

CONF-950887-2

Cooperative Terrain Model Acquisition by Two Point-Robots in Planar Polygonal Terrains †

Nageswara S. V. Rao and V. Protopopescu

Center for Engineering Systems Advanced Research

Oak Ridge National Laboratory

Oak Ridge, Tennessee 37831-6364

{raons,protopopesva}@ornl.gov

November 29, 1994

"The submitted manuscript has been authored by a contractor of the U.S. Government under contract No. DE-AC05-84OR21400. Accordingly, the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes."

Submitted to: *IEEE/RSJ International Conference on Intelligent Robots and Systems, August 5-9, 1995, Pittsburgh, PA*

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

†Research sponsored by the Engineering Research Program of the Office of Basic Energy Sciences, of the U.S. Department of Energy, under Contract No. DE-AC05-84OR21400 with Martin Marietta Energy Systems, Inc.

MASTER

Cooperative Terrain Model Acquisition by Two Point-Robots in Planar Polygonal Terrains

Nageswara S. V. Rao and V. Protopopescu
Center for Engineering Systems Advanced Research
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831-6364
{raons,protopopesva}@ornl.gov

Abstract

We address the model acquisition problem for an unknown terrain by a team of two robots. The terrain may be cluttered by a finite number of polygonal obstacles whose shapes and positions are unknown. The robots are point-sized and equipped with visual sensors which acquire all visible parts of the terrain by scan operations executed from their locations. The robots communicate with each other via wireless connection. The performance is measured by the number of the sensor (scan) operations which are assumed to be the most time-consuming/expensive of all the robot operations. We employ the restricted visibility graph methods in a hierarchical setup. For terrains with convex obstacles the sensing time can be shown to be halved compared to that of a single robot implementation. For terrains with concave corners, the performance of the algorithm depends on the number of concave regions and their depths. A hierarchical decomposition of the restricted visibility graph into 2-connected components and trees is considered. The performance for the team of two robots is expressed in terms of the sizes of 2-connected components, and the sizes and diameters of the trees. The proposed algorithm and analysis can be applied to the methods based on Voronoi diagram and trapezoidal decomposition.

Keywords and Phrases: path planning, Voronoi diagram, trapezoidal decomposition, terrain model acquisition, two robot team.

1 Introduction

The *terrain model acquisition problem* (TMAP) deals with robots autonomously acquiring a complete model of a terrain (or environment) by systematically visiting portions of it. The motivation for this problem is at least two-fold.

- (a) **Efficiency in Future Navigation:** Once the terrain model is completely acquired, the navigation algorithms of known terrains can be employed for path planning with two potential advantages. First, the sensors may be switched off (at least in theory) in future navigation, thereby avoiding the time-consuming sensor operations involved in the acquisition and processing of sensor data. Second, navigation paths with the shortest distance between the start and goal positions may be computed using the terrain model. If the terrain model is not available, no algorithm can always guarantee the shortest paths. Consequently, a robot can recognize a dead-end corner only after it has moved into it and explored it; such situation can be avoided if the terrain map is available.

(b) **Assistance to Human Model Builders:** In applications involving mobile robots in indoor environments for repetitive operations, typically a human operator is in charge of model building (which is tedious and time-consuming). Robots capable of autonomous terrain model acquisition (even in only small parts of the terrain) can be employed to relieve her/him of part of the work.

The focus here is on *non-heuristic* algorithms that are guaranteed to converge within the assumptions of the formulation. The terrain model acquisition problem for three dimensional polyhedral terrains has been solved by using the visibility graph structure by Rao *et al.* [14] for the case of a discrete vision sensor. In the plane, the restricted visibility graph, which is obtained by removing all non-convex vertices from the visibility graph, is shown to suffice [13]. The same problem can also be solved by using a method based on the Voronoi diagram [16]. The terrain model acquisition problem in the case of a robot equipped with a continuous vision sensor has been solved by Lumelsky *et al.* [11]. The terrain model acquisition problem for terrains where the obstacle boundaries consist of circular arcs and straight lines can be solved by the methods of visibility graphs, Voronoi diagram and trapezoidal decomposition using discrete and continuous vision sensors [12]. A survey of non-heuristic algorithms for navigation in unknown terrains and terrain model acquisition can be found in Rao *et al.* [15].

To our knowledge, the problem of acquiring a model of an unknown terrain by a *team* of robots has not been addressed in the formulation of non-heuristic algorithms. This problem, however, has been studied by a number of researchers outside the non-heuristic formulation. For example, Ishioka *et al.* [7] describe a cooperative map generation by heterogeneous autonomous mobile robots (also see Dudek *et al.* [4]). A cooperative recognition system for the environment using multiple robots has been developed by Ishiwata *et al.* [8]. Our orientation is more along the lines of the unknown terrains algorithms pioneered by Lumelsky [10]. In terms of the cooperative terrain model acquisition by two robots, the formulation that comes closest to ours is the maze-searching algorithm by two pebble automata by Blum and Kozen [1]. On the other hand, the navigation of multiple robots in known and unknown terrains has been studied by a number of researchers. Most of the existing papers on this problem are devoted to the case of known terrains, i. e. a terrain model is supplied to the robot (see, for example, Latombe [9] for a comprehensive discussion). For unknown terrains, the recent study by Harinarayan and Lumelsky [6] indicates that the simultaneous navigation of two robots cannot be solved if no “cooperation” is present between them. Note that our higher-level objective is different from theirs in two ways: (a) we are interested in terrain model acquisition, and (b) we wish to explore the cooperation mechanisms so that the objective can be achieved more effectively by a team of robots instead of a single robot.

Each robot is equipped with a visual sensor that detects all visible boundaries of the terrain from the present location by a *scan* operation. We consider that the sensor time carries the overwhelming cost of the robot operations. This is a reasonable assumption in the context of indoor navigation of mobile robots in terrains that are cluttered with unknown obstacles. In systems employing vision systems in this context, each visual scan may take of the order of minutes including the time required to acquire and process the sensory data. The time required for terrain model acquisition is dominated by the total number of scan operations performed, called the sensing time. In our model the robots communicate via an

ideal wireless connection to transfer finite sequences of *real*¹ numbers.

Our formulation deals with strategies used by two robots that start at the same location in an unknown planar polygonal terrain. The objective is to investigate the advantages of using two robots instead of one. In “very bad” cases, e. g. if the robots are initially located at one end of a “long narrow polygonal corridor”, there may not be any advantage in employing two robots. In the present formulation, the robots typically perform a scan operation and decide the next destination from which the algorithm is repeated. Thus in a long narrow corridor the robots are forced to “stay” together. If the terrain has “branches” so that the robots can explore different parts of the terrain, two robots are likely to acquire the terrain faster than one robot.

We show that the terrain model acquisition method based on the restricted visibility graph (RVG) method [13] can be advantageously implemented by two robots. Using a single robot, the terrain model can be obtained in N_c sensor operations, where N_c is the total number of convex obstacle vertices [13]. This method can be used to expedite the terrain model acquisition using two robots. We first show that if all obstacles are convex, the sensing time can be halved. If the terrain consists of non-convex obstacles, RVG can be decomposed into 2-connected components C_1, C_2, \dots, C_{n_c} and trees T_1, T_2, \dots, T_{n_t} . The C_i ’s can be assigned to l different levels in a hierarchical decomposition of RVG (see Section 5 for precise definitions). Some T_j ’s connect C_i ’s of different levels while some are attached to a C_i of a single level. Then the hierarchy tree T_0 is a tree obtained by condensing each of C_i ’s to a node and removing the trees that do not connect C_i ’s of different levels. Informally, T_0 captures the nesting level of the terrain that can, in a worst case, force the two robots to follow each other, thereby obviating the advantage of two robots. Let $|C_i|$, $i = 1, 2, \dots, n_c$, and $|T_j|$, $j = 0, 1, \dots, n_t$ denote the number of nodes of C_i and T_i respectively. The total sensing time for a single robot, based on the RVG method, is given by

$$\sum_{i=1}^{n_c} |C_i| + \sum_{j=1}^{n_t} |T_j|.$$

The sensing time achieved by a team of two robots, based on the proposed method, is upper-bounded by

$$\frac{1}{2} \sum_{i=1}^{n_c} |C_i| + \frac{1}{2} \sum_{j=1}^{n_t} |T_j| + \frac{1}{4} [d(T_0) + |T_0|] + \frac{l}{4} \max_{i \in \{1, \dots, n_t\}} [d(T_i) + |T_i|]$$

where $d(T_i)$ is the diameter of the tree T_i , which is the number of nodes on the longest path of T_i . When all obstacles are convex, RVG does not contain any trees in which case the sensing time is halved using two robots as per the above estimate. On the other hand, if RVG is a single path, typically corresponding to a long corridor, then RVG does not have any 2-connected components and it consists of a single tree T_1 with $d(T_1) = |T_1|$. Then the above bound evaluates to $\frac{3}{2}|T_1|$, compared to the required sensing time of $|T_1|$ (in a worst case where both robots start at one end of the corridor).

¹Since, in general a real number carries an infinite, uncompressible amount of information, this hypothesis may seem unrealistic. However, for the specific aspects of the present problem, this is not crucial. This hypothesis is similar in spirit to the infinite precision arithmetic often assumed to be available in the study of path planning problem [9].

To place our formulation in perspective, we briefly summarize the formulations of Blum and Kozen [1], and Harinarayan and Lumelsky [6], and then distinguish them from ours. Blum and Kozen [1] describe a method by which two pebble automata cover all the free cells of a maze. The communication between the pebble automata is achieved by using the pebbles that are placed on free cells. Also, the pebble automata can recognize the pebbles but are not equipped with computational mechanisms to keep track of the coordinates of the cells. On the contrary, we assume that the robots can store and manipulate real numbers with arbitrary precision. In terms of the terrain, the maze consisting of a two-dimensional arrangement of cells [1] is much simpler than the polygonal terrain considered here. This maze exploration problem is similar to the terrain model acquisition problem here in that the automata are required to visit all free cells of the maze. The problem studied by Harinarayan and Lumelsky [6] deals with two robots that do not communicate with each other. So the robots recognize each other when they “see” each other, whereas in our case one robot knows the past location and next destination of the other robot at all times.

The organization of the paper is as follows. The problem formulation and some preliminaries are described in Section 2. The case of terrain model acquisition in convex polygonal terrains and along tree structures are described in Sections 3 and 4 respectively. The terrain model acquisition in polygonal terrains is considered in Section 5. Several variations of the solution of Section 5 are considered in Section 6.

2 Preliminaries

We consider a finite two-dimensional terrain populated by a finite and non-intersecting set of polygons, $O = \{O_1, O_2, \dots, O_m\}$. An obstacle vertex is called a *convex vertex* if the angle included inside the obstacle by the edges that meet at this vertex is less than 180 degrees, and the obstacle vertex is called *concave* otherwise.

Let the *free-space*, denoted by Ω , be the subset of the plane given by $\bigcap_{i=1}^n O_i^C$, where O_i^C denotes the complement of O_i in the plane. Let $\bar{\Omega}$ denote the closure of Ω . Two points p and q in $\bar{\Omega}$ are *visible* to each other if the line segment joining p and q , denoted by \overline{pq} , is completely contained in $\bar{\Omega}$.

The robot, denoted by R , is point-sized and equipped with a vision sensor. A *discrete vision sensor* is characterized by a *scan* operation: a scan operation performed from a position (point) p returns the *visibility polygon* of p , which is the polygonal region consisting of all points in the terrain visible to p (Fig. 1).

The robots communicate with each other in terms of finite sequences of real numbers such that a real number can be communicated in a small time unit via the wireless connection.

We assume that the most time consuming-part of the robot operation is the scan operation, and hence the time of terrain model acquisition by a single robot is dominated by the total time of scan operations. The total *sensing time* is given by the number of scan operations performed by the robot(s) in sequence.

The *restricted visibility graph* is defined as follows [13]. The vertices of the RVG are the convex obstacle vertices. Two vertices are connected by an edge if and only if they are visible from each other or they are the end vertices of an obstacle edge. The RVG is shown to be connected and satisfies the terrain-visibility property which implies that the union of the

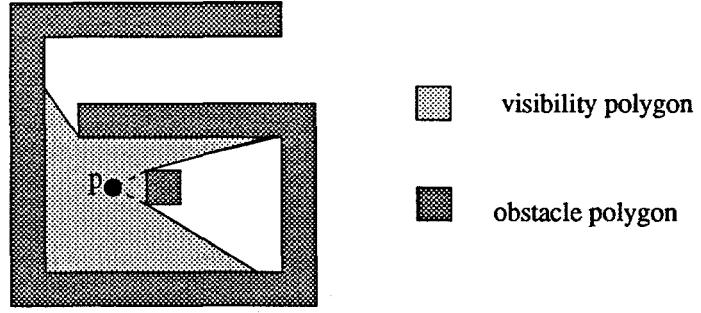


Figure 1: Visibility polygon from location p .

visibility polygons of all the vertices of RVG contains the entire free-space [13]. The latter property implies that any graph search algorithm implemented by a robot will completely acquire the terrain model in a sequence of N_c scan operations, i. e. the sensing time for solving TMAP by a single robot is given by N_c .

We use some terminology from graph theory, e. g. connectivity, condensation, decomposition, etc., which can be found in books on graph theory (e. g. Harary [5]).

3 Convex Polygonal Terrains

In this section, we consider the terrain composed of convex polygonal obstacles. The objective of the terrain model acquisition algorithm is to perform a scan operation from every node of RVG which guarantees that the entire free-space is seen. For solving the TMAP with a single robot, it suffices to perform a graph search, e.g. depth-first search, which has a sensing time of N_c .

The overall algorithm for two robots is based on the robots executing a graph search algorithm in a cooperative manner. At any step, each robot has the same version partial RVG. For the team of robots R_1 and R_2 , let R_1 have a higher priority in the following sense. Each robot performs a scan operation and obtains the resultant visibility polygon. Then each robot computes its own adjacency list and communicates it to the other robot. Then R_1

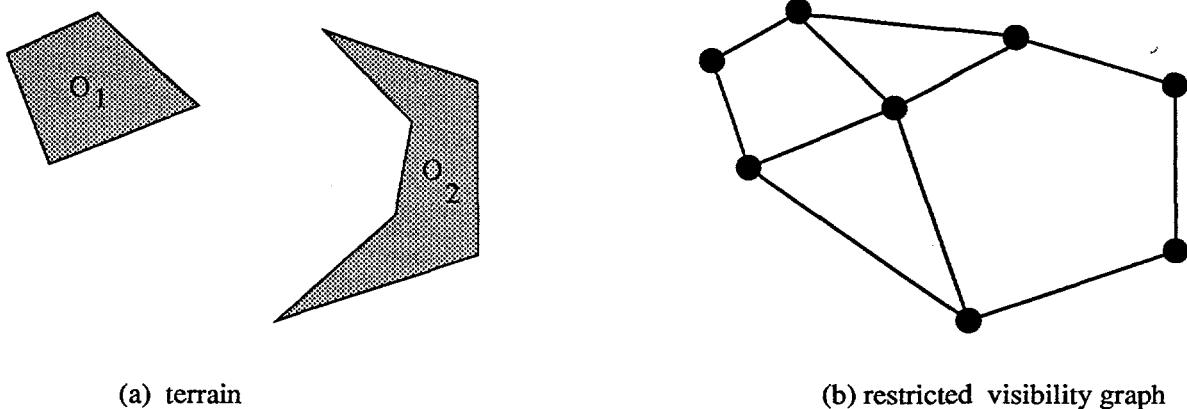


Figure 2: Restricted visibility graph.

sends its next destination which is one of the nodes adjacent to its present location to R_2 . If the destinations of R_1 and R_2 are different, then both move to their destinations and repeat the algorithm. If both destinations are the same, R_2 chooses a different destination and communicates it to R_1 following a certain strategy. For concreteness, we discuss a strategy based on the depth-first search (see, for example, Cormen *et al.* [2] for details on depth first search). R_2 chooses an unvisited vertex adjacent to its present location, if such vertex exists. If not, R_2 backtracks along the path towards its starting vertex until it is located at vertex with an adjacent vertex that had not been visited so far or has been chosen by R_1 ; R_2 then communicates its destination to R_1 . Then R_1 and R_2 move to their chosen destinations and repeat the algorithm.

Note that an adjacency list computed after a scan operation consists of (possibly empty) set of visited vertices and not yet visited vertices. The above algorithm terminates when all known vertices have been visited; the connectivity property together with the terrain-visibility property ensure that the terrain model had been completely acquired.

In order that the above algorithm be executed, we need to establish that R_2 can always find its destination. The required property is the 2-connectedness of the RVG which is established for convex polygonal terrains.

Lemma 3.1 *The restricted visibility graph of a terrain cluttered by a finite number of convex polygonal obstacles satisfies the following properties: (a) there is a path between any two nodes u and v containing a node w , and also there is a path between u and v not containing w , and (b) there are two node-disjoint paths between any two nodes of RVG, i. e. RVG is 2-connected.*

Proof: We show this lemma by induction on the number of obstacles. Both Part (a) and (b) are true for a terrain consisting of a single convex polygon. Assume that the claim true for a terrain of k obstacles; let RVG_k denote the RVG of the k obstacles. Now place the $(k+1)$ th polygon P_{k+1} . The edges of the RVG_k that are intersected by the new polygon are rerouted along the boundary of P_{k+1} . First consider Part (a). The claim is true individually for the sets of vertices of P_{k+1} and vertices of RVG_k . Now consider the properties between the vertices of P_{k+1} and RVG_k . There are at least two edges between the vertices of P_{k+1} and the vertices of RVG_{k+1} as illustrated in Fig. 3. It is clear that any node u can be included in a path between v_1 and v_2 of P_{k+1} by using the path v_1, u_1, u, u_2, v_2 . Similarly any node v can be included in a path between pair w_1 and w_2 of RVG_k as follows. By Theorem 5.14 of Harary [5], there are two node disjoint paths joining w_1 to u_1 and w_2 to u_2 (since by hypothesis RVG_k is 2-connected); then the required path is given by $w_1, u_1, v_1, v, v_2, u_2, w_2$. Almost identical argument shows the second part of (a) that a chosen vertex can be excluded from the path between two vertices.

Now consider the Part (b). The required 2-connectivity among the nodes of RVG_k is preserved since no paths are broken by P_{k+1} . The required 2-connectivity among the nodes of P_{k+1} is trivially satisfied. Now consider the 2-connectivity between a vertex v of P_{k+1} and u of RVG_k . By hypothesis, there is a path P_u between u_1 and u_2 via u going through only the vertices of RVG_k . This path is vertex disjoint from the path v_1, v, v_2 , and thus the paths v, v_1, u_1, u and v, v_2, u_2, u obtained by employing the pieces of P_u , are vertex disjoint. \square

Note that 2-connectivity implies that the above RVG cannot be disconnected by removing a single vertex (or equivalently the graph does not have *cut points* whose removal will reduce

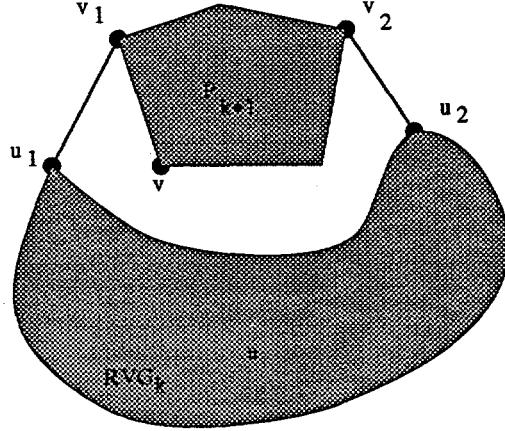


Figure 3: Illustration of inductive step for Lemma 2.1.

the graph into two or more disconnected pieces [5]). Then the sensing time of the above algorithm can be shown to be $\lceil N_c/2 \rceil$, where $\lceil x \rceil$, for real x , denotes the smallest integer larger than or equal to x . Notice that R_1 will move along a path without ever having to backtrack. If we show that R_2 finds its destination at each step, then by the time R_1 performed $\lceil N_c/2 \rceil$ scan operations, R_2 would have performed scan operations from the rest of the nodes of RVG. Now we shall show that R_2 is always guaranteed to find a destination. First notice that RVG is connected which implies that every vertex of RVG is reachable from present location of R_2 . Let the next destination chosen by R_1 at this step be denoted by v . By connectivity, if there is an unvisited node (other than v), then there is at least one unvisited node adjacent to the paths traced by R_1 or R_2 . If not, all the unvisited nodes can only be reached via v , which makes v a cut point; this in turn contradicts the 2-connectedness of RVG. Thus by the time R_1 performs $\lceil N_c/2 \rceil$ scan operations, R_2 would have performed scan operations from rest of the nodes of RVG. In summary we have the following theorem.

Theorem 3.1 *The terrain model of a polygonal terrain cluttered by a finite number of convex obstacles can be obtained by a team of 2 robots in a sequence of $\lceil N_c/2 \rceil$ scan operations, where N_c is the total number of obstacle vertices.*

The above method can be replaced by several other methods. We now briefly consider another algorithm that can be used by a team of two robots. The outline of the algorithm can be visualized as follows: Consider the convex hull of the terrain. The boundary of the terrain is called the *outer path* which consists of polygonal obstacle boundaries separated by straight line segments. Then obtain the *inner path* by (a) identifying the alternative paths for the non-obstacle parts of the outer path, and (b) replacing each obstacle chain of the outer path by the other path around the obstacle. An illustration is shown in Fig. 4. Then all obstacles that are part of the paths at this level are removed, and the procedure is recursively carried out. As a result we obtain layers of inner and outer paths. The algorithm for TMAP is to have the two robots move along different layers as long as possible.

4 Acquisition Along A Tree Structure

The RVG for a polygonal terrain can be decomposed into trees and 2-connected components (see Fig. 5). While the 2-connected components can be easily explored in parallel by two robots, the trees may prevent such explorations. If the RVG consists of single path, then the sensing time cannot be improved by employing a team of robots in a worst case.

We now consider the case where the team of robots explore a tree which is connected. Notice that in a worst case, $d(T)$ is the minimum time required to explore a tree T by a team of two robots. The strategy is for both robots to stay together until the first opportunity to move along two edges of a tree. While the robots are in two different branches of the tree, sensor operations are done simultaneously. At the same time the robots will not be together for more than $d(T)$ time since the diameter is the longest possible distance (in terms of sensor operations) that the robots will stay together without branching off. To see this, assume that it is not true, then we have sequences of paths (without branching) whose total length is longer than $d(T)$; since the tree is connected and has no cycles, the union of these paths constitutes a path of length larger than $d(T)$, which is a contradiction. Thus $|T| - d(T)$ scan operations are performed while the robots are not together. Hence, the sensing time required to explore a tree is upper-bounded by $d(T) + |T|/2 - d(T)/2 = \frac{1}{2}[|T| + d(T)]$. Thus we have the following result.

Lemma 4.1 *The sensing time of exploring a tree T of $|T|$ nodes by two robots is upper-bounded by $d(T)/2 + n/2$, where $d(T)$ is the diameter of T .*

Notice that in a worst case of T being a single path the above lemma yields a sensing time of $|T|$ which is actually the required (worst case) sensor time.

5 Polygonal Terrains

The restricted visibility graph of the terrain can be decomposed into a hierarchy of levels such each level consists of a union of 2-connected parts and trees. Such hierarchy can be defined

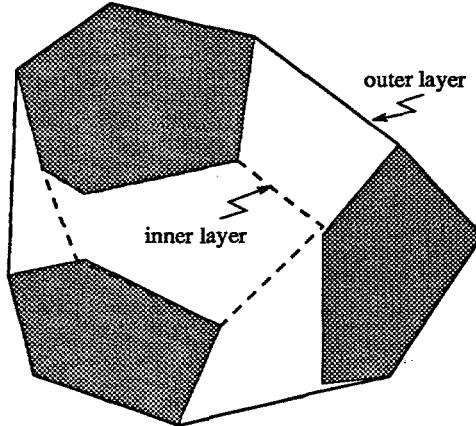


Figure 4: Illustration of inner and outer layers.

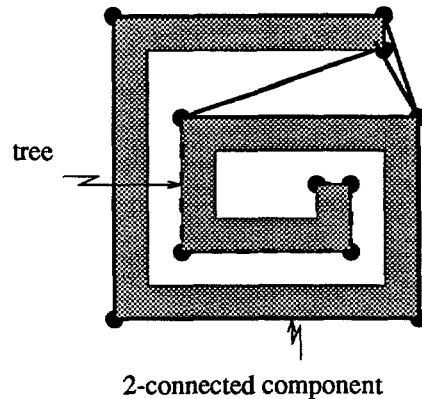


Figure 5: Decomposition of RVG into 2-connected parts and trees.

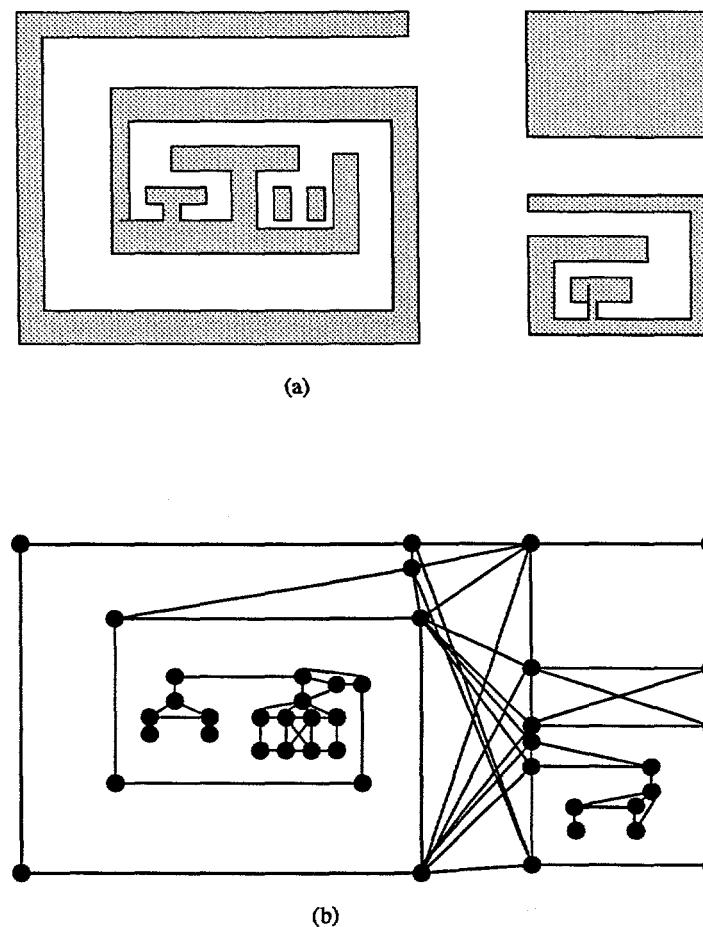


Figure 6: Example of terrain and RVG. RVG of (a) is shown in (b).

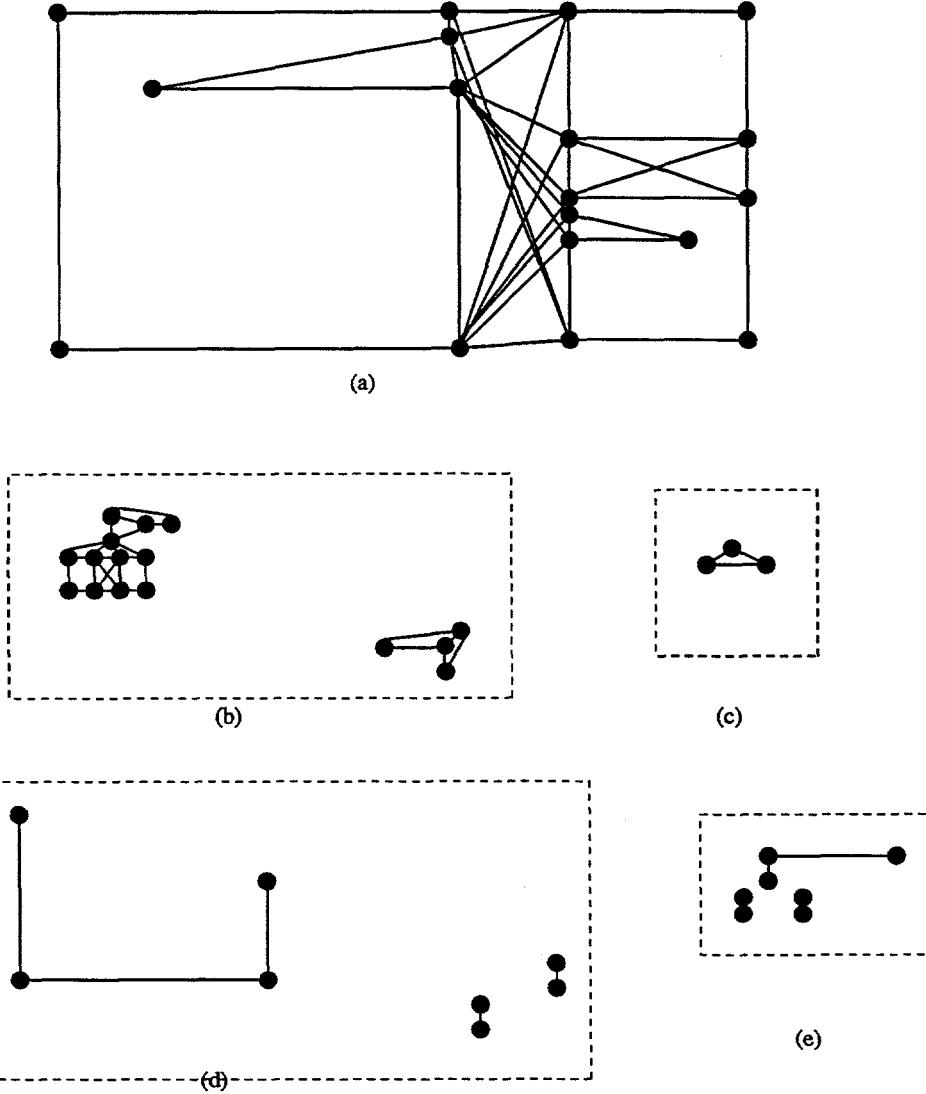


Figure 7: Illustration of hierarchical decomposition of RVG of Fig. 6.

with respect to any starting location. For concreteness, we assume that the initial location is outside the convex hull of the obstacle vertices. For illustration consider Fig. 6. We identify the 2-connected component corresponding to the initial location. This 2-connected component for the RVG of Fig. 6(b) is shown in Fig. 7(a). Then we remove this component and all the trees that are emanating from this component and identify the 2-connected components of the next level as shown in Fig. 7(b). The same process is repeated to identify the next levels of 2-connected components as shown in Fig. 7(c).

Trees of various levels are identified as follows. For any level, we specify the trees that emanate from the nodes of the 2-connected components of that level. Fig. 7(d) and (e) show the trees emanating from 2-connected components of level 1 and 2 respectively. There are two types of trees. The *first type* are the trees that connect the nodes of one hierarchy with nodes at another hierarchy, and the *second type* are the trees that strictly belong to one hierarchy. In Fig. 7(d), the left and right trees belong to the former type and the middle one

belongs to the second type.

We obtain a *hierarchy tree* from RVG by condensing each 2-connected component of the hierarchical decomposition to a node and removing the trees of second kind. The resultant tree is denoted by T_0 . An example of hierarchy tree is shown in Fig. 8 where the hollow circles represent nodes obtained by the condensation of 2-connected components.

The overall strategy for solving TMAP by two robots is to avoid keeping two robots in the same tree T_i to the maximum extent possible. Using this strategy, the robots will explore different trees until there is at most one tree left to be (possibly partially) explored concurrently at the current level of hierarchy. This strategy can be implemented as follows. Notice that the end points of trees can be recognized by a local concavity, but a local concavity does not necessarily indicate the presence of a tree. The strategy is to delay sending the robots into the same local concavity until this becomes the only available option at that particular level.

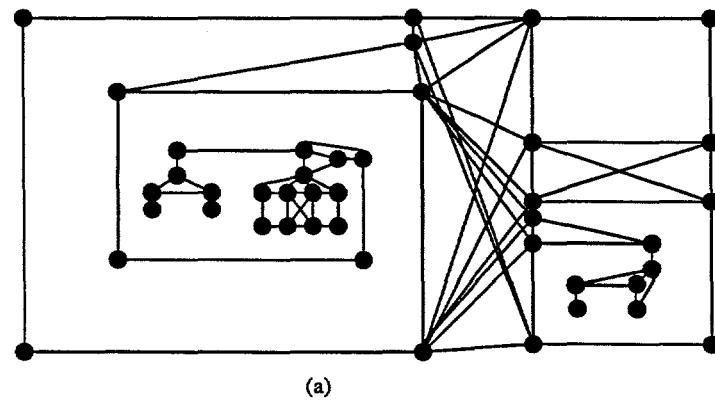
Now consider the performance of the above method. Since the terrain is unknown, the order in which the individual trees are explored is unknown. We carry out a worst case analysis for each of the l levels of the hierarchy. Consider the level k with 2-connected components $C_1^k, C_2^k, \dots, C_{n_t^k}^k$, and the trees $T_{11}^k, T_{12}^k, \dots, T_{1n_t^k}^k$ and $T_{21}^k, T_{22}^k, \dots, T_{2n_t^k}^k$ of first and second type respectively. Also let $T_k = \{T_{11}^k, T_{12}^k, \dots, T_{1n_t^k}^k\} \cup \{T_{21}^k, T_{22}^k, \dots, T_{2n_t^k}^k\}$. The size of the tree that is left to be explored last is upper-bounded by

$$\max_{I,J} \left| \sum_{T \in I} |T| - \sum_{T \in J} |T| \right|$$

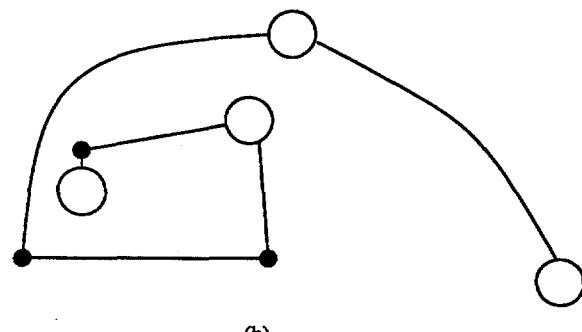
where the maximum is taken over all sets I and J such that $I \cup J = T_k, I \cap J = \phi, ||I| - |J|| = 1$. Now note that this quantity is upper-bounded by $\max_{T \in T_k} |T|$, which in turn implies, from Lemma 4.1, that the sensing time is upper-bounded by $\frac{1}{2} \max_{T \in T_k} [d(T) + |T|]$. For the level k , the number of scan operations that are performed simultaneously by the two robots is at least $\sum_{T \in T_k} |T| - \frac{1}{2} \max_{T \in T_k} [|T| + d(T)]$. Thus, the total sensing time for level k required by a team of two robots is upper-bounded by

$$\begin{aligned} & \frac{1}{2} \sum_{i=1}^{n_t^k} |C_i^k| + \frac{1}{2} \left[\sum_{T \in T_k} |T| - \frac{1}{2} \max_{T \in T_k} [|T| + d(T)] \right] + \frac{1}{2} \max_{T \in T_k} [|T| + d(T)] \\ & \leq \frac{1}{2} \sum_{i=1}^{n_t^k} |C_i^k| + \frac{1}{2} \sum_{T \in T_k} |T| + \frac{1}{4} \max_{T \in T_k} [|T| + d(T)]. \end{aligned}$$

The summation of above quantity over all levels yields an upper-bound on the sensing time of the team. The summation of the first two quantities yields $\frac{1}{2} \sum_{i=1}^{N_c} |C_i|$ and $\frac{1}{2} \sum_{j=1}^{N_t} |T_j|$ respectively. The summation of the third term is handled as follows. At every level only one tree (if any) either of type one or two is last explored by the two robots (in a worst case). Thus the contribution to the upper-bound by the trees of type two is no more than $\frac{1}{4} \max_i [d(T_i) + |T_i|]$, and the contribution to the upper-bound by the trees of first type is upper-bounded by $\frac{1}{4} [d(T_0) + |T_0|]$. Thus we have the following theorem.



(a)



(b)

Figure 8: Illustration of hierarchy tree.

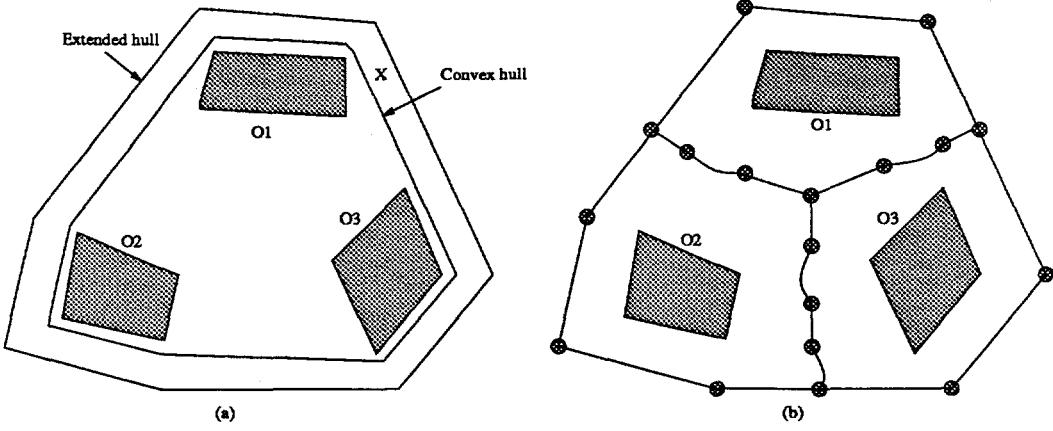


Figure 9: Navigation course based on Voronoi diagram.

Theorem 5.1 *The sensing time for two robots to acquire the model of a terrain of polygonal obstacles is upper-bounded by*

$$\frac{1}{2} \sum_{i=1}^{n_c} |C_i| + \frac{1}{2} \sum_{j=1}^{n_t} |T_i| + \frac{1}{4} [d(T_0) + |T_0|] + \frac{l}{4} \max_{i \in \{1, \dots, n_t\}} [d(T_i) + |T_i|].$$

Notice that for terrains with convex polygonal obstacles, RVG consists of only one 2-connected component. Thus this theorem subsumes Theorem 3.1. If RVG is a single tree T , then $T_0 = T$; for this case, since this theorem yields a weaker upper-bound, it does not precisely subsume Lemma 4.1. The performance of the two robots is decided (in a worst case) by the depth l of the hierarchy described above. For typical office indoor environments l is of the order 2. On the other hand, deeply nested mazes can generate large values for l .

6 Variations

We now consider two geometric structures that are used for terrain model acquisition in unknown terrains. These structures can be employed in place of RVG of last section.

(a) **Voronoi Diagram:** The Voronoi diagram corresponding to a set of line segments and circular arc segments has been studied by Yap [17]. The *distance* $d(p, s)$ between a point p in free-space and a boundary edge s is defined as $\inf\{d(p, q) | q \in s\}$. The *clearance* of a point p in free-space with respect to O is the minimum of $d(p, s)$ for some obstacle edge (segment or an arc) s of O . For $x \in \Omega$, we define *Near*(x) as the set of points that belong to the boundaries of obstacles O_i , $i = 1, 2, \dots, n$ and are closest (among all points on the obstacle boundaries) to x in terms of the metric d . The *Voronoi diagram*, $Vor(O)$, of the terrain populated by O is the set $\{x \in \Omega | Near(x) \text{ is a disconnected set}\}$, (i. e. for each $x \in Vor(O)$ the set *Near*(x) contains more than one topologically connected components or equivalently $x \in Vor(O)$ is nearest two at least two distinct points on the obstacle boundary). This definition implies that for each $x \in Vor(O)$ there are at least two distinct points on the obstacle boundary that are closest to x in the metric d . See Fig. 9 for an example. In this case, $Vor(O)$ is a union of $O(N)$ straight lines and algebraic arcs of degree at most four [17].

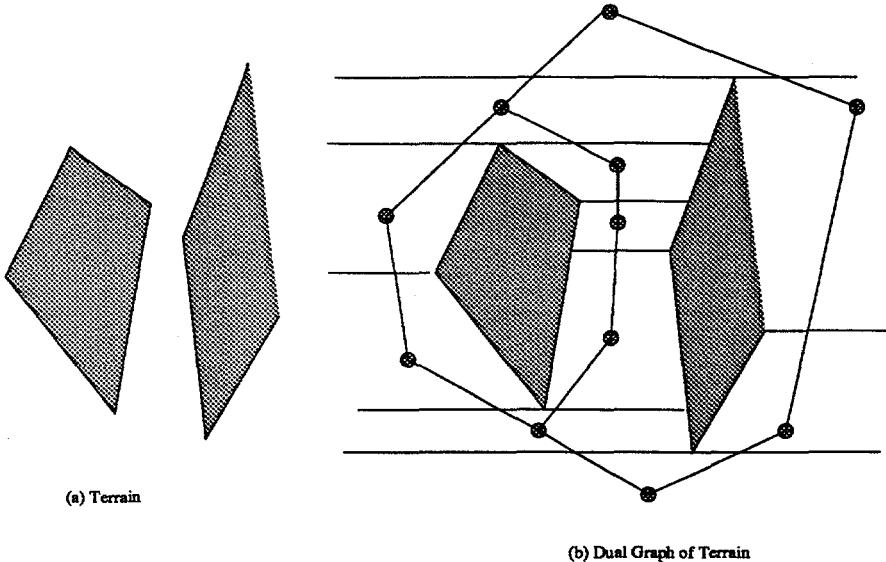


Figure 10: Navigation course based on trapezoidal decomposition.

(b) **Dual graphs based on trapezoidal decomposition:** First, we decompose the free-space into trapezoids by sweeping a line (for example, moving a horizontal line from top to bottom) such that whenever the line passes through a vertex, extend a *sweep-line segment* from this vertex into free-space until it touches an obstacle boundary or extends to infinity as shown in Fig. 10. Now free-space is partitioned into trapezoids. There are many ways to define dual graphs based on the decomposition, and we consider a version that is suited for the present problem. For each sweep-line segment we have one of the two following cases: (a) if the segment is finite, the dual graph node corresponds to the mid point of the segment , or (b) if the segment is not finite, the dual graph node corresponds to a point on the segment at a distance δ from the vertex. Two nodes belonging to the boundary of the same trapezoid are connected by an edge of the dual graph. See Fig. 10 for an example of a dual graph based on the trapezoidal decomposition.

Each of the above structures can be combined with a graph search algorithm to solve TMAP by a single robot [12]. Notice that both the structures can be decomposed into 2-connected components and trees and thus results along the lines of Theorem 5.1 can be derived for the case of two robots. In particular, the structure of the hierarchy tree for these two will be similar to that of RVG.

In terms of the sensing time for a single robot, the Voronoi diagram method could require a larger number of scan operations, whereas trapezoidal decomposition method yields about the same number as required by the RVG method. The RVG method requires that the robots be capable of navigating along the obstacle boundaries, whereas Voronoi diagram method keeps them as much away from the obstacles as possible. The trapezoidal decomposition method could require that the robot navigate close to obstacles, but less frequently than the RVG method.

If the terrain is composed of generalized polygons whose edges are circular arcs and straight lines, the present method can be applied. In such terrains, a complete solution to

TMAP cannot be guaranteed, but if narrow “generalized” corridors of width smaller than ϵ are treated as obstacles, the present method can be applied using the generalized visibility graphs, Voronoi diagrams, and trapezoidal decompositions [12].

7 Discussion

This paper is an initial attempt to identify the terrain model acquisition problems where it would be beneficial to employ a *team* of robots to perform a task rather than a single robot. Only the sensor time is considered here as a measure of performance, and the main discussion is based on the visibility graph methods. In this context, we have identified the parts of the terrain that can be advantageously explored in parallel and the parts in which having more than one robot may not be effective (in a worst case). Similar conclusions are also valid for methods based on Voronoi diagrams and trapezoidal decompositions. The estimates for the sensing time for two robots derived here are conservative. We believe that alternative characterizations and better performance estimates are possible. Also the method discussed is restricted to one particular way of solving the TMAP, namely, using a graph search on a navigation course [12]. These methods in general do not guarantee that the sensing time or the distance traversed by a single robot is close to the optimal achievable if the terrain model is known. The recently studied class of competitive algorithms for TMAP by Deng *et al.* [3] guarantee that the distance traversed by a single robot is bounded within a factor of the minimum possible value achieved if the terrain model is available. Improving the performance of the algorithms of this type by employing a team of robots will be of future interest.

The effectiveness of employing a team of robots for terrain model acquisition might be judged by other measures of performance such as distance traversed, total time of sensor operations, travel time, etc. For example in the RVG method for a single robot, the distance traversed in solving TMAP is a function of the search algorithm employed, whereas the sensor operations is given by N_c (fixed for a terrain). The analysis of the parameter such as the distance appears to be significantly difficult even for the case of RVG. Another direction of future research could be the deployment of a team of more than two robots.

References

- [1] M. Blum and D. Kozen. On the power of compass (or why the mazes are easier to search than graphs). In *18th Annual Symposium on Foundations of Computer Science*, pages 132–142, October 1978.
- [2] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. MIT Press, Cambridge, MA, 1990.
- [3] X. Deng, T. Kameda, and C. H. Papadimitriou. How to learn an unknown environment. In *Proc. 32nd Ann. Symp. on Foundations of Computer Science*, pages 298–303, 1991.

- [4] G. Dudek, M. Jenkins, E. Milios, and D. Wilkes. Robust positioning with a multi-agent robotic system. In *1993 IJCAI Workshop Series: Dynamically Interacting Robots*, pages 118–123, 1993. Working Notes.
- [5] F. Harary. *Graph Theory*. Addison-Wesley Pub. Co., Reading, MA, 1969.
- [6] K. R. Harinarayan and V. J. Lumelsky. Sensor-based motion planning for multiple mobile robots in an uncertain environment. In *Proceedings of IEEE/RSJ/GI Int. Conf. on Intelligent Robots and Systems*, pages 1485–1492, 1994.
- [7] K. Ishioka, K. Hiraki, and Y. Anzai. Cooperative map generation by heterogeneous autonomous mobile robots. In *1993 IJCAI Workshop Series: Dynamically Interacting Robots*, pages 57–67, 1993. Working Notes.
- [8] Y. Ishiwata, M. Inaba, and H. Inoue. Cooperative recognition of environments by multiple robots. In *Proc. of JSME Annual Conf. on Robotics and Mechatronics*, pages 79–84, 1992.
- [9] J. C. Latombe. *Robot Motion Planning*. Kluwer Academic Pub., Boston, 1991.
- [10] V. Lumelsky. Algorithmic and complexity issues of robot motion in uncertain environment. *Journal of Complexity*, 3:146–182, 1987.
- [11] V. Lumelsky, S. Mukhopadhyay, and K. Sun. Dynamic path planning in sensor-based terrain acquisition. *IEEE Transactions on Robotics and Automation*, 6(4):462–472, 1990.
- [12] N. S. V. Rao. Robot navigation in unknown generalized polygonal terrains using vision sensors. *IEEE Transactions on Systems, Man and Cybernetics*, 1994. to appear.
- [13] N. S. V. Rao and S. S. Iyengar. Autonomous robot navigation in unknown terrains: Visibility graph based methods. *IEEE Transactions on Systems, Man and Cybernetics*, 20(6):1443–1449, 1990.
- [14] N. S. V. Rao, S. S. Iyengar, B. J. Oomen, and R. L. Kashyap. On terrain model acquisition by a point robot amidst polyhedral obstacles. *IEEE Journal of Robotics and Automation*, 3:450–455, 1988.
- [15] N. S. V. Rao, S. Karet, W. Shi, and S. S. Iyengar. Robot navigation in unknown terrains: An introductory survey of non-heuristic algorithms. Technical Report ORNL/TM-12410, Oak Ridge National Laboratory, Oak Ridge, TN, July 1993.
- [16] N. S. V. Rao, N. Stoltzfus, and S. S. Iyengar. A ‘retraction’ method for learned navigation in unknown terrains. *IEEE Transactions on Robotics and Automation*, 7(5):699–707, 1991.
- [17] C. K. Yap. An $O(n \log n)$ algorithm for the Voronoi diagram of a set of simple curve segments. *Discrete and Computational Geometry*, 2:365–393, 1987.