TOPOLOGICAL EXPLORATION WITH MULTIPLE ROBOTS

G. DUDEK
Centre for Intelligent Machines, McGill University,
3480 University St., Montreal, QC, Canada, H3Y 2A7,
dudek@cim.mcgill.ca

M. JENKIN, E. MILIOS, D. WILKES
Vision, Graphics and Robotics Laboratory,
Department of Computer Science,
York University, 4700 Keele St., North York, ON, Canada, M3J 1P3
{jenkin,eem,wilkes}@cs.yorku.ca

ABSTRACT

This paper describes a technique whereby a group of mobile autonomous agents explores an unknown graph-like environment and constructs a topological map of it. The key idea is that the mobile agents start at a common node of the environment, explore independently (using a previously published algorithm) and agree to meet after a specified number of moves to exchange information in order to augment each other's partial map of the world. Just after each exchange, robots have the same map of the world, which is a superset of each robot's map just before the exchange. The process of harmonizing each other's maps involves a combination of reasoning (for the areas that consist of paths common to all partial maps before the exchange) and physical movements of the robots, to check whether certain nodes in one map are identical to nodes in the rest of the maps.

INTRODUCTION

The problem of exploring and mapping an unknown terrain in an autonomous way is becoming increasingly important, especially for environments which are hostile or inaccessible. In practice, the progressive accumulation of positional error makes the construction of a map based on an absolute metric coordinate system problematic at best. Furthermore, the definition of a map in terms of specific locations or landmarks instead of x-y coordinates facilitates person-machine interaction; it is much easier to tell a robot to carry mail to the third office down the hall or to the foyer than to x-y coordinate (975,436). Because exploration and mapping can be a time consuming process, it is valuable to contemplate the use of multiple robots to accelerate the exploration procedure.

This paper describes a technique whereby a group of autonomous agents, such as mobile robots, can explore an unknown environment and make a topological map of it. It is assumed that the environment can be represented as a graph, that is, as a fixed set of discrete locations or regions with an ordered set of bidirectional paths between them. In previous work, it has been shown that such terrains can be fully explored and described by a single robot, provided that it is equipped with a single movable marker even if there are no spatial metrics and minimal sensory ability on the part of the robot [2].

We address this "worst-case" problem by assuming that our mobile robots can obtain *no* metric positional information whatsoever. Of particular concern is the *location identification* problem: when visiting a given location in the world, how can the robot determine whether or not it was already visited and therefore already present in the map prepared thus far. Markers are used to resolve position ambiguities. In [2] it was shown that in the worst case, the number of edges of the map that must be traversed by the robot (i.e. the number of mechanical steps) has a dominant term of $6n^3$.

THE WORLD MODEL

The algorithms presented in this paper operate on an augmented graph-like representation of the world, defined in detail in [2] and summarized below. Although the pros and cons of this approach to map representation are beyond the scope of this paper, let us say that the abstraction used here serves as a lowest common denominator for real robotic systems, yet is nevertheless rich enough to represent location in a meaningful way. A robot's "purpose in life" is to use its ability to act and to perceive in the graph domain in order to build an augmented undirected graph (including constraints on how it is embedded), which is isomorphic [3] to the finite world it has been assigned to explore. The robot's inputs are its sensations and it can interact with the world only through its actions. Note that in practice such a graph could be defined within a continuous environment based of landmarks or other features [4].

The world is defined as an embedding of an undirected graph G with set of vertices V and set of edges E. We assume the world has no multiple edges between two vertices or an edge incident twice at the same vertex. The definition of an edge is extended slightly to allow for the explicit specification of the order of edges incident upon each vertex of the graph embedding. This ordering is obtained by enumerating the edges in a systematic (e.g. clockwise) manner from some standard starting direction.

A robot can move from one vertex to another by traversing an edge (a *move*), it can pick up a marker that is located at the current vertex, and it can put down a marker it holds at the current vertex (a *marker operation*). We assume that a sequence of moves is invertible, and can be retraced, by the robot keeping track of the relative order of the edge it took in and out of each vertex. A single move is thus specified by the order r of the edge along which the robot exits the current vertex, where r is defined with respect to the edge along which the robot entered such vertex. Note that in the special case of a planar embedding of a graph, enumeration of edges in a clockwise fashion satisfies the above assumption.

A marker operation is fully specified by indicating whether it is being picked up, put down, or not operated upon. A simple action a is defined as a marker operation accompanied by a move. A path is a nonempty sequence of actions.

Perception A robot's perception is of two kinds, marker-related (whether the marker is present at the current vertex) and edge-related perception. A robot can determine the relative positions of edges incident on the current vertex v_i in a consistent manner, e.g. by a clockwise enumeration. As a result, it can assign an integer label to each edge incident on v_i, representing the order of that edge with respect to the edge enumeration at v_i. The label 0 is assigned arbitrarily to the edge $E_{i,j}$, through which the robot entered vertex v_i. The ordering is local, because it depends on the edge $E_{i,j}$. Entering the same vertex from two different edges will lead to two local orderings, one of which is a permutation of the other. Note that if the graph is planar and a spatially consistent (e.g. clockwise) enumeration of edges is used, then two permutations will be simple circular translations of each other. If the robot visits the same vertex twice, it must relate the two different local orderings produced and unify them into a single global ordering, for example by finding the label of the 0-th edge of the second ordering with respect to the first ordering. Determining when the same vertex has been visited twice and generating a global ordering for each vertex is part of the task of the exploration.

EXPLORATION WITH ONE ROBOT

In earlier work ([1, 2]) we demonstrated that in general it is not possible for a single robot to explore and map an unknown environment with this sort of limited sensory information without some additional information. The basic problem is the location identification problem defined previously.

In this earlier paper it was also demonstrated that as long as the explorer had a single unique marker which could be dropped and picked up at will it was possible for the explorer (or robot) to fully map his environment (an arbitrary graph) within $O(N^3)$ steps (although complexity for most typical cases appears substantially better than this worst-case bound). The basic technique was to use the invertibility of the path taken by the robot, and the fact that the marker is unique, to disambiguate locations which could be confused. The algorithm proceeds by incrementally building a known map out of a known subgraph S. Whenever a potentially new location is encountered, the marker is dropped in the new location and all nodes in the known map are visited. If the marker is found in known graph S, then this location corresponds to an already visited (and mapped) location. On the other hand, if it is not found in S, then this location really is new and can be added to the map. In either case, additional information has been found, and more of the world (graph) has been explored.

The cost of exploring the graph in terms of edges traversed by the robot (mechanical complexity) is $O(N^3)$. This follows from the need to go back and actually visit all of the locations in the known sub-graph to solve the "have I been here before" problem.

EXPLORATION WITH MULTIPLE ROBOTS

In this work, we consider the use of two (or more) robots to solve this environment exploration problem. We assume the same environmental representation as [2] but populate the environment with two or more robots, each of which is equipped with a unique marker. We also assume that the robots can only communicate with each other when they are at the same physical location (i.e the same vertex of the graph).

Algorithm outline

Our multi-robot exploration algorithm has the following outline. Like the existing methods, exploration proceeds by accumulating a known and an unknown environment subgraph, and incrementally accumulating information about the unknown part of the map. The multi-robot version of the algorithm proceeds by having all of the robots start at a single location and initially partitioning the unknown part of the world so that each robot can explore independently. After exploring for a previously agreed-upon time interval, the robots return to a common known location to fuse (or harmonize) their individual partial world models. After fusing the individual environmental representations, the remaining unknown portions of the environment are re-partitioned between the robots and the entire algorithm repeats until the entire environment is explored.

The difficult task in this process is the task of fusing the partially explored environments obtained by each of the robots. This process involves disambiguating possible confusions between the environmental models obtained by each of the robots.

Statement of the algorithm for two robots

Joint exploration proceeds by using alternating phases of independent exploration and subsequent merging of the independently acquired sub-graphs. This latter merging step is referred to as "harmonization". At any time, the robots retain a common representation of some part of the world (the commonly known subgraph) as well as additional information regarding other parts of the world that the other robot may not be aware of. This commonly known map is a subgraph $S^m(t)$ that evolves over time. By abuse of notation, we will refer to it simply as S^m.

The robots start at a common "root" node with a common reference orientation. This single node is the initial definition of S^m. As successive iterations of the exploration and merging process take place, S^m grows monotonically until it is isomorphic to the entire world map G.

The key aspect of the entire process is the following exploration and merging algorithm.

EXPLORE-MERGE:

0\. Start at an arbitrary "root" node in the mutually known subgraph S^m.

1\. Plan partition of work and rendezvous schedule.
   - agreeing which outgoing edges of the root each robot should depart from.
   - after how many edge traversals they should meet and where.

2\. Each robot explores using the single-robot algorithm.
   This allows them to each define its own (evolving) known subgraph S^i.
   These known subgraphs each contain the mutually known subgraph: $S^m \subset S^i$

3\. At the predetermined time, the robots return to the root or another agreed point in S^m

3a.Harmonize nodes that are in both robot's explored subgraphs without moving.
   (i.e. nodes in S^1 and S^2 but not in S^m).

4. Harmonization begins.
   Each robot identifies the perimeter of its known subgraph $P^i = \partial S^i$.
   These are all nodes with an (unexplored) edge leading *out* of the known subgraph

5. Perimeters of robots' known subgraphs are "mentally harmonized" (without moving).
   The following steps are repeated until no more nodes can be mentally harmonized.

   a) consider nodes on the perimeters P^i of each graph in order of distance from S^m.
      (First do robot 1 with respect to 2, then 2 with respect to 1)

   b) if a node v is distance one (edge) from the common map S^m
      and the edge linking to S^m is present and explored in both maps S^i and S^j,
      then the node v and the associated edge are included in S^m.
      This is a form of "spreading activation".

6. For each boundary node u of each explored subgraph S^i not yet harmonized,
   harmonize it by physical robot movements as follows:
   a) place a marker at u.
   b) robot j visits all nodes of explored subgraph S^j looking for the marker
      If marker found at node w of S^j, then u of S^i is harmonized with w.
      Harmonization includes the definition of a common reference edge for the node,
      and the harmonization of unexplored edges emanating from the node.

      If the marker is not found anywhere, node u in $S^i \neq$ node w in S^j and
      node u is known as visited by only one of the two robots,
      and is added to the known subgraph S^m.

7. Once the perimeters have been harmonized (and thus contained in S^m),
   the entire graph can classified into 4 types of nodes (i.e. 4 types of regions):
   a. Nodes that are in the common sub-graph S^m.
   b. Nodes that have only been visited by robot 1
   c. Nodes that have been visited by robot 2
   d. Nodes that have been visited by both robots.

   Nodes visited by both robots (in class d) are mentally harmonized (step 5)
   Nodes in class b and c can be added to S^m trivially (Figure 1).

The outcome of the EXPLORE-MERGE process is a single known subgraph S^m, and a common set of unexplored edges emanating from nodes of S^m known to both robots.

DISCUSSION

An algorithm has been presented for topological exploration by two robots. The two robots explore independently and then meet to exchange information and harmonize their maps. Harmonization happens mentally (without physical movement) as much as possible, with physical movement required otherwise. This algorithm is provably complete and convergent, and can be tuned to trade off parallelism against more

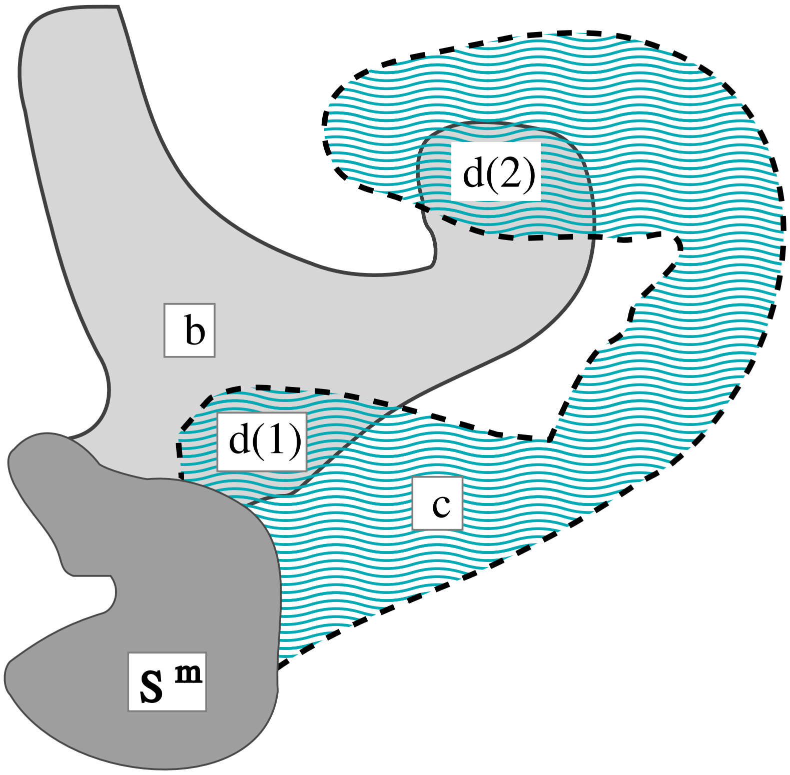

Figure 1: The four types of nodes (regions) are shown, as discussed in step 7 of the algorithm. Regions $d(1)$ and $d(2)$ are both of type d.

frequent meetings. Research on extending the algorithm to more than two robots, and its experimental comparison on random graphs with the single-robot version previously published is currently in progress.

Acknowledgements This work is partly sponsored by the IRIS network.The financial support of NSERC is gratefully acknowledged.

REFERENCES

References

[1] Gregory Dudek, Michael Jenkin, Evangelos Milios, and David Wilkes. Robot exploration as graph construction. RBCV-TR-88-23, Dept. of computer science, University of Toronto, October 1988.

[2] 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.

[3] L. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams. *ACM Trans. on Graphics*, 4, 1984.

[4] B. Kuipers and T. Levitt. Navigation and mapping in large-scale space. *AI Magazine*, pages 25–43, summer 1988.