Ad Hoc Networks 73 (2018) 95–105

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Quality-aware UAV coverage and path planning in geometrically complex environments

Rui Dai*, Sneha Fotedar, Mohammadreza Radmanesh, Manish Kumar

University of Cincinnati, Cincinnati, OH 45221, United States

ARTICLE INFO

Article history:
Received 4 June 2017
Revised 22 December 2017
Accepted 26 February 2018
Available online 28 February 2018

Keywords:
Unmanned aerial vehicle networks
Coverage
Energy efficiency

ABSTRACT

Networks of unmanned aerial vehicles (UAVs), capable of providing flexible aerial views over large areas, are playing important roles in today's distributed sensing systems. Since camera sensors are sensitive to occlusions, it is more challenging for UAVs to provide satisfactory sensing quality in geometrically complex environments, such as dense urban areas and mountainous terrains. This paper proposes a new quality-aware and energy-efficient UAV coverage and path planning scheme with the objective of sensing a geometrically complex target area with satisfactory spatial and temporal resolutions. An occlusion-aware waypoint generation algorithm is first designed to find the best set of waypoints for taking pictures in a target area to satisfy the spatial resolution requirement. The selected waypoints are then assigned to multiple UAVs by solving a vehicle routing problem (VRP) such that all the waypoints are visited within a global deadline to satisfy the temporal resolution requirement. The vehicle routing problem is formulated to minimize the maximum energy for the UAVs to travel through the waypoints within the deadline. A Min-Max energy path planning algorithm is designed to solve this problem in two steps: first, a mixed integer linear programming problem (MILP) is solved to calculate the minimum energy for a UAV to go from one waypoint to other; then, a genetic algorithm is devised to plan the paths for all the UAVs. Evaluation results show that the proposed coverage and path planning scheme results in better coverage and energy consumption than traditional coverage and path planning techniques for UAVs.

1. Introduction

The advent of small Unmanned Aerial Vehicles (UAVs) has enhanced many applications such as remote sensing [1], surveillance [2], border patrol [3], disaster management [4], and wildlife and forest monitoring [5]. Deploying a network of small UAVs for an application can not only reduce the cost but also improve the scalability and quality of sensing than single UAV systems. Networks of UAVs, also known as flying ad-hoc networks or aerial sensor networks, have drawn the attention of many researchers over the recent years [6].

For a wide range of sensing applications, it is essential to achieve full coverage of a target area with satisfactory sensing quality. The problem of coverage for static wireless sensor networks has been extensively studied [7,8]. There have also been many studies on introducing mobile nodes in sensor networks to improve network coverage [9,10]. The problem of coverage in UAV networks, however, requires new considerations due to the high mobility of UAVs, the limited battery support for flying, and the fact that obstacles may affect the coverage of camera sensors on UAVs [11].

The dynamic coverage problem for multiple UAVs is generally approached by two steps: area partitioning where a given area is divided into subareas, and path planning for individual UAVs inside each sub area [11]. This type of approach focuses on addressing the 2D geometrical constraints of a target area. For example, the optimal line-sweep-based decomposition method in [12] divides a coverage region into monotone sub regions which can be efficiently covered by basic back and forth motion along rows perpendicular to the sweep direction. In [13], several geometric models are summarized for path planning, including spiral/spiral-like model, Lawnmower model, Zamboni model, and Dubins path model, and a modified Lawnmower/Zamboni path planning strategy is introduced to accommodate for features that may be required by different missions. In [14], based on a 2D digital map, a genetic algorithm is used to generate paths for UAVs in the existence of unknown obstacle environments.

Many studies on UAV coverage assume that the target area has an ideal flat terrain. In reality, all kinds of obstacles may occlude a camera's field of view. For applications such as smart cities and

* Corresponding author.
E-mail addresses: rui.dai@uc.edu (R. Dai), fotedasa@mail.uc.edu (S. Fotedar), radmanma@mail.uc.edu (M. Radmanesh), kumarmu@ucmail.uc.edu (M. Kumar).

https://doi.org/10.1016/j.adhoc.2018.02.008
1570-8705/

wild fire monitoring, the field of views of cameras can be reduced by buildings in dense urban areas or mountains and rocks in rural areas. This indicates that, to achieve full coverage of all the spots inside a target area, more images should be taken than in an ideal flat terrain case. Furthermore, to meet with sensing resolution requirements, the UAVs should be able to vary its flying height to take images, which demands for a new look at path planning in the 3D domain. The problem of occlusion-aware multi-UAV surveillance is addressed in [15], and the focus of this work is on point coverage, i.e., to cover certain points of interest as often as possible. It does not address the overall area coverage quality nor the spatial and temporal resolutions for sensing.

For the path planning of UAVs, a few studies have jointly considered energy consumption and sensing quality in terms of spatial resolution. An empirical energy model is derived from flight experiments under different speed and acceleration and deceleration parameters in [16], and based on this model, a path planning algorithm is designed to minimize energy consumption while satisfying mission requirements. In [17], a set of sensing points for generating an overview image are found by minimizing the costs in energy and communication bandwidth. However, this approach is designed for single UAV systems and it does not address the path planning problem of multiple UAVs.

The problem of path planning within time deadlines has also been a topic of interest, where the deadline can reflect the temporal resolution of sensing. This problem is typically known as Vehicle Routing Problem with Time Deadline (VRPTD) or Vehicle Routing Problem with Time Windows (VRPTW). VRPTD in Geo-spatial Information Systems is solved in [18] using a routing scheme based on a genetic algorithm. The scheme focuses on minimizing the number of vehicles and the distance between them while satisfying the time deadline. Although the approach is successful in solving the VRPTD, it does not consider the energy consumption of the vehicles. In [19], a combinatorics problem called Swarm Routing Problem (SRP) is studied, which is described as a variant of VRPTW. The problem is formulated to solve multi-UAV routing while meeting constraints on mission time and optimizing path cost using evolutionary computation. To solve the problem of optimality and to incorporate complex timing constraints, Karaman and Frazzoli [20] presents a MILP (Mixed Interger Linear Programing) based algorithm for multi-UAV misson planning. For the optimization problem of minimum time coverage for multiple UAVs, the solution in [21] provides a way to automatically select the number of UAVs used to cover an area. A path planning algorithm for time sensitive cooperative surveillance is proposed considering the maneuver limits of UAVs [22]. These methods were successful in solving the path planning problem with time constraints for multi-UAVs; however, the joint consideration of occlusion-aware coverage along with path planning is missing.

In this paper, we propose a sensing quality-aware coverage and path planning solution for UAV networks for the monitoring of geometrically complex scenes with varying altitudes and occlusions. The goal is to provide overview images for a target area with satisfactory spatial and temporal resolutions under the limited energy supply of UAVs. To achieve this, we first introduce a camera sensing model that considers the effect of occlusions. Based on the sensing model, we find the best set of waypoints for taking pictures in a target area to achieve full coverage with satisfactory spatial resolution. Next, we formulate a vehicle routing problem that finds the paths of multiple UAVs to visit all the waypoints within a global deadline such that the energy consumption for UAV flight is minimized. In our preliminary work [23], we used a simple point mass energy model for the UAV flight and planned the paths for UAVs without considering any time constraint. In this paper, we extend our preliminary work by using a more comprehensive energy model and considering the time constraints for UAV flight, and we propose a new technique to solve the VRPTW optimization problem.

The rest of this paper is organized as follows. In Section 2, we introduce the proposed quality-aware coverage and path planning technique for geometrically complex scenes. Performance evaluation results and discussion are presented in Section 3, and Section 4 concludes the paper.

2. Quality-aware coverage and path planning considering camera occlusion

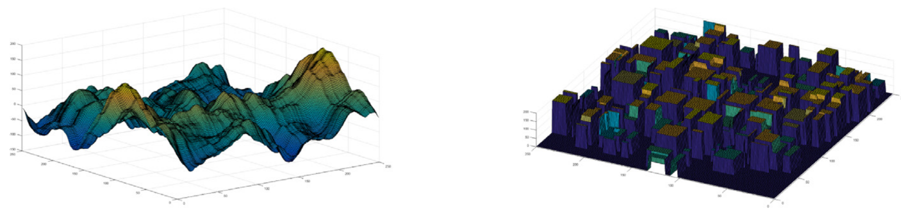
We consider the problem of dynamic UAV coverage for a target area, where each UAV starts from the ground control station (GCS) and move forward to visit a set of pre-defined waypoints. The UAVs keep flying during the mission and take images at these waypoints. After visiting all of its waypoints, the UAVs fly back to the GCS. We consider a periodic sensing scenario: The UAVs repeat the above steps, and by taking images at the waypoints in each round, they provide periodic observations of the target area. For typical distributed sensing applications such as remote monitoring and surveillance, the UAV network is required to provide overview images of the area with satisfactory sensing quality in terms of spatial and temporal resolution.

To achieve full coverage of the area with satisfactory spatial resolution, the effect of obstacles has to be taken into consideration. As an example, Fig. 1(a) presents a non-homogeneous mountainous terrain, and Fig. 3(b) shows the sensing process of a camera sensor on a UAV over this terrain. The camera can only sense objects within its line of sight. The shaded area in Fig. 3(b) is not covered by the camera due to occlusion. Furthermore, the quality of sensing varies for objects at different altitudes. The camera will provide better spatial resolution for an object at the top of the mountain than an object at the bottom of the mountain in Fig. 3(b). In addition, to provide observations of the target area with satisfactory temporal resolution, the UAVs should travel through the waypoints with time constraints. In this paper, we propose a quality-aware coverage solution consisting of the following three components.

1. *Coverage and sensing quality estimator.* Given an altitude map of a target area and the location and sensing parameters of a camera, this component estimates which portion of the area can be observed by the camera with a user-defined spatial resolution requirement.
2. *Waypoint generation algorithm.* Generate a set of waypoints for the UAVs to visit. The algorithm aims to use the minimum number of waypoints to fully cover the target area with satisfactory spatial resolution. By minimizing the number of waypoints for taking pictures, the communication bandwidth for transmitting the images could be saved.
3. *Quality-aware path planning.* A group of UAVs will be scheduled to visit the generated waypoints, take pictures, and report back to GCS within a global deadline, where the purpose to define a deadline is to satisfy the temporal resolution requirement of the sensing application. We formulate a problem to minimize the maximum energy consumption among the UAVs to achieve energy efficiency. We use the mixed integer linear programming technique and a genetic algorithm to solve this problem and generate routes for the group of UAVs.

2.1. Coverage and sensing quality estimator

A camera sensor commonly equipped on a UAV is a directional sensor that can sense objects within its angle of view (AoV). The pinhole camera model in computer vision [24] can be used to describe the sensing process of a camera, which is characterized by



(a) Mountainous terrain (b) Urban area

Fig. 1. Target areas.

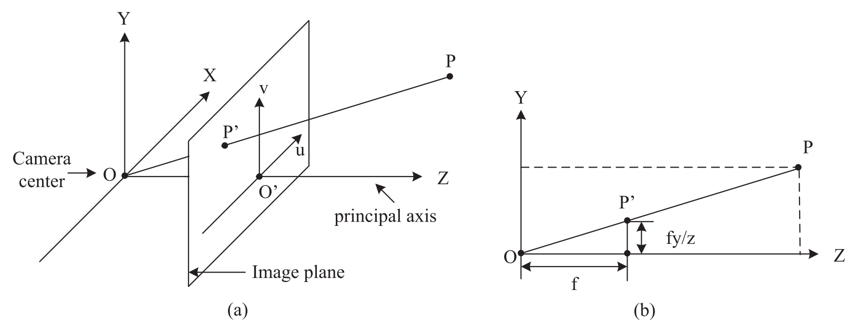
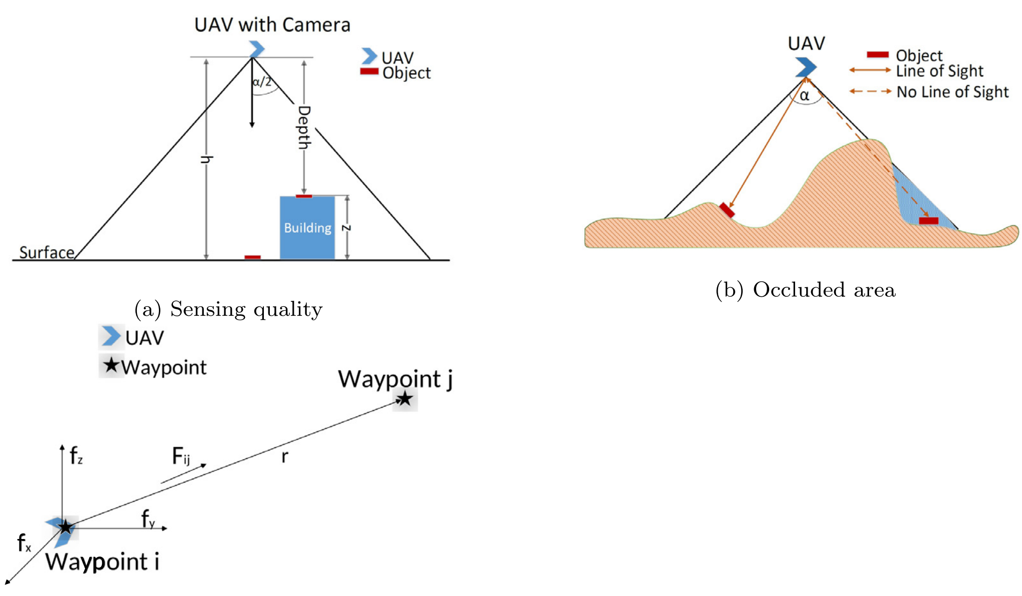


Fig. 2. Camera projection model.



(a) Sensing quality

(b) Occluded area

(c) Net Force

Fig. 3. Sensing model.

3-D to 2-D projection. The same model has also been applied in our previous work in multimedia sensor networks [25]. Fig. 2 illustrates an example of a pinhole camera. The camera's center of projection is at the origin of a Euclidean coordinate system, and its sensing direction is along the z axis. The focal length of the camera is f. A scene point P with coordinates $(x, y, z)^T$ is mapped to $P'(u, v)^T$ on the image plane, where u and v are given by

$$\begin{cases} u = fx/z \\ v = fy/z \end{cases} \tag{1}$$

Here z is the depth of the scene point. Suppose the focal length f is given in the number of pixels. Based on this model, a small object located at depth d will be projected on the image plane with a spatial resolution (pixel/meter) of

$$S_r = \frac{f}{d} \tag{2}$$

When a UAV is deployed for taking pictures, the spatial resolution for sensing an object is dependent on both the camera's sensing parameters and the locations of the camera and the object. We consider the case that the camera's sensing direction is perpendicular to the ground (which could be achieved through a camera stabilizer). The UAV is deployed at a height of h, and the AoV of the camera is α. As shown in Fig. 3(a), two objects are within the camera's sensing range, but the sensing resolutions for them are different. The object on the ground results in a depth of h and a spatial resolution of f/h, while the object on the top of the building has a depth of $h - z$ and a spatial resolution of $f/(h - z)$, where z is the height of the building.

To provide an overview image with satisfactory spatial resolution over a target area, a UAV application typically specifies a minimum spatial resolution requirement S_{r_min} for the entire area. The neighboring images taken by the UAV should also have a minimum percentage of overlap (o) with each other so that they can be stitched together to provide an overview image. To satisfy this condition, we define an adjusted AoV $\tilde{\alpha}$, where $\tilde{\alpha} = \alpha \cdot (1 - o)$, and we use this adjusted value to estimate whether or not an area is within the camera's AoV. Furthermore, for a geometrically complex terrain, parts of the terrain might be blocked due to occlusion. As shown in Fig. 3(b), the shaded area is not covered by the camera. We define that a block on the ground plane can be observed by the camera only when there are no obstacles between them.

Suppose a UAV stays at a location of (x_u, y_u, h) where h is its flying height. To determine if a small block on the ground with coordinates (x, y, z) (z is the altitude of the block) can be covered with the minimum spatial resolution, the following three conditions should be true at the same time: i) The block is within the camera's adjusted AoV; ii) there is a direct line of sight between the camera and the block (no occlusion); and iii) the spatial resolution $S_r = f/(h - z)$ should not be smaller than S_{r_min}, which indicates that the height of the UAV cannot exceed $f/S_{r\_min} + z$. If any of these conditions cannot be satisfied, the block is not covered by the UAV with the minimum spatial resolution.

2.2. Waypoint generation

Based on the sensing model above, we propose an algorithm to generate a set of waypoints for UAVs to visit and take pictures. Given a target area with known altitude information, our goal is to minimize the number of waypoints to reduce the communication load while providing an overview image of the entire area with satisfactory spatial resolution. Suppose that the altitude information of the area is given in terms of small blocks $B = \{b_1, \ldots, b_i, \ldots, b_n\}$, where each block has a 3D coordinate of (x_i, y_i, z_i) with z_i denoting the altitude. While it is possible to generate a waypoint on top of each block with different altitudes, to achieve computational efficiency, we can sample the 2D coordinates on the ground plane with a step size of n_s, i.e., we sample every n_s block along x and y axes on the ground plane. Based on the analysis in the last subsection, at a sampled 2D location of (x_i, y_i), the maximum height for a UAV directly above this location should be $h_i^{\max} = f/S_{r\_min} + z_i$. According to regulations of UAV flight, there should also be a minimum height that the UAV can fly above this area, and we denote this minimum height by $h_i^{\min}$.

The problem of generating waypoints can now be solved in two major steps. The first step is to find the best height h_i for sensing on top of each sampled block with a 2D location of (x_i, y_i), such that a UAV at this height can cover the largest number of blocks. We define S_i as the cover set that results from the best height h_i for block (x_i, y_i). In the second step, the best set of waypoints could be found by solving the following set covering linear programming problem:

$$\begin{aligned} \text{MIN} \quad & \sum_{i \in S} X_i \\ s.t \quad & \sum_{i: e \in S} X_i \geq 1, \quad e \in B \\ & X_i \in \{0, 1\}, \qquad i \in S \end{aligned} \tag{3}$$

where X_i is 1 if S_i is in the solution set, and 0 otherwise. B is the entire set of blocks for the target area, S is the set of waypoints, and C presents the elements of B already covered by elements in S. The constraint ensures that each element of B is covered by at least one waypoint. The aim of the objective function is to minimize the amount of waypoints needed to cover the area.

Algorithm 1 Waypoint generation algorithm.

1: Input: B, S, $C \leftarrow \emptyset$
2: $S = B$, C
3: sample the area of small blocks $\{b_1, b_2, \ldots, b_n\}$ with a step size of n_s
4: **for** `<each sampled 2D location of` (x_i, y_i)`>` **do**
5: Find candidate heights $\{h_i^j\}$ within range $[h_i^{\min}, h_i^{\max}]$ using a step size of Δh
6: **for** `<each` h_i^j`>` **do**
7: Find the cover set $S(h_i^j)$ for h_i^j
8: **end for**
9: Determine the best cover set S_i with height h_i for (x_i, y_i), where $S_i = \max\{S(h_i^j)\}$ and $h_i = argmax\{S(h_i^j)\}$
10: **end for**
11: **while** B contain blocks not in C **do**
12: Find S_{imax} containing the largest amount of uncovered elements in B from all possible S_i
13: Add elements of S_{imax} to C
14: **end while**
15: **for** `<each` S_i `in C>` **do**
16: Select its elements and remove them to get C' and S'
17: **if** $C' == B$ **then**
18: $C = C'$
19: $S = S'$
20: **end if**
21: **end for**
22: Return the set of waypoints $W = \{x_i, y_i, h_i\}$

In Algorithm 1, we present a waypoint generation algorithm following the two steps. Lines 4–10 corresponds to the operations of finding the best heights. The set covering problem is NP-hard [26], and a greedy algorithm is used to solve it, which is presented in lines 11–21 of Algorithm 1.

2.3. Quality-aware path planning

We consider periodic sensing applications: in each sensing round, a group of UAVs are scheduled by GCS to visit the generated waypoints, take pictures, and come back to the GCS. The applications require the observations of the target area to be updated with a temporal resolution of T_u. In other words, if all the UAVs are sent out at time t = 0, they should return to the GCS within a global deadline T_u. We study how multiple UAVs can be scheduled in an energy efficient way such that all the waypoints are traversed within the global deadline. Generally, a UAV has enough energy to support its communication and computing hardware, and the flight is the most energy consuming operation. Fig. 3(c) shows a UAV travelling from waypoint W_i to waypoint W_j. The net force F_{ij} required by the UAV to travel from W_i to W_j is represented as

$$F_{ij} = \sqrt{f_x^2 + f_y^2 + f_z^2} \tag{4}$$

The mathematical statements for UAV movement can be found using the following equation:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \tag{5}$$

$$+\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{1}{m} & 0 & 0 \\ 0 & \frac{1}{m} & 0 \\ 0 & 0 & \frac{1}{m} \end{bmatrix} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix}$$

where m is the mass of UAV and f_x, f_y and f_z are the components of force that exerts on UAV to move from one waypoint to another. For generality, Eq. (5) can be written as:

$$\dot{s} = As + Bu \tag{6}$$

where:

$$s := \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T \tag{7}$$

The result has full compatibility with Newton's second law and the discrete form of Eq. (6) can be written as showed in [27] and [28].

We formulate a Vehicle Routing Problem (VRP) for finding the best routes for a group of UAVs such that all the waypoints can be visited within the deadline. To balance the energy consumption of different UAVs, the objective is to minimize the maximum energy for the UAVs to visit the waypoints within the global deadline. If U_{ij} is the potential energy and F_{ij} represents the force required by a vehicle to move from waypoint W_i to waypoint W_j, then $U_{ij} \propto F_{ij}$ defines the relation between potential energy and force. Since the force directly reflects the potential energy per unit distance, we calculate force in our problem formulation and try to minimize the maximum force. The VRP problem is presented as follows:

$$\mathrm{MIN} \quad \left\{ \mathrm{MAX} \sum_{\mathrm{i=0}}^{\mathrm{N_W+1}} \sum_{\mathrm{j=0, j\neq i}}^{\mathrm{N_W+1}} \sum_{\mathrm{u=1}}^{\mathrm{N_u}} \mathrm{X_{iju} F_{ij}} \right\} \tag{8}$$

$$s.t \quad \sum_{i=1}^{N_W} \sum_{u=1}^{N_u} X_{iju} = 1, \forall j \tag{9}$$

$$\sum_{j=1}^{N_W} \sum_{u=1}^{N_u} X_{iju} = 1, \forall i \tag{10}$$

$$\sum_{j=1}^{N_W} X_{0ju} = 1, \forall u \tag{11}$$

$$\sum_{i=1}^{N_W} X_{i(N_W+1)u} = 1, \forall u \tag{12}$$

$$\sum_{i=0}^{N_W+1} \sum_{j=0, j\neq i}^{N_W+1} X_{iju} T_{ij} \leq T_u, \forall u \tag{13}$$

$$X_{iju} = \begin{cases} 1, & \text{if UAV } u \text{ travels from} \\ & W_i \text{ to } W_j; \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

$$where\ i \neq j; i, j \in \{0, 1, 2, \ldots, N_W\}$$

where N_W is the total number of waypoints and N_u is the number of vehicles. We consider the list of waypoints $\{W_0, W_1, \ldots, W_{N_W}, W_{N_W+1}\}$, where W_0 and W_{N_W+1} both represent the base station (GCS). For simplifying the problem formulation, we say that a vehicle starts from W_0 and after visiting its set of waypoints it returns to W_{N_W+1}. F_{ij} is the cost (force) required to travel from W_i to W_j. T_{ij} is the time taken by vehicle V_u to travel from W_i to W_j and T_u is the global deadline. Constraints (9) and (10) ensure that each waypoint is only visited once in each sensing round. Constraints (11) and (12) ensure that a vehicle starts from the base station and ends at the base station. Constraint (13) specifies that a vehicle V_u visits its set of waypoints within the time deadline. Constraint (14) represents the value of binary variable X_{iju}, where $X_{iju} = 1$ if vehicle V_u travels from waypoint W_i to waypoint W_j, and 0 otherwise.

We propose to solve this problem in two steps. First, we solve a sub-problem of finding the minimum energy needed for a UAV to fly from one waypoint to another. A UAV could fly between a pair of waypoints with different speeds. There is a tradeoff between energy consumption and flight time. For example, a UAV could fly with a higher speed to visit the waypoints in a shorter time, but this could result in higher energy consumption. Given a maximum speed allowed for the UAV and based on the aforementioned energy model, this problem can be discretized over time and solution can be found using the Gurobi optimizer [29]. By giving the control parameters of UAV to Gurobi optimizer, the optimizer solves the Mixed Integer Liner Programming(MILP) problem to generate the optimal/minimum force between each pair of waypoints. The travel time between the two waypoints under this force could be calculated as well.

Second, we use the results from the first step to find the routes for a group of UAVs. The Min-Max Vehicle Routing Problem is an NP-hard problem [30]. We propose to introduce a genetic algorithm (GA) along with mixed integer linear programming (MILP) to find an efficient solution to the problem. A GA is a search heuristic for solving both constrained and unconstrained optimization problems based on a natural selection process that mimics biological evolution by iteratively modifying a population of individual solutions [31]. GA has been used to solve UAV path planning problems in [14,18], and more generally, large-scale wireless network design problems in [32]. The proposed two-step solution could be regarded as a kind of hybrid metaheuristic combining an exact optimization approach based on the introduction of Hybrid Metaheuristics in [33,34].

The detailed steps for solving the proposed VRP problem is presented in Algorithm 2. The algorithm consists of two major parts:

1. Mixed integer linear programming: A MILP problem is solved to calculate the minimum force by UAV between each pair of waypoints (lines 2–4 in Algorithm 2).

The whole procedure for finding the forces between each pair of waypoints, is exactly the same as provided in general forms through references [27] and [28]. The details are provided as follows. Let $t = [1, 2, \ldots, T]$ be the time steps and T be the total flight time to reach the goal position. Now, if b_{final} represents the binary variable for visiting the goal position, then the cost function can be written as:

$$\forall t \in [1, 2, \ldots, T]$$

$$J_{min.time} = \sum_{t=1}^{T} [b_{final} T + \varepsilon_1(|f_{x,t}|) + \varepsilon_2(|f_{y,t}|) + \varepsilon_3(|f_{z,t}|)] \tag{15}$$

where, ε_i are the penalties for the fuel consumption (i.e. force generated) and small enough to ensure that the penalty never exceeds the value of each time step. $f_{i,\,t}$ are the components of forces exerted on the UAV due to the rotors at time step t. The objective of the cost function is to minimize both the energy consumption as well as the time. b_{final} is active (equal to 1) when the UAV reaches its final destination, else it is inactive (equal to zero). Hence, the UAV reaching the final destination is represented as:

$$\sum_{t=1}^{T} b_{final} = 1 \tag{16}$$

Algorithm 2 Min-Max GA Algorithm with MILP.

```
1: Mixed Integer Linear Programming:
2: Input: Waypoints, v_min, v_max
3: Compute the optimal force between each pair of waypoints by
   solving the MILP problem using the Gurobi Optimizer
4: Output: Waypoint matrix with the force and the associated
   travel time between each pair of waypoints
5: Genetic Algorithm:
6: Input: Waypoint matrix, Vehicles, Global deadline
7: Generate initial population (chromosomes) of size P
8: Evaluate fitness of chromosomes
9:     The value of chromosome x, V_x, is given by V_x = max(T_rx)
   for that chromosome
10:    F_cx = B_v/V_x
11: while no change || max iteration not reached do
12:   a. Selection
13:       Select two chromosomes from the current population
      (with a high probability of choosing the fittest chromo-
      somes). Carry both over to the new generation.
14:   b. Crossover
15:       Crossover parts of chromosomes of the two members re-
      sulting in two children using the single point order crossover
      technique.
16:   c. Mutation
17:       With a probability, mutate the newly created children us-
      ing 2 exchange: randomly choose 2 waypoints and exchange
      them.
18:   d. Repeat (a) to (c) until a new population of P members are
      created
19:   Calculate time taken by each chromosome
20:       If time constraint is violated then add penalty
21:   Evaluate fitness of chromosomes
22: end while
23: Choose best (fittest) chromosome
24: if chromosome is found with total time <= global deadline
    then
25:   Output set of routes
26: else
27:   Increase Maximum velocity by Δv and go to 1
28: end if
```

The incorporation of these formulations in Gurobi optimizer concludes step 3 of Algorithm 2.

2. Genetic algorithm: A genetic algorithm is designed to find out the best set of routes by repeatedly performing the operations - Selection, Crossover & Mutation. The best(fittest) route is then selected which satisfies the global deadline.

In the genetic algorithm, genes are represented by waypoints which makes up a route, while a chromosome is made up of the routes R_i of all the vehicles. A chromosome is one possible solution to the VRP. The pseudo-code presented provides an overview of the algorithm. Key steps and operations of the genetic algorithm in context to the proposed vehicle routing problem are explained as below:

Initial Population: The GA starts off with an initial population and moves towards a better solution. Since the paper presents a vehicle routing problem with the aim to generate energy efficient routes for the vehicles, an initial set of routes/chromosomes of size P is randomly generated. Let n be the number of vehicles, R is the route of the vehicle and C_0 be the initial population/ chromosome matrix,

$$C_0 = \begin{bmatrix} R_{11} & R_{12} & \dots. & R_{1j} \\ . & . & \dots. & . \\ . & . & \dots. & . \\ R_{i1} & R_{i2} & \dots. & R_{ij} \end{bmatrix} \tag{17}$$

$$C_0 = R_{ij} \tag{18}$$

where i = 1,2,.,n; j = 1,2,.,P

Fitness Function: Let T_{rx} be the cost of a route r in chromosome x and it is represented by

$$T_{rx} = \sum_{i}^{N_W+1} \sum_{j}^{N_W+1} X_{ijn} F_{ij} \tag{19}$$

where n is the index of a vehicle and r is the index of a route for a vehicle. We define the value of chromosome x, V_x, as the maximum T_{rx}. B_v is the average V_x of all the chromosomes from the initial population. F_{cx} is the fitness function used to judge each chromosome given by,

$$F_{cx} = B_v / V_x \tag{20}$$

Selection: With the aim to obtain energy efficient routes in each iteration, the genetic algorithm uses Roulette Wheel selection method to select a set of individuals from the current population to carry over to the new population, which is used as parents to produce children for the new generation through the methods of crossover and mutation:

- Calculate sum of the fitness of all the chromosomes in the population,
  $S = \sum F_{cx}$
- Generate a random probability from the interval (0,S), rp
- While $P < rp${
  Iterate over the population and keep adding the finesses to the partial sum P, till $P < S$}
- Select the chromosome where $P > S$**.**

Crossover and Mutation: The algorithm uses a single point crossover technique where a crossover point in the parent chromosomes is chosen and the tails of the parent chromosomes are swapped to generate new children chromosomes. The new children chromosomes then undergo mutation. The algorithm uses a mutation probability of 0.4 and with this probability, randomly 2 waypoints are selected in the children chromosomes and are exchanged. This operation ensures genetic diversity in the population of chromosomes.

In Algorithm 2, lines 6–10 are about the initialization of the population, chromosome value, and fitness function, and lines 11–22 illustrates the iteration process.Chromosomes with higher fitness values are selected in each iteration, which corresponds to lower maximum energy of the routes. During the process, the population will evolve towards a better solution with each iteration. At the end of all the iterations, lines 23–28 in the pseudo-code will check whether the fittest route(with minimum energy consumption) is satisfying the global deadline. If no route is found which satisfies the time constraint, then the maximum velocity of the vehicles is increased by a small value (Δv) and the entire algorithm is repeated again.

3. Performance evaluation

We evaluate the proposed work in terms of coverage quality and path planning quality through simulations in Matlab. Two target areas are generated: A mountainous area and an urban area as shown in Fig. 1. Both areas have a size of 250×250 m, and they are divided into 1×1 m blocks. The minimum spatial resolution

Table 1
Sensing parameters.

Angle of view	$\pi/9$	o (Overlap%)	20%
Δh	40 m	$h_{\min}$	100 m
$v_{\min}$	2 m/s	I_x, I_y	600
$v_{\max}$	10 m/s	Δv	2 m/s
Number of UAVs	2, 5, 7	UAV mass	20 kg

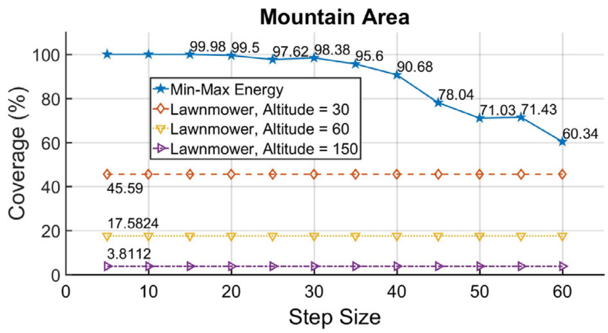


Fig. 4. Coverage for the mountain area.

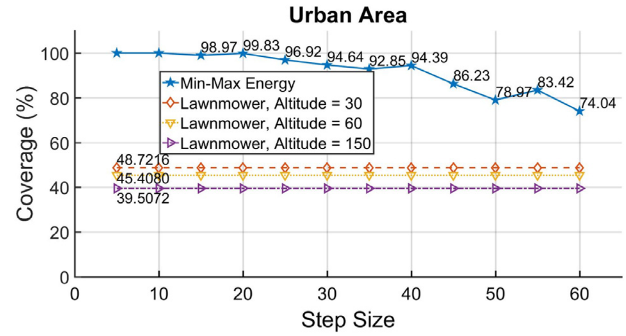


Fig. 5. Coverage for the urban area.

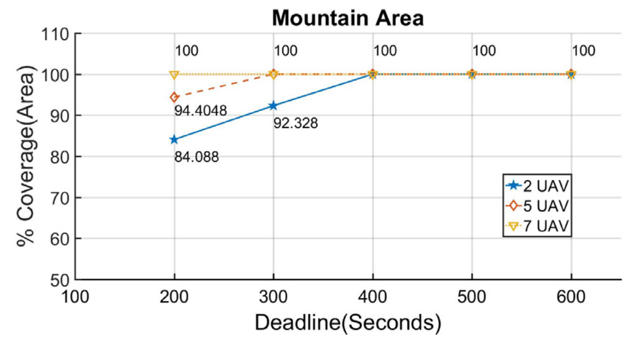


Fig. 6. Coverage vs Deadline for the mountain area.

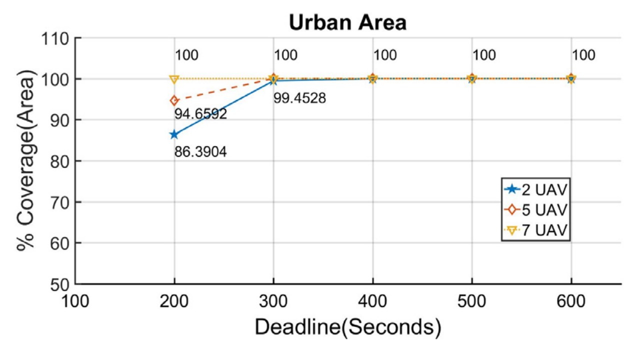


Fig. 7. Coverage vs Deadline for the urban area.

requirement ($S_{r\_\min}$) for each block is set to 8px/meter, and the temporal resolution (global deadline) is varied between 200 and 600 s. Other parameters for camera sensing and UAV properties are given in Table 1. The genetic algorithm is implemented using a population size of 30, a mutation probability of 0.4, and a total number of 10,000 iterations.

3.1. Coverage performance

The performance of coverage has been evaluated with varying step size and with varying deadline. With varying step size, we compare the coverage percentage of our proposed algorithm with lawnmower algorithm. The results of this computation helps in determining the appropriate step size to ensure full coverage of the terrain with the generated waypoints. After the waypoints are generated, the coverage of the terrain is checked with varying deadlines. In both cases, the coverage has been calculated using the sensing model discussed earlier in this paper, considering the positions, heights of the waypoints and the occlusions in the target area.

3.1.1. Coverage performance with varying step size

We first evaluate the quality of coverage assuming that the UAVs could successfully visit all the waypoints within deadline. In this case, the waypoints generated for UAVs to take pictures determine the quality of coverage. With regard to varying step size, there is a tradeoff between computational complexity and accuracy in Algorithm 1. The proposed waypoint generation algorithm is compared to the lawnmower coverage scheme that operates at a constant altitude [13]. The coverage results for the two target areas are shown in Figs. 4 and 5. The vertical axes present the percentage of areas that can be covered with the given image resolution requirement. The step size in the horizontal axes represent the amount of blocks by which we step through the environment in each iteration while searching for waypoints (defined as n_s in the previous section), and step size values ranging from 4 to 60 are tested. There is a trade off between coverage and computational complexity for the proposed waypoint generation algorithm. The percentage of covered area for the proposed algorithm stays high when the step size is not more than 30 and gradually decreases when the step size increases. Since the lawnmower scheme is designed for flat 2D terrains and it considers a constant flying altitude, we test its coverage performance at three different altitudes, which are 30 m, 80 m, and 150 m above the maximum altitude of both terrains. These altitudes are chosen taking into consideration the image quality requirement as well as the distribution of altitudes in the terrains. The coverage performance of the lawnmower scheme has nothing to do with the step size in our algorithm, but it is dependent on the altitude for flying. The coverage performance of the proposed method is better than the lawnmower scheme due to the fact that the latter does not consider the effect of occlusion.

3.1.2. Coverage performance with varying deadline

Given the waypoints that could generate full coverage, we now evaluate the coverage performance of the proposed path planning algorithm with varying deadlines. A set of UAVs is sent out to cover the given set of waypoints within a given deadline. If the UAVs cannot not cover all the waypoints within the deadline, the areas under the missing waypoints are calculated to determine the coverage percentage under that deadline. Figs. 6 and 7 show the effect of deadline over coverage for different set of UAVs. The coverage performance varies for different set of waypoints with changing deadline. We use a range of 200 s–600 s of deadlines for our experiments and coverage is calculated for a set of 2, 5 and 7

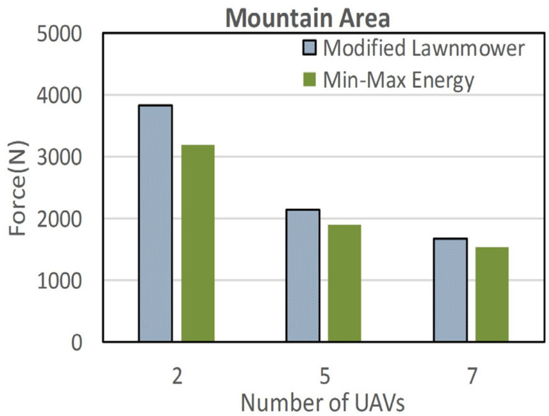


Fig. 8. Comparison of force for the Mountainous area.

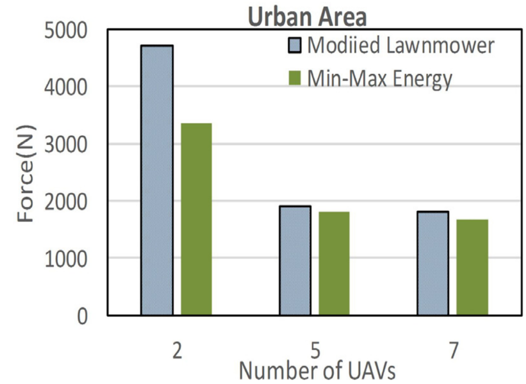


Fig. 9. Comparison of force for the Urban area.

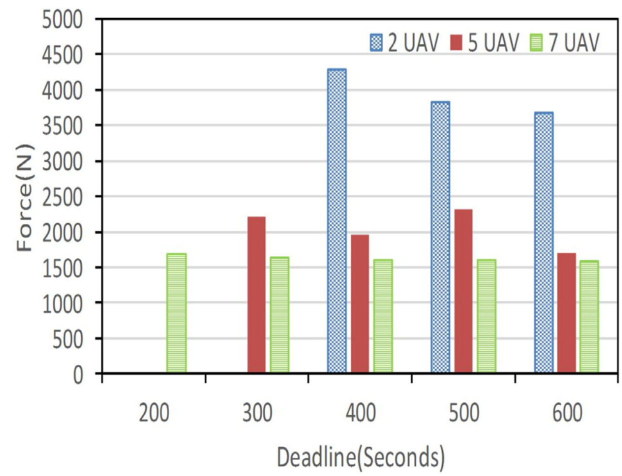


Fig. 10. Force vs. deadline for Mountainous area.

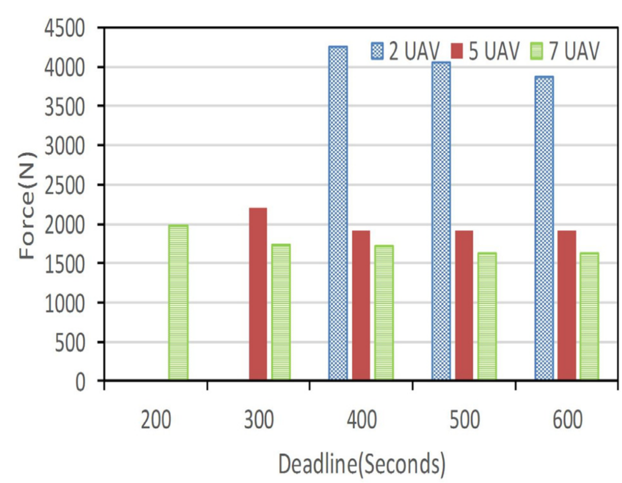


Fig. 11. Force vs. deadline for Urban area.

UAVs. The results are as expected: a small number of UAVs cannot achieve full coverage under a very small deadline, while a larger number of UAVs could be scheduled to meet a shorter deadline. This evaluation provides a way to determine an appropriate global deadline or an appropriate number of UAVs to meet a certain deadline for the terrain under observation.

3.2. Path planning energy efficiency

The energy efficiency of the proposed quality-aware path planning solution is compared with a modified version of the original lawnmower algorithm in [13], which takes the same waypoints that are generated by the proposed waypoint generation algorithm and then visits these waypoints in a lawnmower manner. This enables a fair performance comparison under the same coverage performance, as using the same waypoints indicates that the coverage performance is the same for both algorithms. Specifically, the modified lawnmower algorithm operates in the following steps: i) Divide the area into several subregions; ii) in each subregion, find a sweep direction and generate parallel lanes that includes all the waypoints; iii) let a UAV fly along each lane to visit the waypoints along the lane, and when the UAV reaches the last way point, it turns to the next lane. To facilitate a fair comparison on energy consumption, the lawnmower algorithm was modified to optimize energy consumption between each pair of waypoints. More specifically, the modified lawnmower algorithm calculates the minimum force between the waypoints as the computations are done in our proposed algorithm by solving the MILP problem using the Gurobi optimizer.

The maximum force of the proposed Min-Max Energy path planning algorithm is also compared with the modified Lawnmower algorithm for different numbers of UAVs, and the results are shown in Figs. 8 and 9. Our proposed method is executed with a given deadline whereas the modified Lawnmower algorithm does not consider deadlines. The values of the deadline need to be determined considering the area where the UAVs are deployed, the number of waypoints in that area, and the number of UAVs deployed. As an example, given the same waypoints, a larger number of UAVs could potentially satisfy a shorter deadline than a small number of UAVs. For our experiments, we select the value of deadlines suitable for our target areas. Deadline for 2 UAVs: 550 s, 5 UAVs: 300 s and 7 UAVs: 250 s are chosen for both mountainous and urban terrain.

The results in Figs. 8 and 9 show that the proposed method performs very well compared to the modified Lawnmower algorithm under the time constraints and regardless of the number of UAVs for both target areas. Taking the case of two UAVs as an example, compared to the Modified Lawnmower algorithm, the proposed method can generate 16.71% less force for the mountain area and 28.89% less force for the urban area, respectively. Summarizing all the test cases for the two target areas, the proposed Min-Max Energy path planning algorithm can save 16.23% force than the modified Lawnmower algorithm on average.

We have also evaluated the effect of deadline on force generated by the UAVs. Figs. 10 and 11 demonstrate the influence of deadline on the maximum force generated by UAVs in the mountainous and the urban terrains. The proposed algorithm is executed under multiple deadlines for different number of UAVs. The

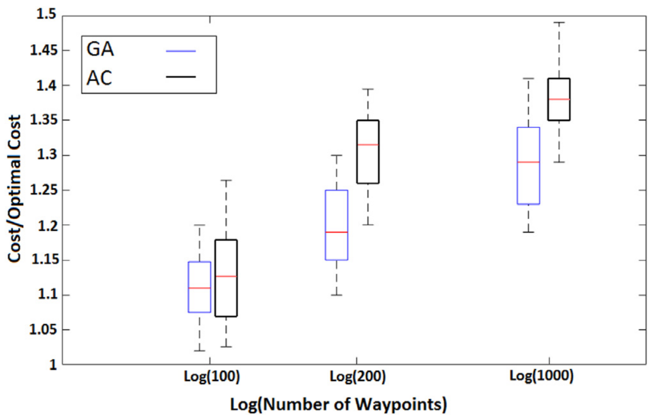


Fig. 12. Comparison on AC and GA performance.

cost(maximum force) of UAV routes depends upon the choice of the deadline. For the same number of UAVs, routes with less cost can be generated when provided with a loose deadline. In other words, the value of force is inversely proportional to the deadline. Absence of bars for 2 UAVs and 5 UAVs for 200 s and 2 UAVs for 300 s in both figures indicate that such narrow deadlines could not be met using the given numbers of UAVs. These results also suggest that the selection of the deadline is largely dependent on the number of available UAVs.

3.3. Analysis of the optimization algorithm

We also compare the proposed path planning algorithm with another optimization technique Ant Colony (AC). AC has been vastly used to solve similar optimization problems [35–37] The general solution of the VRP problem using ant-colony method is represented in [38]. For a fair comparison between our GA solution and the AC algorithm, the energy cost between each of the waypoints has been found via the MILP algorithm and then AC is implemented to find the best solution to the problem. Random 100, 200, and 1000 waypoints are subjected to the comparison and the average of 25 runs are reported in Fig. 12 for analysis of performance. In this figure the optimal solution is found via the Dynamic Programming method. For each scenario the number of UAVs is considered to be 5. Each algorithm runs for exactly one hour and then the route and the path have been recorded. It should be noted that, for each number of waypoints, the left box-whisker plot is for GA and the right one is for AC.

In Fig. 12, the box and whisker plot of the 25 runs are shown for both GA and AC. On the x axis and the y axis, Log_{10} of the number of waypoints and $\frac{Cost}{OptimalCost}$ are represented, respectively. As shown in Fig. 12, the GA has better performance than the ant colony for solving the VRP problem, and as the number of waypoints increases, GA shown the ability to reach to a better solution than AC.

4. Conclusion and future work

In this paper, we have introduced a quality-aware coverage and path planning scheme for UAV networks that can efficiently monitor geometrically complex environments. Utilizing a camera sensing model that takes into account occlusions, we have developed a waypoint generation algorithm to find the best set of waypoints to achieve full coverage of a target area with satisfactory spatial resolution. We have also designed an energy-efficient, deadline-aware path planning solution for multiple UAVs to visit the waypoints. Simulation results have shown that the proposed solution can achieve better coverage performance through lower energy costs than traditional UAV coverage and path planning techniques. In the future, we plan to extend our study in two directions. First, while the proposed scheme plans the waypoints and UAV paths before the beginning of a mission, we will investigate online adjustment strategies to guarantee sensing quality in unpredicted situations. Second, effective communication mechanisms are needed to provide satisfactory visual observations to end users. We will investigate the communication of the sensed images under the proposed coverage and path planning framework. After a set of UAVs are deployed for observing an area, we will introduce additional relay UAV nodes to assist the delivery of images, and we will propose image quality- and deadline-aware communication solutions for all the UAVs. An emulation platform will be used to evaluate the delivery of realistic images in the network of UAVs.

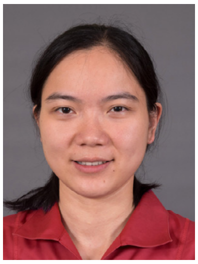
Acknowledgment

This work was supported in part by the US National Science Foundation under Grant No. CNS-1644946.

References

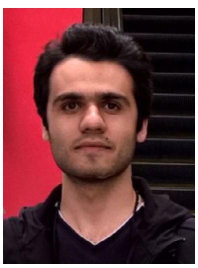
[1] O. Alvear, N.R. Zema, E. Natalizio, C.T. Calafate, Using UAV-based systems to monitor air pollution in areas with poor accessibility, J. Adv. Transp. 2017 (2017).
[2] Z. Zaheer, A. Usmani, E. Khan, M.A. Qadeer, Aerial surveillance system using UAV, in: Wireless and Optical Communications Networks (WOCN), 2016 Thirteenth International Conference on, IEEE, 2016, pp. 1–7.
[3] A.R. Girard, A.S. Howell, J.K. Hedrick, Border patrol and surveillance missions using multiple unmanned air vehicles, in: Decision and Control, 2004. CDC. 43rd IEEE Conference on, 1, IEEE, 2004, pp. 620–625.
[4] M. Erdelj, M. Król, E. Natalizio, Wireless sensor networks and multi-UAV systems for natural disaster management, Comput. Netw. (2017).
[5] K. Alexis, G. Nikolakopoulos, A. Tzes, L. Dritsas, Coordination of helicopter UAVs for aerial forest-fire surveillance, Appl. Intell. Control Eng. Syst. (2009) 169–193.
[6] I. Bekmezci, O.K. Sahingoz, Ş. Temel, Flying ad-hoc networks (FANETs): a survey, Ad Hoc Netw. 11 (3) (2013) 1254–1270.
[7] G. Fan, S. Jin, Coverage problem in wireless sensor network: a survey, J. Netw. 5 (9) (2010) 1033–1040.
[8] R. Mulligan, H.M. Ammari, Coverage in wireless sensor networks: a survey, Netw. Protocols Algorithms 2 (2) (2010) 27–53.
[9] M. Vecchio, R. López-Valcarce, Improving area coverage of wireless sensor networks via controllable mobile nodes: a greedy approach, J. Netw. Comput. Appl. 48 (2015) 1–13.
[10] B. Liu, P. Brass, O. Dousse, P. Nain, D. Towsley, Mobility improves coverage of sensor networks, in: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing, ACM, 2005, pp. 300–308.
[11] Y. Chen, H. Zhang, M. Xu, The coverage problem in UAV network: A survey, in: Computing, Communication and Networking Technologies (ICCCNT), 2014 International Conference on, IEEE, 2014, pp. 1–5.
[12] W.H. Huang, Optimal line-sweep-based decompositions for coverage algorithms, in: Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on, 1, IEEE, 2001, pp. 27–32.
[13] J. Araujo, P. Sujit, J.B. Sousa, Multiple UAV area decomposition and coverage, in: Computational Intelligence for Security and Defense Applications (CISDA), 2013 IEEE Symposium on, IEEE, 2013, pp. 30–37.
[14] S.-Y. Fu, L.-W. Han, Y. Tian, G.-S. Yang, Path planning for unmanned aerial vehicle based on genetic algorithm, in: Cognitive Informatics & Cognitive Computing (ICCI* CC), 2012 IEEE 11th International Conference on, IEEE, 2012, pp. 140–144.
[15] M. Jakob, E. Semsch, D. Pavlıcek, M. Pechoucek, Occlusion-aware multi-UAV surveillance of multiple urban areas, in: 6th Workshop on Agents in Traffic and Transportation (ATT 2010), Citeseer, 2010, pp. 59–66.
[16] C. Di Franco, G. Buttazzo, Energy-aware coverage path planning of UAVs, in: Autonomous Robot Systems and Competitions (ICARSC), 2015 IEEE International Conference on, IEEE, 2015, pp. 111–117.
[17] M. Quaritsch, K. Kruggl, D. Wischounig-Strucl, S. Bhattacharya, M. Shah, B. Rinner, Networked UAVs as aerial sensor network for disaster management applications, e & i Elektrotechnik und Informationstechnik 127 (3) (2010) 56–63.
[18] O. Maeda, M. Nakamura, B.M. Ombuki, K. Onaga, A genetic algorithm approach to vehicle routing problem with time deadlines in geographical information systems, in: Systems, Man, and Cybernetics, 1999. IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on, 4, IEEE, 1999, pp. 595–600.
[19] A.J. Pohl, G.B. Lamont, Multi-objective uav mission planning using evolutionary computation, in: Proceedings of the 40th Conference on Winter Simulation, Winter Simulation Conference, 2008, pp. 1268–1279.

[20] S. Karaman, E. Frazzoli, Vehicle routing problem with metric temporal logic specifications, in: Decision and Control, 2008. CDC 2008. 47th IEEE Conference on, IEEE, 2008, pp. 3953–3958.
[21] G.S. Avellar, G.A. Pereira, L.C. Pimenta, P. Iscold, Multi-UAV routing for area coverage and remote sensing with minimum time, Sensors 15 (11) (2015) 27783–27803.
[22] A. Ahmadzadeh, A. Jadbabaie, V. Kumar, G.J. Pappas, Multi-UAV cooperative surveillance with spatio-temporal specifications, in: Decision and Control, 2006 45th IEEE Conference on, IEEE, 2006, pp. 5293–5298.
[23] K. Scott, R. Dai, M. Kumar, Occlusion-aware coverage for efficient visual sensing in unmanned aerial vehicle networks, in: Global Communications Conference (GLOBECOM), 2016 IEEE, IEEE, 2016, pp. 1–6.
[24] D.A. Forsyth, J. Ponce, Computer Vision: A Modern Approach, Prentice Hall, New Jersey, 2002.
[25] R. Dai, I.F. Akyildiz, A spatial correlation model for visual information in wireless multimedia sensor networks, Multim. IEEE Trans. 11 (6) (2009) 1148–1159.
[26] R. Motwani, P. Raghavan, Randomized Algorithms, Cambridge University Press, 1995.
[27] M. Radmanesh, M. Kumar, Flight formation of UAVs in presence of moving obstacles using fast-dynamic mixed integer linear programming, Aerosp. Sci. Technol. 50 (2016) 149–160.
[28] M. Radmanesh, M. Kumar, A. Nemati, M. Sarim, Dynamic optimal UAV trajectory planning in the national airspace system via mixed integer linear programming, Proc. Inst. Mech. Eng. Part G 230 (9) (2016) 1668–1682.
[29] 2016. Gurobi Optimizer Reference Manual, Gurobi Optimization, Inc. Available: http://www.gurobi.com.
[30] C. Ren, Solving min-Max vehicle routing problem., J. Softw. 6 (9) (2011) 1851–1856.
[31] S. Karakatič, V. Podgorelec, A survey of genetic algorithms for solving multi depot vehicle routing problem, Appl. Soft Comput. 27 (2015) 519–532.
[32] F. DAndreagiovanni, On improving the capacity of solving large-scale wireless network design problems by genetic algorithms, in: European Conference on the Applications of Evolutionary Computation, Springer, 2011, pp. 11–20.
[33] G.R. Raidl, J. Puchinger, Combining (integer) linear programming techniques and metaheuristics for combinatorial optimization., Hybrid Metaheuristics 114 (2008) 31–62.
[34] C. Blum, C. Cotta, A. Fernández, J. Gallardo, M. Mastrolilli, Hybridizations of metaheuristics with branch & bound derivates, Hybrid Metaheuristics (2008) 85–116.
[35] M. Dorigo, L.M. Gambardella, Ant colony system: a cooperative learning approach to the traveling salesman problem, IEEE Trans. Evol. Comput. 1 (1) (1997) 53–66.
[36] R. Karmakar, R. Mitra, A. Dey, V. Chakraborty, A. Nayak, Solving tsp using improved elitist ant system based on improved pheromone strategy and dynamic candidate list, MAYFEB J. Comput. Sci. Vol 1 (2016) 8–15.
[37] F. DAndreagiovanni, J. Krolikowski, J. Pulaj, A fast hybrid primal heuristic for multiband robust capacitated network design with multiple time periods, Appl. Soft Comput. 26 (2015) 497–507.
[38] T.K. Ralphs, Parallel branch and cut for capacitated vehicle routing, Parallel Comput. 29 (5) (2003) 607–629.



Rui Dai is an Assistant Professor in the Department of Electrical Engineering and Computing Systems at University of Cincinnati, Ohio, USA. She received her BS in Electronics and Information Engineering and MS in Communications and Information Systems from Huazhong University of Science and Technology, Wuhan, China, in 2004 and 2007, respectively. She received her Ph.D degree in Electrical and Computer Engineering from Georgia Institute of Technology, Atlanta, GA, USA in 2011, under the supervision of Professor Ian F. Akyildiz in the Broadband Wireless Networking Laboratory. She was a postdoctoral fellow at the Center for Assistive Technology and Environmental Access of Georgia Tech in 2012. She was an Assistant Professor at the Department of Computer Science at North Dakota State University from 2012 to 2014. Her recent research interests include wireless sensor networks, multimedia communications and networking, and cyber-physical systems.

Sneha Fotedar graduated with Master's degree in Computer Science from University of Cincinnati, Ohio, USA in December 2017, where she worked on research problems in coverage and sensing in unmanned aerial vehicle networks & communication in ad hoc networks. She received her Bachelors degree in Computer Engineering from University of Pune, Pune, India in 2012. She is currently working as an Automation Analyst with Huntington National Bank, Columbus, Ohio.



Mohammadreza Radmanesh is a Ph.D candidate in University of Cincinnati after receiving his masters in Mechanical Engineering in 2016 and in Aerospace Engineering in 2017. He has published several journal and conference papers. His current research interests include multi-disciplinary decision-making, complex systems and control in large-scale systems, development of novel techniques for a collaborative control of UAVs, robotics, swarm systems, and multiple robot coordination and control. He is a member of ASME, AIAA and IEEE. Furthermore, he has been served as a coauthor of a book titled, "Multi-Rotor Platform Based UAV Systems".

Manish Kumar received his Bachelor of Technology degree in Mechanical Engineering from Indian Institute of Technology, Kharagpur, India in 1998, and his M.S. and Ph.D. degrees in Mechanical Engineering from Duke University, NC, USA in 2002 and 2004 respectively. After finishing his Ph.D., he worked as a postdoctoral research associate in the Department of Mechanical Engineering and Materials Science at Duke University from 2004 to 2005. In 2005, he received the Research Associateship Award from National Research Council (NRC). This award allowed him to work as a postdoctoral Research Associate with the Army Research Office, NC, USA from 2005 to 2007. As a part of his NRC Associateship program, he was a visiting scholar at General Robotics, Automation, Sensing, and Perception (GRASP) laboratory at the University of Pennsylvania, PA, USA. Subsequently, he worked as an Assistant Professor in the School of Dynamic Systems at the University of Cincinnati, OH, USA where he directed the Cooperative Distributed Systems (CDS) Laboratory and co-directed the Center for Robotics Research. He is currently an Associate Professor in the Department of Mechanical and Materials Engineering in University of Cincinnati. His current research interests include complex systems, decision-making and control in large-scale systems, development of novel techniques to fuse data from multiple sources, robotics, swarm systems, and multiple robot coordination and control. He is a member of American Society of Mechanical Engineers. In addition to coauthoring a book titled, Ant Colony Inspired Robotic Swarm Systems, in 2011, Kumar actively reviews multiple scholarly journals and also publishes his own articles for journals, conferences, and research offices. He also mentors many Ph.D and MS candidates and has served as the chair of multiple professional conferences and organizations as well as worked as a consultant for Edaptive Computing, Inc.