Autonomous Unmanned Aerial Vehicle for Search and Rescue using Software Defined Radio*

Seán Óg Murphy[1] , Cormac Sreenan[2] , Kenneth N. Brown[3]

***Abstract*— To find missing people in a remote area, we propose an autonomous unmanned aerial vehicle (UAV) approach which attempts to locate the target by detecting and localising the radio signals produced by a GSM cell phone. By using a low-weight software defined radio and companion computer, the UAV can act as a GSM base station and induce the missing person's device to attempt to make contact. Through the signal strength values and known UAV location, a series of these contact attempts can be used to quickly and accurately localise their position. As the area in which the missing person might be located may be quite large, and the interaction of radio signals with terrain is potentially complex, an efficient search strategy for exploring the area is required in order to reduce time taken to make contact. We make use of a constraint-based graph-based path planning approach to produce a route for the UAV to traverse in the air passing through expected signals from a large number of possible source locations.**

I. Introduction

Every year, hundreds of people go missing while exploring or hiking in national parks, mountain ranges and forests. When its discovered that someone has gone missing in the wild, search and rescue personnel have a limited time-frame to find them as exposure, dehydration or injury increase risk of death. In cases of a person going missing, they may possess a device that supports the common GSM cellphone protocol, though in many cases there may not be GSM infrastructure in place to call for help. Leveraging this GSM capability for radio-signal based localisation could cut down significantly on the time taken to find the missing person.

The rapid development, sophistication and miniaturisation of unmanned aerial vehicle (UAV) technology coupled with similar developments in wireless sensing technology (particularly Software Defined Radio, or "SDR") opens up a new field of autonomous UAV wireless sensing which can detect these GSM signals and assist search and rescue personnel in locating the missing person. We developed a UAV-mounted GSM-based localisation system which allows for rapid and accurate localisation of a signal source (without requiring the missing person to possess any unusual hardware or make use of any custom cell phone software) using a UAV platform constructed from low-cost, off-the-shelf components (Figure 1). GSM is used for this contact as it is the most commonly present communication protocol on cellular telephones, and has high range characteristics. The localisation principles in this work can also extend to other radio-based protocols (LTE, 5G, Wifi etc.) by replacing the GSM base station software on the SDR with alternatives.

The motivation for automation is clear: it is not always feasible for a human operator to remain in control of the UAV (the UAV may need to explore beyond radio range to base, there may be multiple UAVs in use), and optimised UAV mission plans make the most of limited battery (hence flight time), as well as improving the chances of finding the missing person as early as possible. In this work we investigate a UAV mission planning scheme to efficiently explore the search area, by considering the likely propogation of radio waves from a missing device were it in any one of a large number of positions ("candidate locations"). Using a combination of real-world terrain height data and discretised radio propogation models to form three-dimensional navigation graphs, a Constraint Satisfaction Problem (CSP) is constructed which is used to compute a sub-circuit of these graphs that passes through signals produced by each of the candidate locations (a "Covering Tour" [1]) . The sub-circuits we generate allow the UAV to efficiently traverse the terrain and discover signal sources from a large set of possible locations. Once a signal is discovered, we employ a non-linear, curve-fitting algorithm to home in on the source location and produce an accurate localisation quickly.

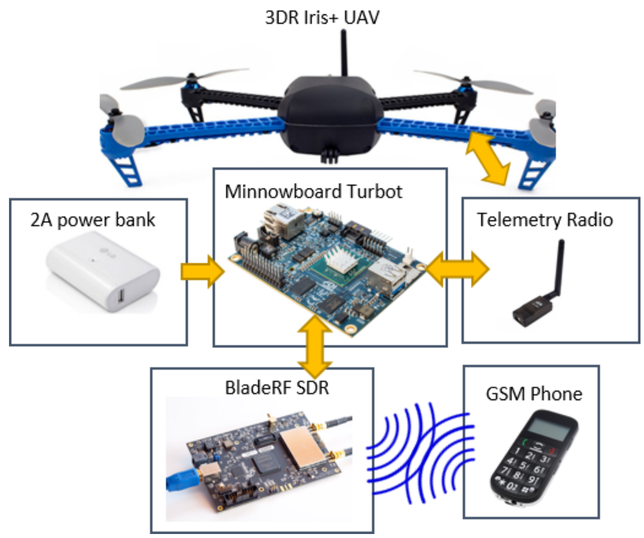


Fig. 1. UAV Search and Localisation Platform hardware

[1]seanog.murphy@ucc.ie Mobile & Internet Systems Laboratory, Dept. of Computer Science, University College Cork
[2] cjs@cs.ucc.ie Mobile & Internet Systems Laboratory, Dept. of Computer Science, University College Cork
[3]k.brown@cs.ucc.ie Insight Centre for Data Analytics, Dept. of Computer Science, University College Cork

*This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) and is co-funded under the European Regional Development Fund under Grant Numbers 10/CE/I1853, 13/RC/2077 and 12/RC/2289

II. RELATED WORK

The increasing capabilities of UAVs and embedded computers have led to a number of new search and rescue solutions exploiting automated UAV flight capability and sophisticated sensing. Most UAV-based search and rescue uses cameras attached to the UAV [2], with wireless data video transmission to an operator on the ground; at first using human operators and more recently using automated planning for flight or real-time response to sensed data. The strategy of exploiting radio signals for localising persons in distress has led to solutions that determine position based on Wi-Fi probe beacons [3] or make use of a user's smartphone with an installed custom app which periodically generates a wireless signal for detection [4]. Carpin et al. [5] use real-time updating of probabilistic utility during fixed wing UAV flight to systematically discover a WiFi signal source. Most of these solutions require human control, and complex real-time computation during the mission using ground-based resources (and hence necessitating real-time communication with the UAV in the air). Baker et al. [6] and Lin et al. [7], deploy video search, and analyse the terrain in a pre-computation stage, which subsequently forms the basis of a graph routing problem. where the aim is to generate space-filling coverage paths.

The class of problems of finding paths or tours in a connected graph of locations is a well-studied topic in Operations Research. Two examples include the Covering Tour problem [1], in which we search for a partial tour (i.e. visiting a subset) of a group of points, such that a set of targets are all within a specified distance of the tour path, and the Orienteering Problem (OP) [8], in which we search for a limited-length partial tour that maximises a reward composed from independent rewards of each node in the tour. Neither of these approaches account for the non-uniform coverage of wireless propagation. Constraint programming [9] is a general technique for combinatorial problem solving and optimisation, which is not limited to linear constraints, and has had particular success in solving routing problems [10]. Li and Xu [11] explored optimised placement of UAVs to form efficient aerial networks, in this work determining *stationary* locations of the UAVs that maximise connectivity and throughput. Ghazzai et al [12] explored using particle swarm and k-means approaches for determining placement of UAVs pre-deployment, so that for an emergency any geographic location within the area, a UAV is within flight range of it; this approach would combine well with our search techniques to ensure that for large geographic areas, UAVs are well-placed to begin a search mission if required.

III. AUTONOMOUS UAV LOCALISATION PLATFORM

A. Hardware

1) Unmanned Aerial Vehicle: We use a 3DRobotics Iris+ Quadcopter UAV (Figure 1) . This UAV features a Pixhawk 4 flight computer which supports MAVLink (Section III-B.1) protocol control. The UAV can carry up to 400 grams in extra equipment and has a flight time of approximately 30 minutes on a full charge. Control of the UAV is performed through MAVLink commands via 433MHz telemetry radio connection.

2) Software Defined Radio: The BladeRF x40 (Figure 1) is a low-cost, light-weight Software Defined Radio produced by Nuand, with an operating range between 300MHz and 3.8GHz. Using YateBTS (Section III-B.2), the BladeRF can be configured to operate as a low-power mobile cellphone Base Transceiver Station (BTS). In these experiments, the BladeRF uses a pair of Vert900 antennae, for a total weight of 120g, making it well-suited to the limited payload capacity of the Iris+ UAV.

3) Embedded Companion Computer: While the Iris+ UAV has a flight computer, it isn't powerful enough to perform complex computation or to manage an SDR. A "Companion Computer" communicates with the flight computer via a MAVlink wireless connection, and is responsible for managing the radio search aspects of the platform, including operating the software BTS and localising user devices based on Received Signal Strength Indication (RSSI). In this work we use a Minnowboard Turbot (Figure 1) due to its light weight (45g), low cost, USB 3.0 support and effective computation capability.

B. Software and Protocols

1) MAVLink: MAVLink provides for communicating flight commands to the UAV Pixhawk flight computer via 433MHz telemetry radio connection. The Companion Computer requests the UAVs current geographical position, altitude, velocity and other metrics. Through this connection, the Companion Computer also provides waypoint destinations for the UAV to visit. Through these communications, we associate radio readings with geographic positions, and use these for localisation.

2) YateBTS: Using the BladeRF as a transceiver, we implement the GSM stack using Yate as the telephony engine and YateBTS for wireless communications and subscriber management. By appearing to be a more powerful Base Transceiver Station, user devices are induced into attempting to associate with the aerial BTS. While the device is not accepted as a subscriber (to ensure it remains available on its original network), this is sufficient to establish an RSSI reading for the cellphone. As each cellphone has a unique International Mobile Equipment Identity (IMEI) number, specific devices can be tracked independently, or a particular device can be targeted in the case that the IMEI of the missing person is known ahead of time.

IV. DATA SOURCES

The data sources used to produce the Constraint Problem and navigation guidance are three-dimensional terrain heightmaps representing the physical space, and Longley-Rice radio propagation models used to characterise the spread and strength of radio signals produced at candidate locations.

A. Terrain Heightmaps

The Shuttle Radar Topography Mission was a radar-based Space Shuttle survey of the Earth's surface conducted in 2000. This survey produced terrain elevation maps ("heightmaps") at a 30 * 30 metre resolution for most of the surface of the earth. These maps are used in this work to determine the straight-line traversability between graph nodes in the Low-Density Planning Graph (Section V-A), and are also input for Radio Path-loss model generation (Section IV-B).

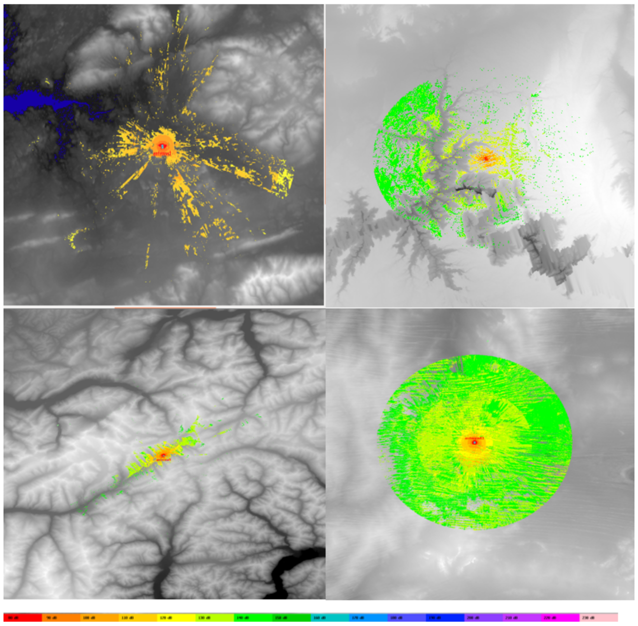


Fig. 2. Example propagation maps for Cork, Grand Canyon, Swiss Alps, South Australia

B. Pre-computed Radio Path-loss models

Pre-computed path-loss models form the basis for the Covering Tour problem to be solved. These models are generated offline using the SPLAT! "Signal Propagation, Loss, And Terrain" tool [13], with each model representing the expected radio coverage area and signal strengths produced by a cell phone located in a particular location (candidate source location), thus representing the distribution of expected signals the UAV+SDR platform would receive were they to traverse that space. These maps are generated for recieving locations from 100 metres up to 600 metres above the ground, at 100 metre intervals. Samples are taken from these maps at node positions to produce the expected signal values from candidate sources were the UAV to visit that node, forming the basis for the determination of "covering power" in the Planning Graph (Section V-A).

V. Covering Tour

A. Planning Graph

The first step is to find a radio signal from the target device. In Section IV-B, we used path-loss models to compute expected signal strengths at various locations (nodes) in space for different candidate locations of the device on the ground. Our first aim is to move around those nodes until we find a signal. First, we form a planning graph, and then we compute the paths through the graph. The traversal plan takes the form of a circuit on a 3-dimensional graph ("Planning Graph"). This graph's bottom layer consists of a grid of nodes spaced apart at 300 metre intervals, with each node positioned 100 metres above the altitude of the terrain directly below it. On top of this layer are four more graph layers each 100 metres above the previous layer, giving a stacked three-dimensional space for consideration when traversing the terrain. Each node in this graph has a "covering" power associated with it, this indicates which candidate sources would be audible were the UAV at that node's position, as determined by the radio propagation models (Section IV-B) – for example; a single node might "cover" candidate locations 5, 11, 12 and 22.

B. Covering Tour Constraints

The problem to be solved is finding a partial tour in a graph in which the set of all visited nodes provides full coverage of the candidate locations (i.e. all potential radio signal sources would be in range if the UAV follows the path). Formally, given a graph (V, E) where $V = \{v_1, v_2, \ldots, v_n\}$ are the vertices and E is a set of weighted directed edges (ordered pairs of vertices with an associated cost), a set $L = \{\lambda_1, \lambda_2, \ldots, \lambda_m\}$ of locations, and an $n \times m$ boolean matrix specifying which vertices cover which locations, find a cycle $v_{i1}, v_{i2}, \ldots, v_{ik} = v_{i1}$ such that every λ_j is covered by at least one v_{ip}, and the path cost of the cycle is minimised. We model the problem using the Choco 4 Constraint Programming Java library [14], with a complete navigation graph, as described below.

C. Variables and Constants

$costs[i][j]$ - the (constant) travel time between each pair of vertices (i, j), with values computed from the low-level planning graph

$covers[i][j]$ - (constant) booleans stating whether vertex i covers location j, with value T derived from the SPLAT! maps if the predicted signal strength at i is above a threshold

$tour[i]$ - integer variables stating the next vertex after i in tour, with $tour[i] = i$ if i is not in tour

$vertex[i]$ - boolean variables, True if vertex i is visited

$edge[i][j]$ - boolean variables stating whether edge (i, j) is traversed in the tour

$covered[i]$ - boolean variables, True if location i is covered

$notCovered$ - integer variable stating the number of locations not covered

$length$ - integer variable for the number of visited vertices

$cost[i]$ - integer variables stating the cost of the tour edge departing from i

$totalCost$ - integer variable stating cost of tour

D. Constraints

$subcircuit(tour, length)$ - the successors in the tour form a subcircuit of length vertices

$\forall i \in vertices : channel(edge, tour)$ - $edge[i][j]$ is True if and only if $tour[i] == j$

$\forall i \in vertices : vertex[i] > 0ifandonlyiftour[i] \neq i$ - vertices in the tour must have departing edges

$\forall \lambda \in L : \sum_{v \in V} covers[v][\lambda] * vertices[v] \geq covered[\lambda]$ - a location is covered only if at least one of its covering vertices are visited

$\sum_{\lambda \in L} covered = m$ - all locations must be covered

$\forall e = ((i, j), c) \in Eedge[i][j] * c = cost[i]$ - the departure cost from a vertex is the cost of the traversed edge

$\sum cost = totalCost$ - total cost sums the selected edge costs

Objective: $Minimise(cost)$

VI. Localisation

A. Levenberg-Marquardt Localisation

As radio propogation is inherently variable, and scattering and shadowing effects can have a significant impact on the utility of RSSI as a distance metric, trilateration of a user device using linear approaches yields ambiguous results. The Levenberg Marquardt (L-M) [15] algorithm is a non-linear least-squares optimisation algorithm. To tune the parameters of our localisation and investigate a number of possible search motion patterns before field tests, we used Longley-Rice path-loss coverage maps generated in SPLAT! in software simulation. Based on simulation results, we arrived at a Gaussian sampling approach and a spiral movement pattern for searching in the vicinity of the expected location and use this approach in outdoor flight experiments.

We select a random sampling of six readings (biased towards choosing readings with stronger signals) and pass these location and signal strength pairs to the L-M algorithm to compute a close-matching model, on which we locate the optimum and consider the coordinates of the optimum to be the current candidate for cellphone location. The UAV then travels towards the estimated location, slightly offset in angle by 20° to produce a spiraling-in motion. As further readings are made and the estimate improves, the squared difference (error) between the actual readings and the L-M generated curve reduces, and a sequence of estimates that are consistently below a threshold constitutes the search termination criteria. The UAV returns home to inform the rescue team of the estimated location of the user device.

VII. Empirical Evaluation

A. Field Testing

To determine the characteristics and confirm viability of the UAV search and locate platform, we conducted field trials using the completed system. In these practical experiments, we configure YateBTS to operate in a low power broadcast mode, limiting the transmission range to 100 metres to allow for testing of localisation algorithms in a limited area. In these experiments, the practical localisation capacity using the Levenberg-Marquardt approach was to within 1m, or 1% of the transmission range. As the practical limit of consumer-level GPS localisation (i.e. available to the UAV) is 1m, this demonstrates the powerful localisation ability of this approach.

B. Simulation Testing

To facilitate algorithm experimentation and evaluating a variety of terrain types, we developed a software simulator for conducting off-line search and localisation experiments (Section VII). This simulation features a simulated UAV which travels around a search area, with the signals it may hear in any one longitude and latitude location position provided by a set of 5 “ground truth” SPLAT path loss maps computed for varying altidudes above that point (at 100 metre intervals). This simulation approach allows us to practically experiment with radio models in a variety of terrain types, which is of great benefit in evaluating the performance and viability of our search and localisation strategies in a greater variety of circumstances. To evaluate our search and localisation strategies, we conducted simulations for four areas: rural Cork (the area where we conduct field trials), the South Australia Outback, the Grand Canyon in Arizona, USA, and the Swiss Alps (as in Figure 2). For each area, we generated 81 terrain path loss models (in 9 by 9 grid of device locations each 1km apart from their neighbours) for a mix of terrain types. The maximum transmission range for the device in these experiments is approximately 5km (determined by the transmitting power levels of the SDR and the simulated user device), and the starting position of the UAV is 10km from the top-left candidate location in each set. In these experiments, two metrics are collected: the time to get a signal (Figure 3), and the time taken after the first signal to localise the device within a 10% confidence threshold (Figure 4). For each area we select one of the 81 candidate locations and remove it from the set: this location acts as the ground truth, with the other locations contributing to the Covering Tour optimisation (Section V used by the UAV in the search phase. Each experiment is repeated, using a different candidate location as ground truth each time.

C. Fixed, Greedy and Covering Tour Strategies

To evaluate our approach, we compared against two other strategies. The Fixed strategy traces an X-shaped path followed by a 10-point star path, making a 72° turn to the right each 20km). The shape of the Fixed Path was chosen to give a favourable combination of early-detection (provided by the initial X-shaped path) with total coverage (provided by the star shape). The Greedy approach directs the UAV as it traverses the search area, selecting the node covering the most un-visited locations at each stage. These two approaches do not consider the variation in terrain: in both cases the UAV travels at 100 metres above the ground.

D. Search Time Results

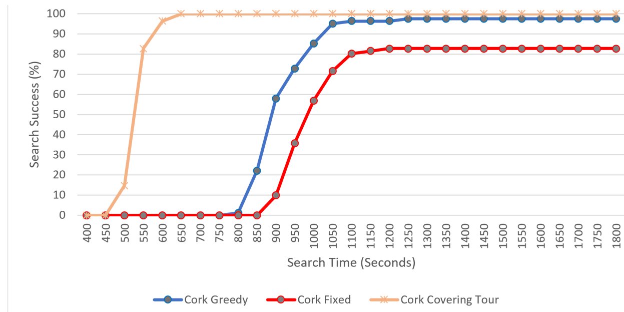
The first point to note is that the constraint model, although slower than the other two approaches, computes optimal tours in under 8 minutes in each case. Since these tours can be precomputed before flight, this is sufficiently fast for practical deployment, and so computation time is not considered further. The results are shown in Figure 3. The covering tour tends to discover signals earlier than the other approaches, as the path avoids excess elevation change, and exploits reception of radio signals at varying altitudes. In contrast, the Fixed and Greedy approaches suffer from inability to plan ahead around changes in terrain elevation, and do not exploit space higher than 100 metres above the ground surface. The Greedy approach performs reasonably well, discovering the signal early enough in the mission to allow for localisation within the battery life of the UAV. However the Greedy approach suffers from an inability to plan: it loses considerable time navigating the Grand Canyon, and the route the UAV takes enters and leaves the canyon area several times. In contrast, the Covering Tour approach enters the canyon far fewer times, instead hopping across the canyon at altitude when possible (i.e. once any signal sources inside the canyon have been “visited” on the covering tour, there is no more need to check inside the canyon itself. Despite this, a very small number of candidate locations are not heard from during the Covering Tour of the Grand Canyon map within the mission lifetime, though substantially more are discovered than in the Greedy or Fixed approaches. The impact of elevation change is also clearly visible in the results for the Swiss Alps; here the Partial Tour is able to more consistently stay at a fixed altitude where the Greedy and Fixed approaches follow the contours of the terrain more closely and lead to unneccessary altitude change while navigating the mountainous terrain. The Partial Tour solutions find a signal in the majority of simulations well within the 30-minute flight time limit of the reference hardware, demonstrating its practical use for search and rescue applications.

E. Nonlinear Curve-fitting Localisation Results

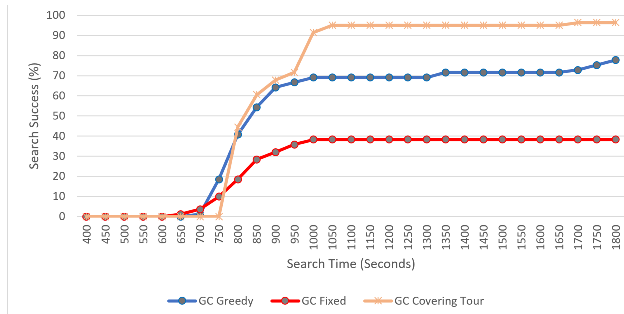
For the localisation phase (Figure 4), we find a range of times, generally in the 5-7 minute range. Here we observe consistent, quick localisation times for the Cork terrain, due to the long distance of radio propagation coupled with the irregular shapes of propagation. A small number of early signals are quickly matched to the unusual curve shape and a confident estimate is made early. For the Alps, the localisation generally takes slightly longer than with Cork, but as its radio characteristics are similarly irregular, the localisation is completed quickly. A small number (10%) of ground truth locations were not localised for a considerable time, a result of the high variance in terrain and often small coverage area of radio signals. For Australia, the localisation times follow are more elongated, but regular curve, a smooth range of results due to the large and consistent radio spread of the flat desert terrain. Again, the Grand Canyon results feature a combination of features from the Alps and Australia terrain. As with Australia, most positions are located in the early period, with an elongated curve due to many positions having regular, radial patterns. As with the Alps, a small number of positions are difficult to locate, due to the small size and irregular shape of some radio sources in confined areas on the canyon floor.

VIII. Conclusions

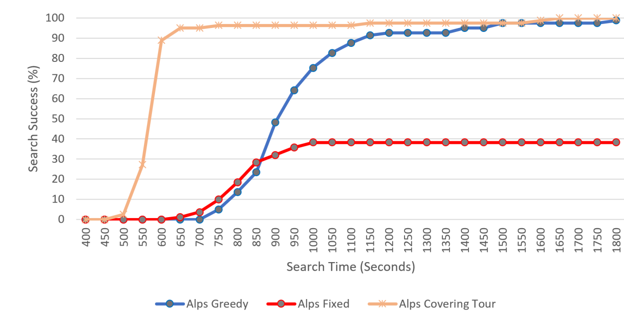
In this work we presented a fully autonomous aerial platform for localising cellphone devices of users in distress, using low-cost off-the-shelf parts. Through the use of an intelligent search strategy accounting for radio propogation characteristics, we can discover a device signal in a short time, and using a powerful curve-fitting approach the device location is identified accurately soon after, all autonomously without requiring human oversight. We developed a terrain-aware Covering Tour solution for discovering a signal source in remote terrain; the basis for autonomous path-planning for efficient coverage of a search area by a radio-equipped UAV. In our experiments, we demonstrated the successful performance of these plans for discovering a signal source in a variety of complex terrains. In future work, we will investigate the coordination of teams of drones, allocating plans between drones and facilitating rendezvous communication during the search for coordinating localisation efforts.



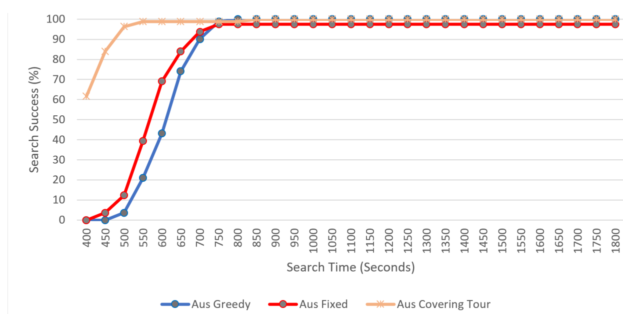
(a) Cork



(b) Grand Canyon



(c) Swiss Alps



(d) South Australia

Fig. 3. Fixed, Greedy, and Covering Tour Experiment Simulation Results

References

[1] M. Gendreau, G. Laporte, and F. Semet, "The covering tour problem," *Operations Research*, vol. 45, no. 4, pp. 568–576, 1997.

[2] M. B. Bejiga, A. Zeggada, A. Nouffidj, and F. Melgani, "A convolutional neural network approach for assisting avalanche search and rescue operations with UAV imagery," *Remote Sensing*, vol. 9, no. 2, p. 100, 2017.

[3] Y.-H. Ho, Y.-R. Chen, and L.-J. Chen, "Krypto: Assisting search and rescue operations using wi-fi signal with UAV," in *Proceedings of the First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use*, pp. 3–8, ACM, 2015.

[4] V. Wolfe, W. Frobe, V. Shrinivasan, and T.-Y. Hsieh, "Detecting and locating cell phone signals from avalanche victims using unmanned aerial vehicles," in *Unmanned Aircraft Systems (ICUAS), 2015 International Conference on*, pp. 704–713, IEEE, 2015.

[5] M. Carpin, S. Rosati, M. E. Khan, and B. Rimoldi, "UAVs using bayesian optimization to locate wifi devices," *arXiv preprint arXiv:1510.03592*, 2015.

[6] C. Baker, G. Ramchurn, L. Teacy, and N. Jennings, "Planning search and

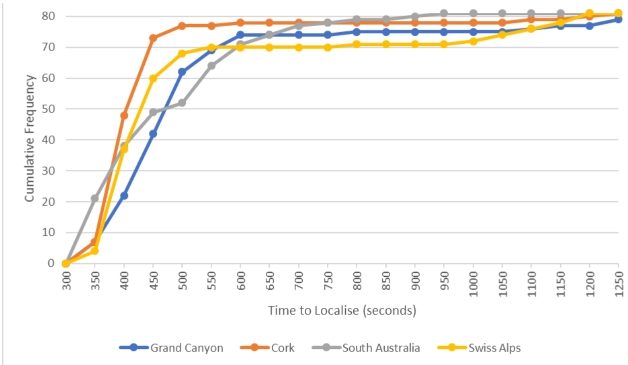


Fig. 4. Simulation localisation time for Grand Canyon, Cork, South Australia, Swiss Alps

rescue missions for UAV teams," 2016.

[7] L. Lin and M. A. Goodrich, "UAV intelligent path planning for wilderness search and rescue," in *Intelligent robots and systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pp. 709–714, IEEE, 2009.

[8] B. L. Golden, L. Levy, and R. Vohra, "The orienteering problem," *Naval research logistics*, vol. 34, no. 3, pp. 307–318, 1987.

[9] F. Rossi, P. van beek, and T. Walsh, eds., *Handbook of Constraint Programming*. Elsevier, 2006.

[10] B. de Backer, V. Furnon, P. Kilby, P. Prosser, and P. Shaw, "Solving vehicle routing problems using constraint programming and metaheuristics," *Journal of Heuristics*, vol. 1, pp. 1–16, 1997.

[11] P. Li and J. Xu, "Placement optimization for UAV-enabled wireless networks with multi-hop backhaul," *arXiv preprint arXiv:1809.04963*, 2018.

[12] H. Ghazzai, H. Menouar, and A. Kadri, "On the placement of UAV docking stations for future intelligent transportation systems," in *Vehicular Technology Conference (VTC Spring), 2017 IEEE 85th*, pp. 1–6, IEEE, 2017.

[13] "Splat! a terrestrial RF path analysis application for Linux/Unix," 2018. http://www.qsl.net/kd2bd/splat.html [Accessed: 26/02/2018].

[14] "Choco solver," 2018. http://www.choco-solver.org/ [Accessed: 26/02/2018].

[15] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.