2021 International Conference on Unmanned Aircraft Systems (ICUAS)
Athens, Greece. June 15-18, 2021

Near-Optimal Coverage Path Planning of Distributed Regions for Aerial Robots with Energy Constraint

Zeba Khanam, Klaus McDonald-Maier and Shoaib Ehsan

***Abstract*— Unmanned Aircraft Vehicles (UAVs) have gained immense popularity for area coverage having applications such as environmental monitoring, demining, search and rescue, among others. Most of the existing studies exploring area coverage have considered only a single region, however, few recent studies have considered coverage of multiple distributed regions. One of the limitations which UAV suffers while covering distributed multiple regions is energy constraints where complete area coverage is not possible. From a strategical point of view, we propose a novel algorithm which solves a variant of area coverage problem where the UAV aims to achieve near-optimal area coverage due to path length limitation caused by the energy constraint. In this paper, a preliminary study is conducted by first formulating the problem and later on presenting a solution. The solution has been partitioned into two inter-dependent sub-problems : i) inter-region coverage, ii) intra-region coverage. The performance of the algorithm has been evaluated by analysing its properties over an exhaustive set of test case scenarios and comparing it against two state-of-the-art area coverage approaches.**

***Keywords* - Coverage Path Planning (CPP), Unmanned Aircraft Vehicle (UAV), Partial Area Coverage, Energy Constraints**

I. Introduction

Recent times have witnessed extensive use of Unmanned Aerial Vehicles (UAVs) in many application domains for surveying and covering large areas. This problem is also known as Coverage Path Planning (CPP) and can be defined as the process of path computation for a UAV to explore every location in any given area. The application of CPP is wide spread across multiple domains like agriculture [1], [2], [3], environmental inspection [4], [5], floor cleaning [6], terrain reconstruction [7], [8], demining [9] and lawn mowing [10]. The vast majority of prior works have focused on computation of coverage path for a single region [11], [12]. However, there can exists scenarios such as post-disaster relief, environmental monitoring, military surveillance, search and rescue missions, where the UAV is required to cover multiple regions.

There are a few handful of works which explore the problem of computation of coverage path for multiple disjoint regions. In addition with the need to compute coverage path for coverage of multiple regions, instead of a single one, the problem with disjoint regions introduces the additional challenge of generating least cost tours covering a set of regions, also known as Travelling Salesman Problem (TSP). The combination of CPP and TSP, both of which are classified as NP-hard, introduces significant challenges.

This problem was first introduced by Xie et al. [13] which proposed a dynamic programming solution to compute coverage path for multiple disjoint rectangular regions. The authors further extended this algorithm for multiple convex region by proposing a genetic algorithm solution [14]. Vasquez-Gomez et al. [15] had attempted to solve this problem by proposing a two step path planning algorithm. Recently, Khanam et. al [4] had computed coverage path for multiple disjoint regions with precedence provision. All these work have assumed that the UAV had sufficient energy to cover all the regions. Only recently potential energy limitation and constraint has been considered when computing coverage path over multiple disjoint regions [16]. This work allowed the UAV to return to the depot to change its battery. However, this might not be a realistic scenario in many cases and limited energy may lead to partial area coverage. Nevertheless, appropriate execution of partial coverage can provide useful information of the site. Recently, few works have explored optimization of coverage path for partial coverage of a single region [17], [18], [19]. To the best of our knowledge, there does not exist a single work addressing the problem of partial area coverage over geographically distributed regions by energy constrained aerial robot.

This paper considers solving variation of TSP-CPP problem where aerial robot cannot achieve full coverage of multiple distributed regions due to the energy constraints on the UAV which impose limitation on the total path length. The algorithm first computes the inter-region path which also decides the start and exit point for each region. Using these points, the algorithm then distributes waypoints to achieve near-optimal coverage taking into consideration energy constraints, where the optimal path is defined as a set of waypoints, if visited by the aerial robot, entire area is covered by robot's sensor. In this context, the near-optimal path is defined as a set of waypoints, if visited by the aerial robot, robot's sensor is able to maximize the coverage area within the given limitation [17]. Fig 1 shows an example of a near optimal coverage path for a single region.

The remainder of the paper is organised as follows. Section II defines the proposed algorithm. The experimental evaluation of the proposed method is shown in Section III. Finally, the conclusion and future work are presented in Section IV.

This work is supported by the UK Engineering and Physical Sciences Research Council through grants EP/R02572X/1 and EP/P017487/1.

Zeba Khanam, Klaus McDonald-Maier and Shoaib Ehsan are with Embedded and Intelligent Systems Lab, School of Computer Science and Electronics Engineering, University of Essex, UK {zeba.khanam,kdm,sehsan} @ essex.ac.uk

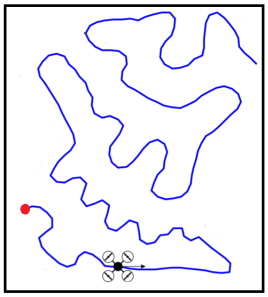


Fig. 1. A near-optimal coverage path considering a path-length constraint. Start point is denoted with a red circle [17]

II. Proposed Algorithm

The algorithm for near optimal coverage of disjoint regions is described next which is applicable to an energy constrained UAV. Considering the kinematics, the following assumptions are made. The path is defined as a sequence of waypoints which the UAV traverses. The velocity and acceleration when moving between waypoints is consistent such that the UAV requires same amount of energy to complete the travel between two pairs of waypoints which are same distance apart.

A. Algorithm Overview

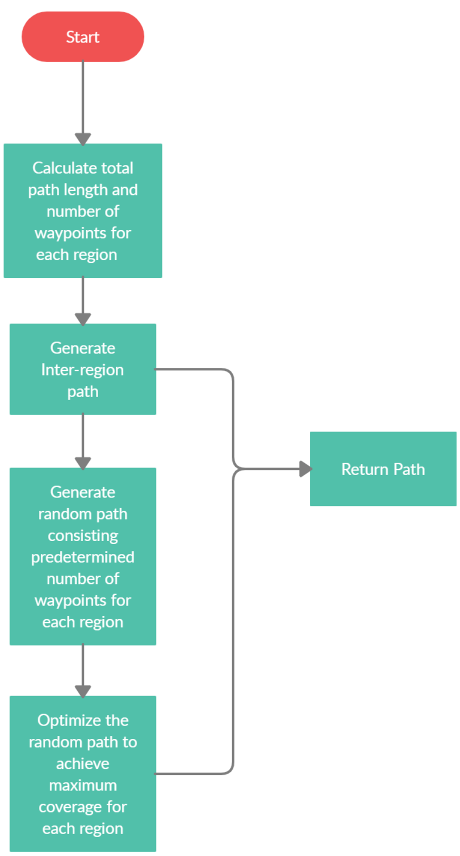


Fig. 2. Flowchart elucidating the proposed algorithm for partial area coverage.

In this section, we present a higher level overview of the algorithm. Fig. 2 illustrates a flow chart of the path generation process.

1) Calculate the total path length and the number of waypoints for each region.
2) Calculate the inter-region path and first and last waypoint for each region.
3) Generate a random sequence of path waypoints for each region with desired spacing ensuring that the path do not intersect each other.
4) Optimize the intra-region path to maximize the coverage area subjected to energy constraints.

B. Calculating Total Path Length

The proposed algorithm requires the number of waypoints in each region to be pre-computed. For each region, the waypoints are all spaced at an equal distance d_i away from the adjacent waypoint. If there are n regions which the UAV has to traverse and for each region r_i, there are m_i ($W_i = \{w_1^i, \ldots, w_{m_i}^i\}$) total waypoints, the path length can be computed as:

$$l_{total} = \sum_{i=1}^{n}(m_i - 1)d_i + \sum_{i=1}^{n-1} Dist(w_{m_i}^i, w_1^{i+1}) \quad (1)$$

where, $Dist(a, b)$ is a function which calculates euclidean distance between two input waypoints a and b. The maximum path length is dependent on the energy consumption of the UAV used for the mission. To compute the energy required to traverse the path length, we require two functions $F_m(d, v)$ and $F_r(t)$. $F_m(d, v)$ outputs the energy required to travel distance d with a velocity v and $F_r(t)$ outputs the energy consumed when resting at a waypoint for sensing for an amount of time t. We assume that the UAV travels at a consistent velocity v_t and the waypoints in a particular region i are spaced equally with distance d, the energy $E_m = F_m(d, v_t)$ will be constant for each region $r_i \in R$. The aerial robot is assumed to stop at each waypoint for sensing for same amount of time t_r, so $E_r = F_r(t_r)$ will also be constant. Another assumption which is made is that, for each region, the first waypoint w_1^i and last waypoint $w_{m_i}^i$ are the vertices of the region i. The energy $E_m^{i,j} = F_m(Dist(w_{m_i}^i, w_1^j), v_t)$ is consumed while traveling from region r_i to region r_j. If there are m_i total waypoints in each region r_i, the energy E_{path} consumed by the UAV while covering the all the regions is:

$$E_{path} = \sum_{i=1}^{n}\{m_i E_r + (m_i - 1)E_m\} + \sum_{i=1}^{n-1} E_m^{i,i+1} \quad (2)$$

As the total energy capacity E_{total} for a particular UAV is already known, the number of waypoints can be computed by imposing the following condition:

$$E_{path} \leq E_{total} \quad (3)$$

In Equation 2, the first and second terms indicate the intra-region energy consumption, E_{iar}, described in Section II-D and the last term indicates the inter-region energy consumption, E_{ier}, described in Section II-C. Solving for m_i, resulting in:

$$\sum_{i=1}^{n} m_i(E_r + E_m) - E_m \leq E_{total} - E_{ier} \qquad (4)$$

Using Equation 4, the waypoints m_i for a region r_i are determined by:

$$\sum_{i=1}^{n} m_i = \frac{E_{iar} + n \times E_m}{E_r + E_m} \qquad (5)$$

We assume that the number of waypoints m_i for each region to be:

$$m_i = \lfloor \frac{A_i}{A_{max}} \rfloor \times D \qquad (6)$$

Using Equation 5 and 6, D can be formulated as:

$$D = \frac{E_{iar} + n \times E_m}{E_r + E_m} \times \frac{A_{max}}{A_{total}} \qquad (7)$$

where, $A_i,\ A_{max},\ A_{total}$ denote area of region r_i, area of region with maximum area and total area of all the regions, respectively.

In the following sub-sections, we describe the computation of inter-region energy E_{ier} and intra-region E_{iar} consumption.

C. Inter-Region Energy Consumption (E_{ier})

The first step in the proposed algorithm is to compute energy consumed by the aerial robot while traveling between different regions. As mentioned in last sub-section, we have assumed that first and last waypoints w_1^i and $w_{m_i}^i$, respectively as the vertices of the regions. To determine the energy consumed inter-region traversal E_{ier}, we need to compute:

1) First and last waypoints (w_1^i and $w_{m_i}^i$) $\forall i \in n$.
2) Inter-region traversal order.

However, to minimize the energy consumption the computation of waypoints should be dependent on inter-region traversal orders. The full algorithm is described in Algorithm 1.

Putting together all the elements mathematically stated in Algorithm 1, the inter-region energy E_{ier} computation algorithm works as follows. To begin, a list $\hat{V}$ listing number of regions visited by aerial robot is initialized $NULL$. In addition, the start depot region r_q is initialized as the current region C_r of traversal by the aerial robot and inter-region energy E_{ier} is initialized as zero. The algorithm then loops through several actions. First, it determines the nearest region to the current region which has not been visited by the aerial robot. The nearest region (r_i) is determined by the distance between the vertices of different regions (D^{C_r,r_i}). Next, the vertex of the current region is designated as the last waypoint of current region ($w_{m_i}^{C_r}$) and the vertex of the nearest region is designated as the start waypoint of next region ($w_1^{r_i}$). Finally, the current region C_r is the added to the visited list $\hat{V}$ and the nearest region r_i is selected as the current region. Using, the distance between two waypoints, energy consumption is computed and added to the total inter-region energy E_{ier}. This process iterates until all the regions are visited.

Algorithm 1: INTER-REGION ENERGY CALCULATION

Input:
1. $R = \{r_1, \dots, r_n\}$: Set of regions;
2. $< Centre, Vertices, Side > CVS = \{< c_1, \{v_1^1, \dots, v_m^1\}, S_1 >, \dots, < c_n, \{v_1^n, \dots, v_m^n\}, S_n >\}$
3. r_q : Starting depot region;
4. S_{r_q} : Starting waypoint of depot region r_q;
5. v_t : Constant velocity;

Output: Energy Consumption E_{ier}

```
INITIALIZATION
V̂ = NULL : List of regions visited;
C_r = r_q : Current region;
w_1^{r_i} : Starting waypoint of region r_i;
w_{m_i}^{r_i} : Ending waypoint of region r_i;
D = {D^{1,2}, ..., D^{1,n}, D^{2,3}, ..., D^{2,n}, ..., D^{n-1,n}} : Set
  of distance matrices where D_{k,l}^{i,j} represents distance
  between vertex k of region r_i and vertex l of region r_j;
E_ier = 0;                  ▷ Initializing Inter-Region Energy
while |V̂| ≠ |R| − 1 do
    for each region r_i ∈ R ∋ (r_i ∉ V̂ and r_i ≠ C_r) do
        min_dist = minimum of D^{C_r,r_i};
        row, col = indices of minimum of D^{C_r,r_i};
        w_{m_i}^{C_r} = row;
        w_1^{r_i} = col;
        V̂ = V̂ ∪ C_r;
        C_r = r_i;
        E_ier = E_ier + F_m(min_dist, v_t);
```

D. Intra-Region Energy Consumption (E_{iar})

After calculating the number of waypoints (m_i), desired spacing (d_i) and inter-region order of traversal with entrance and exit points for each region, we are fully equipped to calculate intra-region energy consumption E_{iar}. The first step is to calculate the intra-region path, randomly. The random intra-region path needs to be optimized to reduce the overall cost which is described in detail in next section.

1) Optimality: Due to the energy-constraint, the aerial robot can only achieve partial coverage, the proposed algorithm aims to generate a sequence of waypoints to cover the regions in best possible manner, attempting to maximize the coverage area. When generating a random path, we ensure that the coverage path between waypoints do not overlap. However, there is a possibility that there can exist two coverage paths which don't overlap and achieve same amount of area coverage but provide different qualities of information. For instance, in Fig. 3, the aerial robot is able to cover same amount of area in both the cases, however, in the first case in Fig. 3 [a], the robot is able to provide detailed coverage information about the upper portion of region without delving information about the bottom region. In the second case as elucidated in Fig. 3[b], the area coverage by the aerial robot provide information which present an overall picture of the region. We assume that having moderate quality information of the entire region is better than having detailed information of a part of the region and no information about the remaining region. To ensure that the aerial robot takes a coverage path similar to Fig. 3[b] following cost function $J(.)$ is imposed on coverage path P_i.

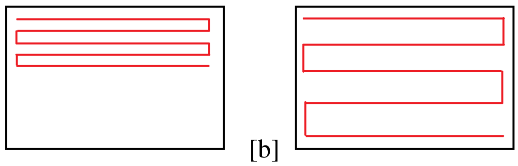
[a]  [b]

Fig. 3. Example of coverage path with same path length [a] where robot is able to collect detailed information about the top of region and no information of bottom side [b] where robot is able to collect moderate information about entire region

$$J(P_i) = \iint_{r_i} \min_{w_j^i} ||x - w_j^i|| dA \tag{8}$$

For region r_i, $W_i = \{w_1^i, \ldots, w_{m_i}^i\}$ is a set of m_i waypoints, **x** is a point lying inside the region r_i represented by a vector $[xy]^T \in r_i$ and dA is the differential area $dydx$. We assume that the region is a set of grid cells where the resolution of each grid cell is of the size of the footprint of the robot's sensor. When the robot is stationed at the center of the grid cell it can cover the entire grid cell using its sensor. In the resumption with our previous assumption, we also assume that the possible set of waypoints which robot can traverse while covering any region is the centre of the grid cell. If g_{n_i} is the total number of grid cells in the region r_i, this assumption allows us to reformulate the cost function in Equation 8 as:

$$J(P_i) = \sum_{x=1}^{g_{n_i}} \min_{w_j^i} ||x - w_j^i|| \tag{9}$$

Applying the above formulated cost function $J(P_i)$, the problem of finding location of waypoints for inter-region coverage path P_i^* for region r_i translates to:

$$P_i^* = \arg\min_{P_i} J(P_i) \tag{10}$$

subject to constraints,

$$||w_j^i - w_{j-1}^i|| = d, \ \forall i = 1, \ldots, n \ , \ \forall j = 2, \ldots, m_i \tag{11}$$

$$||w_{j+1}^i - w_j^i|| = d, \ \forall i = 1, \ldots, n, \ \forall j = 1, \ldots, m_i - 1 \tag{12}$$

$$w_1^i, w_{m_i}^i \in \text{vertex of region } r_i \tag{13}$$

III. Simulations and Results

In this section, we present simulations to assess the performance of proposed algorithm, beginning with several examples of inter-region and intra-region path. We perform series of experiments to characterize the algorithm with respect to various parameters and compare the algorithm to other path generation algorithms. During the simulation, an arbitrary energy model has been used to determine the total path length, which is used to determine number of waypoints for each regions which are used as an input for intra-region path generation. All the algorithms are implemented using MATLAB R2020a, and the experiments are performed on a DELL XPS IDV8QVO with Intel Core i5, 8GB memory and 225GB storage. The built-in MATLAB function $fmincon$ is used for direct optimization.

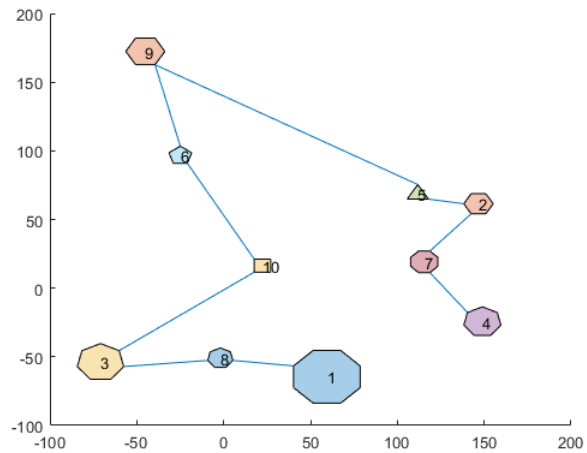
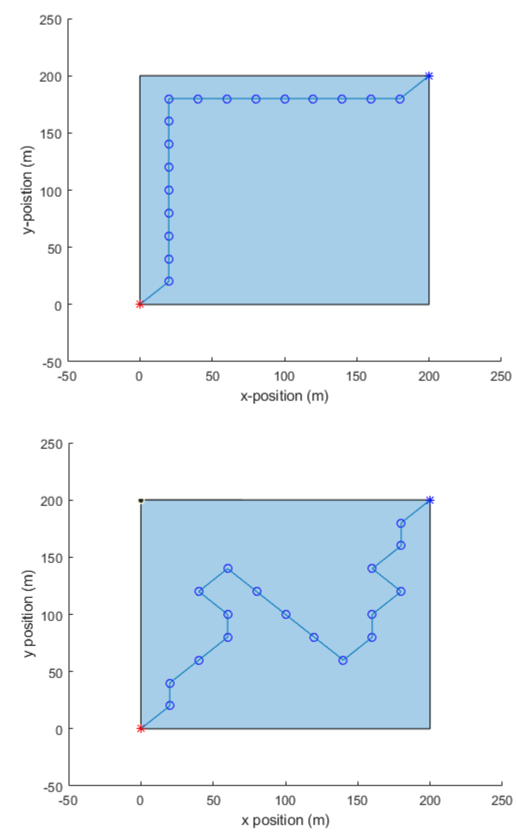


Fig. 4. Example of inter-region path generated to cover 10 regions. Depot is denoted by region 1



[a]

[b]

Fig. 5. Example of coverage path generated by proposed algorithm for intra-region coverage of a 200 m x 200 m square region with starting waypoint (vertex of region) marked as red asterisk and ending waypoint (vertex of region) marked as blue asterisk [a] initial random path for [b] optimized final path

A. *Target Site Generation*

The parameters corresponding to the target site which needs to be covered have been generated randomly. The number of regions in the sites is given by the user with an upper bound of 100 regions. The above-mentioned assumption is based on the energy budget an aerial agent needs to perform area coverage [13]. The regions in any given site are distributed around an origin denoted by the coordinates (0,0). Any new region is created by generating the coordinates of its centroid. These coordinates are generated by first selecting the distance from the origin through a uniform random distribution U1 $[0, 200]$ and then by determining the angle that it subtends on the x axis, using another uniform random distribution U2 $[0, 360]$.

The new region is created using built-in MATLAB function $nsidedpoly()$ [20]. This function takes as input the number of vertices, the coordinates of the centroid, and length of each sides of the region. $nsidedpoly()$ generates a regular polygon based on the input parameters. The number of vertices is generated through a uniform random distribution U3 $[3, 10]$. The length of the side of a region is generated using normal distribution with standard deviation as $\sigma_{side} = 2.5$ and the mean as $\mu_{side} = 7.5m$. The new region is accepted if and only if it does not overlap with already created region; otherwise, the region is discarded. These steps are repeated unless until the desired number of regions have been generated. The size of each grid cell is assumed as $1 \times 1\ m^2$.

B. *Path Generation Examples*

In the first example, the inter-region algorithm was run in a target site with 10 regions. The region 1 is selected as the starting region (depot). The inter-region is shown in Fig. 4. In the second example, an intra-region path is illustrated for a square region with dimension 200 m x 200 m, which is discretized into a 200 × 200 cell grid. The initial random path is depicted in Fig. 5[a]. The algorithm optimizes to the path shown in Fig. 5[b].

C. *Algorithm Characterization*

In order to characterize the proposed algorithm, several experiments were run in simulation. First experiment is conducted to determine the runtime of the algorithm with respect to the number of regions in the target site. The second set of experiments were conducted to compare the algorithm optimality and runtime of the proposed algorithm against two other algorithms focusing on area coverage of distributed regions. The first comparison algorithm is the method described by Xie et al [16]. This algorithm was chosen because it is the only work in literature which determines the path for area coverage of multiple distributed regions by the energy constrained UAV. However, due to the fact that the Xie method performs multiple tours to provide complete area coverage and our the proposed work computes coverage path which perform partial coverage in a single tour. Therefore, a slight modification was made to Xie method where the area covered in a single tour was only considered to bring both the methods on similar platform. The second comparison algorithm is a path generation method which covers the area by Boustrophedon path for intra-region and using nearest neighbour algorithm for inter-region traversal as described in [4]. This algorithm was chosen because it is regularly used for area coverage, with much of the work in the field of CPP generating these sorts of coverage paths for various applications.

1) Average Runtime: To determine the time complexity of the proposed algorithm, an experimental setup is designed by fixing the waypoint spacing d to be 1 grid cell. The number of regions are varied across the range of 25 to 100. For each different number of region, 100 trials of experiment is performed and average runtime per iteration is computed. The results are illustrated in Fig. 6, depicting that as the number of regions increase, the average run time increases. This increment can be attributed to the fact that as the number of regions would increase, the algorithm would require more time to compute inter-region path and optimise the intra-region path to achieve near-optimal area coverage.

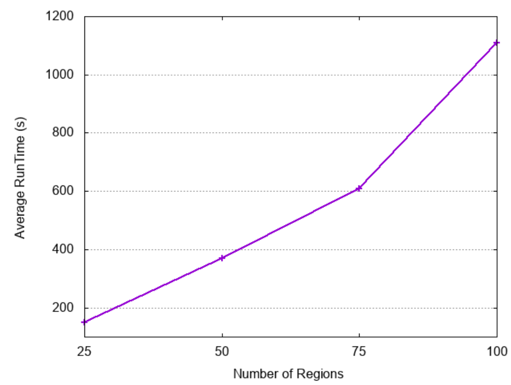


Fig. 6. Average runtime for different number of regions

2) Normalized Area Coverage Comparison: We assess the optimality of the proposed algorithm, by measuring normalized area coverage. Further, this parameter is used to compare the proposed algorithm with the two previously described comparison algorithms. Five different target sites are generated for the experiments by varying the number of regions from the range of $\{10, 20, 30, 40, 50\}$. All the algorithms are executed 10 times for each number of regions.

The normalized area coverage is computed as the ratio of the partial area covered by the UAV and total area of the target site. The results are shown in Fig. 7 . The proposed algorithm generated path is able to achieve normalized area coverage between 55% to 73%, on the other hand, the method [16] achieves partial area coverage between 34% to 47%. The second method [4] is able to cover 26% to 39% area.

IV. Conclusion and Future Work

This paper presents a partial coverage method for planning near-optimal area coverage paths for multiple distributed regions using an aerial robot given energy limitations. The

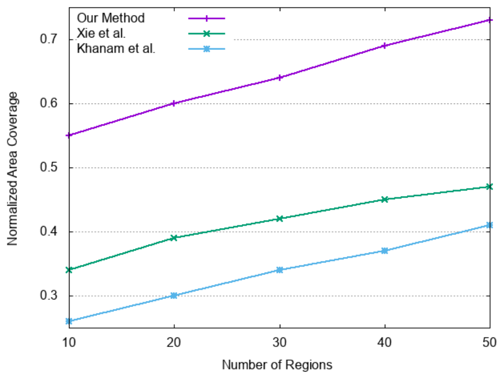


Fig. 7. Comparison of Normalized Area Coverage for different number of regions

coverage path generation algorithm first computes an inter-region order traversal and then distributes path waypoints for intra-region coverage. Little work has been done previously exploring generation of area coverage path for multiple disconnected regions due to energy constraints, and the path-planning algorithm depicted in this paper is a novel approach to solving that problem by generating path which aims to achieve near-optimal partial area coverage with given energy budget.

Simulations are conducted to compare the proposed method with other methods that can generate coverage paths for distributed multiple regions. The algorithm presented in this paper provides paths much closer coverage to the full coverage than the state-of-the-art area coverage methods. The reason behind moderately better performance in terms of area coverage, due to difference in algorithm design which focuses on optimizing intra-region area coverage by traversing a set of waypoints.

The proposed algorithm suffers from two limitations. The first limitation is that the intra-region path is devised by optimizing the cost function using an built-in MATLAB function. This optimal solution strategy will be prohibitively expensive in terms of solution generation times and required storage space. Further, it will also be difficult to deliver satisfactory outputs as the number of regions will increase. Hence, we need to propose a heuristic strategy, which can generate considerably good solutions within reasonable time. The second limitation is that this algorithm require an accurate energy model, assuming that the UAV is able to rotate in place. Few possible direction for future research to overcome this limitation is to physically test this method on a real-world aerial robotic platform.

ACKNOWLEDGMENT

This work is supported by the UK EPSRC through grants EP/R02572X/1 and EP/P017487/1.

REFERENCES

[1] I. A. Hameed, D. Bochtis, and C. A. Sørensen, "An optimized field coverage planning approach for navigation of agricultural robots in fields involving obstacle areas," *International journal of advanced robotic systems*, vol. 10, no. 5, p. 231, 2013.

[2] A. Villa-Henriksen, G. T. Edwards, L. A. Pesonen, O. Green, and C. A. G. Sørensen, "Internet of things in arable farming: Implementation, applications, challenges and potential," *Biosystems engineering*, vol. 191, pp. 60–84, 2020.

[3] K. Zhou, A. L. Jensen, C. Sørensen, P. Busato, and D. Bothtis, "Agricultural operations planning in fields with multiple obstacle areas," *Computers and electronics in agriculture*, vol. 109, pp. 12–22, 2014.

[4] Z. Khanam, S. Saha, S. Ehsan, R. Stolkin, and K. McDonald-Maier, "Coverage path planning techniques for inspection of disjoint regions with precedence provision," *IEEE Access*, 2020.

[5] S. S. Mansouri, C. Kanellakis, E. Fresk, D. Kominiak, and G. Nikolakopoulos, "Cooperative coverage path planning for visual inspection," *Control Engineering Practice*, vol. 74, pp. 118–131, 2018.

[6] A. K. Lakshmanan, R. E. Mohan, B. Ramalingam, A. V. Le, P. Veerajagadeshwar, K. Tiwari, and M. Ilyas, "Complete coverage path planning using reinforcement learning for tetromino based cleaning and maintenance robot," *Automation in Construction*, vol. 112, p. 103078, 2020.

[7] M. Torres, D. A. Pelta, J. L. Verdegay, and J. C. Torres, "Coverage path planning with unmanned aerial vehicles for 3d terrain reconstruction," *Expert Systems with Applications*, vol. 55, pp. 441–451, 2016.

[8] I. A. Hameed, "Intelligent coverage path planning for agricultural robots and autonomous machines on three-dimensional terrain," *Journal of Intelligent & Robotic Systems*, vol. 74, no. 3, pp. 965–983, 2014.

[9] S. Dogru and L. Marques, "Improved coverage path planning using a virtual sensor footprint: a case study on demining," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 4410–4415.

[10] I. A. Hameed, "Coverage path planning software for autonomous robotic lawn mower using dubins' curve," in *2017 IEEE International Conference on Real-time Computing and Robotics (RCAR)*. IEEE, 2017, pp. 517–522.

[11] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous systems*, vol. 61, no. 12, pp. 1258–1276, 2013.

[12] T. M. Cabreira, L. B. Brisolara, and P. R. Ferreira Jr, "Survey on coverage path planning with unmanned aerial vehicles," *Drones*, vol. 3, no. 1, p. 4, 2019.

[13] J. Xie, L. R. G. Carrillo, and L. Jin, "An integrated traveling salesman and coverage path planning problem for unmanned aircraft systems," *IEEE control systems letters*, vol. 3, no. 1, pp. 67–72, 2018.

[14] ——, "Path planning for uav to cover multiple separated convex polygonal regions," *IEEE Access*, vol. 8, pp. 51 770–51 785, 2020.

[15] J. I. Vasquez-Gomez, J.-C. Herrera-Lozada, and M. Olguin-Carbajal, "Coverage path planning for surveying disjoint areas," in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2018, pp. 899–904.

[16] J. Xie and J. Chen, "Multi-regional coverage path planning for robots with energy constraint," in *2020 IEEE 16th International Conference on Control & Automation (ICCA)*. IEEE, 2020, pp. 1372–1377.

[17] K. R. Jensen-Nau, T. Hermans, and K. K. Leang, "Near-optimal area-coverage path planning of energy-constrained aerial robots with application in autonomous environmental monitoring," *IEEE Transactions on Automation Science and Engineering*, 2020.

[18] K. O. Ellefsen, H. A. Lepikson, and J. C. Albiez, "Multiobjective coverage path planning: Enabling automated inspection of complex, real-world structures," *Applied Soft Computing*, vol. 61, pp. 264–282, 2017.

[19] C. Papachristos, K. Alexis, L. R. G. Carrillo, and A. Tzes, "Distributed infrastructure inspection path planning for aerial robotics subject to time constraints," in *2016 international conference on unmanned aircraft systems (ICUAS)*. IEEE, 2016, pp. 406–412.

[20] "Regular polygon-matlab," 2020. [Online]. Available: https://uk.mathworks.com/help/matlab/ref/nsidedpoly.html