OAK RIDGE NATIONAL LABORATORY LIBRARY
3 4456 0003206 0

ORNL/TM-9782

ornl

OAK RIDGE NATIONAL LABORATORY

MARTIN MARIETTA

Robot Navigation Algorithms Using Learned Spatial Graphs

S. S. Iyengar
C. C. Jorgensen
S. V. N. Rao
C. R. Weisbin

OAK RIDGE NATIONAL LABORATORY
CENTRAL RESEARCH LIBRARY
CIRCULATION SECTION
4500N ROOM 175
LIBRARY LOAN COPY
DO NOT TRANSFER TO ANOTHER PERSON
If you wish someone else to see this report, send in name with report and the library will arrange a loan.

OPERATED BY
MARTIN MARIETTA ENERGY SYSTEMS, INC.
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road, Springfield, Virginia 22161
NTIS price codes—Printed Copy: A03; Microfiche A01

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.

ORNL/TM-9782

Engineering Physics and Mathematics Division

ROBOT NAVIGATION ALGORITHMS USING LEARNED SPATIAL GRAPHS*

S. S. Iyengar,+ C. C. Jorgensen, S. V. N. Rao,+ C. R. Weisbin
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831

*Submitted for Journal publication

+Louisiana State University

Date Published - December 1985

Research sponsored by
U.S. Dept. of Energy
Office of Basic Energy Sciences

Prepared by the
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831
operated by Martin Marietta Energy Systems, Inc.
for the U.S. DEPARTMENT OF ENERGY
under Contract No. DE-AC05-840R21400

OAK RIDGE NATIONAL LABORATORY LIBRARY
3 4456 0003206 0

TABLE OF CONTENTS

Abstract iii

Introduction 1

Optimization of Local Object Avoidance 3

Terrain Model 3

Path Planning and Learning 6

Illustrative Example 12

Conclusions 16

Future Directions 17

References 19

Acknowledgments 20

ABSTRACT

Finding optimal paths for robot navigation in known terrain has been studied for some time but, in many important situations, a robot would be required to navigate in completely new or partially explored terrain. We propose a method of robot navigation which requires no pre-learned model, makes maximal use of available information, records and synthesizes information from multiple journeys, and contains concepts of learning that allow for continuous transition from local to global path optimality. The model of the terrain consists of a spatial graph and a Voronoi diagram. Using acquired sensor data, polygonal boundaries containing perceived obstacles shrink to approximate the actual obstacles' surfaces, free space for transit is correspondingly enlarged, and additional nodes and edges are recorded based on path intersections and stop points. Navigation planning is gradually accelerated with experience since improved global map information minimizes the need for further sensor data acquisition. Our method currently assumes obstacle locations are unchanging, navigation can be successfully conducted using two-dimensional projections, and sensor information is precise.

Keywords and phrases: Learning, Spatial Graph Model, Robot Navigation, Local Optimization, Path Problems, Voronoi Diagram.

1. INTRODUCTION

Robotics has become an actively pursued research area of computer science and has proven to be replete with a variety of issues ranging from abstract mathematical to highly pragmatic problems. In many industrial applications which are repetitive and tedious (e.g., normal maintenance or inspection), it would be desirable to utilize mobile robots. Other tasks requiring rapid response in emergency situations are also appropriate for intelligent machines; this is particularly true in hazardous environments. Some of the more active robotics research areas today include knowledge representation, task planning, multi-sensor interpretation, dynamics and control, advanced computer architectures, algorithms for concurrent computation, and coordinated manipulation and navigation.

A robot may be characterized as an autonomous machine capable of decision making and action. To perform complex tasks which cannot be fully programmed *a priori*, effective sensing becomes crucial for monitoring both the robot's environment as well as the status of its own internal system. There have been several efforts to design an automated mobile robot. Examples are SHAKEY,[1,2] the JPL robot,[3] HILARE,[4,5] the Stanford Cart,[6] the CMU Terregator and Neptune robots,[7] Yamabico,[8] and HERMIES.[9]

Navigation planning is one of the vital aspects of any mobile robot. One approach toward navigation, called the find-path problem, addresses itself to determining a collision free path for a robot moving through a terrain cluttered with obstacles whose positions are known. This problem is well understood and solved in many cases.[10-18] The techniques for navigation described in these papers assume that a complete global model of our obstacle laden environment is known. Most of the techniques above model the obstacles and the free space of a robotic environment as mathematical and geometric entities. When a robot must navigate in an unexplored environment, the algorithms are not directly applicable.

Navigation in the more general case calls for the collision free movement of a mobile robot in entirely or partially unexplored terrain. The problem of planning optimal or near optimal paths that avoid collisions with obstacles in such an environment is a challenging task. Contrary to the known environment case, there has not been as much work reported in the literature about navigation problems in unexplored terrain. This can be attributed to the inherent ambiguity of the problem due to the lack of global information about the obstacles. Early attempts to navigate in unexplored terrain were based solely on image understanding.[3,6] More recently, Crowley[19] and Parodi[20] have suggested hierarchical approaches with global and local models updated based on sensor feedback. Chattergy[21] describes some novel heuristic strategies to aid the navigation of a robot in an unexplored terrain. This paper builds upon many of these ideas but specifically aims toward a method for which no pre-learned model is required, information from multiple journeys is explicitly synthesized, all information is used to the maximum extent, and a global path optimization is achieved in a continuous transition from local path optimization as more information is acquired.

In this paper, we assume that the robot (HERMIES)[9] begins his task in a completely unexplored terrain of finite dimensions. HERMIES has to complete a number of different traversals (e.g., carrying objects from place to place) and the goal of this paper is to provide the method by which he can navigate more efficiently with each successive trip, based upon experience acquired to date. The terrain can be randomly populated with obstacles, but the world is assumed to be static. The robot, assumed to be a point in a two-dimensional plane, can recognize line-of-sight distances to objects and detect their edges without imprecision.

ORNL-DWG 85C-14006

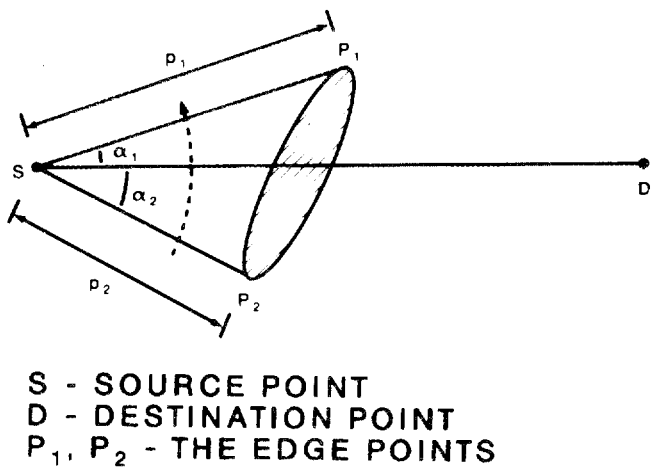
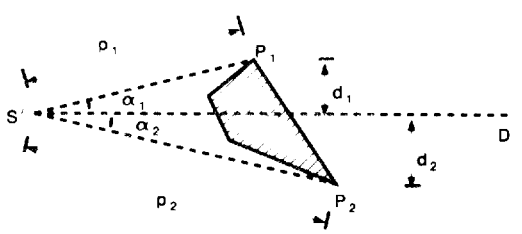
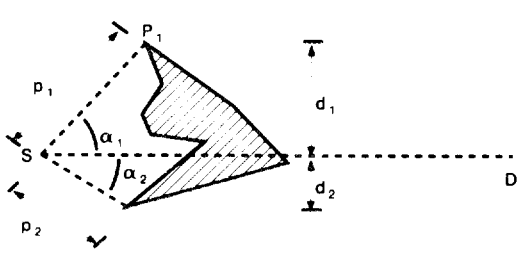


Fig. 1. The sensor readings include α_1, α_2, P_1 and P_2.

ORNL-DWG 85C-14005



(a) CONVEX POLYGON



(b) NON-CONVEX POLYGON

OPTIMALITY CRITERION $\longrightarrow$ $\min(d_1, d_2)$

$\longrightarrow$ $\min(p_1 \sin\alpha_1, p_2 \sin\alpha_2)$

Fig. 2. First case for local optimization.

The terrain is modelled using an attribute graph, called a *spatial graph*, and a *Voronoi diagram*. Initially, both are empty, and they are updated as more and more paths are traversed. Each path of navigation is composed of a sequence of stop points, where the robot stops to take sensor readings, or to access the terrain model to compute the next stop point. The robot travels in straight lines in between two successive stop points. Initially, obstacle avoidance techniques use local optimization for the navigation of the robot. By local optimization we mean optimal path selection based only on sensor information at the time of the decision.

Traversal of paths includes sensor exploration of the regions in which the robot navigates. Information gained while on new paths is consolidated into the existing graph structures. In planning any path, the content of the current graphs is made use of to the maximum possible extent, and local optimization occurs in the regions where no model is available. Initially, since no graph is available, the paths are only locally optimal. As more and more paths are traversed, the graphs become more complete ('learned') and gradually improve from local optimality to global optimality.

2. OPTIMIZATION OF LOCAL OBJECT AVOIDANCE

When a robot navigates in new terrain with no *a priori* information, its path of navigation is completely decided by the sensor readings and presumed goal destination. The localized nature of the sensors makes a true globally optimal path determination impossible in a terrain with arbitrary distribution of obstacles. Thus, a local optimization scheme must be used to determine the path of navigation in the immediate proximity of obstacles.

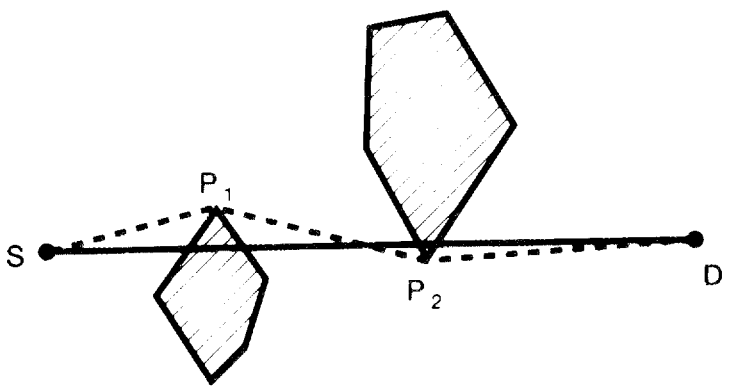
We consider the obstacle that is nearest to the source point S in the direction of the robot's goal destination. The sensor readings obtained allow for determination of the distance from the source to the edges of the obstacle, and also the corresponding edge angles relative to the line between the robot center and the goal. In Fig. 1, the angles α_1 and α_2 and the distances p_1 and p_2 are obtained from sensor readings. Our local optimization approach considers two cases. Figure 2 depicts the first case for which no part of the obstacle extends beyond the source point in the direction opposite to the direction of the next robot destination. The local optimization criterion is to minimize the distance traversed in the direction perpendicular to the line joining the source point S to the destination point D. That is, the locally optimal path is given by the condition min (d_1,d_2) or min $(p_1 \sin \alpha_1, p_2 \sin \alpha_2)$. This method may not yield a globally optimal path as shown in Fig. 3. The path SP_1P_2D will be followed according to the local optimality criterion, but the path SP_3D will be globally optimal. The second case of local optimality involves the obstacles that extend beyond the source in the direction opposite to the direction of the destination point as shown in Fig. 4. In this case the distance traversed (f_1,f_2) in the direction opposite to that of destination point D also has to be minimized. Referring to Fig. 4, the criterion for local optimization is given by min $(\sqrt{d_1^2 + f_1^2}, \sqrt{d_2^2 + f_2^2})$. Again, it is to be noted that this method may not give rise to globally optimal solution.

3. TERRAIN MODEL

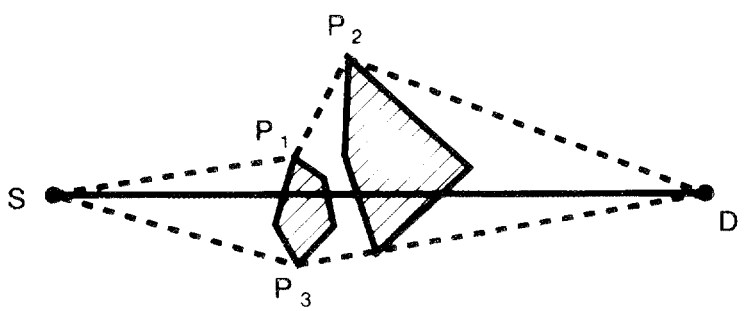
Figure 5 shows an illustrative rectangular terrain populated with four obstacles. Four paths are traversed using local optimization. The paths start at S_1, S_2, S_3, and S_4 and end at D_1, D_2, D_3, and D_4, respectively. The terrain in which the robot navigates is represented by both a *spatial graph* and a *Voronoi diagram* (Figs. 6 and 7, respectively).

LOCAL OPTIMIZATION ALGORITHM

```
ALGORITHM NAVIGATE-LOCAL (S, D);
   S THE SOURCE POINT.  D IS THE DESTINATION POINT
BEGIN
1.   IF D IS DIRECTLY REACHABLE
2.   THEN GO STRAIGHT
3.   ELSE
     BEGIN
4.     SCAN THE TERRAIN AROUND THE DIRECTION OF SD̄;
5.     P* ← OPTIMUM (P1, P2);
6.     GO STRAIGHT TO P*
7.     IF P* ≠ D
8.     THEN  NAVIGATE-LOCAL (P*, D);
     END;
END;
```



(a) BOTH LOCALLY AND GLOBALLY OPTIMAL



(b) ONLY LOCALLY OPTIMAL

Fig. 3. Local optimality does not mean global optimality.

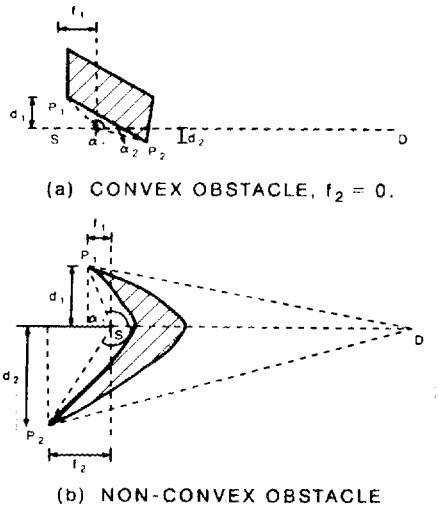
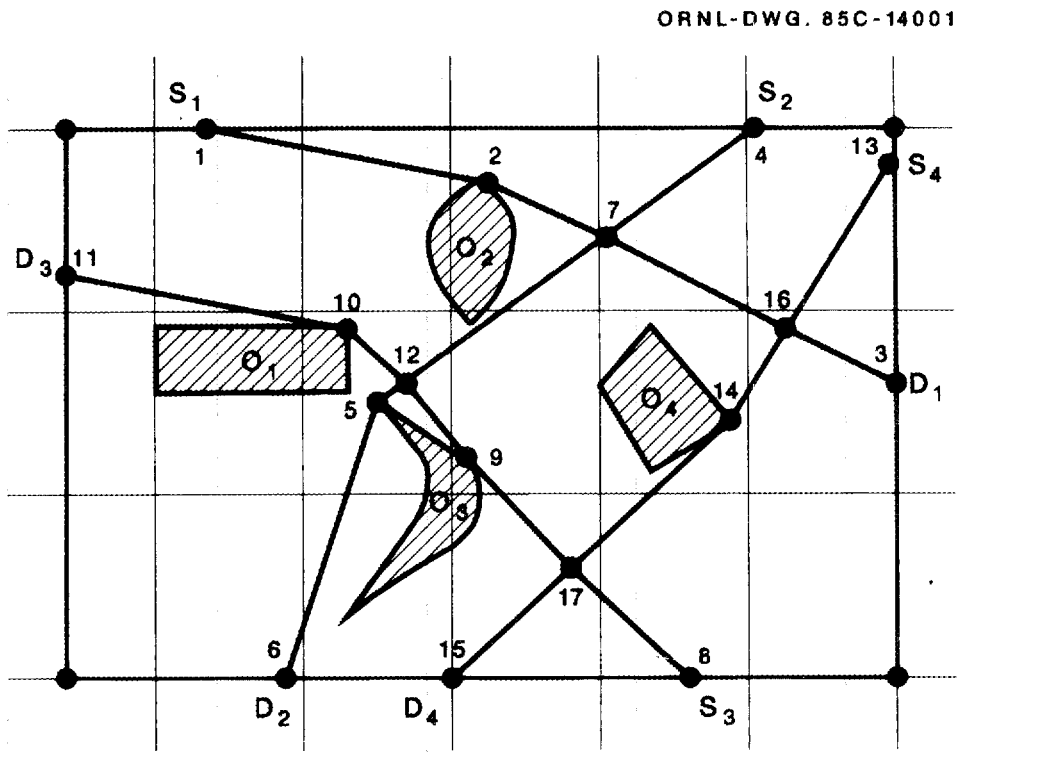


Fig. 4. Second case for local optimization.



FOUR PATHS FROM S_1, S_2, S_3 AND S_4 TO D_1, D_2, D_3 AND D_4 RESPECTIVELY USING LOCAL OPTIMIZATION. ALL PATHS ARE CONSOLIDATED BY FINDING INTERSECTION POINTS SUCH AS 7, 16, 12, AND 17

Fig. 5. The terrain.

A *spatial graph* G is defined as the ordered triple (V,E,ψ), where V is the set of nodes, E is the set of edges, and ψ is an *attribute mapping* that defines a pair of attributes (e.g., coordinate locations) for each vertex. For an edge $e = (v_i,v_j) \subset E$, we say that v_i and v_j are *connected* to each other. We also have a *distance* d(e) defined for each edge $e = (v_i,v_j) \subset E$, $\psi(v_i) = (i_1,i_2)$, $\psi(v_j) = (j_1,j_2)$, as

$$d(e) = \left[(i_1 - j_1)^2 + (i_2 - j_2)^2 \right]^{1/2} . \tag{1}$$

Initially a uniform grid is superimposed on the terrain of navigation. The granularity or grid size is chosen to be smaller than the expected size of the smallest obstacle of interest. The grid cells are numbered in the usual manner using x and y coordinate systems. Any path of navigation on the grid consists of straight lines and stop points. Each stop point corresponds to a node of the spatial graph, and each path joining two adjacent stop points corresponds to an edge. The pair of attributes of a node corresponds to the coordinates of the cell in which the node lies. The distance of an edge, $e = (v_i,v_j) \subset E$, is the euclidian distance between the nodes v_i and v_j. Figure 6 illustrates the spatial graph corresponding to the terrain and local optimization path planning of Fig. 5.

We next obtain a *Voronoi diagram* for the set of vertices, V, of the spatial graph given a set S of n points of $\{p_1,p_2,...,p_n\}$. The *Voronoi diagram* of S, Vor(S), partitions the plane into n equivalence classes, each of which corresponds to a point. Specifically, the equivalence class corresponding to point p_i is the *Voronoi polygon* $VP(p_i)$, defined[22] such that any point x in $VP(p_i)$ is closer to p_i than to any other point in S. Figure 7 illustrates the *Voronoi diagram* corresponding to the spatial graph of Fig. 6.

Initially, when HERMIES is first placed in a new terrain, the spatial graph is empty or null and the Voronoi diagram contains no points. The new paths are integrated into the terrain models when they are traversed. The spatial graph is updated for every new path as follows: (i) create new nodes corresponding to new stop points, (ii) create new edges corresponding to the paths in between two adjacent stop points, (iii) create new intersection nodes corresponding to the intersection points of new edges with the existing edges. When this process is complete, the Voronoi diagram is updated accordingly.

4. PATH PLANNING AND LEARNING

In this section we develop an algorithm that plans safe paths to navigate from a new arbitrary source point to a new arbitrary destination point. At each stop point on the path, either sensor readings are taken or graph computation is performed based on the existing terrain models to compute the next stop point. The terrain model is appropriately updated at each stop point.

Consider the navigation of the robot from the source point S to the destination point D. We compute *virtual source* S′ and *virtual destination* point D′, such that $S \epsilon VP(S')$ and $D \epsilon VP(D')$. In other words, S′ and D′ are the nodes of the spatial graph that are nearest to S and D, respectively. The paths from S to S′ and D′ to D are traversed according to the local optimization described in Section 2. The path S′D′ is planned using the spatial graph model and sensor readings, as will be described below.

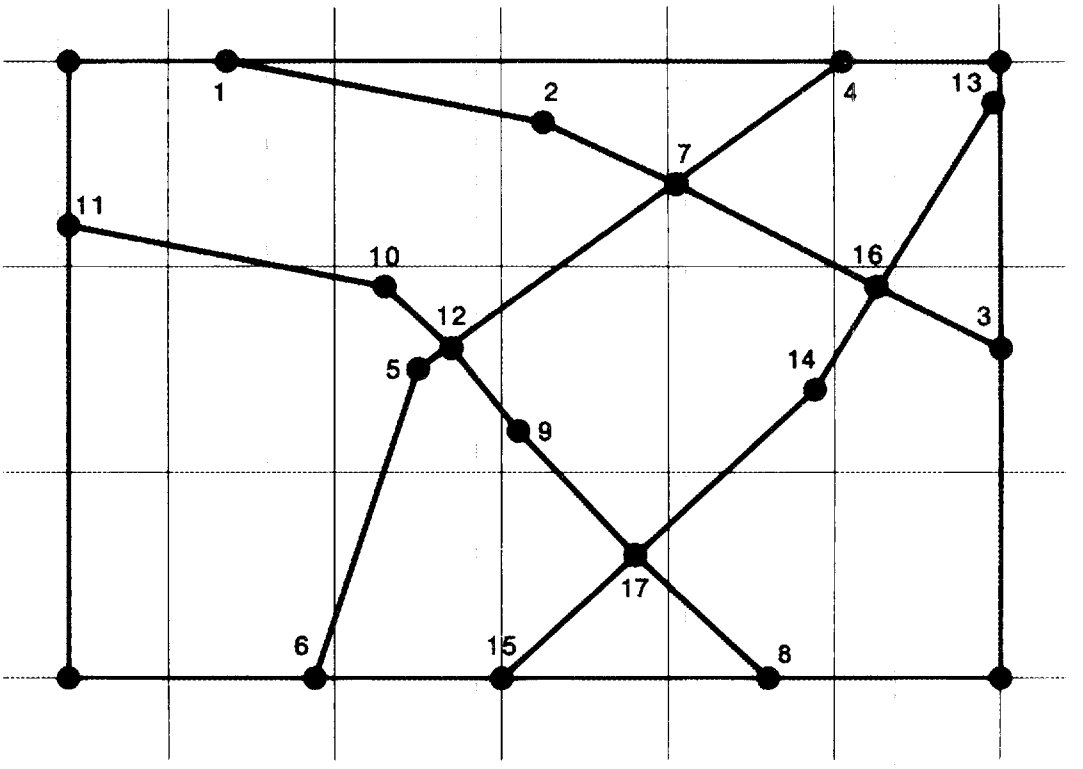


Fig. 6. The spatial graph.

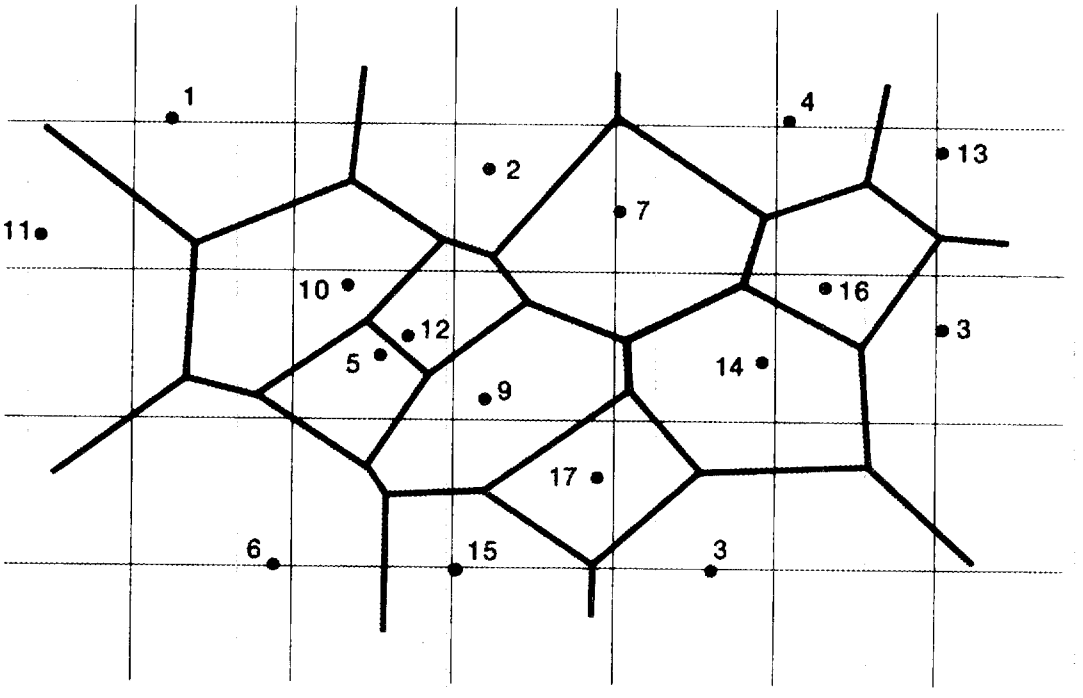


Fig. 7. The Voronoi diagram.

The paths from S to S′ and D′ to D can be navigated directly or constructed using the minimal distance to the spatial graph and following the graph to reach S′ (and D′) from the intersection point. The latter approach involves the creation of new nodes for the stop points and the appropriate edges. Also, the *Voronoi diagram* should be updated by creating new Voronoi regions for the new nodes. But the process of finding the virtual points should be carried out only after the graph is reasonably complete. That is to say, initially, until a considerable number of nodes are inserted into the spatial graph, all the navigation should be determined using sensor based algorithms. The basic algorithm is as follows:

COMPLETE NAVIGATION ALGORITHM

ALGORITHM NAVIGATE (S,D);
S IS THE SOURCE POINT, D IS THE DESTINATION POINT.
BEGIN
1. FIND S′ AND D′ SUCH THAT SϵVP(S′),
   AND DϵVP(D′);
2. NAVIGATE-LOCAL (S,S′);
3. NAVIGATE-GLOBAL (S′,D′);
4. NAVIGATE-LOCAL (D′,D);
   END.

The algorithm **NAVIGATE** (S′,D′) plans the path S′D′. This algorithm tests the polygon P, in which the source end of S′D′ lies. A polygon is said to be an *obstacle polygon* with respect to S′ if the obstacle or obstacles contained in P entirely fill the sensor range from S′, as shown in Fig. 8.

A polygon is a free polygon if it does not contain any obstacles. If the polygon P is unexplored with respect to S′, then the algorithm **EXPLORE (P,S′)** is involved. Sensor readings from S′ distinguish two types of regions — visible and invisible — as shown in Fig. 9. The *invisible* regions are the regions of the polygon that are not reachable by the sensor when the obstacles contained in the region are absent. The regions that are not invisible are called the *visible regions*. Based on the sensor readings, the polygon P can be partitioned into regions as shown in Fig. 10.

A region could be an unexplored polygon, a free-polygon, or an obstacle-polygon. The invisible regions are declared as unexplored with respect to the vertices on the line that limits the range of sensor from S′. The visible region is partitioned into obstacle polygons and free polygons. In Fig. 10 the region R_1 is unexplored with respect to the vertices P_1 and P_2. The regions R_2 and R_4 are free polygons, and the region R_3 is an obstacle polygon with respect to S′. It is to be noted that, in general, a polygon can be an obstacle polygon with respect to the other vertices. But, a polygon is a free polygon with respect to all the vertices of the polygon.

ORNL-DWG 85C-13992

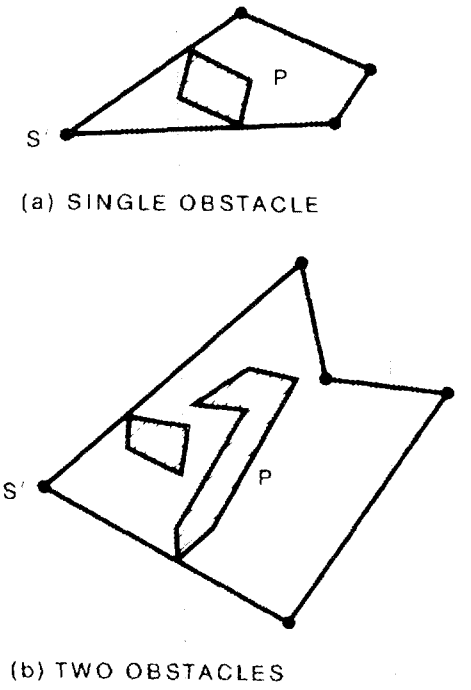
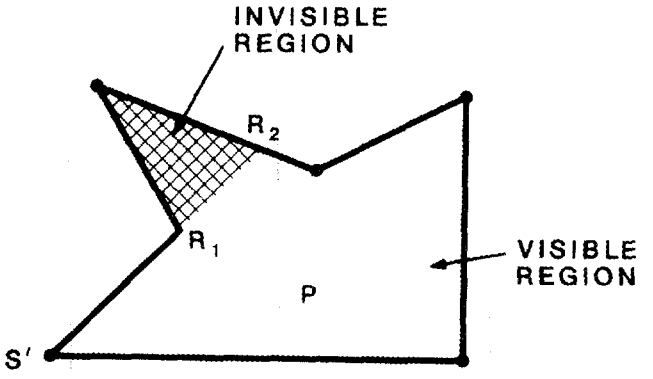


Fig. 8. Polygon P is an obstacle Polygon with respect to S′.

ORNL-DWG 85C-14000



VISIBLE REGION: THE REGION REACHABLE BY A SWEEPING SENSOR, WHEN ALL OBSTACLES ARE REMOVED

INVISIBLE REGION: THE REGION NOT REACHABLE BY SWEEPING SENSOR, WHEN ALL OBSTACLES ARE REMOVED

Fig. 9. Visible and invisible regions with respect to S′.

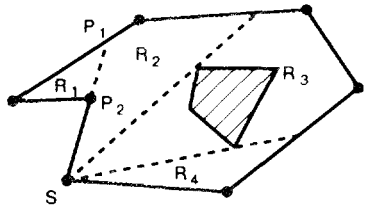


Fig. 10. Partitioning a Polygon based on sensor readings.

The algorithm **CONSOLIDATE** checks for any adjacent free regions from a convex region. If they form a convex region then it combines them and forms a single free polygon. The consolidation algorithm is described as follows:

CONSOLIDATION ALGORITHM

```
ALGORITHM CONSOLIDATE (P, S);
  P IS AN EXPLORED POLYGON WITH RESPECT TO VERTEX S
BEGIN
1.  FOR EACH FREE POLYGON P1, BELONGING TO THE
    PARTITION OF P        DO
    BEGIN
2.    FIND ALL ADJACENT FREE POLYGONS OF P1;
3.    FIND THE MAXIMAL SUBSET OF THEM THAT FORMS A
      CONVEX POLYGON AND COMBINE THEM INTO A SINGLE
      POLYGON;
    END;
  END;
```

The complete navigation algorithm for S′D′ is described in the Pascal-like syntax. The overall effect of this navigation algorithm is summarized as follows:

1. In general, all free polygons are convex and these polygon increase in size as learning proceeds.

2. Initially, all the obstacles are bounded by larger polygons, and as learning proceeds the bounding polygons are reduced in size to enclose the obstacles more closely.

3. If the path of navigation runs through all free polygons, then the complete path from S′ to D′ can be directly computed.

4. If the path contains unexplored polygons, then the robot halts at the appropriate stop point to explore the regions, and then the next stop point is computed only after the information about the currently explored region is incorporated into the terrain model.

5. Learning is incorporated along with path planning.

6. The paths are locally optimal initially, and they gradually become globally optimal as learning proceeds.

```
ALGORITHM NAVIGATE-GLOBAL (S', D');
  S' AND D' ARE THE SOURCE AND DESTINATION POINTS, RESPECTIVELY,
  ON THE SPATIAL GRAPH
  S'D' STANDS FOR THE STRAIGHT LINE JOINING S' AND D'
BEGIN
  FIND THE POLYGON P THAT CONTAINS SOURCE END OF S'D';
  IF (P IS AN OBSTACLE POLYGON)
  THEN
  BEGIN
    FIND THE NEAREST INTERSECTION POINT s OF S'D' AND P;
    FIND S*, SUCH THAT s ∈ VP(S*);
    MOVE TO S* ALONG EDGES OF P;
    NAVIGATE-GLOBAL (S*, D');
  END
  ELSE IF (P IS A FREE POLYGON)
  THEN
  BEGIN
    FIND THE INTERSECTION POINT s OF S'D' AND P;
    GO DIRECTLY TO s;
    NAVIGATE-GLOBAL (s, D');
  END
  ELSE IF (P IS UNEXPLORED WITH RESPECT TO S')
  THEN
  BEGIN
    EXPLORE (P, S');
    CONSOLIDATE (P, S');
    NAVIGATE (S', D');
  END;
END;
```

In the above algorithm we assumed the robot to be a point. However, the same can be applied to any finite sized robot by allowing suitable leeway in computing α_1, α_2, P_1 and P_2 from the sensor readings as shown in Fig. 11. However, a more generalized problem would be to consider the exact shape of the robot and plan the motion that involves both translation and rotation. Other natural extensions of the problem include the use of more than one sensor, and also taking into account the errors in distance measurement.

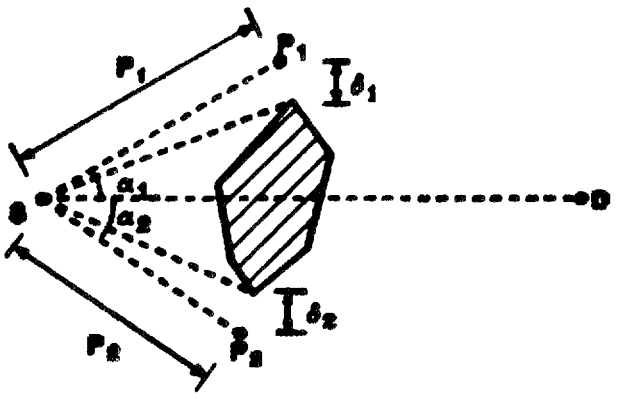


Fig. 11. Modification for a finite-sized robot: δ_1 and δ_2 account for the finite robot dimensions.

5. ILLUSTRATIVE EXAMPLE

In this section we illustrate our technique by tracing the algorithm of the previous section using a sample terrain. Figure 12 shows an unexplored terrain that contains four obstacles O_1, O_2, O_3, and O_4. Initially, four paths are traversed using local optimization from the source points S_1, S_2, S_3, and S_4 to D_1, D_2, D_3 and D_4, respectively. These paths are shown in Fig. 5, and the corresponding spatial graph and Voronoi diagram are shown in Fig. 6 and Fig. 7, respectively. Now consider applying the method of this paper to determining a path from S_5 to D_5. First, the virtual-source S' and virtual-destination D', are found as the nearest graph vertices corresponding to S_5 and D_5, respectively, as in Fig. 13.

The path from S_5 to S'_5 is traversed according to the local optimization method. The polygon P_2 contains the source end of the line $S'_5D'_5$. The polygon P_2 is unexplored and hence algorithm EXPLORE (P_2, S'_5) is invoked. The region P_2 is scanned using the sensor, and the polygon P_2 is partitioned into the regions P_2^1, P_2^2 and P_2^3 as in Fig. 14.

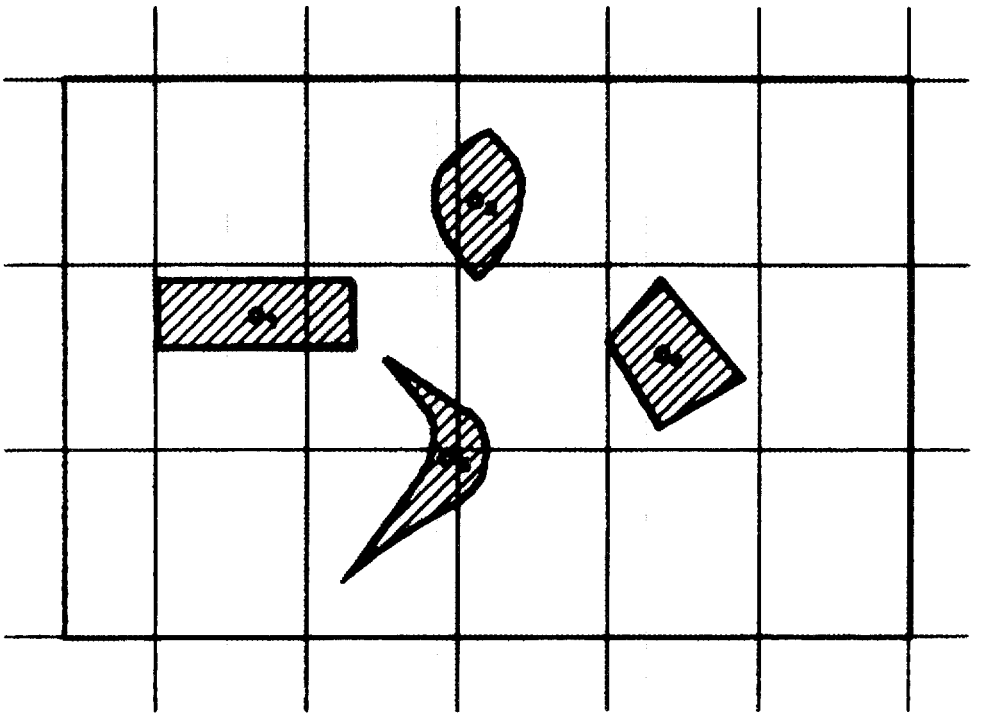
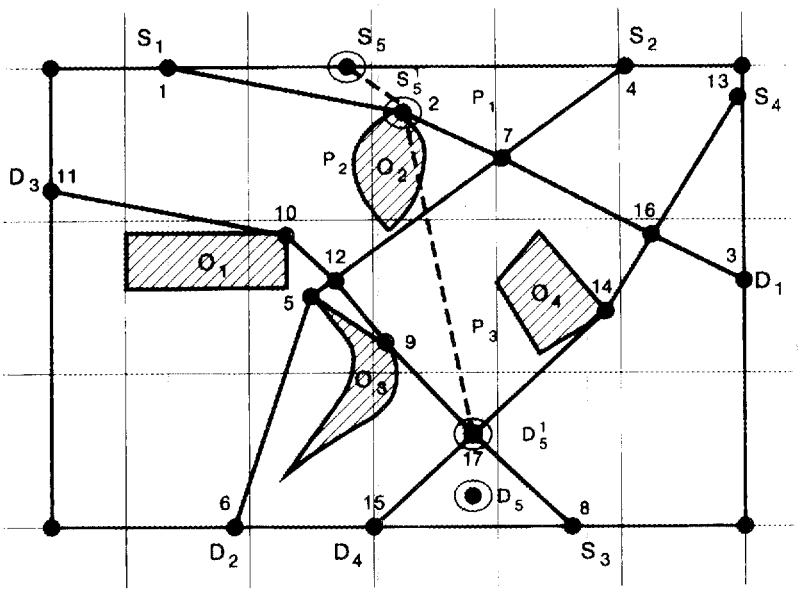
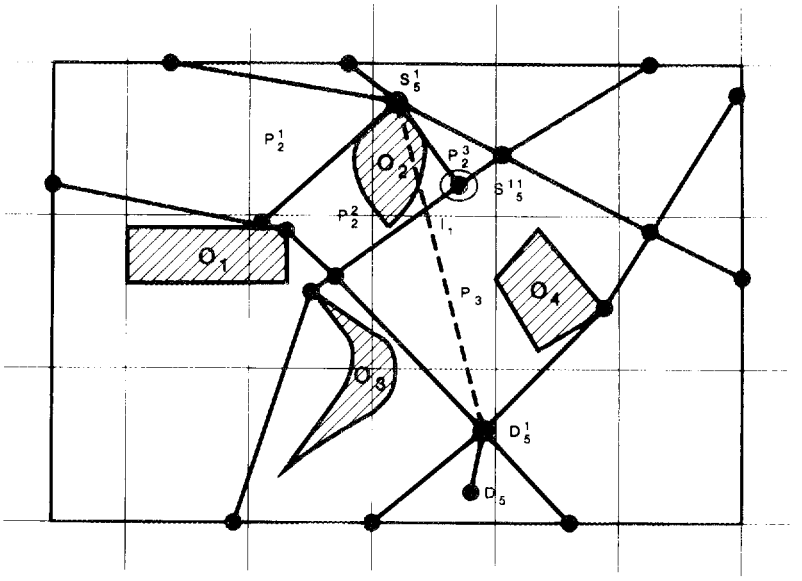


Fig. 12. Unexplored terrain.



CONSIDER NAVIGATION FROM S_5 TO D_5
S_5^1 IS THE VIRTUAL SOURCE $S_5 \varepsilon$ VPC (S_5^1)
D_5^1 IS THE VIRTUAL DESTINATION $D_5 \varepsilon$ VPC (D_5^1)
PATH FROM S_5 TO S_5^1 IS ACCORDING TO LOCAL OPTIMIZATION

Fig. 13. S_5 source point, D_5 destination point.



SOURCE END OF S_5^1 D_5^1 LIES IN POLYGON P_2
POLYGON P_2 IS EXPLORED
P_2 IS PARTITIONED INTO POLYGONS P_2^1, P_2^2, P_2^3
P_2^1, P_2^3 - FREE POLYGONS
P_2^2 - IS AN OBSTACLE POLYGON WITH RESPECT TO S_5^1
THE POLYGON P_2^2 IS PROCESSED, SINCE SOURCE END OF S_5^1 D_5^1 LIES IN P_2^2
THE INTERSECTION POINT I_1 IS COMPUTED, AND S_5^{11} IS
FOUND, SUCH THAT $I_1 \varepsilon$ VPC(S_5^{11})
PATH S_5^1 TO S_5^{11} IS TRAVERSED ALONG THE MINIMAL LENGTH
PATH ALONG THE EDGES OF P_2^2

Fig. 14. Exploration of Polygon P_2.

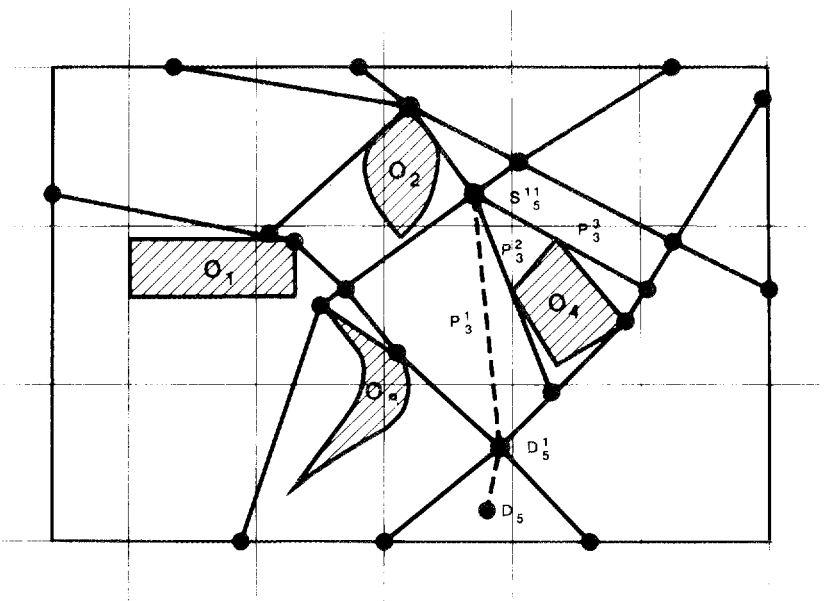


Fig. 15. Exploration of Polygon P_3.

The regions P_2^1 and P_2^3 are free-polygons, and the region P_2^2 is an obstacle-polygon with respect to the vertex S'. At this point, the source end of $S'_5D'_5$ is contained in the polygon P_2^2. The intersection point I_1 of $S'_5D'_5$ with the farther edge of P_2^2 is computed, and its nearest vertex S''_5 of the spatial graph is found. Then, the nearest path to S''_5 via the edges of the polygon P_2^2 is computed by finding the corresponding euclidian distance. The robot navigates along the edges of the polygon P_2^2 to reach S''_5. Next the path is planned from $S''_5D'_5$. The polygon P_3 contains the source end of $S''_5D'_5$ and is unexplored. Based on sensor readings, the polygon P_3 is partitioned into the regions P_3^1, P_3^2 and P_3^3. P_3^1 and P_3^3 are free polygons and P_3^2 is an obstacle polygon. At this stage, P_3^1 contains the source end of $S''_5D'_5$. The intersection point of $S''_5D'_5$ with P_3^1 is D'_5. The path of $S''_5D'_5$ is directly traversed as in Fig. 15. No update of the model is carried out since $S''_5D'_5$ is entirely contained in a free polygon. The navigation from D'_5D_5 is based on local optimization. The final spatial graph of the terrain is given in Fig. 16. Note that the obstacles O_2 and O_4 are bounded by smaller polygons than those shown in Fig. 5. Also, the polygons P_2^1, P_2^3, P_3^1 and P_3^3 are declared to be free polygons. Regions P_2^3 and P_3^3 are combined to form a single free polygon. Clearly, the information about the obstacles and free space of Fig. 16 is more consolidated and available for utilization than that of Fig. 5. Consider another navigation path from S_6 to D_6. The result of this traversal is shown in Fig. 17. Now the regions P_4, P_6 and P_9 are declared to be free polygons. The objects O_3 and O_4 are bounded by much smaller polygons than the ones in Fig. 16. Thus, the example illustrates the shrinking of the bounding polygons of the obstacles and widening of the free polygons as learning proceeds. Again, as more paths are traversed, more and more polygons are explored and the spatial graph becomes consolidated.

ORNL-DWG. 85C-13994

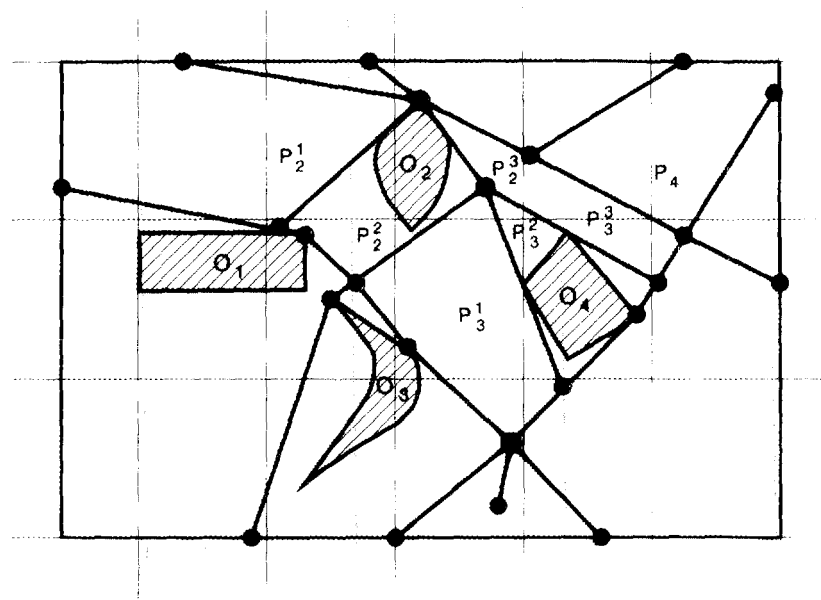
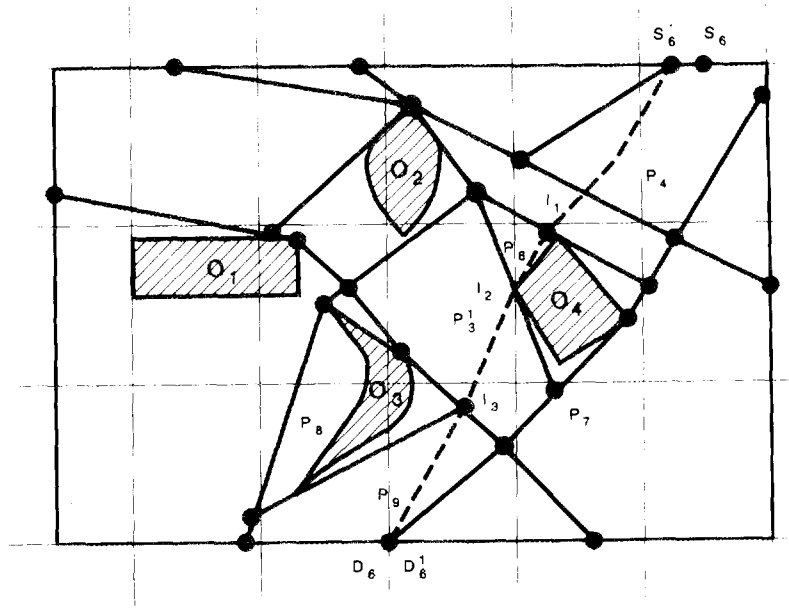


Fig. 16. Terrain model after the path from S_5 to D_5 is consolidated.

ORNL-DWG. 85C-13993



TRAVERSAL OF YET ANOTHER PATH FROM S_6 TO D_6

1. THE OBSTACLES O_4 AND O_3 ARE BOUNDED BY SMALLER POLYGONS
2. POLYGONS P_4, P_6, P_9 ARE DECLARED FREE POLYGONS
3. POLYGONS P_7 AND P_8 ARE DECLARED OBSTACLE POLYGONS WITH RESPECT TO I_1 AND I_3 RESPECTIVELY
4. PATH IS GLOBALLY OPTIMAL FROM S_6^1 TO D_6^1

Fig. 17. Terrain model after the path from S_6 to D_6 is consolidated.

6. CONCLUSIONS

In this paper, we describe a method that enables a mobile robot to navigate in an unexplored terrain and learn more about the terrain as it navigates paths. Our method requires no pre-learned model, makes maximal use of available information, records and synthesizes information from multiple journeys, and contains concepts of learning that allow for continuous transition from local to global optimality. The model of the terrain consists of a spatial graph and a Voronoi diagram. As more information is consolidated into the terrain model, the bounding polygons of the obstacles fit more closely and the polygons representing free space grow larger. In this way, the robot learns and applies the results of dynamically acquired sensor information to improve performance and relax navigational ambiguity on a continual basis up to the point where the environment is fully described; i.e., all obstacle-polygons are tightly bounded.

This paper has introduced the concept of learning in the domain of robot navigation and movement, namely path traversal and planning through a two-dimensional Cartesian environment. The utilization of concepts of spatial graphs has much broader implications however. For example, the rates at which sensor data updates are applied to the spatial graph directly affect the potential of the robot to navigate in a changing environment. Voronoi regions under a learning navigation paradigm can expand or shrink as a result of changing environmental conditions. The present spatial graph reflects only decisions arrived at from analysis of sensor data, but the method also permits the fusion of multiple sensor sources such as simultaneous use of line-of-sight (visual) and sonar to compose a simple graph space.

Similarly, there is no reason to confine the dimensionality of the graph to an N of two. For example, by extending the two-dimensional polygons to three-dimensional volumes, traversal in three dimensions and the learning of three-dimensional spaces become possible. A typical extension could be three-dimensional path planning of a robot end effector during grasping behavior scenarios. Further, the spatial graph nodes do not have to represent a single value. They can, for example, be pointers to complex data structures which contain a variety of relational data about a robot environment. In this way during path planning a spatial graph can serve as a context sensitive procedure for data base search by limiting the potential set of world data to local Voronoi regions and their associated data sets. In this manner, decisionmaking can be aided through "context focusing" which makes use of the spatial localization of the robot. Details of data structure and complexity analysis of the proposed algorithms are covered in a different paper.

7. FUTURE DIRECTIONS

Research is currently underway to extend the completeness of learning concepts to HERMIES navigation. In reality, true learning involves the utilization of more extensive sets of information such as those contained in complex data structures. Typical data include time tags, inter-object relations, tentative object classifications or labels. At present, we are extending learning to demonstrate performance on the HERMIES-II robot at CESAR by incorporating consolidation, abstraction, and forgetting processes. The latter deserves some comment. Forgetting or selective removal of information becomes more important for dynamic navigation if environments change to prevent the accumulation of useless data such as graph locations of moving objects in the environment. We propose to explicitly consider "forgetting" of spatial graph information by attaching a reinforcement or extinguishing time-based value to polygons. Values are decremented (i.e, extinguished by a fixed amount) unless a polygon is reinforced (confirmed) by additional sensor contacts. Such additions represent a more complete implementation of learning mechanisms traditionally associated with human psychological research.

REFERENCES

1. P. Hart et al., "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Trans. Sys. Sci. Cyber*, **SSC-4(2)**, 1968, pp. 100-107.

2. N. Nilsson, "Mobile Automation: An Application of Artificial Intelligence Techniques," Proceedings of First International Joint Conference on Artificial Intelligence, May 1969, pp.509-520.

3. A. M. Thompson, "The Navigation System of the JPL Robot," Proceedings of the Fifth International Joint Conference on Artificial Intelligence, August 22-25, 1977, Cambridge, MA, pp. 749-757.

4. G. Giralt et al., "A Multilevel Planning and Navigation System for a Mobile Robot," Proceedings of the Sixth International Joint Conference on Artificial Intelligence, August 20-23, 1979, Tokyo, Japan, pp. 335-338.

5. G. Giralt et al., "An Integrated Navigation and Motion Control System for Autonomous Multisensory Mobile Robots," appearing in *Robotics Research*, edited by Michael Brady and Richard Paul, MIT Press, 1984, pp. 191-214.

6. H. P. Moravec, "Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover," Carnegie-Mellon Robotics Institute, Pittsburgh, PA, Technical Report CMU-RI-TR-3, September 1980.

7. R. Wallace et al., "First Results in Robot Road Following," Proceedings of the Ninth International Joint Conference on Artificial Intelligence, August 18-23, 1985, Los Angeles, CA, pp. 1089-1095.

8. Y. Kanamaya et al., " A Mobile Robot with Sonic Sensors and Its Understanding of a Simple World," Proceedings of Seventh International Joint Conference on Artificial Intelligence, 1981 (see also IECON '84, 1984, p. 303).

9. C. R. Weisbin, J. Barhen, G. de Saussure, W. R. Hamel, C. C. Jorgensen, J. L. Lucius, E. M. Oblow, and T. E. Swift, "Machine Intelligence for Robotics Applications," Proceedings of the 1985 Conference on Intelligent Systems and Machines," Oakland, MI, April 22-24, 1985.

10. R. A. Brooks, "Solving the Find-Path Problem by Good Representation of Free-Space," *IEEE Trans. Systems, Man and Cybernetics*, vd. **SMC-13**, No. 3, March/April 1983.

11. R. A. Brooks, "Planning Collision-Free Motions for Pick-and-Place Operations," *The International Journal of Robotics Research*, **Vol. 2**, No. 4, Winter 1983, pp. 19-44.

12. R. A. Brooks and T. Lozano-Perez, "A Subdivision Algorithm in Configuration Space for Find Path with Rotation," *IEEE Trans. Systems, Man and Cybernetics*, **Vol. SMC-15**, No. 2, March/April 185, pp. 224-233.

13. T. Lozano-Perez, "Automatic Planning of Manipulator Transfer Movements," *IEEE Trans. Systems, Man, and Cybernetics*, **Vol. SMC-11**, October 1981, pp. 681-698.

14. T. Lozano-Perez, "Spatial Planning: A Configuration Space Approach," *IEEE Trans. Computers*, **Vol. C-32**, February 1983, pp. 108-120.

15. T. Lozano-Perez, and M. A. Wesley, "An Algorithm for Planning Collision-Free Paths Among Polyhedral Obstacles," *Commun. ACM*, **Vol. 22**, No. 10, October 1979, pp. 560-570.

16. J. E. Hopcroft, J. T. Schwartz and M. Sharir, "On the Complexity of Motion Planning for Multiple Independent Objects,; PSPACE Hardness of the Warehouseman's Problem," *Robotics Research* 3 (4), pp. 76-88, (1984).

17. M. Sharir and E. Ariel-Sheffi, "On the Piano Movers' Problem: IV. Various Decomposable Two-Dimensional Motion Planning Problems," *Comm. Pure and Applied Mathematics*, **37**, pp. 479-493 (1984).

18. J. T. Schwartz and M. Sharir, "On the Piano Movers' Problem: V. The Case of a Rod Moving in Three-Dimensional Space Amidst Polyhedral Obstacles," *Comm. Pure and Applied Mathematics*, **37**, pp. 815-848 (1984).

19. J. Crowley, "Navigation for an Intelligent Mobile Robot," *IEEE Journal of Robotics and Automation*, **Vol. RA-1**, No. 1, March 1985, p. 31, ff.

20. A. Parodi, "Multi-Goal Real-time Global Path Planning for an Autonomous Land Vehicle Using a High-Speed Graph Search Processor," IEEE International Conference on Robotics and Automation, St. Louis, Missouri, 1985, pp. 161-167.

21. A. Chattergy, "Some Heuristics for the Navigation of a Robot," *The International Journal of Robotics Research,* **Vol. 4**, No. 1, Spring, 1985, pp. 59-66.

22. D. T. Lee and F. P. Preparta, "Computational Geometry — A Survey," *IEEE Transactions on Computers*, **Vol. C-33**, No. 12, December 1984, pp. 1072-1101.

ACKNOWLEDGMENTS

The authors gratefully acknowledge comments by the ORNL reviewers. Special thanks to G. deSaussure for improving the presentation of the paper. We sincerely appreciate the support of O. Manley of the Office of Basic Energy Sciences and the encouragement of Alex Zucker and F. C. Maienschein. This manuscript was expertly prepared by A. B. Weil, E. S. Howe, and C. Zeigler.

INTERNAL DISTRIBUTION

1. Abbott, L. S.
2. Alsmiller, R. G., Jr.
3. Anderson, T. D.
4. Babcock, S.
5. Barhen, J.
6. Barnett, D.
7. Bowman, K.
8. Burch, W.
9. Cacuci, D. G.
10. Clark, F.
11. de Saussure, G.
12. Eads, B. G.
13. Einstein, J. R.
14. Feldman, M.
15. Flanagan, D.
16. Flanagan, G. F.
17. Ford, W. E., III
18. Fulkerson, W.
19. Funderlic, R.
20. Furth, W. F.
21. Geist, G. A.
22. Gosslee, D.
23. Gray, L.
24. Haas, P. M.
25. Hall, M. C. G.
26. Hamel, W. R.
27. Heath, M.
28. Honea, R.
29–32. Jorgensen, C. C.
33. Killough, S.
34. Lucius, J. L.
35. Maerker, R. E.
36. Maienschien, F. C.
37. Martin, H.
38. Maskewitz, B. F.
39. Meacham, S.
40. Mitchell, T.
41. Mynatt, F. R.
42. Oblow, E. M.
43. Otaduy, P.
44. Peelle, R. W.
45. Perey, F. G.
46. Pin, F. G.
47. Postma, H.
48. Protopopescu, V.
49. Reed, J.
50. Richmond, C. R.
51. Ricks, E.
52. Roberto, J. B.
53. Rosenthal, M. W.
54. Santoro, R. T.
55. Solomon, A. D.
56. Swift, T.
57. Uppuluri, V. R.
58. Ward, R. C.
59. Wehe, W.
60–63. Weisbin, C. R.
64. Wilson, D. G.
65. Wiltshire, R. S.
66. Worley, B. A.
67. Wright, R. Q.
68. Wright, T.
69. Zucker, A.
70. P. W. Dickson, Jr. (Consultant)
71. G. H. Golub (Consultant)
72. D. Steiner (Consultant)
73–74. Central Research Library
75. ORNL Y-12 Technical Library Document Reference Section
76–77. Laboratory Records
78. ORNL Patent Office
79. Laboratory Records - RC

EXTERNAL DISTRIBUTION

80. Office of Assistant Manager for Energy Research & Development, DOE-ORO, Oak Ridge, TN 37830
81. David Abraham, Analytic Disciplines, Inc., 1370 Piccard Dr., Rockville, MD 20850
82. James Allen, Dept. of Computer Science, Univ. of Rochester, Rochester, NY 14627
83. S. Amarel, Rutgers University, New Brunswick, NJ
84. Prof. Michael Arbib, Computer & Information Science Dept., Univ. of Massachusetts, Amherst, MA 01003
85. Artificial Intelligence Report, 3600 W. Bayshore Rd., Palo Alto, CA 94303
86. Dr. Robert M. Balzer, Information Sciences Institute, 4676 Admiralty Way, Marina Del Rey, CA 90291
87. Dr. Antal K. Bejczy, Robotics and Teleoperator Group, Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Dr., Pasadena, CA 91109
88. Dr. Herbert Bernstein, Courant Institute, New York Univ., 251 Mercer St., New York, NY 10012
89. Griff Bilbro, CCSP, North Carolina State Univ., Box 7911, Raleigh, NC 27695
90. Prof. Thomas O. Binford, Artificial Intelligence Laboratory, Margaret Jackson Hall, Stanford Univ., Palo Alto, CA 94305
91. Ted Blank, Editor-in-Chief, Smart Machines, PO Box 459, Sharon MA 02067
92. W. W. Bledsoe, The University of Texas, Austin, TX 78712
93. Daniel G. Bobrow, Editor-in-Chief, Xerox Corp., Palo Alto Research Center, 3333 Coyote Hill Rd., Palo Alto, CA 94304
94. Dr. Wayne Book, Dept. of Mechnical Engineering, Georgia Institute of Technology, Atlanta, GA 30332
95. Dr. Judy Bostock, Office of Management & Budget, Executive Office Bldg., 17th & Pennsylvania Avenues, Washington, DC 20503
96. Ron Brachman, Vice Chairman, Laboratory for Artificial Intelligence Research, Fairchild Camera and Instrument Corp., MS 30-888, 4001 Miranda Ave., Palo Alto, CA 94304
97. Michael Brady, Artificial Intelligence Laboraory, Mass. Institute of Technology, 545 Technology Sq., Cambridge, MA 02139
98. Prof. Rodney Brooks, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, Cambridge, MA 02139
99. Phyllis Campbell, Dept. of the Navy, Naval Air Systems Command, AIR 5511J2, Jefferson Plaza 2, Room 1164, Washington, DC 20361
100. Richard Cantone, Automated Reasoning Corp., 290 W. 12th St., Suite 1-D, New York, NY 10014
101. Jaime Carbonell, Chairman, Dept. of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213
102. Dr. John F. Cassidy, General Electric, Corporate R&D, Bldg. 5 - Room 249, Control Technology Branch, Schenectady, NY 12345
103. Raymond GA Cate, Editor, Robotics Age, Strand Bldg., 174 Concord St., Peterborough, NH 03458

104. B. Chandrasekaran, Laboratory for Artificial Intelligence Research, Dept. of Computer & Information Science, 2036 Neil Ave. Mall, Columbus, OH 43210-0923
105. Raja Chatila, Laboratoire d'Automatique et d'Analyse des Systems du CNRS, 7 Avenue du Colonel Roche, 31 077 Toulouse, Cedex, France
106. Peter Cheeseman, National Aeronautics and Space Administration, Ames Research Center, Mail Stop 244-7, Moffett Field, CA 94035
107. Don Ciffone, National Aeronautics and Space Administration, Ames Research Center, MS 244-7, Moffett Field, CA 94035
108. J. Robin B. Cockett, Asst. Professor, Dept. of Computer Science, 8 Ayers Hall, The University of Tennessee, Knoxville, TN 37916
109. Dr. J. S. Coleman, Div. of Engineering, Mathematical, and Geosciences, Office of Basic Energy Sciences, ER-15, U.S. Dept. of Energy Washington, DC 20545
110. Prof. Jim Crowley, Carnegie-Mellon University, Robotics Institute, Pittsburgh, PA 15213
111. Dr. Randy Davis, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139
112. Chia P. Day, GMF Robotics, 5600 New King St., Troy, MI 48098
113. Dr. George G. Dodd, General Motors Research Laboratories, Computer Science Dept. - RANB-254, Warren, MI 48090-9057
114. Prof. Max Donath, Dept. of Mechanical Engineering, Univ. of Minnesota, Minneapolis, MN 55455
115. Richard Duda, Syntelligence Company, 1000 Hamlin Ct., Sunnyvale, CA 94088
116. Dr. Alberto Elfes, The Robotics Institute, Carnegie Mellon University, Schenley Park, Pittsburgh, PA 15213
117. Robert Engelmore, Editor-in-Chief, Teknowledge, Inc., 525 Univ. Ave., Palo Alto, CA 94301
118. Lt. Kristin A. Farry, Flight Dynamics Lab - Bldg. 146, AFWAL/FIGC/58683, Wright-Patterson Air Force Base, OH 45433
119. Michael R. Fehling, Advanced Information & Decision Systems, 201 San Antonio Circle, #286, Mountain View, CA 94040
120. Ken Forbis, Dept. of Computer Science, Univ. of Illinois, 1304 Springfield Ave., Urbana, IL 61801
121. David Fox, Mathematical & Information Sciences Directorate, Air Force Office of Scientific Research, Bolling Air Force Base, Washington, DC 20332
122. Jude Franklin, Administrator, Navy Center for Applied Research in Artificial Intelligence, Naval Research Laboratory, Washington, DC 20375
123. Dr. Richard R. Gawronski, Univ. of West Florida, Systems Sciences Dept., Pensacola, FL 32514-003
124. Dr. Stanley B. Gershwin, Laboratory for Information and Decision Systems, Room 35-308, Massachusetts Institute of Technology, Cambridge, MA 02139
125. Dr. Frederick R. Glickman, Director, Ergonomic Systems, Office of Advanced Technology, Technology Resource Dept., Washington, DC 20260-8123
126. Prof. R. C. Gonazlez, Dept. of Electrical Engineering, Univ. of Tennessee, Knoxville, TN 37916
127. Prof. James H. Graham, Electrical, Computer & Systems Engineering Dept., Rensselear Polytechnic Institute, Troy, NY 12180-3590

128. Walter L. Green, Dept. Head, Univ. of Tennessee, Electrical Engineering Dept., 406 Ferris Hall, Knoxville, TN 37916
129. Ernest L. Hall, Dept. of Mechnical & Industrial Engineerig - MS 72 Univ. of Cincinnati, Cincinnati, OH 45221
130. Dr. Robert Haralick, Vice President, Research, Machine Vision International Burlington Center, 325 East Eisenhower, Ann Arbor, MI 48104
131. Scott Y. Harmon, Code S442, Naval Ocean Systems Center, San Diego, CA 92152
132. P. J. Hayes, Assoc. Editor, Dept. of Computer Science, Univ. of Rochester, Rochester, NY 14627
133. Dr. Frederick Hayes-Roth, Chief Scientist, Teknowledge, Inc., 525 Univ. Ave., Palo Alto, CA 94301-1982
134. Ewald Heer, 5329 Crown Avenue, La Canada, CA 91011
135. Dr. David B. Hertz, Director, Intelligent Computer Systems Res. Inst., Univ. of Miami, Miami, FL
136. Michael Hitchcock, AFWAL/MLTC, Wright-Patterson Air Force Base, OH 45433
137. Steven W. Holland, General Motors Research Laboratories, Computer Science Dept., GM Technical Center, Warren, MI 48090-9057
138. Frank Holley, Human Engineering Lab., Attn+ ANXHE-FH (Holley), Aberdeen Proving Grounds, MD 21005
139. Larry L. Hollinghead, Manager, AI Development, Corporate Research & Development, Cincinnati, OH 45209
140. Dr. Ralph L. Hollis, Research Staff Member, Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, NY 10598

140–143. Sithaima Iyengar, Dept. of Computer Science, Louisiana State Univ., Baton Rouge, LA 70803-4020

144. Ramesh Jain, Assoc. Prof., Univ. of Michigan, Dept. of Electrical Engineering & Computer Science, Div. of Computer Science and Engineering, Ann Arbor, MI 48109-1109
145. Dr. Leslie Kaebling, SRI International, 333 Ravenswood Ave., Menlo Pak, CA 94025
146. Avi Kak, Dept. of Electrical Engineering, Purdue University, Lafayette, IN 47907
147. Dr. Malvin H. Kalos, AEC Computing & Applied Mathematics Center, Courant Institute of Math Science, New York Univesity, 251 Mercer St., New York, NY 10012
148. Yutaka Kanayama, Stanford University, Cedar Hall, Artificial Intelligence Laboratory, Stanford, CA 94305
149. Kazuhiko Kawimura, Assoc. Prof., Electrical Engineering, Vanderbilt Univ., Nashville, TN 37235
150. David Keirsey, Hughes AI Center, 23901 Calabasas Rd., Calabasas, CA 91302
151. Dr. Robert B. Kelley, Electrical, Computer, and Systems Engineering Dept. Rensselaer Polytecnic Institute, 15th St., Troy, NY 12180
152. Dr. Robert Leighty, U.S. Army Engineering Topographic Laboratory, Bldg. 2592, Attn+ ETL-R1 (Dr. Leighty), Fort Belvoir, VA 22060
153. Miron Livny, Univ. of Wisconsin, Computer Sciences, 4218 Computer Sciences & Statistics, 1210 W. Dayton St., Madison, WI 53706

154. Prof. Nan Loh, Center for Robotics & Advanced Automation, School of Engineering & Computer Sciences, Oakland University, Rochester, MI 48063
155. Tomas Lozano-Perez, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139
156. Dr. Azad M. Madni, Perceptronics, Vice President and Director, Artificial Intelligence and Man-Machine Systems Division, 6271 Variel Ave., Woodland Hills, CA 91367
157. Dr. Oscar P. Manley, Div. of Engineering, Mathematical, and Geosciences, Office of Basic Energy Sciences, U.S. Dept. of Energy, Washington, DC 20545
158. Stephen H. Maslen, Associate Director, Martin Marietta Laboratories, 1450 S. Rolling Rd., Baltimore, MD 21227
159. Dr. John McDermott, Senior Research Computer Scientist and Associate Department Head, Carnegie Mellon University, Pittsburgh, PA 15213
160. Drew McDermott, Dept. of Computer Science, Yale University, New Haven, CT 06520
161. Harvey B. Meieran, H. B. Meieran Associates, 458 S. Dallas Ave., Pittsburgh, PA 15208
162. Alexander Meystel, Drexel University, ECE, Philadelphia, PA 19104
163. M. R. Mikkilineni, P.E., President, MRM Engineers, P.C., 918 Park Ave., Pittsburgh, PA 15234
164. David Miller, Dept. of Computer Science, Yale University, Box 2158, Yale Station, New Haven, CT 06520-2158
165. Dr. Robert L. Moore, Vice President, Process Systems, LISP Machine, Inc. Suite 900, 6033 W. Century Blvd., Los Angeles, CA 90045
166. Dr. Howard Moraff, Program Director, Automation, Instrumentation & Sensing Systems, National Science Foundation, 1800 G. St., N.W., Washington DC 20550
167. Prof. Trevor N. Mudge, Univ. of Michigan, Dept. of Electrical Engineering and Computer Science, Div. of Computer Science & Engineering, Ann Arbor, MI 48109-1109
168. Scott D. Myers, Staff Engineer, Systems Analysis & Simulation, Martin Marietta Aerospace, Baltimore Div., 103 Chesapeake Park Plaza, Baltimore, MD 21220
169. H. -H Nagel, Coordinator Europe, Fraunhofer - Institut fur Informations - und Datenverarbeitung (IITB), Sebastian-Kneipp-Strasse 12-14, 7500 Karlsruhe 1, F. R. Germany
170. David Nitzan, Director, Robotics Dept., SRI International, 333 Ravenswood Ave Menlo Park, CA 94025
171. Prof. David E. Orin, Dept. of Electrical Engineering, The Ohio State University, 2015 Neil Ave., Columbus, OH 43210
172. Dr. John F. Palmer, Chairman, NCUBE Corporation, 915 E. La Vieve Land, Tempe, AZ 85284
173. Alexandre Parodi, Artificial Intelligence Center, FMC Corp., Central Engineering Laboratories, 1185 Coleman Ave., Box 580, Santa Clara, CA 95052
174. Richard Paul, Dept. of Computer & Information Science, Univ. of Pennsylvania, Prof. Judea Pearl, 4731 Boelfer Hall, UCLA, Los Angeles, CA 90024 Philadelphia, PA 19104

175. Dr. Alex Penland, SRI International, Room EK243, 333 Ravenswood Ave., Menlo Park, CA 94025
176. Dr. Jack E. Pennington, Automation Research Branch, National Aeronautics and Space Administration, Langley Research Center, MS 152D, Hampton, VA 23665
177. David Pessel, SOHIO, 4440 Warrensville Center Rd., Cleveland, OH 44128
178. Dr. Harry E. Pople, Decision Systems Laboratory, Univ. of Pittsburgh, School of Medicine, Pittsburgh, PA 15261
179. Dr. Mike Rabins, Dean of Engineering, Mechanical Engineering Dept., Wayne State University, Detroit, MI 48202
180. Prof. Marc H. Raibert, Dept. of Computer Science, Carnegie Mellon Univ., Schenley Park, Pittsburgh, PA 15217
181. Prof. G. B. Raju, Dept. of Electrical & Computer Engineerig, Ohio Univ., Athens, OH 45701
182. S. V. N. Rao, Dept. of Computer Science, Louisiana State Univ., Baton Rouge, LA 70803
183. Prof. Raj Reddy, Director, Robotics Institute, Carnegie Mellon Univ., Pittsburgh, PA 15213
184. Prof. Karl N. Reid, Head, School of Mechanical & Aerospace Engineering, Oklahoma State Univ., Stillwater, OK 74074
185. Howard Resnikoff, Vice President and Director of Research, Thinking Machine Corp., 245 First St., Cambridge, MA 02142-1214
186. Elaine Rich, Micro-Computer Systems, 9430 Research Blvd., Eichleon Bldg. No.1, Austin, TX 78759
187. Dr. Bill Richard, Exploratory Development, Division 5628, Sandia National Laboratories, Albuquerque, NM 87185
188. John Roach, Dept. of Computer Science, 562 McBryde Hall, Virginia Polytechnic Institute, Blacksburg, VA 24061
189. Dr. Guillermo Rodriguez, Machine Intelligence Systems, Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Dr., Pasadena, CA 91109
190. Prof. Azriel Rosenfeld, Center for Automation Research, Univ. of Maryland, College Park, MD 20742
191. Stan Rosenschein, Stanford Research Institute, 333 Ravenswood Ave., Menlo Park, CA 94025
192. James Sachs, Elfin Technologies, 3130 Coronado Dr., Santa Clara, CA 95054-3205
193. Prof. George Saridis, Electrical, Computer, and Systems Engineering Dept., Rensselaer Polytechnic Institute, 15th St., Troy, NY 12180
194. Roger T. Schappell, Aerospace Division, Martin Marietta Aerospace, P.O. Box 179, Denver, CO 80201
195. Jacob T. Schwartz, Courant Institute of Mathematical Science, New York Univ., 251 Mercer St., New York, NY 10012
196. Dr. Linda Shapiro, Machine Vision International, Burlington Center, 325 E. Eisenhower, Ann Arbor, MI 48104
197. Prof. Thomas Sheridan, Dept. of Mechnical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139
198. Dr. Robert Simmons, Dept. of Computer Sciences, Univ. of Texas, Austin, TX 78712

199. Dr. Stephen Squires, Information Processing Technology Office, DARPA, 1400 Wilson Blvd., Arlington, VA 22209
200. Robert N. Stouffer, Editor, Robotics Today, One SME Drive, PO Box 930, Dearborn, MI 48121
201. Prof. Delbert Tesar, Carol Cockrell Curran Chair in Engineering, Univ. of Texas at Austin, Room ETC 4.146C, Austin, TX 78712
202. Perry W. Thorndyke, Director, Artificial Intelligence Center, FMC Corp., Central Engineering Laboratories, 1185 Coleman Ave., Box 580, Santa Clara, CA 95052
203. Robert B. Tilove, General Motors Research Laboratories, Computer Science Dept., GM Technical Center, Warren, MI 48090-9057
204. Director, US Army Human Engineering Laboratory, Attn: AMXHE-CS (Shoemaker) Aberdeen Proving Grounds, MD 21005-5001
205. Prof. Leonard Uhr, Dept. of Computer Science, Univ. of Wisconsin, Madison, WI 53706
206. Steven Vere, Jet Propulsion Laboratories, Pasadena, CA 91109
207. Prof. C. C. Wagner, 13F Dodge Hall, Oakland Univ., Rochester, MI 48063
208. Alan Waxman, Thinking Machine Corporation, 245 First St., Cambridge, MA 02142-1214
209. Prof. William G. Wee, Dept. of Electrical & Computer Engineering, University of Cincinnati, Cincinnati, OH 45221
210. Chester A. Winsor, Martin Marietta Aerospace, 6801 Rockledge Dr., Bethesda, MD 20817
211. Pat H. Winston, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, Cambridge, MA 02139
212. Prof. Lofti Zadeh, Prof. of Electrical Engineering & Computer Sciences, Div. of Computer Sciences, University of California, Berkeley, CA 94720

216–242. Technical Information Center (TIC)