Using local information in a non-local way for mapping graph-like worlds*

Gregory Dudek†, Paul Freedman‡, Souad Hadjres†

† McGill Research Center for Intelligent Machines
3480 University St., Montreal, Québec, Canada H3A 2A7

‡ Centre de recherche informatique de Montréal
3744, rue Jean-Brillant, Bureau 500, Montréal, Québec, Canada H3T 1P1

Abstract

This paper describes a technique whereby an autonomous agent such as a mobile robot 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 paths between them. In previous work, it has been shown that such worlds can be fully explored and described using a single movable marker even if there are no spatial metrics and almost no sensory ability on the part of the robot. Here we present an approach to the exploration of unknown worlds without such a movable marker which is simply based on the structure of the world itself.

Locations in the world are identified by a non-unique "signature" that serves as an abstraction for a percept that might be obtained from a robotic sensor. While the signature of any single location may not be unique, under appropriate conditions the distinctiveness of a particular set of signatures in a neighborhood increases with neighborhood size. By using a collection of non-unique local signatures we can thereby construct an "extended" signature that uniquely determines the robot's position (although in certain degenerate worlds additional information is required).

1 Introduction

The problem of exploring and mapping an unknown world in an autonomous way is becoming increasingly important especially for environments which are hostile or inaccessible [Almeida and Melin, 1989]. This exploration/mapping task appears to be simple if one imagines an idealized error-free robot with error-free (albeit limited) perceptual capacities. 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 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)".

In this paper, we address the "worst-case" problem by assuming that our mobile robot can obtain *no* metric positional information whatsoever. We consider worlds that can be abstracted as graphs composed of distinct locations identified by limited sensory cues and connected by paths that provide only connectivity information. 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 ? We demonstrate that by augmenting local information about the location by information about its neighbours and their neighbours up to a given distance, it becomes possible in many non-worlds to answer the location identification problem correctly [Corneil and Kirkpatrick, 1980; Dudek *et al.*, 1991b]. In this way, a map of the world may be obtained which faithfully models the locations and their connectivity.

2 Problem context

We assume that the robot world is composed of a finite number of distinguishable *locations* connected by bi-directional paths. Such an world may be represented as a graph where *vertices* correspond to locations and *edges* correspond to paths[1].

For example, the world shown in Figure 1 may be represented by a graph $G = (V, E)$ where V is the set of vertices $v_1, v_2, .., v_6$ and E is the set of edges where $e_{i,j}$ or (v_i, v_j) denotes the edge connecting vertices v_i and v_j.

Locations are identified by a non-unique *signature*, i.e. a set of characteristic features which may be reliably obtained from the robot's percepts. For our purposes here, we will further assume that upon arriving at a location, the robot can identify the number of paths leading to other locations. To simplify the exposition, we will consider that the signature of a location is the *degree* of

*This paper appears in the *Proceedings of Int'l Joint Conference of Artificial Intelliegence,* Chambery, France, Aug. 1993.

The financial support of NSERC is gratefully acknowledged.

[1] As consequence of this graph representation, there may only be one edge between adjacent vertices and thus worlds in which there are multiple paths between adjacent locations cannot be considered eg. graphs with cycles of length 2.

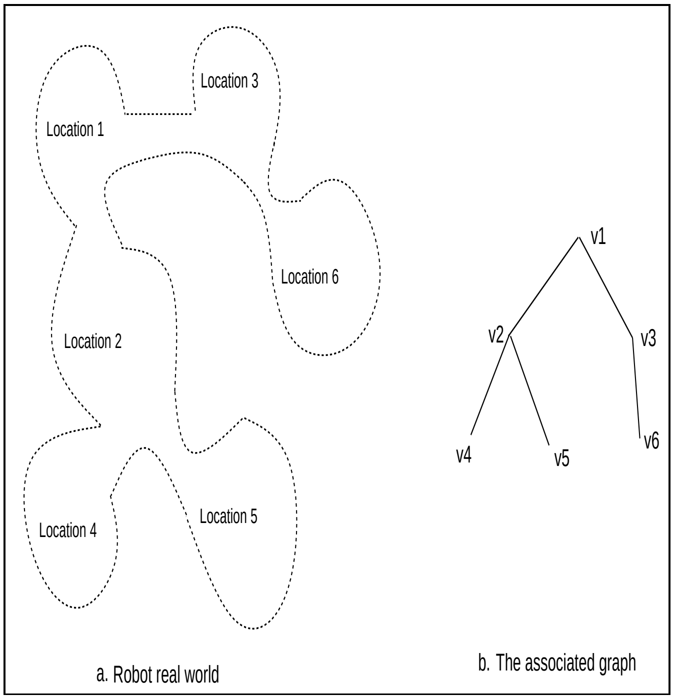

Figure 1: An example of a real world (a), and its associated graph (b).

the corresponding vertex (another example might associate signature with vertex colour). In addition, we shall assume that at each location, the robot is able to enumerate the incoming/outgoing paths in a systematic way (eg. clockwise), relative to the path by which it arrived at the location.

As the robot performs the exploration, it records all the information obtained whenever any action, sensing or motion (path traversal), is performed. By 'remembering' all motion sequences, the robot may retrace any previously performed motion.

3 Related work

In contrast to a geometric map, a *topological* map can be defined [Davis, 1986] as a map including all fixed entities of the world such as distinguishable locations and regions, linked by topological relations eg. connectivity, containment. Advantages of such an approach include its qualitative nature and attractive links to theories of human cognition and mapping. Such a map is often represented as a graph where vertices are locations and edges their adjacency relations.

Metric and topological information can be hierarchically related, within the context of a multilevel representation theory of a *large scale* space [2] based on the observation and re-acquisition of distinctive visual events called *landmarks* [Chatila and Laumond, 1985; Kuipers and Byun, 1988]. It is often assumed that distinctive locations can be robustly found, that they are not too numerous, and that no two locations can be confused [Schwartz and Yap, 1987; Leonard and Durrant-Whyte, 1991]. Clearly, this last assumption is an idealization of a real robot exploring a real world; not making it leads to serious complications [Basye and Dean, 1990].

For example, the TOTO robot [Mataric, 1990] is a real device that creates a topological map (a graph) as it explores its world. As landmarks are detected, they become nodes in the graph along with their qualitative properties, i.e. type (left wall, right wall, corridor) and associated compass bearing. A clever "truth maintenance" protocol is invoked to ensure that the same landmark does not become multiple nodes in the graph. This approach illustrates several important components of topological mapping and involves substantial domain-dependent processing.

The work of [Dudek *et al.*, 1991a] deals with exploration using markers and is directly related to the approach described here. The same assumptions apply as to the nature of the world to be explored and the perceptual capabilities of the robot but in addition, the robot is equipped with at least one recognizable marker which can be put down or picked up. This physical marker makes v_i distinctive according to the others vertices, and therefore represents a 'temporary' signature of that location. It is shown that such worlds can be fully explored and described in limited complexity using a single movable marker (like a pebble) even if there are no spatial metrics and almost no sensory ability on the part of the robot.

The marker-based analysis requires that the robot be able to reliably place, identify and recover the markers it uses for exploration. In this paper, we show how mapping can be accomplished without such markers even though individual locations may not be uniquely identifiable.

4 The mapping algorithm

The exploration and mapping algorithm has essentially two stages. First, the robot, starting from an arbitrary location, explores its (unknown) world by visiting all locations. Information thus obtained is then used to generate a *model* of the observed connectivity[3].

As described above, we shall assume that the robot is equipped with a sensing device which is used, at each location, to determine the number of incoming/outgoing paths and enumerate them in a consistent manner eg. clockwise. The number of paths then defines the *signature* of the location, i.e. the degree of the corresponding vertex in the map. Since this is often inadequate to uniquely identify a vertex (or equivalently, to uniquely specify connectivity) we shall also exploit signature information about the location's neighborhood.

While the exploration takes place, the robot constructs a data representation called the "*exploration tree*" which includes, at the end of the exploration, the set S of all possible world models (i.e maps) consistent

[2] A "large-scale" space is a space whose structure is at a significantly larger scale than the observations available at an instant [Kuipers and Byun, 1988].

[3] Recall that between any two adjacent locations, there is at most one edge.

with the robot's observations. This set of solutions is called the "*solution universe*".

If S contains more than model, then the robot must rely on additional knowledge about the world such as the total number of locations, information about the probability distribution of location signatures, or perhaps some compass measurements, to identify that model which best represents the connectivity information in the world.

4.1 The exploration tree

The exploration tree refers to the collection of possible hypotheses about the world the robot is exploring, given the data accumulated thus far in the exploration. It is incrementally constructed while the exploration takes place. The *root* of the tree is just the initial location from which the exploration began. A *level* in the tree corresponds to the traversal of a previously unexplored edge. The *nodes* belonging to a given level of the exploration tree represent possible partial models of connectivity in the world, according to the locations visited thus far. Nodes corresponding to the current level of exploration are called *frontier* nodes. *Leaf* nodes represent possible models (complete configurations) of world connectivity. A given node in the exploration tree is considered to be a leaf node (i.e. a possible model) if there are no paths still to be traversed[4].

Our notation is as follows:

- vertices corresponding to locations in the world are denoted by v_1,v_2, ...
- vertices associated with nodes in the exploration tree are denoted by v_1^j, v_2^j, ..., where v_i^j corresponds to the j^{th} visit of location v_i. Since a given location may be visited several times as part of the exploration, we can have several associated vertices in the exploration tree nodes corresponding to different visits of that vertex.

A correct model is characterized by the fact that when the robot visits a given location multiple times, it 'recognizes' that these are all visits to the *same* location. That is, there exists a correspondence between v_i^1 (the first visit to the location corresponding to vertex v_i) and v_i^k (the kth visit) for all k, and an absence of other (incorrect) correspondences. To guarantee successful exploration, i.e. exploration leading to the creation of a solution universe S which necessarily contains a model of existing connectivity in the world, two problems must be addressed:

1. How can the robot know when all of the locations in the world have been visited ?
2. When a location is visited, how can the robot know whether or not it represents a location previously visited and is therefore already present in the exploration tree ? We shall call this the *location identification* problem.

[4]This is because in the exploration tree, the robot only 'knows' whether or not there are paths still to be traversed associated with a location previously visited, not whether or not there are extra locations to visit.

Many possible exploration strategies are possible (even stochastic ones) but for simplicity, we will consider a FIFO (first-in first-out) traversal of new edges. This guarantees that all edges will be explored, although perhaps not optimally. In Figure 2, we illustrate how the robot explores the world shown in Figure 1, by showing the robot motions and the associated actions to construct the exploration tree. In this example, the exploration tree has only one branch and the solution universe contains just one possible solution which is the leaf of the tree, the framed node.

The second problem is more complex since location identification must be performed with very limited information. Indeed, by associating the signature of a location with vertex degree, the robot cannot always know when it is visiting a location for the first time or not. For example, in a world which contains cycles, the robot will inevitably re-visit some locations. That is, when the robot visits a given location, it could either be the first visit to a new (previously un-visited) location, or a revisit to any of the locations that have the same signature (i.e. that appear the same). Thus, when the robot visits a location, it must consider all possible ways of adding vertices to the frontier nodes in the exploration tree.

Three classes of errors or mis-identifications can be defined when the robot visits a given vertex v_i. (Errors associated with paths (edges in the exploration tree) are subsumed within this classification.) These are as follows.

E1 Errors of type OLD-LOOKS-NEW. A vertex v_i is assumed to be a new vertex even though it has been visited before (i.e. a failure in correspondence). In this case, an additional vertex is added to represent the current location even though a vertex for the current location has already been created.

E2 Errors of type MIS-CORRESPONDENCE. A vertex v_i is "recognized" as a known vertex v_j ($j \neq i$) even though, in reality, it is another old vertex v_k (i.e. the robot has confused two existing nodes). Thus, an erroneous edge is added to the world.

E3 Errors of type NEW-LOOKS-OLD. A vertex v_i is assumed to be an already visited vertex – an old vertex – even though it is new. In this case, the map will have a missing vertex relative to the real world.

Branches in the exploration tree are created as a result of modelling the true topological structure of the world, or by making one or more correspondence errors of different types. In most cases, branches arising from errors eventually terminate due to inconsistencies resulting from the incorrect topology induced by the error(s). For example, the robot connects two vertices v_1 and v_2 during the exploration of v_1 and then realizes that it is a wrong connection during the exploration of v_2. Of course, the development of any branch is halted once the frontier node has no more paths to traverse.

The exploration tree will always contain a branch for which no errors are committed, i.e. a branch leading to a leaf which faithfully describes the connectivity in the world.

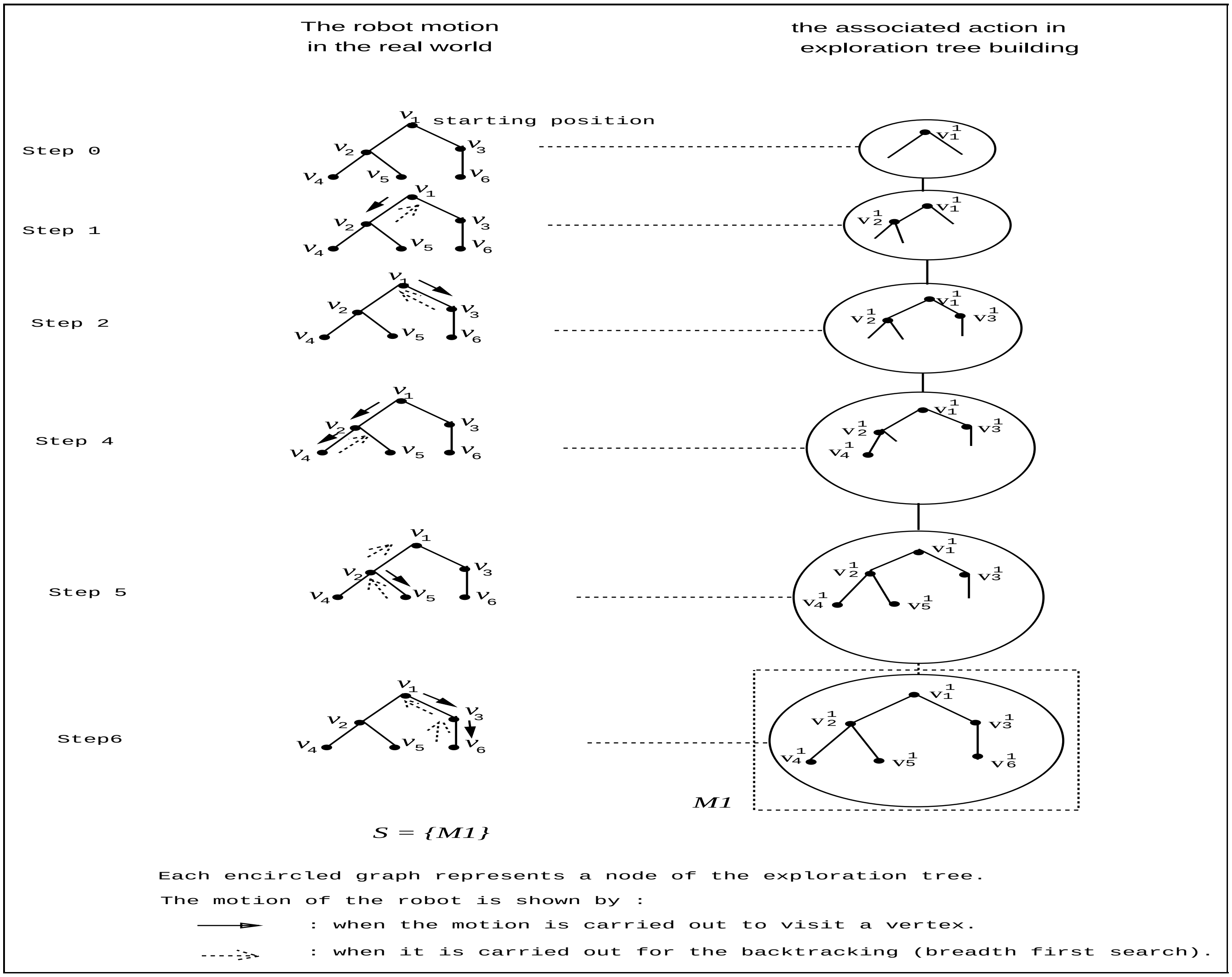

Figure 2: An exploration tree for the world shown in Figure 1.

4.2 The extended signature

Typical exploration trees usually include branches that are subsequently pruned (i.e. they develop inconsistencies before they lead to a complete model). This can be observed in the tree shown in Figure 3. The major reason for this is the weakness of the signature information used by the robot for addressing the location identification problem; incorrect hypotheses regarding vertex correspondences cannot be avoided based on local perceptual input. To make the exploration more robust and effective, we shall now exploit non-local information by defining an *extended signature* incorporating signature information about a location's neighbours.

We begin by defining the initial (zero'th) neighbours $N_0(u)$ of a vertex u as follows:

$$N_0(u) = \{u\} \tag{1}$$

$$\tag{2}$$

Then its immediate neighbours $N_1(u)$ may be defined as follows:

$$N_1(u) = \{v \in V | (u,v) \in E\} \tag{3}$$

We can then define the immediate 'outgoing' neighbours $N_2(u)$ of those vertices in $N_1(u)$ as follows:

$$N_2(u) = \{v \in V \mid v \in \{[N_1(N_1(u)) - u]\}\} \tag{4}$$

Note that we take care to exclude u from this list, in the case where there exists a cycle involving u and some vertex in $N_1(N_1(u))$.

More generally, we define $N_m(u)$ to be the m^{th} neighbours of u:

$$N_m(u) = \{N_1(N_{m-1}(u)) - \bigcap N_{m-2}(u)\}. \tag{5}$$

To make the vertex matching in the exploration tree more robust, we will now go beyond local signature information (vertex degree) to consider an *extended signature* defined in terms of the signatures of the neighbours of a location. For example, suppose that we have two vertices u and v and we wish to establish whether they refer, in fact, to the same physical location. If they have identical signatures (degrees), then we consider their immediate neighbours, $N_1(u)$ and $N_1(v)$. If the immediate neighbours also have identical signatures and appear in the

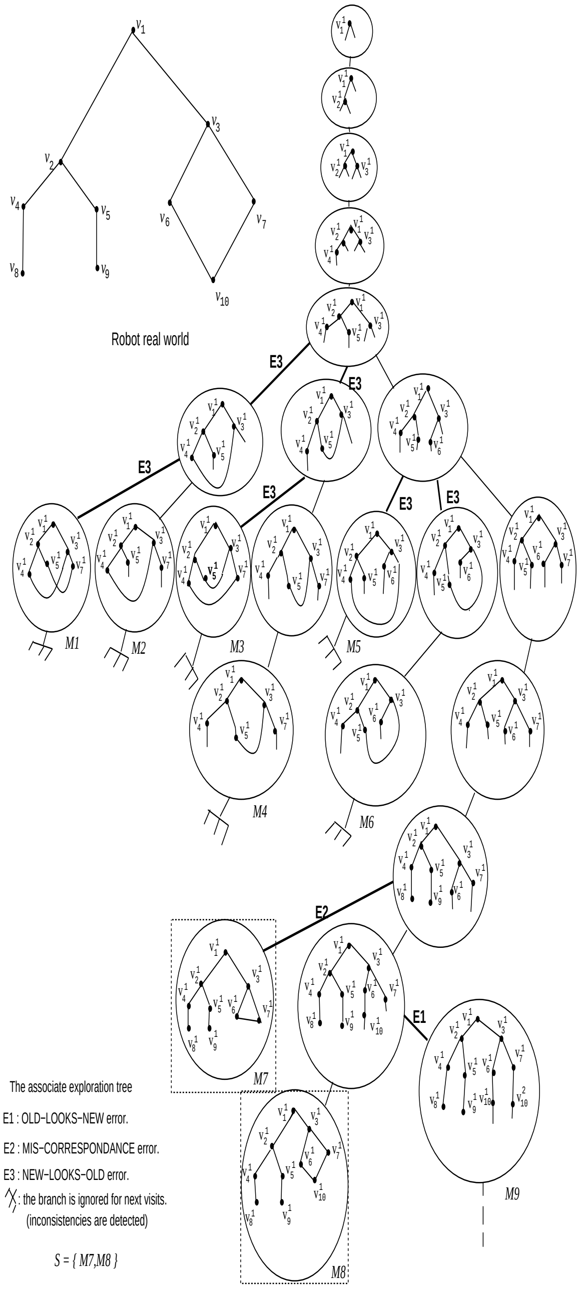

Figure 3: An example of an exploration tree illustrating the three types of errors.

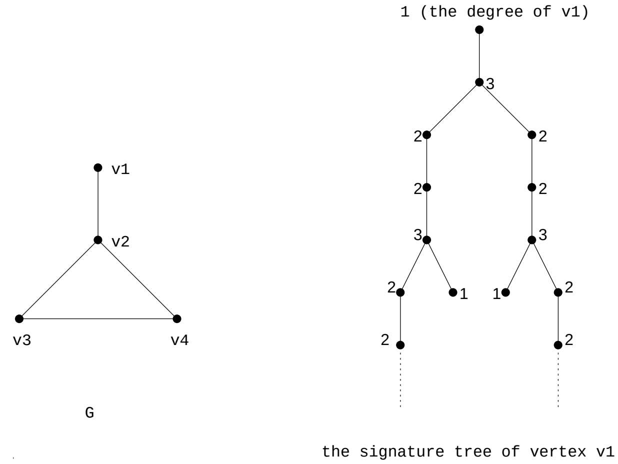

Figure 4: An example of signature tree.

same configuration, then we consider $N_2(u)$ and $N_2(v)$, and so on.

We may define the extended signature of a vertex u in terms of the signatures of its neighbours up to a given distance m as follows:

$$Sig_0(u) = (deg(u)) \tag{6}$$

$$Sig_m(u) = (Sig_{m-1}(u), (deg(v))) \quad \forall v \in N_m(u) \tag{7}$$

where $deg(v)$ denotes the degree of vertex v. For example, for the graph in Figure 1, the extended signatures of the vertex v_1 are as follows:

$$Sig_0(v_1) = (2) \tag{8}$$

$$\begin{aligned} Sig_1(v_1) &= (2,(3,2)) \\ Sig_2(v_1) &= (2,(3,2),(1,1,1)) \end{aligned} \tag{9}$$

This extended signature may be also be viewed as a *signature tree*, where the root represents the degree of vertex u and nodes belonging to a level i in this tree represent the degrees of the i^{th} neighbours of u; see Figure 4.

Note that a node's extended signature is generally only unique with respect to a specific reference edge. Consider as root a vertex with degree d, with d possible extended signatures. In Figure 5, we illustrate how any extended signature may be obtained from any other extended signature by circularly re-ordering its edges. Consequently, when comparing two vertices, the robot must take into consideration all possible re-orderings of their extended signatures. More precisely, if the extended signatures are *isomorphic*, then the two vertices might correspond to the same location, i.e. we must consider extended signatures involving more neighbours.

4.3 Algorithm details

The exploration algorithm functions by creating a subsection of the world for which there exists a known map. Initially, this is simply the robot's starting location. As the exploration proceeds, this map is gradually expanded

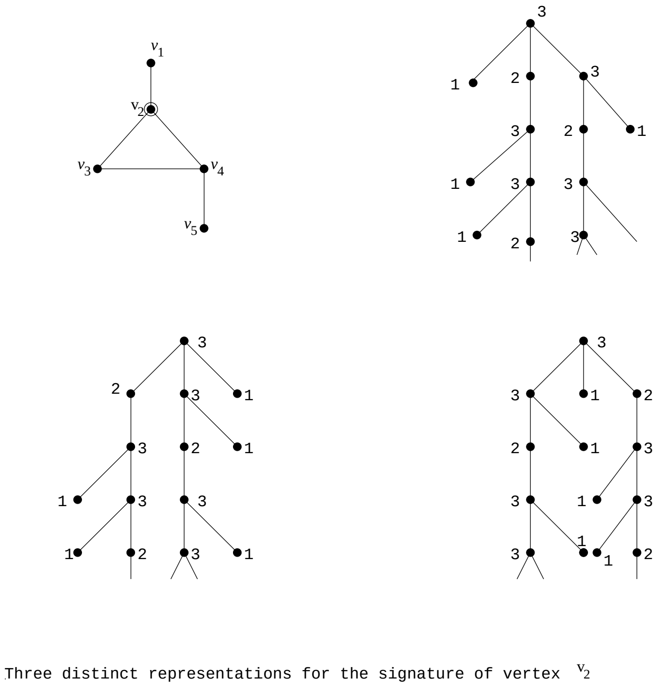

Figure 5: An example illustrating that the signature tree is invariant under circular re-ordering.

by adding new vertices and their connectivity to the map already established. When a location is visited and a new vertex in the map is postulated, its relationship to the set of known vertices must be established and the correspondence (if any) with any other vertex must be verified (using the signature and extended signature analysis).

When the robot visits a location corresponding to a new vertex u, it starts by ordering the incident edges e_i according to the "reference" edge e_0 by which it arrived at u (eg. using a clockwise ordering). (Note that the "reference edge" is only defined by the robot's own history – the reference edge is not perceptible in the graph itself.) The robot then examines these edges (except e_0) sequentially by traversing each one to visit the vertex at the other end. The process of traversing the ith edge e_i may be described as follows:

1. Traverse edge e_i to reach the other vertex v and compute $deg(v)$.
2. Verify if edge e_i is already connected to another vertex w in the exploration tree (for each model). If yes, verify the validity of the connection by comparing the extended signatures of v and w: $Sig(v) and Sig(w)$; if they are different, then reject the proposed connection.
3. If edge e_i is 'free', i.e. not already connected to another vertex, compute $C(e_i)$, the set of all possible connections, which includes:
   - connections to previously visited vertices v_i with $Sig(v_i) = Sig(v)$ and with a 'free' incident edge. For each such v_i found, create a new node in the exploration tree by adding an edge connecting v to v_i.
   - a connection to a new vertex w. A new node is created in the exploration tree by adding an edge connecting v to w.

The cost of this exploration process in terms of actual edge traversals by the robot (mechanical complexity) depends only to the search strategy used by the robot in moving through the world. The algorithm is compatible with almost any strategy that progressively traverses new edges. Breath-first search provides a simple example.

The computational cost of the mapping algorithm is a function of the number of nodes (possible world models) in the exploration tree; the actual generation, maintenance and comparison operations for extended signatures have low-order polynomial complexity. The number of nodes in the exploration tree depends on the distinctiveness of the perceptual information extracted at each location. When insufficient perceptual information is available to constrain the growth in the exploration tree various pruning or deferred expansion strategies are possible.

Full details about the algorithm with some complete examples may be found in [Dudek *et al.*, 1993].

5 Coping with ambiguity in location identification

Despite the availability of an extended signature, ambiguity may still remain in location identification. As a result, the universe of possible solutions S may contain various models which are equivalent insofar as the extended signature is concerned, of which just one faithfully reflects the connectivity in the world (for example a simple cycle of either three or four vertices). In this section, we consider additional information (in addition to location signature) that makes it possible to identify the the correct model in S.

As we have shown, the presence of cycles in the world gives rise to an exploration tree of infinite size, due to the existence of a branch of infinite length, corresponding to the case where every visited vertex is considered to be new. Hence, the exploration procedure must elect to cease exploration even though some possible models have not been fully explored.

One cue as to when to stop exploration and select appropriate model(s) is prior knowledge of the number of locations N in the world. For example, this might be the case when the robot is exploring a multi-floor building where all floors have the roughly the same number of offices. This implies that the exploration process can terminate as soon as all nodes in the exploration tree have at least N vertices, i.e. as soon as nodes which contain edges still to be traversed have more than N vertices.

Another possible cue is prior knowledge of the *planarity* of the world being explored. We may often assume that the world to be explored is planar (topologically speaking) and hence a simple planarity test may suffice to distinguish between equivalent models. Indeed, if N is also known, then we demonstrate elsewhere [Dudek *et al.*, 1993] that our algorithm leads to a unique (correct) solution (or multiple isomorphic correct solutions).

Perhaps most important is the fact that alternative branches in the exploration tree correspond to assumptions regarding the existence (or non-existence) of multiple locations in the world that are perceptually indistinguishable. If the likelihood of such occurences can be estimated then alternative worlds in the exploration tree can be ranked in terms of their overall likelihood.

6 Discussion and Conclusions

In this paper, we have described how a robot with limited perceptual capacities may explore and faithfully map the topology of unknown graph-like worlds. The approach is based on aggregating non-local information to compensate for potentially ambiguous local perceptual information. Locations in the world are identified by a non-unique "signature" that serves as an abstraction for a percept that might be obtained from a robotic sensor. While the signature of any single location may not be unique, under appropriate conditions the distinctiveness of a particular set of signatures in a neighborhood increases with neighborhood size. By using a collection of non-unique local signatures we can thereby construct an "extended" signature that uniquely determines the robot's position (although in certain insufficiently rich worlds additional information is also required). The algorithm makes use of no metric information such as the distances of the paths traversed, but the availability of such measurements would simplify the mapping problem.

The worst case behavior of the algorithm is clearly problematic. For example, there can be multiple embeddings of the same graph, leading to multiple topological models of the unknown world. For example, for regular graphs every location is identical to every other and the number of possible models grows initially as $O(k!)$ for level k of the exploration tree (although keeping only one or two models is sufficient to express both all the structure that the robot has observed and all that it can accomplish given the limited percepts it has made). This initial explosive growth is reduced once the tree depth exceeds the vertex degree (i.e. very early for planar graphs). This difficulty is not surprising since under such circumstances we are attempting to construct a map from no knowledge about where we are or how we are moving – anything is possible. Thus the difficulty is not intrinsic to this algorithm but rather to the impoverished stimuli. In worlds where more information is available and various places are distinguishable the tree grows much more slowly and there are several significant sub-cases of interest. One notable class of worlds are those where a pair of uniquely distinguishable vertices exist. In such cases a single unique solution will be produced [Dudek *et al.*, 1993]. We are now using an implementation of our algorithm written in C to examine issues relating to the growth of the exploration tree and the solution space for different kinds of environments.

7 Acknowledgments

The idea of using an extended signature evolved partly from discussions with M. R. Jenkin, E. Milios, and D. Wilkes.

The authors gratefully acknowledge the financial support of the Natural Sciences and Engineering Research Council and the Centre de recherche informatique de Montréal.

References

[Almeida and Melin, 1989] R. De Almeida and C. Melin. Exploration of unknown environments by a mobile robot. *Intelligent Autonomous Systems 2*, pages 715–725, 1989.

[Basye and Dean, 1990] Kenneth Basye and Thomas Dean. Map learning with indistinguishable locations. In M. Henrion L. N. Kanal J. F. Lemmer, editor, *Uncertainty in Artificial Intelligence 5*, pages 331–340. Elsevier Science Publishers, 1990.

[Chatila and Laumond, 1985] Raja Chatila and Jean-Paul Laumond. Position referencing and consistent world modeling for mobile robots. *IEEE J. Robotics and Automation*, Vol. 1:138–145, 1985.

[Corneil and Kirkpatrick, 1980] D. G. Corneil and D. G. Kirkpatrick. A theoretical analysis of various heuristics for the graph isomorphism problem. *SIAM J. Computing*, 9(2):281–297, May 1980.

[Davis, 1986] Ernest Davis. *Representing and Acquiring Geographic Knowledge*. Morgan Kaufmann Publishers, Inc, Los Altos, California, 1986.

[Dudek *et al.*, 1991a] Gregory Dudek, Michael Jenkin, Evangelos Lilios, and David Wilkes. Robotic exploration as graph construction. *IEEETrans Robotics and Automation*, Vol. 7, NO. 6:859–865, 1991.

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

[Dudek *et al.*, 1993] Gregory Dudek, Paul Freedman, and Souad Hadjres. Signatures and extended signatures for topological mapping using graphs. TR-193-232-34, McGill Research Center for Intelligent Machines, April 1993.

[Kuipers and Byun, 1988] Benjamin J. Kuipers and Yung-Tai Byun. A robust qualitative approach to a spatial learning mobile robot. In *SPIE Advances in Intelligent Robotics Systems*, 1988.

[Leonard and Durrant-Whyte, 1991] J. J. Leonard and H. F. Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *IEEE Transactions on Robotics and Automation*, 7(3):376–382, 1991.

[Mataric, 1990] Maja J. Mataric. A distributed model for mobile robot environment-learning and navigation. Technical Report AI-TR 1228, MIT Artificial Intelligence Laboratory, Artificial Intelligence Laboratory, 545 Technology square Cambridge, MA 02139 USA, 1990.

[Schwartz and Yap, 1987] J. Schwartz and C.-K. Yap. *Algorithmic and Geometric Aspects of Robotics*. Lawrence Erlbaum Assoc., Hillsdale, N. J., 1987.