Title	Rapid Coverage of Regions of Interest for Environmental Monitoring
Author(s)	Pinkam, Nantawat; Newaz, Abdullah Al Redwan; Jeong, Sungmoon; Chong, Nak Young
Citation	Intelligent Service Robotics, 12(4): 393-406
Issue Date	2019-09-20
Type	Journal Article
Text version	author
URL	http://hdl.handle.net/10119/16765
Rights	This is the author-created version of Springer, Nantawat Pinkam, Abdullah Al Redwan Newaz, Sungmoon Jeong, Nak Young Chong, Intelligent Service Robotics, 12(4), 2019, 393-406. The original publication is available at www.springerlink.com, http://dx.doi.org/10.1007/s11370-019-00290-x
Description	

Noname manuscript No.
(will be inserted by the editor)

Rapid Coverage of Regions of Interest for Environmental Monitoring

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

Received: date / Accepted: date

Abstract We present a framework for rapidly determining regions of interest (ROIs) from an unknown intensity distribution, particularly in radiation fields. The vast majority of studies on area coverage path planning for mobile robots do not investigate the identification of ROIs. In a radiation field, the use of ROIs can limit the required range of exploration and mitigate the monitoring problem. However, considering that an unmanned aerial vehicle (UAV) has limited resources as a mobile measurement system, it is challenging to determine ROIs in unknown radiation fields. Given a 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 a large-scale environment exploration, entire areas are initially adaptively decomposed using two hierarchical methods based on recursive quadratic subdivision and Voronoi-based subdivision. Once an informative decomposed subarea is selected by maximizing a utility function, the robot heuristically reaches contaminated areas, and a boundary estimation algorithm is adopted to estimate the environmental boundaries. The properties of this boundary estimation algorithm are theoretically analyzed in this paper. Finally, the detailed boundaries of the ROIs of the target area are approximated by ellipses, and a set of procedures are iterated to sequentially cover all areas. Simulation results demonstrate that our framework allows a single UAV to efficiently explore a given target area and maximize the localization rate for ROIs.

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

[1]School of Information Science, Japan Advanced Institute of Science and Technology,
1-1 Asahidai, Nomi, Ishikawa 923-1292 Japan
Tel.: +81-761-51-1248
Fax: +81-761-51-1149
E-mail: {pinkam.nantawat, jeongsm, nakyoung}@jaist.ac.jp
[2]Department of Computer Science, Rice University,
Houston, TX 77005, USA
E-mail: redwan.newaz@rice.edu

1 Introduction

In a large radiation field, it is important to localize radiation-contaminated areas (hereinafter region of interest [ROIs]) to monitor radiation effects and localize hotspots, sources, and so on. Recent advances have enabled unmanned aerial vehicles (UAVs) to access and navigate unstructured or cluttered environments. Therefore, a quadrotor-type UAV equipped with dedicated sensors can be an attractive platform for environmental monitoring tasks, including radiation field monitoring. Although many area coverage path planning methods for mobile robots have been proposed, it is still difficult to efficiently monitor a large field with a single robot. Therefore, we aim to design a path planner that can rapidly localize ROIs to reduce the exploration path of a robot.

Radiation field monitoring has been commonly studied in robotics [14, 17, 34, 35, 40]. The goal of radiation field monitoring is to plan a path from which the robot can localize all contaminated locations in a given target area. Considering that the contaminated locations could be spatially distributed throughout the target area, a search is needed to localize all of them. The required tasks associated with such a search have inspired various methods for addressing the problem of coverage. Spatial search techniques should be adjusted according to the number of robots used for this application. In cases wherein multiple robots are used, the target area can be partitioned into smaller subregions to reduce the search space for each robot. The search strategy is benefited exclusively by the number of robots and the communication among robots. However, in the case of a single robot, the partitioning of the target area benefits neither the exploration cost nor the accuracy.

The majority of coverage planning work has been proposed for known environments [31, 43, 45, 46]. These approaches are often used to minimize the uncertainty metric of a given map. A common choice is to utilize an exploration method, such as a frontier-based method [6, 44] or rapidly exploring random tree method [21, 47], in a location where the uncertainty metric, such as the entropy or mutual information metric, is high. However, in many situations, a radiation map for the target area is not available *a priori*; therefore, the entire target area must be covered to localize the contaminated locations. Hence, complete coverage algorithms are often used [13]. Even though complete coverage algorithms ensure complete terrain visitation, they lack the opportunity to optimize the localization rate of contaminated locations.

A method that estimates environmental boundaries instead of aiming for complete coverage can decrease the search time and reduce energy consumption [26, 37]. Here the path planning problem involves estimating the boundaries of contaminated areas and allowing the robot to sense the ROIs. However, when the environment is unknown, it is difficult to plan a path that identifies the interesting and not interesting areas. In conventional algorithms for coverage planning with obstacles, the path is usually generated to cover the free space of the environment in an optimal fashion. In our problem, we want to rapidly identify the locations and geometrical size of ROIs rather than avoiding ROIs. When the robot finds contaminated areas in an opportunistic fashion, it can expedite the boundary estimation process for determining ROIs and can bypass the need to exhaustively cover entire regions.The identification of ROIs in a radiation field allows us to prioritize the search area in a way that minimizes the exploration time of the robot.

This work aims to achieve coverage with a single robot. Hence, we investigate an additional component to the coverage problem by incorporating a localization rate factor for the contaminated locations. The localization rate factor should be taken into account when exploring with a single robot because the target area is sometimes too large for the robot to cover completely with a limited exploration budget. Considering that the goal of the robot is to localize all contaminated locations as quickly as possible, the algorithm must behave as if it is performing complete coverage over long operation periods. This problem might be thought of as a target acquisition problem [9]. However, there is an important caveat. Target acquisition problems assume that the robot is equipped with a sensor that has a wide field of view. In our problem, the robot sensor works in a pointwise fashion. Therefore, the robot needs to travel to a location to obtain a measurement.

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

The contributions of this work are as follows:

1. We formulate a method for the localization of ROIs that requires no *a priori* information.
2. Our algorithm can rapidly localize ROIs and minimizing the exploration time of a robot.
3. The proposed algorithm is complete, *i.e.*, all contaminated locations are identified during the operation of the robot.
4. By focusing on the limited computational capabilities of a UAV serving as the mobile robot, the proposed algorithm was made to robustly determine ROIs.

To discuss the aforementioned topics, this paper is organized as follows. In Section 2, we investigate the work related to this area. In Section 3, we describe the formulation of the problem. In Section 4, we present two heuristic coverage algorithms based on adaptive hierarchical area decomposition. In Section 5, we briefly explain the generalization process for the ROIs. Finally, in Sections 6 and 7, we present the simulation results and conclude our findings.

2 Related Work

Area coverage planning has been extensively studied in robotics to establish a path over a target area in which the robot covers all the locations. A traditional area coverage algorithm has been modified and applied in several applications. The proposed algorithm can solve a different area coverage problem through generation of paths that make efficient use of a limited travel time and maximizing the probability of finding the radiation-contaminated locations serving as the ROIs. This problem is somewhat similar to complete area coverage problems. Choset [7] conducted an early survey on coverage algorithms and 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 approaches include lawnmower pattern, raster scanning, inward spiral search, wall following, etc. Heuristic search is computationally less expensive than cell decomposition, but cannot guarantee the optimal performance.

On the other hand, cell decomposition decomposes the target area into smaller areas. Galceran and Carries [12] conducted 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(TSP) (considered NP-hard). Subsequently, the Hamiltonian path is determined using the spanning tree algorithm, which visits each cell once [11]. Recently, a variant of Hamiltonian path has been used for the persistent coverage problem [16, 27, 32, 33, 41]. Nonetheless, if there are obstacles in the target environment, it is impossible to generate the Hamiltonian path. In such cases, the boustrophedon-cellular decomposition solves this issue for bounded planar with known obstacles [38]. The key 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, 18, 30, 31]. If unknown obstacles exist in the environment, the Morse decomposition becomes essential for determining critical points in the target area, and incrementally creating a Reeb graph that optimally solves the online coverage problem [2]. Alternatively, the unknown obstacle problem can be solved using a temporal logic specification constituting the safety components in a partly unknown environment [20].

Regarding adversarial coverage in environmental monitoring problems, Yehoshua [45] suggests a modification of the spanning tree algorithm; the target environment is divided into connected areas constituting the safe/dangerous cells, and then covers safe areas before moving to dangerous one. Disadvantageously, this work requires a map of threats prior, which may not be possible. However, if we make an assumption on the distribution of phenomena, for instance, a mine-laying pattern in the minefield is a priori available, the probabilistic demining algorithm [2] can solve our problem.

The shape of environmental boundaries has been recently studied [3, 5, 25, 42]. Nevertheless, applying a boundary estimation algorithm to our intended problem is complex. The primary challenge is that the coverage planning assumes that all focused locations contain similar features, whereas the boundary assessment concludes that a cluster containing the contaminated sites can be monitored by approximating only a closed boundary. Several boundary estimation methods comprise two parts: one that defines a threshold to be utilized as gradient information, and another that minimizes the square error between the sampling location and the desired threshold to find the robot's trajectory that close to the level curve.

Although boundary estimation has not been applied to the problem of area coverage [36, 39], some methods have been assessed with the intention of reducing the search area. For example, split-merge cells have been utilized for the trapezoidal decomposition of the cell, which is anticipated to be useful in agricultural applications. In multi-robot coverage methods, the partitioning of the targeted area is standard, e.g., the Voronoi-based coverage control problem introduced in Refs. [4, 8, 15, 22]. Similarly, the recursive geometric subdivision was proposed to monitor a spatial temporal sensitive area [19]. A Voronoi-based coverage control is closely associated with our work in the sense that it considers the sizes of various search robots and partitions the search area. However, our method differs; the sizes of the contaminated sections are utilized in place of the sizes of the various robots, and the section sizes are sampled from the exploration of a single robot and are computed online.

An important aspect of path planning is the optimization of resource costs. Since robots have limited endurance and sensing range, the coverage plan needs to be optimized for finite resources, especially UAVs. Apart from the traditional coverage planning where resource costs are overlooked. An optimal persistent coverage plan was proposed in [28], wherein the authors obtain a collection of tours for multiple robots that every target is visited by the robots and the minimum frequency of which a target is visited is maximized. When a single robot is used, a hierarchical planner was proposed in [23] to compute the mode food ratio heuristic and prioritize search regions. [10] proposed energy-aware path planning algorithm that minimizes energy consumption while meeting a set of demands by using actual measurement to derive energy model of the UAV. Then, the UAV calculates velocity that minimizes energy consumption over a specific distance.

Our work presents the opportunistic and iterative environmental boundary estimation method for the area coverage problem. The methodology is evaluated using two strategies; boundary estimation and coverage planning, within a novel framework that localizes unknown ROIs and uses an arbitrary initial robot position. The novelty of the proposed framework is twofold. First, we proposed a novel online framework to integrate the environmental boundary estimation and area coverage problem. Second, we demonstrate the performance of our two algorithms: recursive geometric subdivision and Voronoi-based coverage.

Although the proposed framework is applied in the context of field radiation monitoring with a UAV, our approach is general, can be used with other mobile robots, and can be scaled to other domains in which an opportunistic collection of environmental phenomena is present.

3 Problem Formulation

We are given a target area T, that contains radiation sources. The strength of the target area can be sensed by the UAV, and we assume that T can be decomposed into a regular grid with n cells. Let us denote this grid by G. Considering that radiation sources that modelled using Gaussian mixture model (GMM) might be spatially distributed, G contains two types of cells, namely, free and contaminated cells. Furthermore, nearby sources cumulatively affect the target area, thus resulting in a joint distribution of measurement attributes. Let us assume that cell $(c_1, c_2, ..., c_n)$ is associated with a nonnegative measurement attribute $(z_1, z_2, ..., z_n)$. The UAV is equipped with a sensor to make a pointwise measurement $z(t)$ at its position $x(t)$ at time t. The ROIs in T are those cells $\mathbf{J} := \{c_i | z_i > \epsilon\}$ in which the UAV finds $z_i > \epsilon$, where ϵ is a threshold value near zero, *i.e.*, $\epsilon \approx 0$. The contaminated areas are contiguous. Therefore, the robot can trace such areas by tracking only to their boundaries. Hence, the definitions of the contaminated and free cells are quantified by a binary variable given by the following:

$$p_{c_i} = \begin{cases} 0, & \text{if } z_i \leq \epsilon \\ 1, & \text{otherwise} \end{cases} \tag{1}$$

Fig. 1 shows an example of a 50×50 m^2 world map. Depending on the spatial locations of the radiation sources, measurement attributes are also spatially distributed throughout T. The dark blue cells are the cells in which $p_{c_i} = 0$. The other colored cells represent the fact that measurement attributes are available such that $p_{c_i} = 1$. We can then find multiple ROIs while splitting $\mathbf{J}$, subject to spatial distances.

Definition 1 ROIs: Sets of cells that correspond to a set of contaminated locations in a given target area T.

The global mission of the robot can be defined in two different ways, that imply two different objective functions:

- The minimum time to localize an ROI

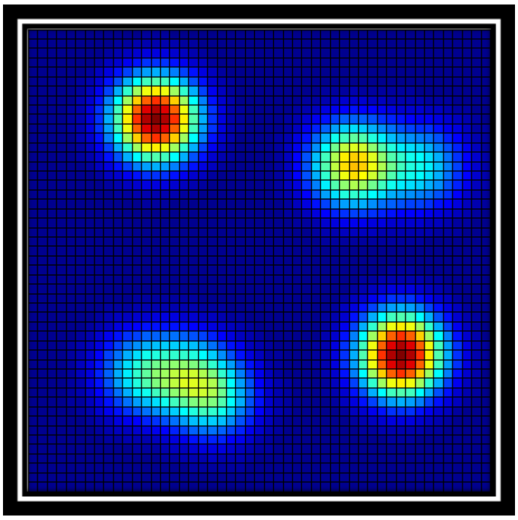


Fig. 1 The dark blue cells have no measurement attributes, whereas the other colored cells represent the measurement attributes.

- The total time to localize all ROIs in T

Without loss of generality, we assume that the travel time is proportional to the travel distance. Thus, we will first use the boundary estimation technique that minimizes robot exploration to localize an ROI. Second, we will use the heuristic area coverage technique that ensures the localization of all ROIs in T. The total time is taken into account by summing the lengths of the boundary estimation and 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 $\mathcal{P}$. We assume that $|\mathbf{J}| << n$, *i.e.*, the contaminated cells are far fewer in number than the total number of cells. We define the event $S_{\mathcal{P}}$ as the event wherein the robot reaches any ROI, which is not localized beforehand. Therefore, the cost of finding an ROI $B(S_{\mathcal{P}})$ can be expressed as follows:

$$B(S_{\mathcal{P}}) = \sum_{c \in \mathcal{P}} (1 - p_{c_i}), \tag{2}$$

Therefore, the first objective is to find an online coverage path that minimizes $B(S_{\mathcal{P}})$. 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 used to determine the ROI size.

For the second objective, we denote the sequence of newly discovered ROIs along the coverage path $\mathcal{P}$. If k ROIs exist in T, we discretize $\mathcal{P}$ to a set $Q = \{q_1, q_2, ..., q_k\}$, where Q is the set of subseqences of path q_i that the robot used visit each ROI along $\mathcal{P}$. Considering that the travel time is proportional to the length of q_i, we want to find the minimum-length paths in set Q to localize all ROIs. Therefore, the total cost $C(\mathcal{P})$ that the robot use to localize a finite set of ROIs is given by the followings:

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

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

Definition 2 PI: PI of the robot is evaluated with respect to the minimum explored path to localize all ROIs.

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

4 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, namely, the *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 by using the *Finding Subregions* step. Once the subregions have been determined, we examine the utility of traversing each subregion as explained in the *Utility Function Design* in the third step. In the subregion that is determined to have maximum utility, we plan a coverage path through the set of unvisited cells. The robot then progresses along this path. If the robot notices an ROI along its path, it will cease its exploration more and iterate whole steps. Otherwise, the whole steps will be iterated after traveling along the entire path.

4.1 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 this objective can be achieved by the adaptive partitioning of the target area in hierarchical order. Given the position of the ROIs, the hierarchical order is determined by a local minimum distance with respect to the relative position of the robot. Therefore, we propose the use of recursive quadratic subdivision and Voronoi-based partition to limit the search space. Fig. 3 shows the overall overview of each algorithm. With a given partition, our goal is to find an ROI through limited exploration.

Recursive quadratic subdivision (RQS): The RQS algorithm follows a greedy approach, wherein each step leads the robot to the ROI nearest to its current location that has not been covered yet. The main idea is that, an optimal path is generated initially to include every cell in T, which is induced from the grid cells. A simple traveling salesman problem (TSP) algorithm is used to generate this a type of path [24] because it minimizes the path length between the current location of the robot and all other unvisited cells in the grid. The robot then starts to explore along this path. When a contaminated cell is found, it switches to the boundary estimation planner. An ROI is then computed from its estimated boundary. Therefore, the robot determines a minimum route from its location even if it executes the TSP path in its entirety. If there are no contaminated cells in the target area, the robot always optimally explores the whole target area. However, in the presence of multiple ROIs, this coverage path can be further optimized on the basis of a simple heuristic search.

In the second phase, RQS finds a coverage path that minimizes the travel distance needed to connect the desired number of ROIs. Finding such a path is made possible by subdividing the area into four quadrants from the center of an ROI. As we iteratively localize the ROIs, the geometric partitions are also subdivided recursively. As a result, if the area turns out to be either the entire target area or a subdivision of the previous decomposition, it is further decomposed into four divisions on the basis of the center of the ROI. The three basic operations of this decomposition are as follows. First, we generate a TSP path to explore the unexplored cells optimally. Second, when an ROI is determined, we terminate the exploration and decompose the area. Finally, the region of each division is determined by Alg. 1.

We demonstrate the RQS while the robot is covering its free space by using an example depicted in Fig. 3(a). The robot starts to cover the space in a vast cell by generating a TSP path over the target area from its starting point at the bottom left corner; the target area is shown as the red rectangle in Fig. 3(a). When the

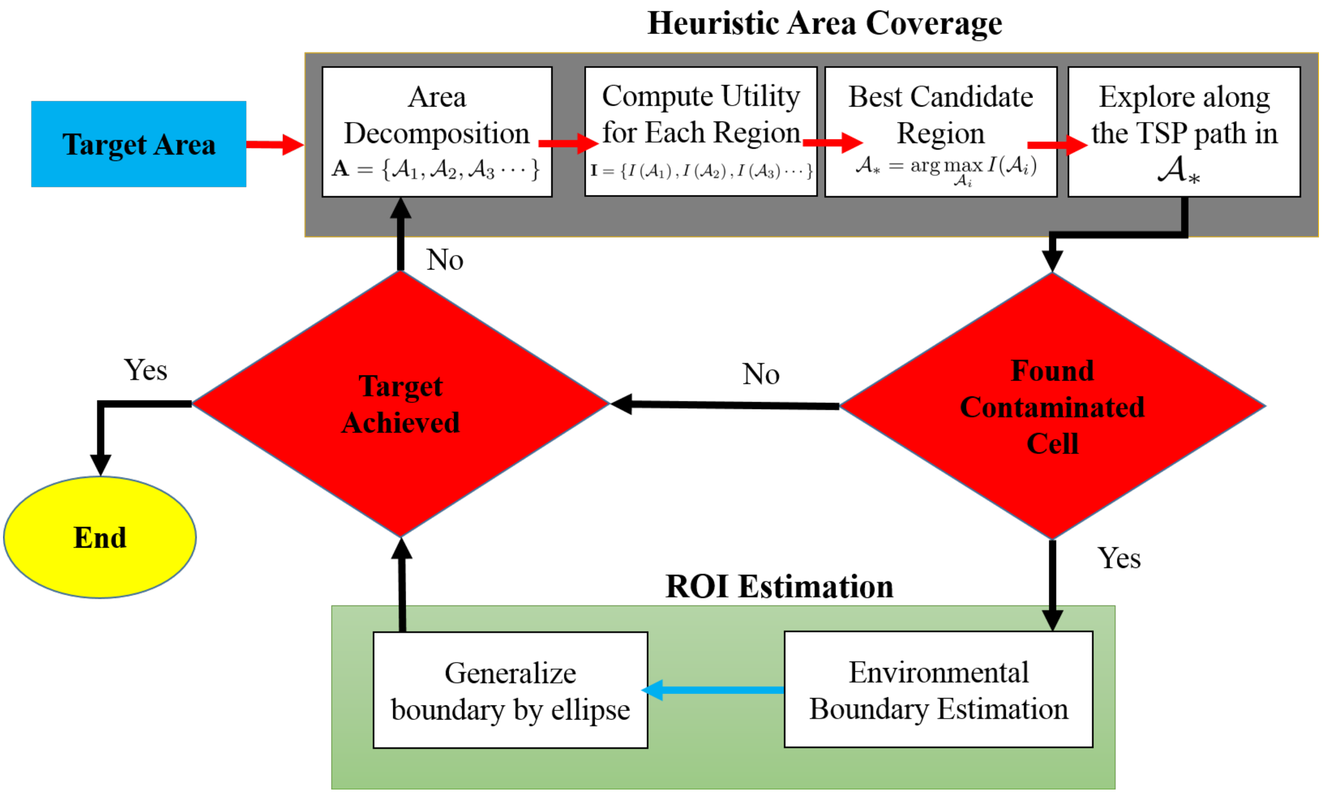
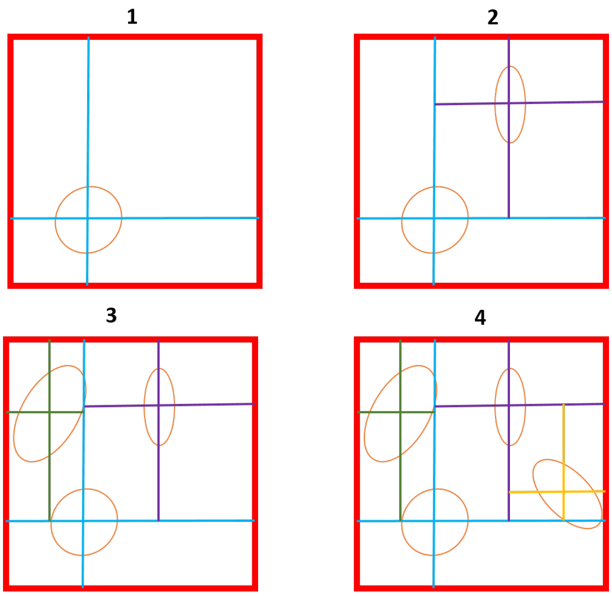


Fig. 2 System Overview: The figure shows all the steps performed during the heuristic area coverage and those performed by the ROI estimation algorithms. Starting from an arbitrary location, the robot can iteratively localize the desired number of ROIs by using this framework.

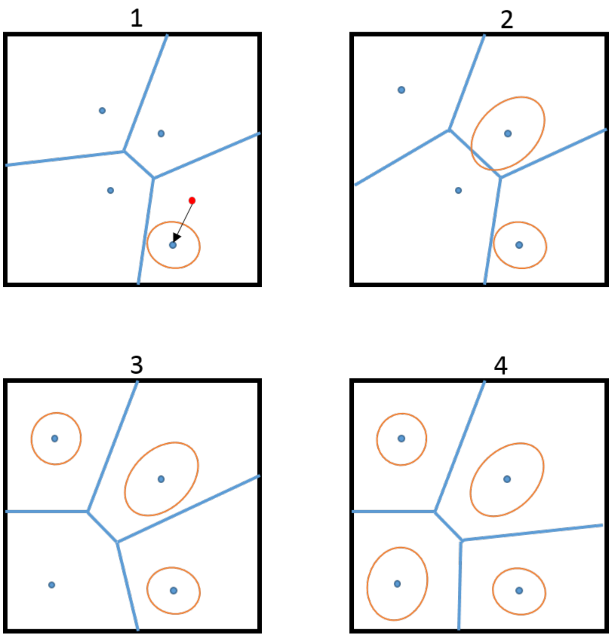
robot reaches the cell in which $p_{c_i} = 1$, which is the unvisited location of a contaminated area, it finishes covering the TSP path by moving along the path from its current position to the end of the path. Given that the contaminated area is unknown *a priori*, the robot follows the boundary-tracking algorithm to cover it. The robot then constructs an ellipse around the estimated boundary to represent the ROI (orange ellipse in Fig. 3[a]). At this point, it encounters the quadratic subdivision at the center point of the ellipse (blue lines in Fig. 3[a]). The robot chooses the subdivision that maximizes the utility function and repeats the step described above (Fig. 3[a]). Considering that the hierarchical quadratic subdivisions are connected to each other, the robot is guaranteed to visit all subdivisions in the target area and completely cover the space.

Voronoi-Based Subdivision (VBS): The 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 Voronoi centroids. In our case, the ROI centers are not available a priori; thus, we introduce a few changes to the original Voronoi-based partition algorithm described in [29]. First, our proposed version randomly partitions the target area by using four random points inside the target area(the number four was chosen to facilitate comparison with the four quadrants from RQS). Second, it leads the robot to the centroid nearest its initial location. Finally, similar to RQS, the TSP algorithm generates the coverage path. The robot starts to explore along this route. However, when a contaminated cell is found, it switches to the boundary estimation planner. An ROI is then computed from its estimated boundary. Unlike in RQS, the robot finds a minimum route to an ROI from its location either while traveling to the Voronoi centroid or while executing the TSP path. Although these paths increase the probability of finding an ROI, if there are no contaminated cells in the subregion, the complete coverage path will be larger than the RQS coverage path owing to the traveling activity to the centroid. Note that, the initial search space is limited by the random partition in VBS, whereas the initial search space is the whole target area in RQS. The partition of the target area is 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 a path is made possible by iteratively updating the Voronoi centroids. The iterative updates of the centroids lead VBS to generate an optimal partition of the



(a) **RQS decomposition:** The area is decomposed into four subdivisions on the basis of the center positions of ROIs.



(b) **VBS decomposition:** Starting with the random partitions, the partitions are updated by the center positions of ROIs.

Fig. 3 Area decomposition: Two different algorithms are proposed to decompose the search space into smaller regions. The RQS decomposes the area in a greedy manner, whereas the VBS iteratively approaches optimal decomposition.

search space without changing the region that will be explored. However, when the number of ROIs is greater than the number of random initial points, the partition centroids are not only updated iteratively but also constructed incrementally. The four basic operations of this decomposition are as follows. First, we generate randomized incremental construction of the partitions to reduce the search space. Second, the robot moves to the Voronoi centroid, and the TSP algorithm creates a coverage path to optimally explore the unexplored cells of a given subdivision. Third, when an ROI is determined, we terminate the exploration and update the Voronoi centroids. Finally, the region of each division is determined by Alg. 1.

Algorithm 1 Finding subregions

Require: Graph, $G = (V, E, B)$
Ensure: Subregions, Λ
for all $e \in E$ **do**
  $\psi_G \leftarrow intersect(e, B)$
  $E_\psi \leftarrow trim(e, \psi_G)$
end for{Shorten initial edges}
for all $b \in B$ **do**
  $\psi_b \leftarrow intersect(b, E)$
  $E_b \leftarrow combination(\psi_b, 2)$
  $E_b \leftarrow unique(E_b)$
end for{Finding box edges}
$E \leftarrow \{\{E_b\} \cup \{E_\psi\}\}$ {update graph}
for all $p \in \psi_c$ **do**
  $\Lambda \leftarrow \cup NeighborEdges(p, E)$
end for{update partition area}

We demonstrate the VBS while the robot is covering its free space by using an example depicted in Fig. 3(b). The Voronoi diagram is the partitioning method used to partition a plane with n points into specific subsets of the plane such that each subset contains exactly one generating point. In a typical Voronoi diagram, the set of generating points is known *a priori*. The Voronoi polygons are then constructed in such a manner that every point in a given polygon is closer to its generating point than to any other point. 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 in the rightmost corner; the target area is depicted as the black rectangle in Fig. 3(b). Thereafter, the robot constructs a TSP path to cover the given region. Whenever the robot reaches a cell in which $p_{c_i} = 1$, *i.e.*, an unvisited location in a contaminated area, it finishes covering the centroid path or the TSP path. Given that the contaminated area is unknown *a priori*, similar to that in RQS, the robot follows the boundary-tracking algorithm to cover it. The robot then constructs an ellipse around the estimated boundary to represent the ROI (orange ellipse in Fig. 3[b]). At this point, it encounters the update of the Voronoi centroid. The Voronoi centroid of the current region is replaced by the center point of the ellipse (blue dot in Fig. 3[b]). If there are more ROIs than the number of Voronoi centroids 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 (Fig. 3 [b]). Considering that the Voronoi regions are connected, the robot is guaranteed to visit all subdivisions in the target area and completely cover the space.

4.2 Finding Subregions

At the end of the second phase, each algorithm finds the subregions on the basis of its partition method. The algorithm begins by creating the graph $G = (V, E, B)$. We represent the target area as a rectangular box B in G. The initial partitions are edge set E which includes edges with infinite lengths. To find subregions Λ, we first 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. Furthermore, let ψ_b be the set of vertices that represents the corner points of B, and let ψ_c be the centroid of the ROIs. Once we trim the long edges, the new partition is represented by E_ψ. Second, we find all possible combinations of edges on B and represent them by E_b. G is updated by combining these two sets of edges such that $E \leftarrow \{\{E_b\} \cup \{E_\psi\}\}$. Finally, we group all subregions Λ by finding the neighboring edges. Finding such neighbors is straightforward. Given ψ_c, an anticlockwise walk along the E can sort such neighbors.

4.3 Utility Function Design

In the third phase, each algorithm finds the best search space among all the subdivisions of the target area. For this action, it computes the utility of each of the subdivisions. The utility is designed to favor destinations that 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 in which 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 must perform starting from its current position x_t. During the execution of a_t, if the robot finds a contaminated cell along its path, it estimates an ROI on the map. Therefore, the explored trajectory of the robot indicates some of the cells in m:

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

In case the robot finds an ROI on the map, we must treat the ROI cells differently. We assume that traveling inside an ROI is redundant; thus, we want to avoid such a region. Therefore, the cells bounded by an ROI are considered similar to 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 the other cells, 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, the coverage path should include all cells when computing the expected information gain because the robot does not know when it will find an ROI along its path. Therefore, entropy of the target subdivision can be expressed 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 world model of the robot 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 the highest expected information:

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

There are some studies on exploration and mapping problems that consider another quantity in addition to the information gain in Eq. (10). This quantity is the cost for reaching the subdivision. However, we observe that adding such a quantity to the utility function decreases the overall performance of both algorithms. Therefore, every time the robot has to make a decision as to where to go next, it uses only an information maximization metric to determine the action a_t^*.

5 Finding ROIs

We employ a boundary estimation algorithm to locate the ROIs by using the proposed exploration method. In this section, we first explain how to generalize an arbitrary boundary. Thereafter, we will explore generalization properties. Considering that the ROIs are functions of the boundaries, we will focus only on boundary estimation.

5.1 Environmental Boundary Generalization

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

Definition 3 Boundary line: A line is said to be a boundary line if it represents the intersection between a contaminated area and noncontaminated areas.

Assume that a contaminated area $\delta\mathcal{A}$ is a nonconvex set in which the continuous boundary is defined by a level set $\delta\mathcal{A}$ as follows:

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

where β is the measurement threshold.

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

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

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

5.2 Analysis of Boundary Estimation Algorithm

Proposition 1 *Given a contaminated area $\mathcal{A}_i \in \mathbf{A}$, the measurement attribute of a location x is not available if and only if $x \notin \mathcal{A}_i$.*

Proof Assume that the cardinality of $\mathbf{A}$ is one for a given T. To prove this proposition, we will first show that all locations bounded by boundary $\delta\mathcal{A}_i$ are important and all locations outside the boundary are negligible.

The sufficiency of this proposition is trivial, and we can easily prove it by using Definition 1. After using Definition 1, $z(x) > \beta$ (where β represents the boundary threshold) if x is in close proximity to the sources. By using Definition 3, we can say that all sources are covered by the boundary lines. Given that β represents the boundary threshold, we can conclude that $\forall x \in \mathcal{A}_i$, where $z > \beta$ is bounded by $\delta\mathcal{A}_i$. Similarly, the necessity of this proposition also follows from Definition 1. Let x_h be the hotspot location for $\mathcal{A}_i$, where $z(x_h)$ is a maximum subject to $\forall x \neq x_h$. As the robot travels far from x_h, the following relationship holds by Definition 1: $z(x_h) > z\,(\forall x \neq x_h)$. When $z(x) << z(x_h)$ *i.e.*, the measurement of x is also less than x_h, we neglect these locations by drawing the boundary line. Therefore, for any location outside the boundary line, the measurement is $z(x) < \beta$ and it is negligible.

Proposition 2 *Let $\delta\mathcal{A}_i$ be the estimated area generated by the boundary-tracking algorithm. Then $\mathcal{A}_i$ is a contaminated area if the effects of all nearby sources can be described jointly by the area shape.*

Proof This is true by the construction of Eq. (11). In particular, given that $\delta\mathcal{A}_i$ is continuous, the contaminated area $\mathcal{A}_i$ must be continuous. Thus, the continuous $\mathcal{A}_i$ represents the joint effect of all nearby sources.

It is evident from Proposition 2 that the boundary-tracking algorithm can estimate an ROI. Once the boundary-tracking algorithm computes the environmental boundary, we set $\delta\mathcal{A}$. We then determine the ROI by fitting an ellipse to the sample points of $\delta\mathcal{A}$. This generalization is performed by minimizing the least squares distance between the sample points and the conic representation of the ellipse. We extract the parameters from the conic representation by removing the tilt. Thereafter, we will explain the manner in which to extend this approach to find multiple ROIs.

Lemma 1 *Given a target area T, if $\delta\mathcal{A}_1$ and $\delta\mathcal{A}_2$ are the two boundaries, then $\delta\mathcal{A}_1 \cap \delta\mathcal{A}_2 = \emptyset$.*

Proof Suppose $\delta\mathcal{A}_1 \cap \delta\mathcal{A}_2 \neq \emptyset$. Thereafter, the contaminated area $\mathcal{A}_1$ must be overlapped with $\mathcal{A}_2$, and this contradicts Proposition 2.

Lemma 1 1 implies that the boundary-tracking algorithm can efficiently separate the contaminated areas.

Theorem 1 *Given a target area T, the boundary algorithm can estimate the boundary of every contaminated area, i.e., $\forall \delta\mathcal{A}_i \in \mathbf{A}$.*

Proof Suppose that there are n contaminated areas in T such that $n = |\mathbf{A}|$. We then use mathematical induction to prove this theorem.

– Base case: If $n = 1$, it is trivial that the boundary algorithm can estimate a unique boundary of the contaminated area $\delta\mathcal{A}_i$ without overlapping any other contaminated areas.
– Induction step: Suppose that $n = 2$. There are two cases for which the estimated boundaries are not equal to n. In the first case, the estimated boundaries are greater than n, which is impossible because this contradicts Definition 3. In the second case, the estimated boundaries are less than n, but this is impossible because this situation contradicts Lemma 1. Therefore, the estimated boundaries are exactly equal to n as required.

6 Simulation Results

To find the shortest coverage path, we perform four different experiments. We assume that the target area contained five ROIs at most. The performance of each algorithm is evaluated by the distance used to cover the area. To demonstrate efficiency, we start by localizing two out of the five ROIs and conclude by localizing all five ROIs. We also analyze the worst-case performance, and we present a statistical analysis of two algorithms using 20 trial runs. The performances of the algorithms vary significantly from each other. In particular, we observe a noticeable difference in the algorithms when localizing uniformly distributed random ROIs. To compute efficiency, the ROI shapes must remain fixed for each algorithm. We then overlook the additional path cost required to localize the ROIs.

6.1 Finding the Coverage Path that Connects the Desired Number of ROIs

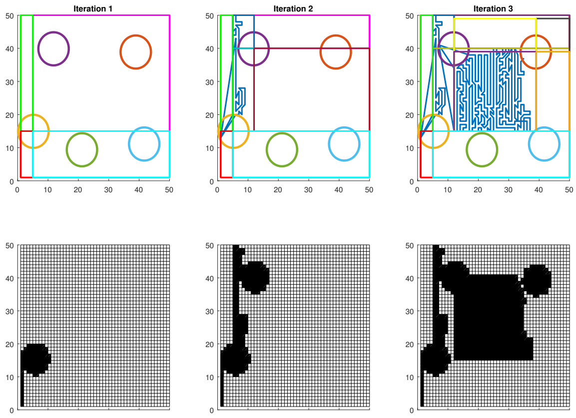
We now consider the case of finding the number of ROIs that meets the desired level of exploration. Therefore, we focus on the shortest coverage path for a given number of ROIs. We consider a $50 \times 50m^2$ grid area in which five uniformly distributed random ROIs are located. Starting from an initial location $(1, 1)$, the robot must find the minimum coverage path that connects the desired number of ROIs. The coverage path can be found by adjusting the cost to be inversely proportional to the unexplored area. In other words, the robot explores the most unexplored region first.

Fig. 4(a) shows an example for the RQS algorithm. In RQS, the initial search space is fixed as the whole target area. Once an ROI is found in a subregion, the search space is subdivided into four regions on the basis of the center position of the detected ROI. The robot avoids exploring the cells bounded by the ROI and starts its exploration from the nearest corner position of a new subregion. When the next ROI is found, only that subregion is divided into four more divisions again. These processes are iterated until the end of the mission. We observe that a smaller search space leads to a more efficient RQS. Although the complexity of the TSP algorithm increases with the dimensions of the search space, a basic reason for this is that RQS sequentially narrows down the search space, thus resulting in faster convergence when the ROIs are located close together. However, this type of heuristic subdivision may cause the traveling time of the robot from its current position to the unexplored region to increase.

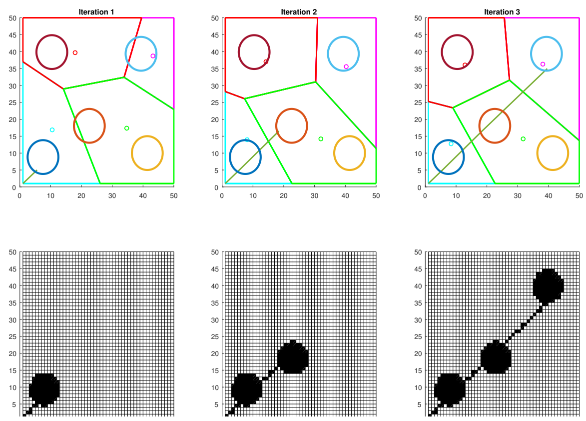
Fig. 4(b) shows an example for the VBS algorithm. In VBS, the initial search space is generated by randomly choosing four points, namely, the Voronoi centroids, inside the target area. The initial search space is then subdivided into four regions on the basis of Voronoi centroids. The robot moves to the centroid of a Voronoi region first and exhaustively searches for an ROI within that region. When an ROI is found while traveling to the centroid or searching the entire subregion, the robot updates the Voronoi diagram. Similar to the RQS, the robot avoids exploring the cells bounded by an ROI. These processes are iterated until the end of the mission. We have observed that VBS is more efficient than RQS because of the initially smaller search space. However, the VBS requires at least three points to partition the entire search space optimally. When there are less than three ROIs in the total area and when the robot has to localize all of them, the performance of the VBS is not stable compared with that of the performance of the RQS.

6.2 Performance Comparison

Fig. 5 shows a performance comparison. To access the long-term performance of each algorithm, we perform



(a) RQS coverage paths on a sample map with uniformly distributed random ROIs. The blue lines in the upper figures show the coverage path, and the colored lines are the partitions of the target areas. A region is divided into four subregions on the basis of the centers of ROIs. For a new region, the searching process is started from the corner. The search spaces are iteratively reduced according to the positions of the centers of ROIs.



(b) VBS coverage paths on a sample map with uniformly distributed random ROIs. The dark green lines in the upper figures show the coverage path, and the colored lines are the partitions of the target areas. 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 on the basis of the true positions of the centers of ROIs.

Fig. 4 Coverage path: The robot starts the coverage in cell $(1, 1)$ and detects any three out of five ROIs. The shape of each ROI is elliptical and is represented in a unique color. The lower grid map represents the coverage map. The measured cells are indicated in black. A cell is called to be measured if it is either included in the coverage trajectory or bounded by the detected ROIs. In general, the coverage path of VBS is shorter than that of the RQS.

the same experiment 20 times while gradually increasing the target numbers. Figs. 5(a), (b), (c), and (d) show the results in terms of the percentage of the area covered. We divide the given target area into three different regions: (1) explored by the robot, (2) covered by the ROIs, and (3) unexplored. The goal is to minimize the explored region. To make a fair comparison, we use five uniformly distributed random ellipses and try to find the shortest path that connects two, three, four, and five ROIs. For these, the regions covered by ROIs represent 6%, 9%, 13%, and 16% of the target area. The unexplored region is then determined by subtracting the covered and explored regions from the total area.

The reduction in search space directly influences the explored areas. When the target number of ROIs is less than the total ROIs in the target area, the robot dramatically reduces the region explored. In the worst-case scenario, in which the robot must localize all five ROIs, the robot is required to travel to more locations to find the ROIs, thus resulting in the exploration of more regions. However, the performance of each algorithm is not stable, and the error bars on the bars in Fig. 5 represent their standard deviations. For both algorithms, the deviation increases with the number of target ROIs.

It is evident from Fig. 5 that the VBS always outperforms the RQS because of its 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 compared with the RQS. We report the numeric performance comparison between the VBS and RQS in Fig. 5. Furthermore, Fig. 6 shows the performance of the same algorithms with the same number and size of the ROIs, the only difference being that the grid area has been expanded to $100 \times 100m^2$. The results are similar to those of the $50 \times 50m^2$ grid area.

7 Conclusion

In this paper, we discuss the ROI-determination problem for a large environment and its various aspects. First, we propose 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 propose two different adaptive area decomposition and search algorithms to rapidly localize the desired number of ROIs: RQS, which uses a greedy-based approach for reducing the search space, and VBS, which uses an optimal partitioning strategy for updating the search space. Fourth, we demonstrate these algorithms

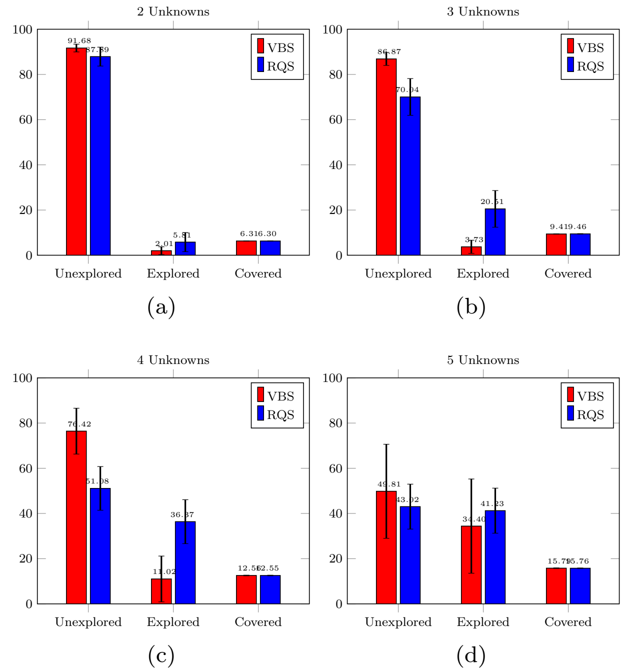


Fig. 5 Area coverage: Every bar chart is generated from 20 trial runs of each algorithm in a $50 \times 50m^2$ grid area. The performance is evaluated by comparing the size of following areas: the unexplored, covered, and explored areas. The error bars represent the standard deviations for each area.

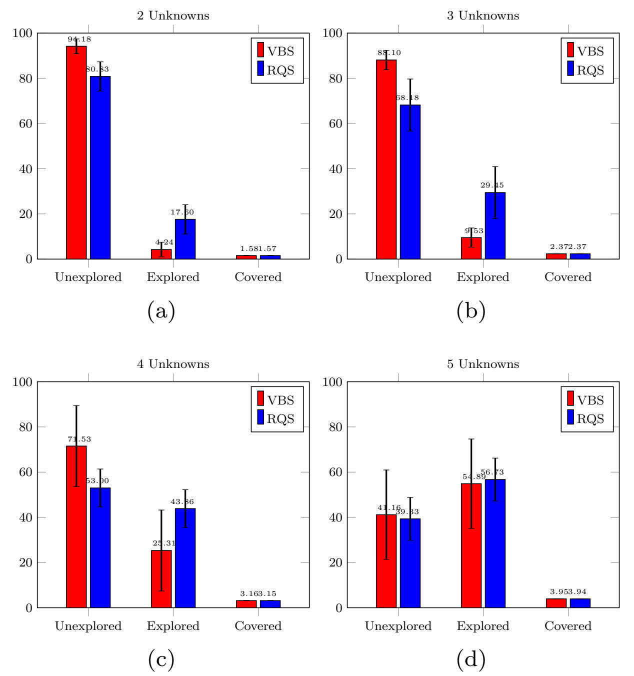


Fig. 6 Area coverage performance in a $100 \times 100m^2$ grid area with five unknown ROIs. The result is similar to those with the $50 \times 50m^2$ grid area.

in a simulated environment and statistically analyze their relative performances.

The simulation results show that VBS generally creates a coverage path that is shorter than the coverage path produced by RQS. VBS has clear benefits when handling fewer ROIs because it performs the global planning of the coverage according to the size of the target area. By contrast, RQS plans only local-best decomposition, thus resulting in an overall poor performance. Both algorithms do not require the complete coverage of the target area and significantly reduce redundant exploration. By comparing the result of all experiments, we show that the total area that the robot is required to explore is smaller than the total unexplored area. Furthermore, the robot does not need to visit the areas covered by ROIs. As a result, the required exploration distance to determine ROIs is always less than that used by complete area coverage algorithms even in worst-case scenarios.

In the future, we would like to extend the algorithms to multi-robot systems. We would also like to consider the problem associated with more complex environment and nonstationary environmental boundaries.

Acknowledgments

The authors would like to thank Ministry of Education, Culture, Sports, Science and Technology (MEXT) of Japan for the financial support via the MEXT scholarship. 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**, 441–466 (2003)
3. Andersson, S.B.: Curve tracking for rapid imaging in AFM. IEEE Transactions on Nanobioscience **6**, 354–361 (2007)
4. Arslan, O., Koditschek, D.E.: Voronoi-based coverage control of heterogeneous disk-shaped robots. In: IEEE International Conference on Robotics and Automation, pp. 4259–4266 (2016)
5. Barat, C., Rendas, M.J.: Benthic boundary tracking using a profiler sonar. Iros 2003: Proceedings of the 2003 Ieee/Rsj International Conference on Intelligent Robots and Systems **1-4**, 830–835 (2003)
6. Burgard, W., Moors, M., Stachniss, C., Schneider, F.E.: Coordinated multi-robot exploration. IEEE Transactions on robotics **21**(3), 376–386 (2005)

7. Choset, H.: Coverage for robotics - A survey of recent results. Annals of Mathematics and Artificial Intelligence **31**, 113–126 (2001)
8. Cortes, J., Martinez, S., Karatas, T., Bullo, F.: Coverage control for mobile sensing networks. IEEE Transactions on Robotics and Automation **20**, 243–255 (2004)
9. 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)
10. Franco, C.D., Buttazzo, G.: Energy-aware coverage path planning of uavs. In: 2015 IEEE International Conference on Autonomous Robot Systems and Competitions, pp. 111–117 (2015). DOI 10.1109/ICARSC.2015.17
11. Gabriely, Y., Rimon, E.: Spanning-tree based coverage of continuous areas by a mobile robot. Annals of Mathematics and Artificial Intelligence **31**(1), 77–98 (2001)
12. Galceran, E., Carreras, M.: A survey on coverage path planning for robotics. Robotics and Autonomous Systems **61**, 1258–1276 (2013)
13. Garcia, E., de Santos, P.G.: Mobile-robot navigation with complete coverage of unstructured environments. Robotics and Autonomous Systems **46**(4), 195 – 204 (2004)
14. 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)
15. Guruprasad, K.R., Ghose, D.: Automated multi-agent search using centroidal voronoi configuration. IEEE Transactions on Automation Science and Engineering **8**, 420–423 (2011)
16. Guruprasad, K.R., Ranjitha, T.D.: ST-CTC: A spanning tree-based competitive and truly complete coverage algorithm for mobile robots. In: Proceedings of the 2015 Conference on Advances In Robotics, AIR '15, pp. 43:1–43:6. ACM, New York, NY, USA (2015)
17. 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)
18. Horváth, E., Pozna, C., Precup, R.E.: Robot coverage path planning based on iterative structured orientation. Acta Polytechnica Hungarica **15**, 231–249 (2018)
19. Kim, Y.H., Shell, D.A.: Distributed robotic sampling of non-homogeneous spatio-temporal fields via recursive geometric sub-division. Proceedings - IEEE International Conference on Robotics and Automation pp. 557–562 (2014)
20. Lahijanian, M., Maly, M.R., Fried, D., Kavraki, L.E., Kress-Gazit, H., Vardi, M.Y.: Iterative temporal planning in uncertain environments with partial satisfaction guarantees. IEEE Transactions on Robotics **32**(3), 583–599 (2016)
21. Lavalle, S.M.: Rapidly-exploring random trees: A new tool for path planning. Tech. rep. (1998)
22. Lee, S.K., Fekete, S.P., McLurkin, J.: Structured triangulation in multi-robot systems: Coverage, patrolling, voronoi partitions, and geodesic centers. The International Journal of Robotics Research **35**, 1234–1260 (2016)
23. Lin, L., Goodrich, M.A.: Hierarchical Heuristic Search Using a Gaussian Mixture Model for UAV Coverage Planning. IEEE transactions on cybernetics **44**, 2532–2544 (2014)
24. Little, J.D., Murty, K.G., Sweeney, D.W., Karel, C.: An algorithm for the traveling salesman problem. Operations research **11**, 972–989 (1963)
25. Marthaler, D., Bertozzi, A.L.: Tracking Environmental Level Sets with Autonomous Vehicles, pp. 317–332. Springer US (2004)
26. 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)
27. 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)
28. Mitchell, D., Corah, M., Chakraborty, N., Sycara, K., Michael, N.: Multi-robot long-term persistent coverage with fuel constrained robots. In: 2015 IEEE International Conference on Robotics and Automation (ICRA), pp. 1093–1099 (2015)
29. Okabe, A.: Spatial tessellations: concepts and applications of Voronoi diagrams. Wiley series in probability and statistics: Applied probability and statistics. Wiley (2000)
30. 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)
31. 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)
32. Pham, H., Moore, P.: Robot coverage path planning under uncertainty using knowledge inference and hedge algebras. Machines **6**, 46 (2018)
33. Pham, H.V., Moore, P., Truong, D.X.: Proposed smooth-stc algorithm for enhanced coverage path planning performance in mobile robot applications. Robotics **8**(2) (2019)
34. Pinkam, N., Jeong, S., Chong, N.Y.: Exploration of a group of mobile robots for multiple radiation sources estimation. In: 2016 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS), pp. 199–206 (2016)
35. Pinkam, N., Newaz, A.A.R., Jeong, S., Chong, N.Y.: Rapid coverage of regions of interest for environmental monitoring. In: J.H. Kim, H. Myung, J. Kim, W. Xu, E.T. Matson, J.W. Jung, H.L. Choi (eds.) Robot Intelligence Technology and Applications 5, pp. 195–209. Springer International Publishing, Cham (2019)
36. Saldaña, D., Assunção, R., Campos, M.F.M.: Predicting environmental boundary behaviors with a mobile robot. IEEE Robotics and Automation Letters **1**, 1133–1139 (2016)
37. 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)
38. Strimel, G.P., Veloso, M.M.: Coverage planning with finite resources. In: IEEE International Conference on Intelligent Robots and Systems, pp. 2950–2956 (2014)
39. Susca, S., Bullo, F., Martinez, S.: Monitoring environmental boundaries with a robotic sensor network. IEEE Transactions on Control Systems Technology **16**, 288–296 (2008)
40. Tiwari, K., Jeong, S., Chong, N.Y.: Point-wise fusion of distributed gaussian process experts (fudge) using a fully decentralized robot team operating in communication-devoid environment. IEEE Transactions on Robotics **34**(3), 820–828 (2018)

41. Van Pham, H., Asadi, F., Abut, N., Kandilli, I.: Hybrid spiral stc-hedge algebras model in knowledge reasonings for robot coverage path planning and its applications. Applied Sciences **9**(9) (2019)
42. Willett, R., Nowak, R.: Minimax Optimal Level Set Estimation. IEEE Transactions on Image Processing **16**, 2965–2979 (2007)
43. Xu, L.: Graph Planning for Environmental Coverage. Carnegie Mellon University (August), 135 (2011)
44. Yamauchi, B.: A frontier-based approach for autonomous exploration. In: Computational Intelligence in Robotics and Automation, 1997. CIRA'97., Proceedings., 1997 IEEE International Symposium on, pp. 146–151. IEEE (1997)
45. Yehoshua, R., Agmon, N., Kaminka, G.A.: Robotic adversarial coverage of known environments. The International Journal of Robotics Research pp. 1–26 (2016)
46. Zafar, M.N., Mohanta, J.: Methodology for path planning and optimization of mobile robots: A review. Procedia Computer Science **133**, 141 – 152 (2018). International Conference on Robotics and Smart Manufacturing (RoSMa2018)
47. Zucker, M., Kuffner, J., Branicky, M.: Multipartite rrts for rapid replanning in dynamic environments. In: Robotics and Automation, 2007 IEEE International Conference on, pp. 1603–1609. IEEE (2007)