On Multiagent Exploration

Ioannis M. Rekleitis[1] Gregory Dudek[1]
Evangelos E. Milios[2]

[1]Centre for Intelligent Machines, McGill University,
3480 University St., Montreal, Québec, Canada H3A 2A7
[2]Department of Computer Science, York University,
North York, Ontario, Canada M3J 1P3

Abstract

This paper describes a technique for multi-agent exploration of an unknown environment, that improves the quality of the map by reducing the inaccuracies that occur over time from dead reckoning errors.

We present an algorithmic solution, simulation results, as well as a cost analysis and experimental data. The approach is based on using a pair of robots that observe one another's behaviour, thus greatly reducing odometry errors. We assume the robots can both directly sense nearby obstacles and see one another. We have implemented both these capabilities with actual robots in our lab. By exploiting the ability of the robots to see one another, we can detect opaque obstacles in the environment independent of their surface reflectance properties. [1]

1 Introduction

In this paper we discuss how a large environment can be explored, mapped and modified to facilitate subsequent navigation. Our approach is sufficiently robust to be able to cope with environments that may have uneven or slippery terrains, or whose surface reflectance properties are not well suited to conventional sensors.

Observe that conventional approaches to robotic mapping and navigation are typically applied to test environments of rather limited size. Further, the sensing techniques used to both explore the environment and position the robot often make rather optimistic assumptions about the environment: diffuse visual reflectors, substantial reflectivity, etc. In practice, some surfaces may either be specular (mirror-like) reflectors or be hard to detect due to low reflectance and some parts of the environment may have frictional properties that make large-scale odometry difficult.

We deal with these issues in two ways, both based on a polygonal approximation to the environment and the detection of convex (reflex) vertices. The presence of reflex vertices is critical since it is these reflex vertices that determine the occlusion of regions of the environment with respect to one another. The two aspects of our approach are as follows. 1) We use a pair of robots observing one another to build a map and circumvent problems of object visibility. 2) We efficiently deposit markers while exploring to facilitate subsequent navigation. The exploration process is based on triangulation using an environment decomposition attached to reflex vertices. The marker deposition process depends on the deposition of these visual cues at positions associated with reflex vertices.

In practice, a non-polygonal environment can always be described using a polygonal approximation. Such an approximation can be readily computed so that it is either conservative in the sense that the interior of the approximated free space is assured to be free, or it can be designed to be accurate in a least-squared sense, so that for a given number of vertices in the approximation the discrepancy between the polygonal model and the actual environment is minimized [11, 8].

In the next subsection, we will briefly discuss relevant background research. In Section 2 we discuss multi-robot localization and exploration including, in Subsection 2.1, an example of a visual "tracker" that we have used to implement the class of algorithm described in the paper. In Section 4 we consider a strategy for placing beacons while explor-

[1]Appeared in "Vision Interface '98", pages 455-461, Vancouver, Canada, June. 1998.

ing, and discuss the complexity of using the optimal number of beacons, as opposed to a good but suboptimal arrangement.

1.1 Background

Several authors have examined the issue of exploring space with one of more robots [14, 13, 6, 20, 5, are illustrative]. In general, multi-robot exploration techniques have tended to focus on models with limited coordination or communication between the robots [1]. In contrast, we consider a tight coupling between the exploring robots in the interest of greater accuracy of more efficient behaviour. Related work deals with exploring spaces large enough that the robots cannot see one another across the environment [17]. In this work, we consider the case where the robots do not lose visual contact so long as their view of one another is not occluded.

The use of beacons or landmarks for robot positioning has been examined by several authors [19, 2, 12, for example]. The problem of placing a set of landmarks such that from any given place inside the polygon at least one of them would be visible is equivalent to placing a set of *"point guards"* to cover a hypothetical art gallery, as posed by Klee [15]. A key issue is then how many such landmarks are required to assure a given environment is navigable without losing sight of a landmark). It can be shown that as many as $n/3$ landmarks are needed in order to cover completely the interior of a polygon with n vertices. One example could be seen in Fig. 1 [3].

The maximum number of landmarks needed for a *simple polygon* with n vertices is $\lfloor n/3 \rfloor$. This can be proven as follows: Consider a triangulation of the polygon (see Fig. 2), such as the one that is returned as the map of our exploration algorithm. Then by using three colours mark every vertex in such a way that two vertices that share an edge of a triangle would have different colours. Finally choose the colour with the smallest number of vertices, and on every vertex with such colour place an ideal landmark. As every triangle would have one vertex of that colour then for any point in the interior of that triangle the landmark placed in the coloured vertex would be visible. As the triangles cover completely the free space, any point in the interior of the environment would be able to see at least one landmark [9].

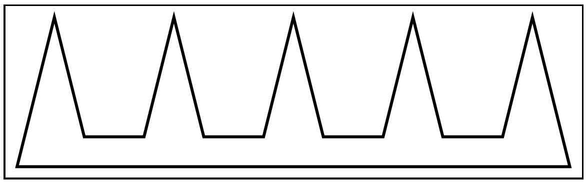


Figure 1: A simple environment that requires N/3 ideal landmarks.

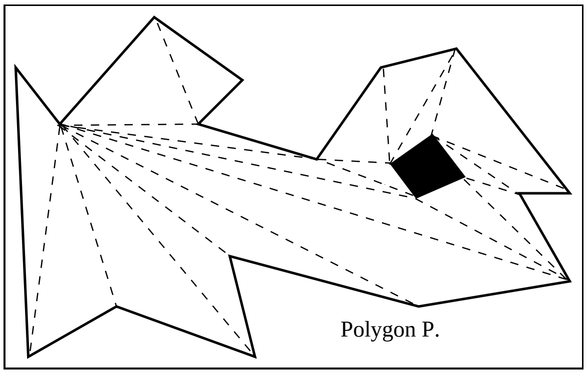


Figure 2: Triangulation of a simple polygon with holes.

2 Cooperative Localization

Since sensing is being used to correct position estimation errors, the sole source of error in the selective localization of the robots is the inaccuracy of the "robot tracker" sensor that is used to update/correct the position of the moving robot relative to the position of the stationary one. Therefore, if the two robots start with one stationary robot in an initial position P_{origin} then the moving robot could localize itself with respect to that position, (see Fig. 3). Note that, in practice, information from both sensing and odometry could be combined using a technique such as Kalman filtering.

2.1 Tracker implementation

There are many sensors that could be used for the robot tracker. Our preliminary implementation is based on visual observation of a geometric target on the robot [7]. (Alternative possible implementations use retroreflectors or laser light striping – our actual robot is equipped with alternative such technologies.) Each robot is equipped with a camera that allows it to observe its partner. The robots are both marked with a special pattern for pose estimation. The first part of the pattern is a series

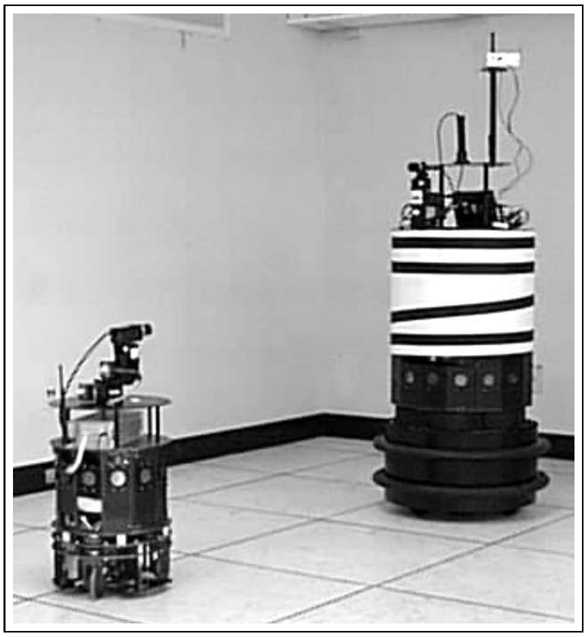


Figure 3: The visual robot tracker system (camera mounted on one robot, helix target pattern mounted on the second robot.

of horizontal circles (that project into almost linear pattern in the image) that allows the robot to be discriminated from background objects: the ratio of spacing between the circles is extremely unlikely to occur in the background by chance. Thus, the presence of the robot is established by a set of lines (curves) with the appropriate length-to-width ratio, and the appropriate inter-line ratios, as well as the correct position. The second component of the pattern is a helix that wraps once around the robot. The elevation of the center of the helix allows the relative orientation of the robot to be inferred (see Fig. 4). In practice, this allows the robot's pose to be inferred with an accuracy of a few centimeters and a 3 to 5 degrees.

3 Outline of the algorithm

In [17] we presented an algorithm for mapping the interior of an environment similar to an art gallery. The size of the area should be small enough to be covered by the range of the tracker sensor. Two mobile robots equipped with two different types of sensors, are used in close cooperation, to completely map the free space. Both robots use a traditional range finder in order to detect obstacles that are very close to them and subsequently to follow the object perimeter during the exploration. In addition each robot has a robot tracker sensor that provides the position and pose of the other robot if the line of visual contact is uninterrupted, or a signal that an obstacle exists between the two robots.

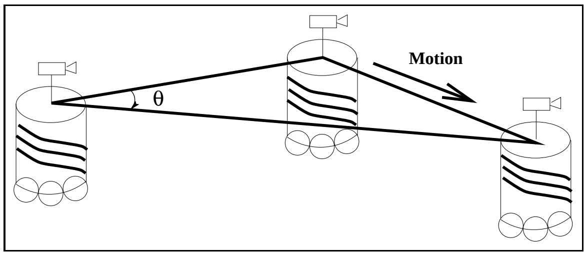


Figure 4: Schematic representation of the tracking of a moving robot and simultaneously mapping a triangle of free space.

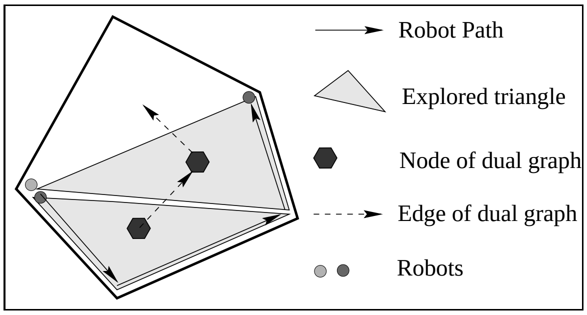


Figure 5: Two robots mapping a simple environment.

The exploration algorithm is based on the following idea. At any single time one robot is positioned at a vertex (corner) of the environment operating as an intelligent landmark, while the other robot moves across the perimeter of the environment maintaining visual contact with the stationary robot. More precisely, as the moving robot follows one wall of the environment it "sweeps" the line of visual contact across the triangle defined by, the corner the stationary robot is positioned and the two ends of the wall. Thus, the robot establishes the position of the wall and the occupancy of the swept free space inside the triangle. To achieve high precision on the mapping the moving robot employs the "just-in-time" sensing strategy developed earlier [5] where the robot uses a fast and inaccurate sonar sensor to follow the wall and employs a slow but accurate laser range finder[2] to map the end-

[2]For example the Quadris platform developed in our

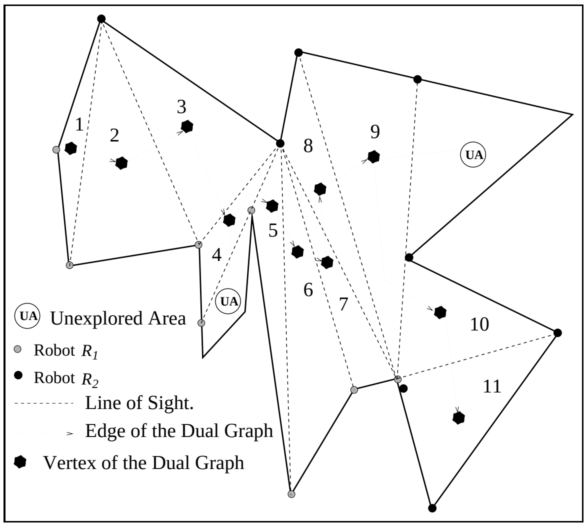


Figure 6: Two robots mapping a complicated room.

points/corners of the wall. Before accurately mapping the corners, the moving robot uses the stationary robot as a landmark in order to correct its position thus reducing the error in its position estimate from odometry error. In Fig. 5 the moving robot has moved along three of the five walls of the room, thus mapping two of the three resulting triangles. The two robots are using the spatial decomposition of a polygon into triangles and the resulting dual graph to ensure full coverage of the entire free space without any repetition of exploration.

The exploration of a complicated environment is demonstrated in Fig. 6. The triangles that tessellate free space and the dual graph are numbered in the order they were explored. The locations where a decision was made due to a reflex vertex of the polygon and an area was temporarily left unexplored are marked by UA. The two robots explore down a branch of the dual graph and then they would return to map the unexplored areas and finally return to the initial position.

3.1 Complexity analysis

The two robots use the dual graph of the triangulation as a guide to explore the entire free space. For simplicity we assume that the two robots return to the starting position after exploring the environment. The total path travelled would be the sum of the perimeter of the environment and the enclosed obstacles, plus the shortest path traversing the dual graph plus the maximum diagonal times the number of reflex vertices. In particular, as the triangulation covers the total area of the free space without any overlapping every triangle explored contributes to the total path travelled the exploration cost, which consists of the length of the perimeter of the environment, plus the shortest path travelled when the two robots return to the initial position. In addition each reflex vertex could contribute to the total path travelled up to the maximum diameter travel in repositioning. The order of exploration is given by a depth first traversal of the dual graph which guarantees that each node is traversed once the first time and a second time during the return. At each branch of the dual graph the decision is made as illustrated in Appendix A. The only situation that any of the robots has to travel inside the polygon during the exploration phase is when a reflex vertex interrupts the line of visual contact, and the robot has to make a decision which branch of the dual graph would follow.

3.2 Experimental results

Simulation experiments were performed to establish the improvement from the cooperative localization over odometry-only positioning during the exploration. The noise properties of the odometry error were modelled after experiments with the real robot, without considering systematic error. The noise model of the robot tracker was based on the visual robot tracker as described in Section 2. A series of one hundred experiments were performed first without cooperative localization and the two robots alternatively explored fifty triangles of free space. Then the visual robot tracker was used after the exploration of every triangle in order to update the position estimation of the mapping robot. As can be seen in Fig. 7, the improvement with cooperative localization is of a factor of 10.

4 Optimal landmark positioning

After the exploration and mapping of an unknown environment the mobile robots need to be able to navigate through the known part safely, and efficiently. The mere existence of, even an accurate, map does not guarantee safe travelling. The accumulation of odometry error could cause accidents while the possibility to mistake between different but similar areas is substantial. The most common solution is the use of landmarks (physical or artificial) for self localization. Most of the existing

lab [4] [16].

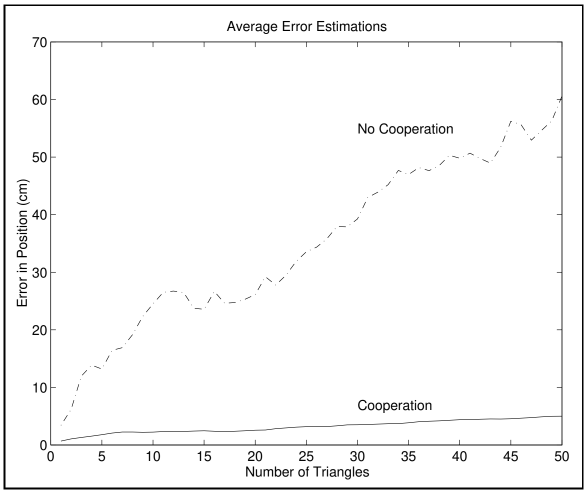


Figure 7: Average error in position for 50 triangle (over 100 experiments).

solutions deal with the localization process per se, while few proceed to suggest methods for the selection of the landmarks. The maximum number of landmarks for completely covering an environment has not been investigated up to now. Moreover, no algorithm has been proposed for the automatic selection of landmark positioning that does not require human interaction.

4.1 Ideal landmark

We now consider how many landmarks we need to place, as a function of the number of vertices n defining the environment boundary. Our constraint is that no matter where we are, we need to be able to see at least a single landmark. In order to provide bounds on the number of landmarks necessary for a complete coverage of the environment the environment is modelled as a simple polygon with holes. For the moment, we assume ideal landmarks which, if any one is visible, could provide adequate information to a mobile robot for accurate self-localization.

In the triangulation exploration algorithm it is clear that the places where the line of visual contact is interrupted are the reflex vertices. The placing of landmarks needs to take into consideration the position of reflex vertices. Unfortunately, there are polygons that would require as many landmarks as the number of reflex vertices [3] [15]. To sum up, the maximum number of landmarks needed for any environment modelled as a simple polygon with r reflex vertices and n total vertices, is $\min(\lfloor n/3 \rfloor, r)$

[3]when the number of reflex vertices is less or equal to $\lfloor n/3 \rfloor$, where n is the total number of vertices.

4.2 Optimal (minimal) number of landmarks required for complete localization

For any practical application the important issue is to find the minimum number of landmarks needed and where to place them. Unfortunately, the problem of *optimally* placing the landmarks is equivalent to the art gallery problem and is NP-hard [18] [15]. The problem becomes impossible when the mobile robot has to locate the optimal landmark positions and place the landmarks online during the exploration phase without backtracking. Moreover, in many cases two landmarks should be visible from every point in the environment in order for the localization to be robust, in such a case it is impossible to compute the positions. For the of-line version of the problem there is a competitive algorithm that guarantees at most $O(\log(n))$ times the minimum number of landmarks in $O(n^5 \log(n))$ time [10].

4.3 Online landmark positioning during exploration

The triangulation exploration algorithm can be extended to provide the locations for online landmark positioning. As the two robots explore the free space the non-stationary robot moves up until it encounters a reflex vertex that interrupts the line of visual contact. As the moving robot progress along an edge the equivalent triangle of free space is mapped. We define a polygon vertex as *completely mapped* if all the triangles of free space that share that vertex are mapped. After a triangle of free space is mapped, if all three vertices of the triangle are *completely mapped* then they are considered for landmark placement and the vertex of the triple that is shared by the largest number of triangles is selected for positioning a new landmark. As there are only $n-2$ triangles for a polygon with n vertices then the above algorithm would insert at most $\lfloor n/3 \rfloor$ landmarks which is the upper limit. The extra cost for the placement of landmarks is at most the number of landmarks times the diameter of the bounding circle.

5 Conclusions

In this paper, we have described an approach to exploring and navigating in *large scale spaces* where positioning and sensing might be difficult. In fact, such difficulties are likely to arise in many real-world environments.

Our approach is based on exploiting a line-of-sight constraint between two robots to achieve exploration with reduced odometric error. This approach can also cope with obstacles with hard-to-sense reflectance characteristics. In the second part of our paper we describe how these exploring robots might place visual beacons to facilitate subsequent navigation, again by exploiting a line-of-sight constraint between a robot and the beacons. While the use of an optimal number of landmarks in shown to be infeasible, we describe how an on-line algorithm can get by with a small number of beacons.

Open issues are how beacons might be placed inside to environment to provide bounded odometry error, by assuring that a beacon or a combination of beacons is always sufficiently close. Another issue for future consideration is how to achieve this type of result using beacons with different sensory characteristics. Moreover our algorithm would be extendible for positioning the landmarks in such a way that from any given point in free space at least two landmarks would be visible. In ongoing work, we are examining the experimental characteristics of this type of strategy.

6 Appendix A

A sketch of the exploration algorithm is presented next. Both robots run the same exploration algorithm, taking turns moving thus mapping the free space and being stationary thus providing a fixed localization reference for the moving robot. In the following we assume no three points are co-linear if they are, it would involve a minor but tedious change to the algorithm. There are four different cases where the line of visual contact is interrupted (Fig. 8a,b and Fig. 9a,b), in these cases the moving robot can not continue its previous course and it has to make a decision where to move next in order to maintain visual contact with the stationary robot. The environment is explored in regions of free space composed by neighboring triangles. The algorithm is summarized below.

```
While Unexplored Areas Do
 {
  Cover Nearest Unexplored Area
   {
    While No Occlusion Do
     Explore the next triangle of
     free space
    If Occlusion Then
     If Case 1 Then
      The two robots exchange roles.
     Else If Case 2 Then
      The Moving Robot goes to
      the Stationary Robot. Marking
      the reflex vertex as an opening to
      an Unexplored Area.
     Else If Case 3 Then
      The Moving Robot marks its position
      as a temporary vertex and moves
      towards the Stationary Robot untill
      it encounters the occluding
      Reflex Vertex. The line
      between the occluding vertex and
      the temporary vertex is an opening
      to the an Unexplored Area.
     Else If Case 4 Then
      The two robots exchange roles
      The new Moving Robot follows
      the occluding edge to the next
      corner, then the two robots exchange
      roles again.
    Continue The Exploration.
   }
  If No Triangle of free space Then
  Move to the clossest Unexplored
  Area.
 }
```

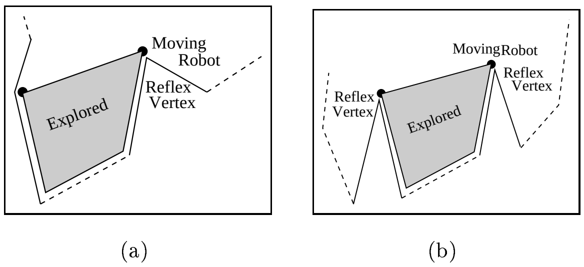
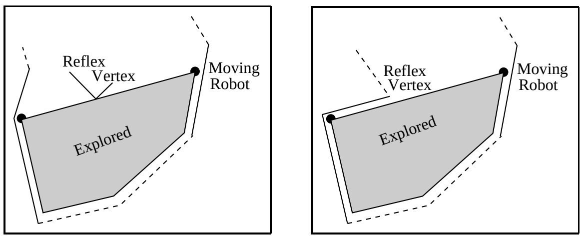


Figure 8: (a) **Case 1:** The stationary robot is at a non-reflex vertex and the moving robot encounters a reflex vertex that would interrupt the line of visual contact (b) **Case 2:** Both robots are placed at reflex vertex such that any further exploration would break the line of visual contact.

References

[1] Tucker Balch and Ronald C. Arkin. Communication in reactive multiagent robotic systems. *Autonomous Robots*, 1(1):27–52, 1994.



(a) (b)

Figure 9: (a) **Case 3:** Occluding Vertex between the two robots. (b) **Case 4:** Occluding Edge next to the stationary robot.

[2] M. Betke and L. Gurvits. Mobile robot localization using landmarks. *IEEE Trans. on Robotics and Automation*, 13(2):251–263, April 1997.

[3] V. Chvatal. A combinatorial theorem in plane geometry. *J. Combin. Theory Ser. B*, 18:39–41, 1975.

[4] Gregory Dudek, Paul Freedman, and Ioannis M. Rekleitis. Environment exploration using "just-in-time" sensor fusion. In *Vision Interface*, pages 175–182, Toronto, May 1996.

[5] Gregory Dudek, Paul Freedman, and Ioannis M. Rekleitis. Just-in-time sensing: efficiently combining sonar and laser range data for exploring unknown worlds. In *International Conference in Robotics and Automation*, volume 1, pages 667–671. IEEE, April 1996.

[6] Gregory Dudek, Michael Jenkin, Evangelos Milios, and David Wilkes. Robotic exploration as graph construction. *Transactions on Robotics and Automation*, 7(6):859–865, December 1991.

[7] Gregory Dudek, Michael Jenkin, Evangelos Milios, and David Wilkes. Experiments in sensing and communication for robot convoy navigation. In *Proc. IEEE International Conference on Intelligent Robots and Systems (IROS)*, vol. 2, pages 268–273, Aug. 1995.

[8] David Eu and Godfried Toussaint. On approximating polygonal curves in two and three dimensions. *CVGIP: Graphical Models and Image Processing*, 56(3):231–246, May 1994.

[9] S. Fisk. A short proof of Chvàtal's watchman theorem. *J. Combin. Theory Ser. B*, 24:374, 1978.

[10] S. K. Ghosh. Approximation algorithms for art gallery problems. In *Proc. Canadian Inform. Process. Soc. Congress*, 1987.

[11] H. Imai and Masao Iri. Polygonal approximations of a curve – formulations and algorithms. In G. T. Toussaint, editor, *Computational Morphology*, pages 71–86. Elsevier Science Publishers B. V., New York, N.Y., 1988.

[12] C. Lin and R. Tummala. Mobile robot navigation using artificial landmarks. *Journal of Robotic Systems*, 14(2):93–106, 1997.

[13] Vladimir J. Lumelsky, Snehasis Mukhopadhyay, and Kang Sun. Dynamic path planning in sensor-based terrain acquisition. *IEEE Transactions on Robotics and Automation*, 6(4):462–472, August 1990.

[14] B. Oommen, S. Iyegar, S. Nageswara, S. Rao, and R. Kashyap. Robot navigation in unknown terrains using learned visibility graphs, part i: The disjoint convex obstacle case. *IEEE J. of Robotics and Automation*, 3(6):672–681, 1987.

[15] J. O'Rourke. *Art Gallery Theorems and Algorithms.* Oxford University Press, New York, NY, 1987.

[16] Ioannis Rekleitis, Gregory Dudek, and Paul Freedman. Mobile robot exploration by using fused data from two sensors. In Proc. of *PRECARN/IRIS Conference* Vancouver, B.C., June 14-16, 1995.

[17] Ioannis Rekleitis, Gregory Dudek, and Evangelos Milios. Multi-robot exploration of an unknown environment, efficiently reducing the odometry error. In IJCAI, editor, *International Joint Conference in Artificial Intelligence*, volume 2, pages 1340–1345, Nagoya, Japan, August 1997. Morgan Kaufmann Publishers, Inc.

[18] T. C. Shermer. Recent results in art galleries. *Proc. IEEE*, 80(9):1384–1399, September 1992.

[19] K. Sugihara. Some location problems for robot navigation using a single camera. *Computer Vision, Graphics, and Image Processing*, 42:112–129, 1988.

[20] Jun Tani. Model-based learning for mobile robot navigation from the dynamical systems perspective. *IEEE Trans. System, Man and Cybernetics*, 26(6), 1996.