Title	Rapid Coverage of Regions of Interest for Environmental Monitoring
Author(s)	Pinkam, Nantawat; Newaz, Abdullah Al Redwan; Jeong, Sungmoon; Chong, Nak Young
Citation	Advances in Intelligent Systems and Computing, 751: 195-209
Issue Date	2018-05-31
Type	Journal Article
Text version	author
URL	http://hdl.handle.net/10119/16103
Rights	This is the author-created version of Springer, Nantawat Pinkam, Abdullah Al Redwan Newaz, Sungmoon Jeong, Nak Young Chong, Advances in Intelligent Systems and Computing, 751, 2018, 195-209. The original publication is available at www.springerlink.com, http://dx.doi.org/10.1007/978-3-319-78452-6_17
Description	International Conference on Robot Intelligence Technology and Applications, RiTA 2017: Robot Intelligence Technology and Applications 5.

Rapid Coverage of Regions of Interest for Environmental Monitoring

Nantawat Pinkam[1], Abdullah Al Redwan Newaz[2], Sungmoon Jeong[1], and Nak Young Chong[1]

[1] School of Information Science, Japan Advanced Institute of Science and Technology, 1-1 Asahidai, Nomi, Ishikawa 923-1292 Japan,
{pinkam.nantawat, jeongsm, nakyoung}@jaist.ac.jp
[2] Graduate School of Information Science, Nagoya University, Furo-cho, Chikusa-ku, Nagoya 464-8603, Japan,
redwan.newaz@jaist.ac.jp

Abstract. In this paper, we present a framework to solve the problem of rapidly determining regions of interest (ROIs) from an unknown intensity distribution, especially in radiation fields. The vast majority of existing literature on robotics area coverage does not report the identification of ROIs. In a radiation field, ROIs limit the range of exploration to mitigate the monitoring problem. However, considering the limited resources of Unmanned Aerial Vehicle (UAV) as a mobile measurement system, it is challenging to determine ROIs in unknown radiation fields. Given the target area, we attempt to plan a path that facilitates the localization of ROIs with a single UAV, while minimizing the exploration cost. To reduce the complexity of exploration of large scale environment, initially whole areas are adaptively decomposed by the hierarchical method based on Voronoi based subdivision. Once an informative decomposed sub area is selected by maximizing a utility function, the robot heuristically reaches to contaminated areas and then a boundary estimation algorithm is adopted to estimate the environmental boundaries. Finally, the detailed boundaries are approximated by ellipses, called the ROIs of the target area and whole procedures are iterated to sequentially cover the all areas. The simulation results demonstrate that our framework allows a single UAV to efficiently and explore a given target area to maximize the localization rate of ROIs.

Keywords: Environmental Monitoring, Regions of Interest Coverage, Energy-Efficient Path Planning, UAV

1 Introduction

In a large radiation field, it is important to localize Regions of Interest (ROIs) to monitor the radiation effects, to localize the hotspots, the sources, and so on. Recent advances in Unmanned Aerial Vehicle (UAV) offers the ability to access and navigate in unstructured or cluttered environments. Therefore, a single UAV equipped with dedicated sensors makes an attractive platform for such kind of tasks. However, it is difficult to monitor a large field with single UAV. In such situations, it becomes necessary to design a path planner that can localize the ROIs rapidly.

Radiation field monitoring has been commonly studied in robotics [6,7]. The goal is to plan a path in which the robot can localize all the contaminated locations in a given target area. Since the contaminated locations could be spatially distributed throughout the target area, a search is needed to localize all of them. Thus, required tasks associated with the search inspire various methods in addressing the coverage problem. Spatial search techniques should be fitted according to the number of robots used for this application. In the case of multiple robots, the target area can be partitioned into smaller subregions to reduce the search space for each robot. The search strategy is exclusively benefited by the number of robots and the communication among them. However, in the case of a single robot exploration, the partitioning of the target area benefits neither the exploration cost, nor the accuracy.

The early survey on coverage algorithms was provided by Choset [3], where he classified the solution approaches either based on heuristic or cell decomposition. Heuristic methods explore the target area with predefined rules or a set of behaviors. The widely used heuristic methods are lawnmower pattern, raster scanning, inward spiral search, wall following, etc. Heuristic search is computationally less expensive, but cannot guarantee the optimal performance. On the other hand, in cell decomposition, the target area is decomposed into smaller areas. Galceran and Carries [5] provided a survey of an exact and uniform decomposition of the target area by a grid of equally spaced cells. Then, the coverage problem can be solved as the Traveling Salesman problem and is known to be NP-hard. Usually, in that case, a Hamiltonian path is determined using the spanning tree algorithm, which visits each cell exactly once. In recent year, a variant of Hamiltonian path utilized for the persistent coverage problem [10]. However, if there are obstacles in the target area, it is not possible to generate the Hamiltonian path in all the cases. The Boustrophedon cellular decomposition can then solve this problem for bounded planar environments with known obstacles [14]. The key idea is to construct a graph by decomposing the target area subject to obstacle positions and finding a minimal cost tour through all regions. In literature, we have seen an extension of that algorithm while respecting sensor feedback [1,11,12]. When unknown obstacles exist in the environment, the Morse decomposition used for determining critical points in the target area, and then incrementally construct the Reeb graph to solve the online coverage problem optimally [2]. Another way is to satisfy a temporal logic specification consisting of safety components in a partially unknown environment [8].

The majority of coverage planning work has been proposed for known environments [12,15,16]. Often these approaches are motivated to minimize the uncertainty metric of a given map. A common choice is to add an exploration to that location where the uncertainty metric such as entropy or mutual information is high. However, in many situations, a radiation map for the target area may not be *a priori* available. The problem can then be closely related relation to covering the entire target area for localizing the contaminated locations. Hence, complete coverage algorithms are often used. Even though complete coverage algorithms ensure the complete terrain visitation, they lack the opportunity to optimize the localization rate of contaminated locations.

Considering estimation on environmental boundaries instead of the complete coverage provides a useful abstraction that reduces the energy consumption [9,13]. Here, the path planning problem consists of estimating boundary of contaminated areas that allow the robot to sense the ROIs. However, when the environment is unknown, it is hard to plan a path that identifies which areas are interesting and which are not. In conventional algorithms for the coverage planning with obstacles, the path is usually generated to cover the free space of the environment in an optimum fashion. In our problem, rather than avoiding ROIs, we want to identify locations and geometrical size of them rapidly. For example, when the robot opportunistically finds contaminated areas, firstly, it can expedite the boundary estimation process to determine ROIs, and then it can bypass exhaustively covering the entire regions. Determining ROIs in a radiation field allows us to prioritize the search area in such a way that minimizes the exploration of the robot.

In this work, motivated by a single UAV coverage, we investigate an additional component to the coverage problems by incorporating a localization rate factor for the radiation contaminated locations. Taking account of the localization rate factor which is important in a single UAV exploration, sometimes the target area is too large for the UAV to completely cover with limited exploration budget (maximum exploration time). Since it is also of the interest that the UAV is to localize all the contaminated locations as quickly as possible, the algorithm must behave as the complete coverage over long periods of operation. This problem might be thought of as target acquisition problems [4]. However, there is an important caveat. Target acquisition problems assumed that the robot equipped with a sensor that has a wide field of view, whereas in our problem, the robot sensor works in a point-wise fashion. Therefore, the robot needs to travel to a location to get a measurement.

In this paper, we discuss the online version of this problem, in which the coverage path of the robot is to be determined based on the information gain metric from the past exploration. To reduce the search space, we initially partition the target area in a random manner. Next, we update the partition size based on the size of ROIs. We propose an optimal path planner, which extends the complete coverage algorithm to reason about a localization rate factor. Under the assumption that there exist multiple ROIs in a given target area, the proposed algorithm can increase the localization rate of contaminated locations while guaranteeing a complete coverage path over long periods of operation.

The contributions of this work are as follows:

1. We have formulated the localization of ROIs which does not require *a priori* information at all.
2. Our algorithm can localize ROIs in a fast manner by minimizing the exploration of UAV.
3. The proposed algorithm is complete, which means all contaminated locations are identified for the long operation of UAV.
4. Focusing on the limited computational capabilities of the UAV, the proposed algorithm can robustly determine ROIs.
5. To best of our knowledge, this is the first approach that integrates the environmental boundary estimation problem to the area coverage problem.

To discuss the aforementioned topics, this paper is organized as follows: in Section 2, we describe the problem formulation; Section 3, we present the heuristic coverage algorithm based on adaptive hierarchical area decomposition. Section 4, we briefly explain generalization process of ROIs. Finally, in Section 5 and 6, we present simulation results and conclude our findings.

2 Problem Formulation

We are given a target area T, which contains radiation sources, strength of can be sensed by the robot. We assume that T can be decomposed into a regular grid with n cells. Let us denote this grid by G. Since radiation sources might be spatially distributed. Thus, G contains two type of cells, i.e., free cells and contaminated cells. Furthermore, nearby sources cumulatively affect the target area, resulting in a joint distribution of measurement attributes. Let us assume that each cell c is associated with a measurement attribute z. The robot is equipped with a sensor to make a point-wise measurement $z(t)$ at its position $x(t)$ at time t. The Regions of Interest (ROIs) in T are those cells $\mathbf{J} := \{c|z > 0\}$ where the robot finds $z > 0$. The contaminated areas are contiguous. Therefore, the robot can trace such areas by tracking only to the boundaries. Therefore, the definitions of the contaminated and the free cell are quantified through a binary probability value given by

$$p_c = \begin{cases} 0, & \text{if } z \approx 0 \\ 1, & \text{otherwise} \end{cases} \tag{1}$$

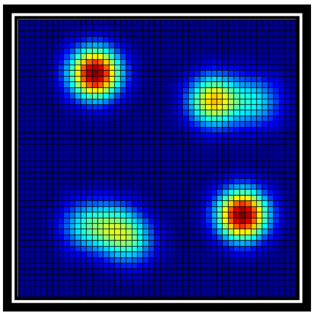

Fig. 1. The dark blue cells have no measurement attributes whereas other colored cells represent the measurement attributes.

Fig. 1 shows an example world map of size 50×50. Depending on the spatial locations of the radiation sources, measurement attributes are also spatially distributed throughout T. The dark blue cells are the cell where $p_c = 0$. The other colored cells represent the fact that measurement attributes are available such that $p_c = 1$. We can then find multiple ROIs while splitting $\mathbf{J}$ subject to spatial distances.

Definition 1. ***Regions of Interest (ROIs):*** *A collection of cells corresponds to a set of contaminated locations in a given target area T, i.e. the set $\{\mathbf{J} \in T | p_c = 1\}$.*

The global mission of the robot can be defined in two different ways, which implies two different objective functions as follows

- the minimum time to localize an ROI,
- the total time to localize all the ROIs in T.

Without loss of generality, we assume that the travel time is proportional to the travel distance. Therefore, firstly, we will use the boundary estimation technique that minimizes the robot's exploration to localize an ROI. Secondly, we will use the heuristic area coverage technique that ensures to localize all the ROIs in T. The total time is taken into account by summing up the boundary estimation paths and the heuristic area coverage paths.

Let us formally define these objective functions. First, starting from an initial cell, we denote the coverage path followed by the robot throughout the free cells by $\mathscr{P}$. We assume that, $|\mathbf{J}| << n$ *i.e.*, the contaminated cells are far fewer than the number of free cells. We define the event $S_{\mathscr{P}}$ as the event that the robot reaches to any ROI which is not localized beforehand. The complete coverage path $\mathscr{P}$ can be then discretized by the presence of ROI. Therefore, the probability to find an ROI can be expressed as follows

$$E[S_{\mathscr{P}}] = \sum_{c \in \mathscr{P}} (1 - p_c) \,. \tag{2}$$

Thus, the first objective is to find an online coverage path that minimizes $E[S_{\mathscr{P}}]$. Note that, in this objective, the heading of the path is not important, once the robot heuristically reaches any location of an ROI, the boundary tracking algorithm is followed to determine the ROI size.

For the second objective, we denote the sequence of newly discovered ROIs along the coverage path $\mathscr{P}$. if there exists k number of ROIs in T, we discretize $\mathscr{P}$ into a subset $Q = \{q_1.q_2, ..., q_k\}$. Since the travel time is proportional to the length of q_k, we want to find the minimum length paths in the set Q to localize all the ROIs. Therefore, the total events $C(\mathscr{P})$ that the robot is experienced to localize a finite set of ROIs given by

$$C(\mathscr{P}) = \sum_{q_k \in Q} S_{q_k} \; s.t. \; |Q| \leq |ROI|, \tag{3}$$

where $|Q|$ is the cardinality of set Q and $|ROI|$ is the number of ROIs are detected in T. If $|ROI|$ is *a priori* given, our focus is to find the minimum exploration time to achieve that number. We then derive the performance index of the robot from eq. (3). A formal definition of the performance index as follows.

Definition 2. ***Performance Index (PI):*** *The performance index of the robot is evaluated with respect to the minimum explored path to localize all of ROIs, i.e.* $PI = \operatorname{argmin} C(\mathscr{P}) \; s.t. \; |Q| \leq |ROI|$.

Since we do not know the exact number of ROIs exists in T, it is not possible to stop the robot's exploration when all ROIs are localized. Then, the robot exploration can be terminated by exploration budget. Otherwise, the robot's task is to plan an online path through T such that every ROIs is rapidly localized while subject to complete area coverage.

3 Adaptive Hierarchical Area Decomposition And Coverage

Fig. 2 shows the overall schematic of our proposed system. The algorithm we propose can be broken down into three steps. In the first step, *Adaptive Hierarchical Area Decomposition*, we adaptively partition the target area in hierarchical order to reduce the search space of the robot. We then find the subregions given by the partition using the *Finding subregions*. When the subregions are determined, we examine the utility to traverse each subregion that explained in the *Utility function design*. The subregion which has maximum utility, we plan a coverage path through the set of unvisited cells. The robot progresses through this path. If the robot notices an ROI along its path, it will drop exploring more and iterates whole steps. Otherwise, the whole steps iterated after traveling along the entire path.

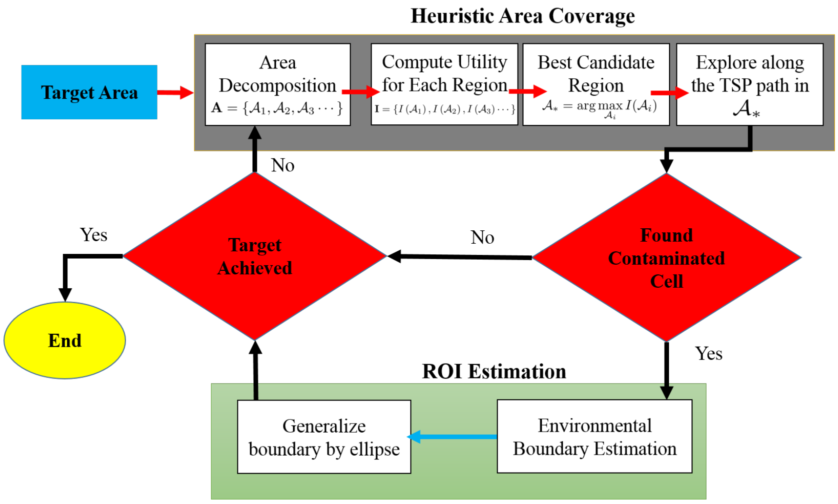

Fig. 2. System Overview: The figure shows all the steps performed by the heuristic area coverage, and ROI estimation algorithms. Starting from an arbitrary location, the robot can iteratively localize the desired number of ROIs using this framework.

3.1 Adaptive Hierarchical Area Decomposition

To reduce the computational complexity while navigating a large environment, the search space for the path planning needs to be at a tractable level. We argue that these objectives can be achieved by adaptive partitioning of the target area in hierarchical order. Given the position of ROIs, the hierarchical order is determined by a local minimum distance with the respect to the robot's relative position. Therefore, we propose Voronoi-based partition in the sense of limiting the search space. Fig. 3 shows the overall overview of each algorithm. With a given partition, our goal is to find an ROI through the limited exploration.

The Voronoi-based subdivision (VBS) uses the Voronoi-based approach to partition the target area. The main idea is to partition the area by representing the ROI centers as the Voronoi centroids. Since in our case the ROI centers are not apriori available, we have introduced a few changes to the original Voronoi-based partition algorithm.

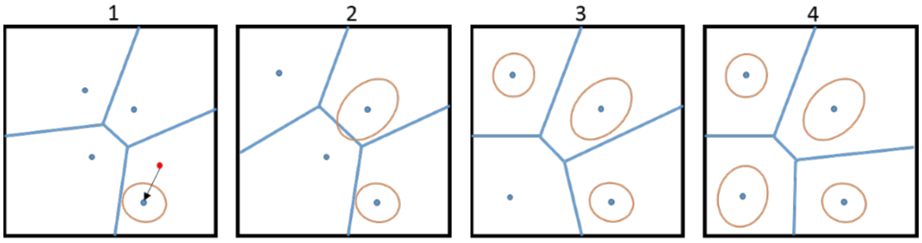

Fig. 3. VBS decomposition: Starting with the random partitions, the partitions updated by the center position of ROIs. The algorithm iteratively approaches to optimal decomposition.

Firstly, it randomly partitions the target area using four random points inside the target area. Secondly, it leads the robot to the nearest centroid from its initial location. Finally, TSP algorithm generates the coverage path. The robot starts to explore along this route when a contaminated cell found; it switches to the boundary estimation planner. An ROI then computed from the estimated boundary. The robot finds a minimum route to ROI from its location either while traveling to the Voronoi centroid or while executing TSP path. Although these paths increase the probability of finding ROI, if there are no contaminated cells in the subregion, then the complete coverage path would be large because of traveling to the centroid. Note that in VBS, the initial search space limited by the random partition. The partition of the target area updated by the center position of the detected ROI.

In the second phase, VBS finds a coverage path that connects the desired number of ROIs. Finding such path is possible by iteratively updating the Voronoi centroids. The iterative updates of centroids lead VBS to generate an optimal partition of the search space. However, when the number of ROIs is greater than the number of random initial points, the partition centroids are not only iteratively updated but also incrementally constructed. The four basic operations of this decomposition are as follows. Firstly, we generate randomized incremental construction of partitions to reduce the search space. Secondly, the robot moves to the Voronoi centroid, and TSP algorithm creates a coverage path to explore the unexplored cells optimally of a given subdivision. Thirdly, when an ROI is determined, we terminate the exploration and update the Voronoi centroids. Finally, the region of each division is determined.

We demonstrate the Voronoi-based subdivision while the robot is covering its free space using an example depicted in Fig. 3. Voronoi Diagram is the partitioning method of a plane with n points into a specific subset of the plane such that each subset contains exactly one generating point. In typical Voronoi diagram, the set of generating points is *apriori* known. The Voronoi polygons are then constructed such that every point in a given polygon is closer to its generating point than to any other. However, in our case, we randomly initialize the generating points and iteratively update their positions.

The robot starts to cover the space in a vast cell by moving into the centroid of the current Voronoi region (red dot) which is located at the rightmost corner; the target area is shown as the black rectangle in Fig. 3. Then, the robot constructs a TSP path to cover the given region. Whenever the robot reaches the cell where $P_c = 1$, which is the unvisited location of a contaminated area, it finishes covering the centroid path or the TSP path. Since the contaminated area is unknown *a priori*, the robot follows the boundary tracking algorithm to cover it. The robot then constructs an ellipse over the estimated boundary to represent the ROI, shown as the orange ellipse in Fig. 3. At this point, it encounters the update of Voronoi centroid. The Voronoi centroid of the current region is replaced by the center point of the ellipse, shown as blue dots in Fig. 3. If there are more ROIs than the Voronoi centroids which are chosen initially, the overall Voronoi partitions are reconstructed with updated centroids. Note that the minimum number of subdivisions in this case is four, and the algorithm can also cover more than four subdivisions. The robot chooses the subdivision that maximizes the utility function and repeats the step described above as shown in Fig. 3. Since the Voronoi regions are connected, the robot is guaranteed to visit all the subdivisions in the target area, and thus completely cover the space.

3.2 Finding subregions

At the end of the second phase, each algorithm finds the subregions based on its partition method. For this purpose, it begins by creating the graph $G = (V, E, B)$ induced from above mentioned methods. We represent the target area as a rectangular box B in G. The initial partitions are the edge set E that includes edges with infinite lengths. To find subregions Λ, firstly, we shorten each edge $e \in E$ subject to B. Let V be the set of vertices that includes three types of subsets such that $V = \{\{\psi_G\}, \{\psi_b\}, \{\psi_c\}\}$. Let ψ_G be the first subset of V that represents the vertices at the intersection between B and E. Also, let ψ_b be the set of vertices that represents the corner points of B, and let ψ_c be the centroid of ROIs. Once we trim the long edges, the new partition represented by E_ψ. Secondly, we find all the possible combination of edges on B and represent by E_b. The G is then updated by combining these two set of edges such that $E \leftarrow \{\{E_b\} \cup \{E_\psi\}\}$. Finally, we group all subregions Λ by finding the neighbor edges. Finding such a neighbors is straightforward. Given ψ_c, an anti-clockwise walk along the E can sort such neighbors.

3.3 Utility function design

In the third phase, each algorithm finds the best search space among all subdivisions of the target area. For this action, it computes the utility between each of subdivisions. The utility is designed to favor destinations which offer higher expected information gain. Throughout this work, we use an explored grid map, m, to model the environment. This map is a binary map where each cell represents visited or unvisited information. Let i be the index of each subdivision and the division of such a map satisfies the following equation

$$m = \sum_i m^{[i]}. \tag{4}$$

An action a_t generated at time step t is represented by a sequence of relative movements $a_t = \hat{u}_{t:T-1}$ which the robot has to carry out starting from its current position x_t. During the execution of a_t, if the robot finds a contaminated cell along its path, then it estimates an ROI in the map. Therefore, the explored trajectory of the robot indicates some of the cells in m as follows

$$x_{1:t} = \exists c \in m. \tag{5}$$

In the case when the robot finds an ROI in the map, we have to treat the ROI cells differently. We assumed that traveling inside an ROI is redundant, and want to avoid such a region. Therefore, the cells bounded by an ROI considered as similar as visited cells. Let d_t be the set that represents these cells as follows

$$d_t = \{\forall c \in ROI1, \forall c \in ROI2 \cdots\}. \tag{6}$$

Assuming that each cell c in m is independent of each other. Then the posterior entropy of m can be computed as follows

$$\begin{aligned} H(p(m|x_{1:t}, d_t)) = -\sum_{c \in m} & p(c) \log p(c) + \\ & (1 + p(c)) \log(1 - p(c)). \end{aligned} \tag{7}$$

Given a subdivision, since the robot does not know when it will find an ROI along its path, the coverage path should include all cells to compute the expected information gain. Thus, the entropy of target subdivision can write as follows

$$\begin{aligned} H(p(m^{[i]}|x^{[i]}_{t+1:T}, d_t, a_t)) = -\sum_{c \in m^{[i]}} & p(c) \log p(c) + \\ & (1 + p(c)) \log(1 - p(c)). \end{aligned} \tag{8}$$

To compute the information gain of a subdivision, we calculated the change in entropy caused by the integration of posterior and predicted prior into the robot's world model as follows

$$I(m^{[i]}, a_t) = H(p(m|x_{1:t}, d_t)) - H(p(m^{[i]}|x^{[i]}_{t+1:T}, d_t, a_t)). \tag{9}$$

After computing the expected information the utility for each action under consideration, we select the action a_t^* with highest expected information

$$a_t^* = \arg\max_{a_t} I(m^{[i]}, a_t). \tag{10}$$

There are some works in exploration and mapping problems that consider another quantity besides the information gain in Eqn. (10). That is the cost to reach the subdivision. However, we observed that adding such a quantity with the utility function decreases the overall performance of both algorithms. Thus, every time the robot has to make the decision where to go next, it uses only information maximization metric to determine the action a_t^*.

4 Finding ROIs

We employ a boundary estimation algorithm to determine the ROIs by using the proposed exploration method. ROIs over the target area T are dependent on the boundary line estimated by environmental boundary algorithm. Memorizing a complex boundary is computationally expensive, therefore to obtain the tractable level of computation, we require the parametric estimation of the boundary.

Definition 3. ***Boundary line:*** *The line is said to be boundary line if it represents the intersection between the contaminated area and non-contaminated areas.*

Assume an contaminated area $\delta\mathscr{A}$ is a non-convex set where the continuous boundary is defined by a level set $\delta\mathscr{A}$ as follows

$$\delta\mathscr{A} = \left\{x \in \mathbb{R}^2 | z(x) = \beta\right\}, \tag{11}$$

where β is the measurement threshold.

Boundary algorithm ensures that an environmental boundary can be estimated by tracking the robot states such that $\delta\mathscr{A} = \{x_{1:t}\}$. When the exploration is terminated, this set $\delta\mathscr{A}$ can be used to estimate of the best fit to an ellipse. This generalization is done by the least squares criterion from the set $\delta\mathscr{A}$. We also consider the possible tilt of the ellipse from the conic ellipse representation as follows

$$ROI(\delta\mathscr{A}) = a{S_x}^2 + bS_xS_y + c{S_y}^2 + dS_x + eS_y + f = 0, \tag{12}$$

where $\{S_x, S_y\} \in \delta\mathscr{A}$ and a, b, c, d, e, f are the parameter for a second degree polynomial equation. After the estimation, the tilt is replaced by a rotation matrix from the ROI, and then the rest of parameters are extracted from the conic representation.

5 Simulation Results

To find the shortest coverage path, we perform 4 different experiments using MATLAB. We assume that the target area contains at most 5 ROIs. The performance of each algorithm was evaluated by the distance of coverage paths. To demonstrate the efficiency, we start localizing 2 out of 5 ROIs and conclude by 5 out of 5 ROIs. We also have analyzed the worst case performance and we present a statistical analysis of two algorithms from 20 trial runs. The performance of algorithms significantly varied from each other. In particular, we have observed a noticeable difference of the algorithms on localizing uniformly distributed random ROIs. It is noteworthy that to compute the efficiency, the ROIs shape should remain fixed for each algorithm, we then overlook the additional path cost required to estimate ROIs.

5.1 Finding coverage path that connects the desired number of ROIs

We now consider the case of finding ROIs that meet the desired level of exploration. Therefore, we focus on the shortest coverage path for a given number of ROIs. We

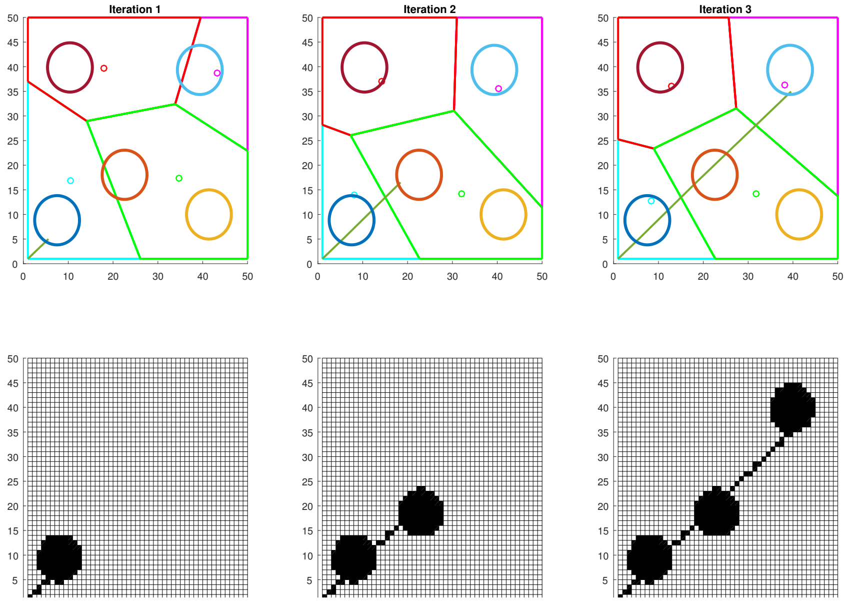

Fig. 4. The robot starts the coverage in cell (1,1) and detects any 3 ROIs out of 5. The shape of each ROI is elliptical and is represented in unique color. The lower grid map represents the coverage map. The measured cells are represented by black color. A cell is called to be measured if it is included either in coverage trajectory or it is bounded by the detected ROIs. VBS coverage paths on a sample map with uniformly distributed random ROIs. The dark green line in upper figure shows the coverage path, while the colored lines are the partition of the target area. The centroid of each region is represented by the same colored cycle. For a new region, the searching process is started from the centroid. The partitions are iteratively updated based on the true position of the center of ROIs.

consider a $50m \times 50m$ grid area where 5 uniformly distributed random ROIs are located. Starting from an initial location $(1,1)$, the robot has to find the minimum coverage path that connects the desired number of ROIs. The coverage path can be found by adjusting the cost to the inversely proportional to the unexplored area. In another word, the robot explores the mostly unexplored region first.

Fig. 4 shows a toy example of VBS algorithm. In VBS, the initial search space is generated by randomly choosing 4 points bounded in the target area. We will call these points as the Voronoi centroids. The initial search space is then subdivided into four regions based on the Voronoi centroids. The robot moves the centroid of a Voronoi region first and exhaustively search for an ROI within that region. When an ROI is found whether traveling to the centroid or searching the entire subregion, the robot updates the Voronoi diagram. The robot avoids exploring the cells bounded by the ROI. These processes are iterated until the end of the mission. The VBS requires at least 3 points to partition the entire search space optimally. When there are less than 3 ROIs in total area and the robot has to localize all of them, the VBS performance is not stable.

5.2 Performance comparison

We compare VBS algorithm to recursive quadratic subdivision (RQS) which follows a greedy approach, wherein each step it leads the robot to the nearest ROI to its current location that has not been covered yet. The three basic operations of RQS are as follows. Firstly, we generate a TSP path to explore the unexplored cells optimally. Secondly, when an ROI is determined, we terminate the exploration and decompose the area. Finally, the region of each division is determined.

Fig. 5 shows a performance comparison. To access the long-term performance of each algorithm, we ran the same experiments for 20 times by gradually increase the target numbers. Fig. 5 shows the results in area coverage percentage metric. We divided the given target area into three different regions- 1) explored by the robot 2) covered by the ROIs 3) remained unexplored. Our goal is to minimize the explored region as small as possible. To make a fair comparison, we use five uniformly distributed random ellipses and try to find the shortest path that connects 2, 3, 4, and 5 ROIs. For them, the covered regions by ROIs are 6, 9, 13, and 16 percentage of the target area. The unexplored region then determined by subtracting the covered and explored regions from the total area.

The reduction of search space directly influences of the explored areas. When the number target of ROIs is less than total ROIs existed in the target area, the robot dramatically reduces the amount of explored region. In worst case scenario, when the robot needs to localize all five ROIs, it requires traveling more locations to find the ROIs, resulting in higher exploration regions. However, the performance of each algorithm is not stable, and we use the error bar of the bar chart to represent their standard deviation (SD). For both algorithms, the SD increases with the increment of the number of target ROIs.

It is evident from the Fig. 5, the VBS always outperforms the RQS because of the optimal search space division strategy. Furthermore, when the number of target ROIs is less than the total number of ROIs, the VBS significantly reduces the explored region than the RQS. We reported the numeric performance comparison between the VBS and the RQS in Fig. 5.

6 Conclusion

In this paper, we have discussed the ROIs determining problem for a large environment and its various aspects. First, we have proposed a novel online framework to integrate the environmental boundary estimation and area coverage problems. Second, we theoretically analyze the properties of the boundary estimation algorithm which is deemed to best satisfy such conflicting requirements. Third, we proposed the adaptive area decomposition and search algorithm to localize the desired number of ROIs rapidly: VBS, which uses an optimal partitioning strategy for updating the search space. Fourth, we demonstrate these algorithms in a simulated environment, and statistically analyze their relative performance.

The simulation results show that, in general, VBS creates coverage path is shorter than the coverage path by RQS. VBS has clear benefit when handling fewer ROIs since

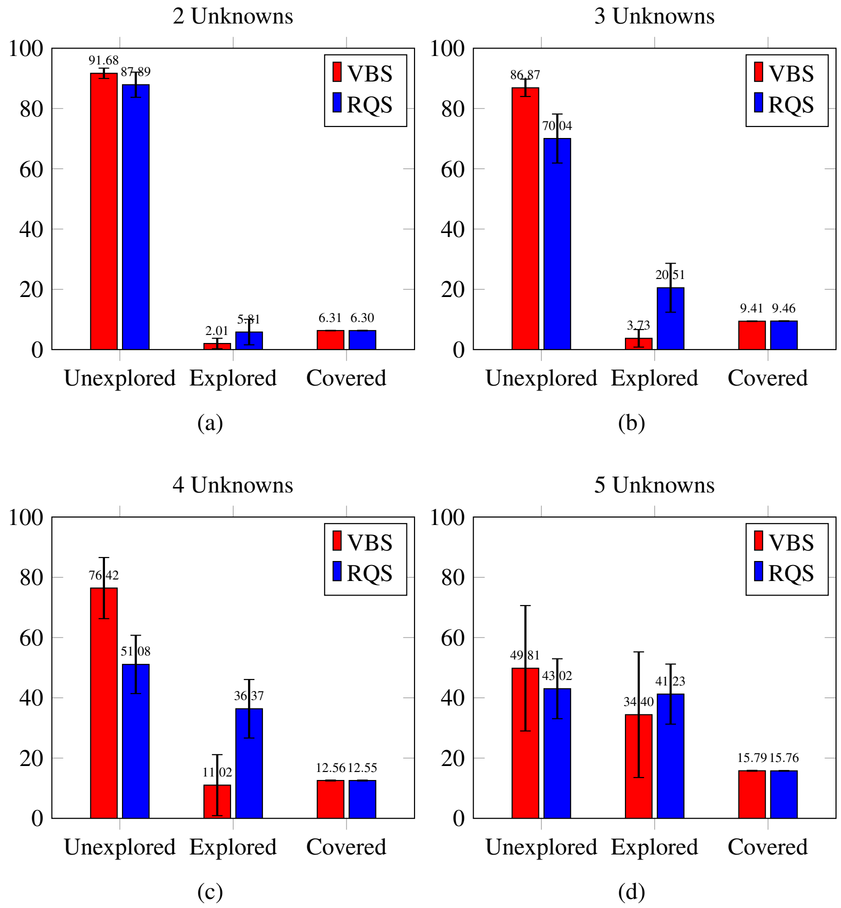

Fig. 5. Area coverage: Every bar chart is generated from 20 trial runs of each algorithm. The performance is evaluated by comparing the size of following areas: unexplored, covered and explored area. The error bar of the bar chart represents the standard deviation of each area.

it performs a global planning of the coverage according to the size of the target area. On the other hand, RQS plans only local best decomposition, resulting in overall poor performance. Both algorithms do not require to complete coverage of the target area and save a significant amount of redundant exploration. Comparing all the experiments, we have shown that, in general, required explored areas are less than unexplored areas. Furthermore, the robot does not need to visit the covered areas by ROIs. As a result, even in worst case scenarios, the required exploration to determine ROIs is always less than complete area coverage algorithms.

In future, we would like to extend the algorithms for multi-robot systems. We would also consider the problem associated with non-stationary environmental boundaries.

Acknowledgments

The authors would like to thank Ministry of Education, Culture, Sport, Science and Technology (MEXT) - Japan, for the financial support through MEXT scholarship. In addition, this work was supported by the Industrial Convergence Core Technology Development Program (No. 10063172) funded by MOTIE, Korea.

References

1. Acar, E.U., Choset, H., Lee, J.Y.: Sensor-based coverage with extended range detectors. IEEE Transactions on Robotics **22**, 189–198 (2006)
2. Acar, E.U., Choset, H., Zhang, Y., Schervish, M.: Path planning for robotic demining: Robust sensor-based coverage of unstructured environments and probabilistic methods. The International Journal of Robotics Research **22**(7-8), 441–466 (2003)
3. Choset, H.: Coverage for robotics - A survey of recent results. Ann. Math. Artif. Intell. **31**(1-4), 113–126 (2001)
4. Dames, P., Kumar, V.: Autonomous localization of an unknown number of targets without data association using teams of mobile sensors. IEEE Transactions on Automation Science and Engineering **12**, 850–864 (2015)
5. Galceran, E., Carreras, M.: A survey on coverage path planning for robotics. Robotics and Autonomous Systems **61**, 1258–1276 (2013)
6. Guillen-Climent, M.L., Zarco-Tejada, P.J., Berni, J.A.J., North, P.R.J., Villalobos, F.J.: Mapping radiation interception in row-structured orchards using 3d simulation and high-resolution airborne imagery acquired from a uav. Precision Agriculture **13**, 473–500 (2012)
7. Han, J., Xu, Y., Di, L., Chen, Y.: Low-cost multi-uav technologies for contour mapping of nuclear radiation field. Journal of Intelligent & Robotic Systems **70**, 401–410 (2013)
8. Lahijanian, M., Maly, M.R., Fried, D., Kavraki, L.E., Kress-gazit, H., Member, S., Vardi, M.Y.: Environments With Partial Satisfaction Guarantees **32**(3), 583–599 (2016)
9. Matveev, A.S., Hoy, M.C., Ovchinnikov, K., Anisimov, A., Savkin, A.V.: Robot navigation for monitoring unsteady environmental boundaries without field gradient estimation. Automatica **62**, 227–235 (2015)
10. Mitchell, D., Chakraborty, N., Sycara, K., Michael, N.: Multi-Robot Persistent Coverage with stochastic task costs. In: IEEE International Conference on Intelligent Robots and Systems, pp. 3401–3406 (2015)
11. Paull, L., Seto, M., Li, H.: Area coverage planning that accounts for pose uncertainty with an AUV seabed surveying application. Proceedings - IEEE International Conference on Robotics and Automation pp. 6592–6599 (2014)
12. Paull, L., Thibault, C., Nagaty, A., Seto, M., Li, H.: Sensor-driven area coverage for an autonomous fixed-wing unmanned aerial vehicle. IEEE Transactions on Cybernetics **44**, 1605–1618 (2014)
13. Soltero, D.E., Schwager, M., Rus, D.: Decentralized path planning for coverage tasks using gradient descent adaptive control. The International Journal of Robotics Research **33**, 401–425 (2014)
14. Strimel, G.P., Veloso, M.M.: Coverage planning with finite resources. In: IEEE International Conference on Intelligent Robots and Systems, pp. 2950–2956 (2014)
15. Xu, L.: Graph Planning for Environmental Coverage. Carnegie Mellon University (August), 135 (2011)
16. Yehoshua, R., Agmon, N., Kaminka, G.A.: Robotic adversarial coverage of known environments. The International Journal of Robotics Research pp. 1–26 (2016)