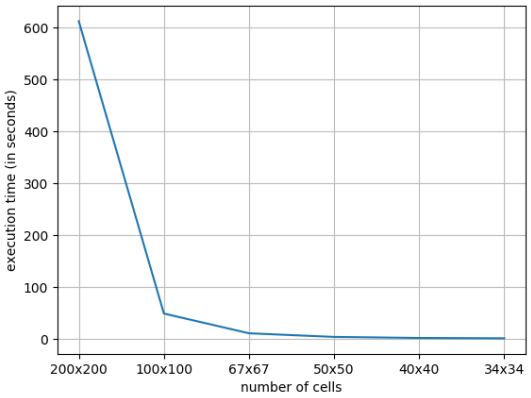

Fig. 4. Graph showing the execution time of SPVT while reducing the number of cells in a 200 x200 square meters area (ρ values ranging from 1 to 6 meters).

B. Execution time

To study the efficiency of SPVT and the effect of the number of cells on the execution time of the algorithm, we measure the time to run SPVT for different ρ values on an environment consisting of a 200x200 square meters area, resulting in different number of cells. The UAV height h is set to 15 meters and the UAV range r_s to 60 meters. The algorithm runs on an Intel Core i7 6700K processor. The results, presented in Figure 4, show that a high number of cells can be very detrimental to the execution time of the algorithm, hence the importance of avoiding relatively small ρ values. The execution time drops by more than 90% when ρ increases from 1 to 2 meters resulting in reducing the number of cells from 200x200 to 100x100.

C. Performance Improvement through SPVT

To evaluate the localization accuracy of our solution and the improvement caused through running SPVT rather than SPDS alone, we develop a simulation environment with $200m \times 200m$ area divided into a grid of cells, each with side length $\rho = 10m$. All cells (i, j) require equal k_{ij} readings that vary between 3 and 8. 400 users are randomly placed in the search space according to a uniform distribution and each user casts a number of WiFi beacons over time. RSSI readings are captured by the UAV whenever it is within the coverage range of the user. RSSI readings are generated according to (3), where X_σ follows a Gaussian distribution with zero mean and standard deviation σ set to 4 dB [33], [34]. Upon capturing an RSSI reading by the UAV, two bounding distances are generated, as described in Section III-C, one with $X_\sigma = X_{\sigma\,\min}$ and the other with $X_\sigma = X_{\sigma\,\max}$. $X_{\sigma\,\min}$ is chosen to be equal to the signal level minus 2σ and $X_{\sigma\,\max}$ to be equal to the signal level plus 2σ. The values of the main simulation parameters are listed in Table II unless otherwise specified. The localization results are then generated as shown in Figure 5 to evaluate the performance improvement of applying SPVT compared to applying SPDS to solve the scanning points identification subproblem.

TABLE II
KEY SIMULATION PARAMETERS

Parameter	Value
h (UAV height)	15 meters
r_s (UAV range)	60 meters
α (pathloss exponent)	4
σ (shadowing standard deviation)	4 dB
P_0 (transmission power)	-60 dB
ρ (cell side)	10 m

As per Figure 5, the localization error drops as the number of readings per cell increases. Applying SPVT, however, consistently offers superior performance over SPDS. In terms of localization error, the SPVT algorithm results in an average error of around 6 m for the scenario with five readings per cell and 2 m for eight readings per cell, while SPDS results in 9 m for five readings per cell and 6 m for eight readings per cell. The middle graph of Figure 5 shows that SPVT does not require additional number of scanning points as compared to SPDS as both curves are almost overlapping. The right graph presents the percentage of accurately localized devices within a given distance for the case of five readings per cell and demonstrates the improvement of localization accuracy when SPVT is applied. For example, using SPVT, 60% of the devices are accurately localized within six meters compared to less than 50% for SPDS. This graph demonstrates the improvement in localization accuracy when the geometrical requirements of trilateration are respected.

In Figure 6, an example UAV trajectory is presented for each of SPDS and SPVT when the number of readings per cell is set to five and the number of devices scattered over the area is set to 20 for better visualization. The black dots mark the generated scanning points. It is clear that the SPVT scanning points are more uniformly spread across the search space compared to SPDS where several scanning points are relatively close to each other.

D. Example Disaster Scenario

In this subsection, we intend to apply our proposed SPVT localization solution to a realistic scenario. We extract $320m \times 320m$ area from a picture showing the damage caused after the tsunami disaster in Indonesia and divide it into a grid of cells as shown in Figure 7. The number of readings per cell depends on the assigned importance level and may be set according to the undertaken damage, where it varies between 3 and 6. We run SPVT and plot in Figure 8 the average localization accuracy with respect to the UAV's range. For each value of the UAV range, we generate 30 runs and compute average accuracy results. The figure demonstrates that smaller UAV ranges reduce the error at the expense of increasing the number of scanning points required to cover the whole area.

Finally, to better assess the efficacy of SPVT, we compare its performance against an alternative approach. We

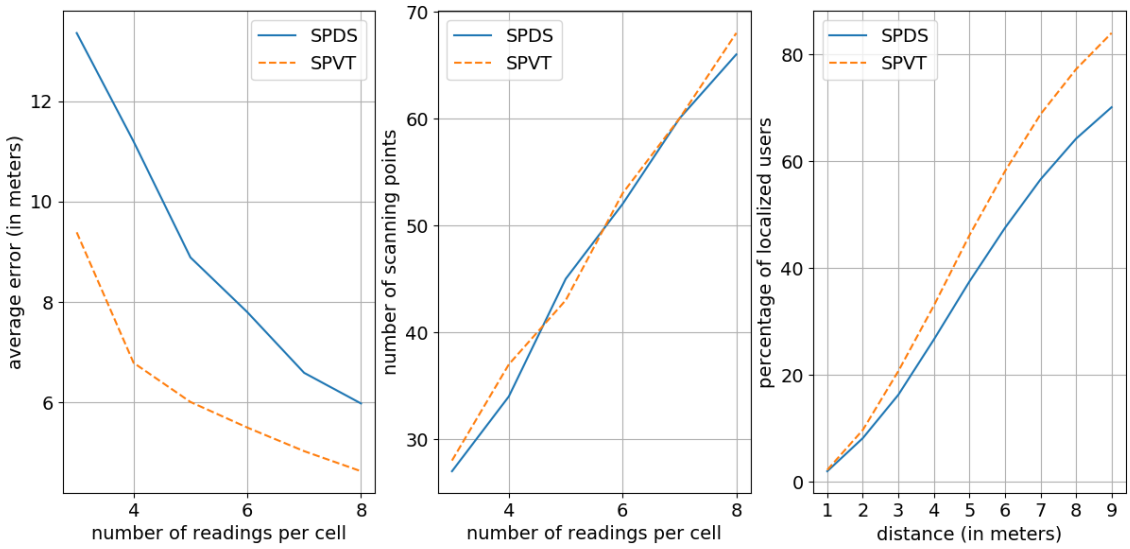

Fig. 5. Performance improvement of SPVT compared to SPDS. The left graph shows the average localization error versus the number of readings per cell. The middle graph plots the number of generated scanning points versus the number of readings. The right graph depicts the percentage of users localized within a certain distance of their actual location for the scenario with five readings per cell.

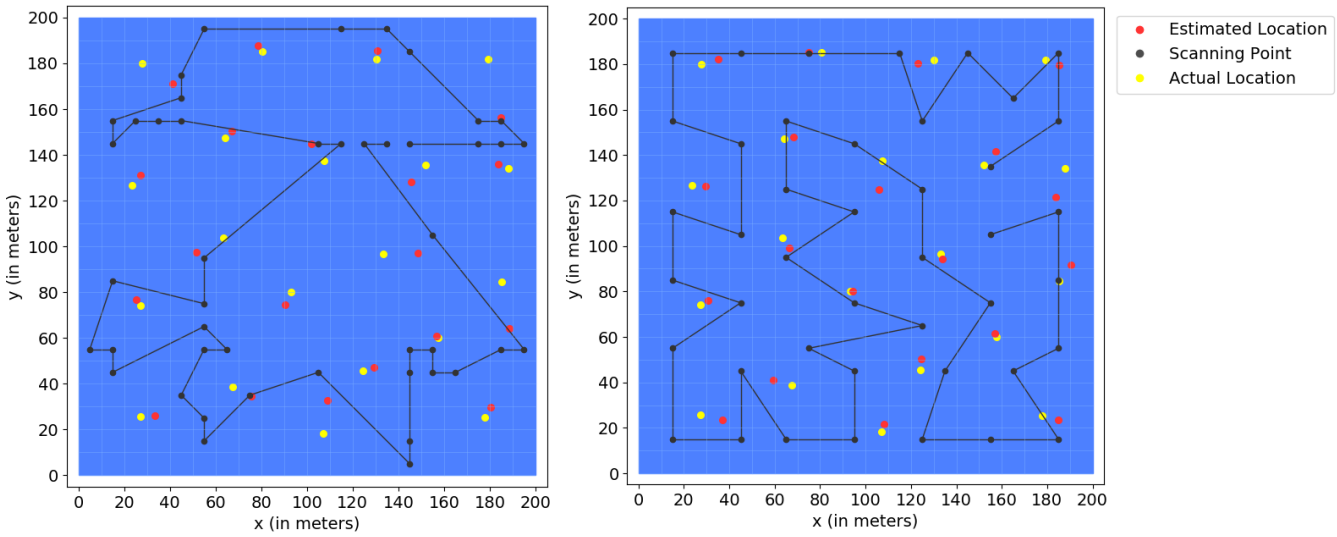

Fig. 6. The left figure shows the UAV's trajectory using the SPDS algorithm. The right figure shows the UAV's trajectory using the SPVT algorithm. All cells are assumed to have the same importance with five required readings.

select a UAV-based localization solution proposed in [17] and referred to HAWK, as being very related to this work. HAWK, however, aims at identifying the populated areas and then intensifies the UAV hovering above those areas to better localize the users. To do so, the work in [17] generates an initial Moore's curve to cover the entire search space. The level of the Moore's curve and the UAV speed are selected in a way to capture at least one packet from each possibly existing user during the UAV's flight. Once the UAV receives a strong RSSI signal from one user, it marks the area as a hot area and recursively traces higher degree Moore's curves to improve the localization accuracy. The location of the user is then determined to be the UAV's position when it sensed its strongest signal. While HAWK is customized to localize lower number of users, we adapted their algorithm to evaluate the performance of our proposed solution. Cells with higher importance are assigned higher degree Moore's curve for increased localization accuracy.

The adapted HAWK is applied on the same disaster scenario presented in Figure 7 and the resulting trajectory to scan the damaged area is traced in Figure 9 . It is

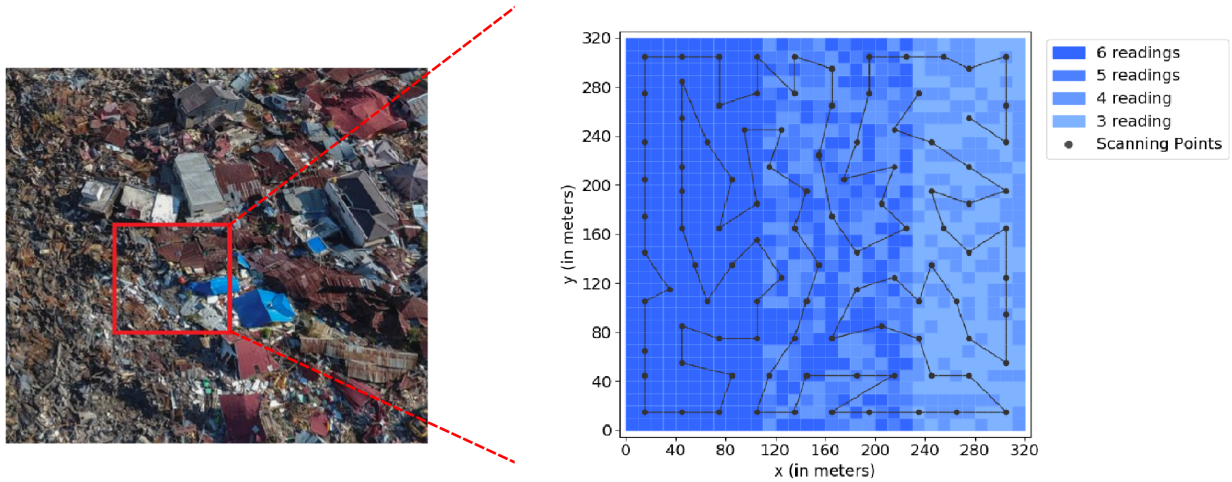

Fig. 7. The left figure shows a selected area with damage by the tsunami in Indonesia in 2018 [source: Yahoo News]. The right figure shows the number of readings required by each cell of the damaged area based on its importance, in addition to the trajectory taken by the UAV when SPVT is applied.

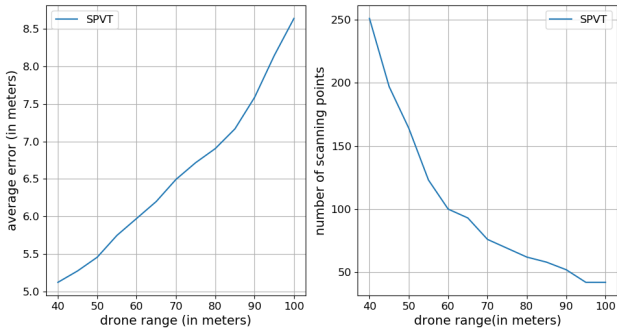

Fig. 8. The left figure plots the average localization error of SP versus the UAV range. The right figure plots the resulting t number of scanning points versus the UAV range.

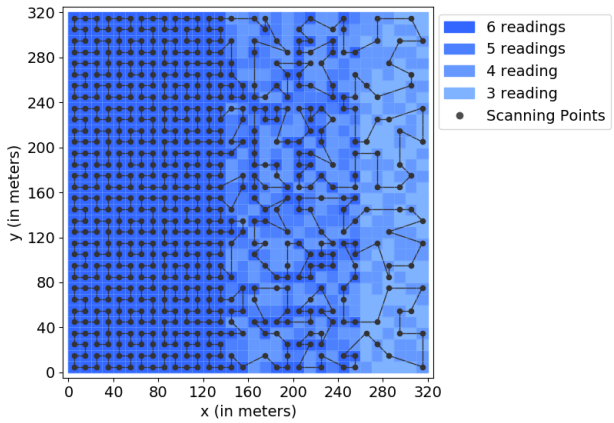

Fig. 9. The UAV trajectory generated by HAWK to scan the damaged area.

evident that the latter approach utilizes a much longer trajectory as compared to SPVT and this is due to increasing Moore's curve level in areas with higher required accuracy. Figure 10 presents the performance of the adapted HAWK in terms of the average localization error and the average flight time. We note that the performance of HAWK is dependent on the UAV speed since the UAV collects signals while traveling along its path. The slower the UAV is, the more signals can be collected and thus higher accuracy is expected as demonstrated in the figure. For a low speed of 1 m/s, HAWK achieves its lowest average error of 8.6 m that steadily increases as the UAV increases its speed. The flight time needed to traverse the whole trajectory dictated by HAWK at 1 m/s exceeds 8000 s (equivalent to 2.22 hours) as per Figure 10. SPVT localization accuracy, on the other hand, is not based on the UAV speed and, thus, the UAV can fly at its maximum speed to travel from one scanning point to another. For a fair comparison, we estimate the flight time needed by SPVT to achieve a similar accuracy of 8.6 m. This is attained when the UAV's range reaches 100 m causing 42 scanning points as per Figure 8. At each scanning point, the UAV hovers for 30 s in our simulations to collect multiple RSSI readings for each device to average out the effect of fading; thus, the flight time of SPVT constitutes 30 s $\times$ 42 = 21 min in addition to the time needed to travel between scanning points at maximum UAV speed.

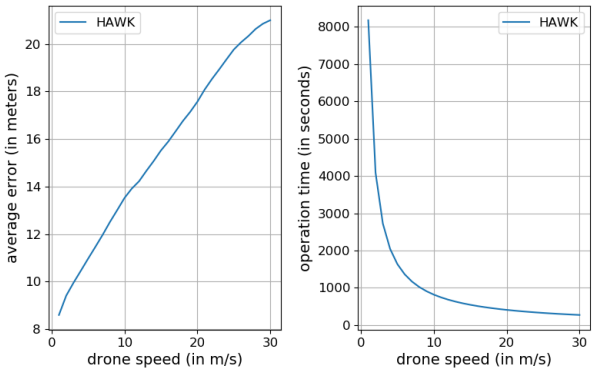

Fig. 10. HAWK performance results. The left figure plots the average localization error versus the UAV speed. The right figure plots the total number of scanning points versus the UAV speed.

V. Testbed Implementation and Evaluation

In order to validate the practicality and effectiveness of the proposed localization approach, we have designed and implemented an experimental testbed. This testbed serves as a proof of concept that demonstrates the solution to be complete, practical, and can be implemented using available hardware and software tools. We used the testbed to conduct experiments in the basketball court at the Lebanese American University, where we flew a UAV to localize smartphones distributed in different locations on the ground. We used a Parrot Bebop 2 UAV [35] to capture RSSI readings of WiFi signals emitted from the smartphones. Cellular-based signal scanning can also be performed; however, this would require installing a cellular transceiver with scanning capability on the UAV. Without loss of generality, we adopt WiFi-based scanning in this testbed due to availability and cost considerations. To do so, we considered the following two methods:

1) Using the original WiFi adapter of the UAV to capture RSSI readings in *managed mode*: this requires adding bash scripts and C programs to the UAV's operating system (Arch Linux) to transform it into a server capturing signals.The drawback of this method is that the devices need to be in *hotspot mode* so their signals could be captured. This might be a limitation in practice since victims might not be able to change their mobile devices' mode after a disaster takes place.
2) Adding a new USB WiFi adapter to the UAV: this requires adding and loading the necessary drivers in the UAV's kernel to support the new adapter, as well as adding bash scripts and C code to setup the interface and make the UAV a server capturing RSSI readings. The main motivation behind this method is that the devices do not need to be in *hotspot mode* for their signals to be captured by the UAV, as the newly added adapter can run in *monitor mode*.

For practical reasons, we resorted to the second method. For our implementation, we used the official Parrot SDK for Android smartphones. Moreover, we developed an Android mobile application that takes as input the dimensions of the area of interest and the number of readings required per cell. The mobile application then computes the trajectory, connects to the UAV's server to initiate WiFi signal capture, and communicates the trajectory to the UAV so it visits each designated scanning point and captures all available RSSI signals during a given period of time. The latter information is sent back to the application that executes the localization technique described in Section III-C to obtain estimates for the locations of all discovered devices. A screen shot of the developed Android mobile application is shown in Figure 11.

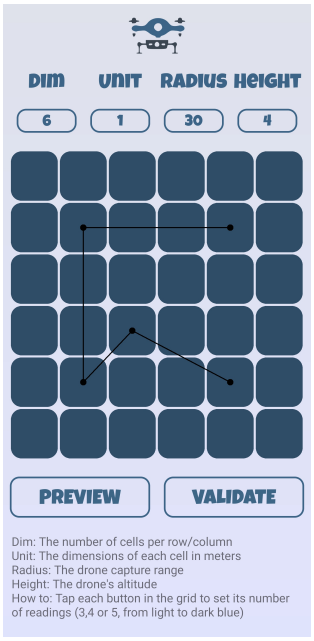

Fig. 11. Screen shot of the developed Android mobile application to compute the UAV trajectory. In this screen shot, a $6m \times 6m$ grid is set with each cell having a side of $1m$, and assigned five readings. The black dots represent the identified UAV's scanning points.

A. Testbed Results

The implemented testbed is run in a basketball court as shown in Figure 12. To apply (4), we need to determine an estimate of the value for α in our testing environment. To do so, we placed one device at different distances from the UAV and measured the RSSI values; we then used the data to estimate α empirically to be equal to 2.7.

As an example testing scenario, we have placed two Samsung S5 smartphone devices on a $6m \times 6m$ area in the basketball court at the following coordinates: ($2m$,$1m$) and ($2m$,$3m$). We divided the area into multiple cells with each cell assigned five required readings. The mobile application computed the trajectory and the UAV was flown accordingly at a height of $4m$ above the ground level. After completing the trajectory and collecting all RSSI readings, the mobile application performed localization. The two devices' locations were estimated to be at the following coordinates: ($2.4m$, $1.6m$) and ($2.6m$, $2.1m$), leading to a

Fig. 12. Running the testbed in a basketball court using one UAV and smartphone devices.

localization error of $0.72m$ and $1m$, respectively. Several other tests were conducted and led to similar accuracy outcomes. This demonstrates the practicality of the proposed solution using existing smartphone devices under regular configuration and operation.

VI. Conclusions

In this work, we addressed the problem of using UAVs to localize users in disaster scenes having regions with varying importance that may be set according to the damage and/or population level. The whole scene is discretized into cells and cells are assigned different number of RSSI readings according to their importance level. The problem jointly considers identifying scanning points that can cover all cells with the required number of readings, constructing an efficient UAV trajectory that visits all scanning points, and developing a localization mechanism. We proposed an RSSI-based localization technique that uses trilateration while accounting for the variability of the wireless channel. The technique attempts to improve the localization accuracy with every additional RSSI reading. To collect the needed number of readings from all cells, we decomposed the problem into two subproblems, the first identifies the scanning points using dominating set and centroidal Voronoi tesselation and the second connects the scanning points in an efficient path that forms the UAV's flight trajectory. Simulation results demonstrated the effectiveness of our solution and its ability to adapt to regions with varying importance levels. Moreover, a small-scale testbed is developed in which the proposed localization solution is implemented together with its corresponding algorithms to validate its applicability in realistic settings. An interesting extension of this work is to deploy the proposed solution in larger testing scenarios with more users distributed in different areas subject to varying levels of importance.

Acknowledgements

This work has been jointly funded with the support of the National Council for Scientific Research in Lebanon CNRS-L and the Lebanese American University (LAU).

References

[1] A. F. G. Ferreira, D. M. A. Fernandes, A. P. Catarino, and J. L. Monteiro, "Localization and positioning systems for emergency responders: A survey," *IEEE Communications Surveys Tutorials*, vol. 19, pp. 2836–2870, Fourthquarter 2017.

[2] A. E. Redondi and M. Cesana, "Building up knowledge through passive WiFi probes," *Computer Communications*, vol. 117, pp. 1 – 12, 2018.

[3] Z. Yang, J. Schafer, and A. Ganz, "Disaster response: Victims' localization using bluetooth low energy sensors," in *2017 IEEE International Symposium on Technologies for Homeland Security (HST)*, April 2017.

[4] S. Liu, Y. Jiang, and A. Striegel, "Face-to-face proximity estimationusing bluetooth on smartphones," *IEEE Transactions on Mobile Computing*, vol. 13, pp. 811–823, April 2014.

[5] Y. Ma, N. Selby, M. Singh, and F. Adib, "Fine-grained RFID localization via ultra-wideband emulation," in *Proceedings of the SIGCOMM Posters and Demos*, SIGCOMM Posters and Demos '17, (New York, NY, USA), pp. 116–118, ACM, 2017.

[6] L. Yang, Y. Chen, X.-Y. Li, C. Xiao, M. Li, and Y. Liu, "Tagoram: Real-time tracking of mobile RFID tags to high precision using COTS devices," in *Proceedings of the 20th annual international conference on Mobile computing and networking*, pp. 237–248, ACM, 2014.

[7] W. Huang, Y. Xiong, X.-Y. Li, H. Lin, X. Mao, P. Yang, and Y. Liu, "Shake and walk: Acoustic direction finding and fine-grained indoor localization using smartphones," in *INFOCOM, 2014 Proceedings IEEE*, pp. 370–378, IEEE, 2014.

[8] Y.-S. Kuo, P. Pannuto, K.-J. Hsiao, and P. Dutta, "Luxapose: Indoor positioning with mobile phones and visible light," in *Proceedings of the 20th annual international conference on Mobile computing and networking*, pp. 447–458, ACM, 2014.

[9] Z. Yang, Z. Wang, J. Zhang, C. Huang, and Q. Zhang, "Wearables can afford: Light-weight indoor positioning with visible light," in *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '15, pp. 317–330, ACM, 2015.

[10] S. He and S. H. G. Chan, "Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Communications Surveys Tutorials*, vol. 18, pp. 466–490, Firstquarter 2016.

[11] United Nations Office for the Coordination of Humanitarian Affairs (OCHA), "The future of technology in crisis response." https://www.unocha.org/story/future-technology-crisis-response, April 2017.

[12] S. Sharafeddine and R. Islambouli, "On-demand deployment of multiple aerial base stations for traffic offloading and network recovery," *Computer Networks*, vol. 156, pp. 52–61, 2019.

[13] R. Islambouli and S. Sharafeddine, "Autonomous 3D deployment of aerial base stations in wireless networks with user mobility," in *2019 IEEE Symposium on Computers and Communications*, June/July 2019.

[14] M. Samir, S. Sharafeddine, C. Assi, T. M. Nguyen, and A. Ghrayeb, "Trajectory planning and resource allocation of multiple UAVs for data delivery in vehicular networks," *IEEE Networking Letters*, 2019.

[15] United Nations Office for the Coordination of Humanitarian Affairs (OCHA), "Unmanned aerial vehicles in humanitarian response," Policy and Studies Series, 2014.

[16] K. Goss, "Unmanned aerial systems emergency management." https://www.domesticpreparedness.com/resilience/unmanned-aerial-systems-emergency-management/, October 2017.

[17] Z. Liu, Y. Chen, B. Liu, C. Cao, and X. Fu, "HAWK: An unmanned mini-helicopter-based aerial wireless kit for localization," *IEEE Transactions on Mobile Computing*, vol. 13, no. 2, pp. 287–298, 2014.

[18] A. Wang, X. Ji, D. Wu, X. Bai, N. Ding, J. Pang, S. Chen, X. Chen, and D. Fang, "GuideLoc: UAV-assisted multitarget localization system for disaster rescue," *Mobile Information Systems*, vol. 2017, 2017.

[19] V. Acuna, A. Kumbhar, E. Vattapparamban, F. Rajabli, and I. Guvenc, "Localization of WiFi devices using probe requests captured at unmanned aerial vehicles," in *IEEE Wireless Communications and Networking Conference (WCNC)*.

[20] P. Perazzo, F. Betti Sorbelli, M. Conti, G. Dini, and C. Pinotti, "Drone path planning for secure positioning and secure position verification," *IEEE Transactions on Mobile Computing*, vol. PP, pp. 1–1, 11 2016.

[21] H. Wang, G. Ren, J. Chen, G. Ding, and Y. Yang, "Unmanned aerial vehicle-aided communications: Joint transmit power and trajectory optimization," *IEEE Wireless Communications Letters*, vol. 7, no. 4, pp. 522–525, 2018.

[22] J. Grigulo and L. B. Becker, "Experimenting sensor nodes localization in wsn with uav acting as mobile agent," in *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*, vol. 1, pp. 808–815, 2018.

[23] G. Ding, Q. Wu, L. Zhang, Y. Lin, T. A. Tsiftsis, and Y. Yao, "An amateur drone surveillance system based on the cognitive internet of things," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 29–35, 2018.

[24] F. Koohifar, A. Kumbhar, and I. Guvenc, "Receding horizon multi-UAV cooperative tracking of moving RF source," *IEEE Communications Letters*, vol. 21, pp. 1433–1436, June 2017.

[25] F. Koohifar, I. Guvenc, and M. L. Sichitiu, "Autonomous tracking of intermittent RF source using a UAV swarm," *IEEE Access*, vol. 6, 2018.

[26] L. A. Sanchis, "Experimental analysis of heuristic algorithms for the dominating set problem," *Algorithmica*, vol. 33, pp. 3–18, May 2002.

[27] Z. Yang and Y. Liu, "Quality of trilateration: Confidence-based iterative localization," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, pp. 631–640, May 2010.

[28] J. C. Hateley, H. Wei, and L. Chen, "Fast methods for computing centroidal voronoi tessellations," *Journal of Scientific Computing*, vol. 63, no. 1, pp. 185–212, 2015.

[29] J. Louis Bentley, "Experiments on traveling salesman heuristics.," pp. 91–99, 01 1990.

[30] N. A. Alrajeh, M. Bashir, and B. Shams, "Localization techniques in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 9, no. 6, p. 304628, 2013.

[31] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, 2014.

[32] R. M. Sandoval, A. Garcia-Sanchez, and J. Garcia-Haro, "Improving RSSI-based path-loss models accuracy for critical infrastructures: A smart grid substation case-study," *IEEE Transactions on Industrial Informatics*, vol. 14, pp. 2230–2240, May 2018.

[33] Y. Chapre, P. Mohapatra, S. Jha, and A. Seneviratne, "Received signal strength indicator and its analysis in a typical WLAN system," in *38th Annual IEEE Conference on Local Computer Networks*, pp. 304–307, Oct 2013.

[34] B. J. Dil and P. J. M. Havinga, "RSS-based self-adaptive localization in dynamic environments," in *2012 3rd IEEE International Conference on the Internet of Things*, pp. 55–62, Oct 2012.

[35] Parrot, "Parrot Bebop-2: The lightweight, compact HD video drone." https://www.parrot.com/us/drones/parrot-bebop-2.