Robust Positioning with a Multi-Agent Robotic System

G. Dudek[1], M. Jenkin[2], E. Milios[2],and D. Wilkes[3]

[1]McGill Centre for Intelligent Machines, McGill University,
Montreal, Quebec, Canada

[2]Department of Computer Science, York University,
North York, Ontario, Canada

[3]Department of Computer Science, University of Toronto,
and Ontario Hydro Research Division, Toronto, Ontario, Canada

Abstract

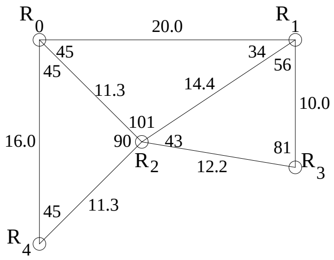
A collection of interacting autonomous robots can define a local coordinate system with respect to one another without reference to environmental features. This simplifies tasks requiring robots to occupy or traverse a set of positions in the environment, such as mapping, conveyance and search. We argue for an approach to positioning in which sensing errors remain localized, and dead-reckoning plays no role. This involves a robot-based representation for the environment, in which metric information is used locally to determine the relative positions of neighbouring robots, but the global map is a graph, capturing the neighbour relations among the robots. We show that many tasks can be solved without reference to a global coordinate system, but that global metric maps may be constructed as desired, with small errors in the vicinity of any chosen position.

1 Introduction

In this paper we deal with spatial representation using a collection of robots. In tasks that require an implicit or explicit long-range representation of space, dead reckoning is inadequate. Robots that attempt to use environmental landmarks have been proposed, but are highly dependent on the detectability of the landmarks. Here we present an approach to describing space by having a collection of robots determine their locations relative to one another. This allows for robust positioning *that is independent of the features of the environment.* Thus, algorithms may be constructed that are also independent of the particular environment occupied by the robots.

We assume a set of robots with the following properties, on the classification axes proposed in [1]:

- *Swarm size:* There are enough robots available that the number does not constrain our proposed solutions.
- *Communication range and topology:* Robots can only communicate with others that are nearby, in some sense (e.g. line-of-sight only).
- *Communication bandwidth:* Communication is cheap, relative to motion of one or more robots.
- *Swarm reconfigurability:* The individual robots may or may not be mobile. Mobility is not essential to all of the tasks mentioned herein.
- *Swarm unit processing ability:* The individual robots have general-purpose computers onboard (i.e. equivalent to Turing machines).



Adjacency matrix for E_d

$$\begin{bmatrix} 0 & 1 & .57 & inf & .80 \\ 1 & 0 & .72 & .50 & inf \\ 1 & 1.27 & 0 & 1.08 & 1 \\ inf & 1 & 1.22 & 0 & inf \\ 1 & inf & .71 & inf & 0 \end{bmatrix}$$

Adjacency matrix for E_a

$$\begin{bmatrix} 0 & 0 & 315 & inf & 270 \\ 0 & 0 & 34 & 90 & inf \\ 0 & 259 & 0 & 216 & 90 \\ inf & 0 & 81 & 0 & inf \\ 0 & inf & 315 & inf & 0 \end{bmatrix}$$

Figure 1: A sample configuration and its representation

- *Swarm composition:* The individual robots may differ in their physical capabilities and algorithms. In particular, each robot has, and is aware of, its own unique identifier.

In the next section, we describe our robot-based map representation. Section 3 shows how this representation may be used to solve a variety of robotics problems that involve a traversal of the space occupied by the robots. Section 4 shows how to get from our representation to a conventional Cartesian map, in which the errors are smallest near the position of interest in the map. Finally, we conclude with a discussion of future work.

2 Robot-relative mapping

Our problem is to use a configuration of robots deployed in a particular, finite 2-D space to map the space. The issue of how the robots came to be deployed in the space is the subject of a companion paper [2].

Let the set of robots be $\mathcal{R}$, where $\mathcal{R} = \{R_0, R_1, ... R_{n-1}\}$.

We let each robot define a unique local 2-D coordinate system in the space. The coordinate system used by robot R_i is a Cartesian system whose origin is at R_i and whose unit x-axis is the line segment joining robots R_i and R_{jmin}. R_{jmin} is the robot neighbouring R_i with minimum subscript value. The y-axis is chosen to be to the left of the positive x direction.

Data is exchanged among robots in each local neighbourhood to determine the positions of the neighbours of each robot. A robot is considered a neighbour of another if there is reliable communication between the pair.

Positions of interest in the space are described relative to the positions of the neighbours. Positioning errors in each local neighbourhood are independent of those in other neighbourhoods, because no reference is made to a global coordinate frame. Instead, only topological (graph-like) relationships are used to relate one neighbourhood to another.

Definition of the representation The map representation kept at each R_i is illustrated in figure 1. It consists of two graphs with labelled edges, $G_d(V, E_d)$ and $G_a(V, E_a)$. The two graphs are isomorphic, with $V = \mathcal{R}$, but have different edge labels. Each edge $e_{ij} \in E_d$ is given the weight ∞ if robots i and j are not neighbours. Otherwise, e_{ij} is the distance between R_i and R_j, in R_i's coordinate system. Each edge $e_{ij} \in E_a$ is similarly labelled ∞ if R_i and R_j are not neighbours, but otherwise is labelled with the direction from R_i to R_j, measured counter-clockwise from the x-axis of R_i's coordinate system.

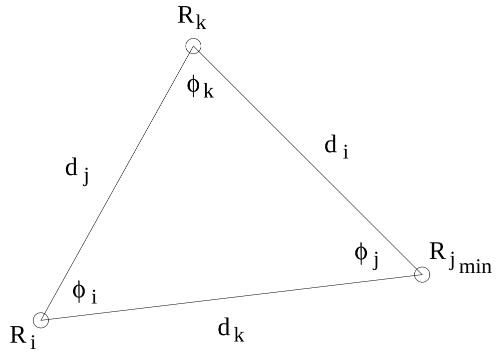


Figure 2: Finding parameters in the local neighbourhood

Construction of the representation Each robot R_i participates in the construction of the graph by first determining the set of robots with which it has reliable 2-way communication. Then, information is exchanged with the other robots in this set to determine the labellings of edges e_{ij}, for all j, in each of E_d and E_a. Finally, this information is propagated over the entire set of robots, to be integrated into the complete edge sets. The details of the necessary communications protocols are complicated, but part of the existing literature (e.g. [3]).

The details of the edge construction depend on what can be sensed by each robot. We consider two models, *neighbour distance only* and *neighbour azimuth (i.e. orientation) only*. The former corresponds to an ability to measure the signal propagation delay or phase shift with each robot in the neighbourhood. The latter corresponds to having line-of-sight communications, with direction-sensitive receivers. In both cases, our problem reduces to one of finding the unknown parameters in triangles such as the one shown in figure 2.

In the case in which only the distance to each neighbour is known, our unknown parameters are the angles ϕ_i, ϕ_j and ϕ_k, which may be found from the known parameters d_i, d_j and d_k.

In the case in which only the azimuth to each neighbour is known, our unknown parameters are the distances d_i, d_j and d_k. We may find only relative distances, since all similar triangles share the given values for ϕ_i, ϕ_j and ϕ_k. Setting $d_k = 1$ puts the distances in R_i-based coordinates.

3 Applications

The main reason for using the above representation instead of a global, metric map, is that there is the potential for more accurate localization of features of the environment. The location of a feature, like the positions of the robots, need only be determined relative to the closest robots in the space. No reference need be made to fixed landmarks in the environment. No use need be made of dead-reckoning to learn or return to a position in the space, since all positioning may be done via triangulation with multiple robots.

The problems of mapping an environment with some sensors, conveying objects from place to place, and searching for a lost object, all require that at least one robot pass near each point in the space.

Figure 3 describes in high-level terms, a simple algorithm for doing this traversal. Once again, the many details of the communications necessary to coordinate the robots have been omitted. The algorithm makes greedy choices of strategy at each step. As such, its performance is not optimal, but the algorithm serves as a demonstration of the principles underlying our map representation. The idea is to divide the space into non-overlapping triangles with a robot at each vertex. For each triangle,

```
1. Select a set of non-overlapping triangles among the
                    robot triangles found.
2. While there remain triangles to traverse
                    Do a serial traversal step as follows:
                              a. while possible to do so:
                                        i. select an untraversed triangle with
                                                  no corners selected to traverse other
                                                  triangles and at least one corner
                                                  free to move in this step
                                        ii. select a corner to traverse the
                                                  interior of the triangle
                                        iii. disallow the remaining corners
                                                  from moving in this step
                              b. do the selected traversals in parallel.
```

Figure 3: A traversal algorithm

one of the robots is chosen to traverse the interior, while the other two robots maintain fixed positions. This allows the moving robot to determine its position by triangulation with the other two. Interesting locations within the triangle may be stored as a pair of angles with the fixed robots. At all times, each triangle must have two fixed corners, in order that the third corner may resume its correct position following the traversal. This constrains the degree of parallelism in the traversals of different triangles.

Figure 4 illustrates the use of the algorithm on an example configuration of robots. The robots are labelled with integers, the triangles are labelled with letters. Triangles are shaded once they have been traversed. Five serial steps are required for the search of 22 triangles by 17 robots. Each step consists of the traversal, in parallel, of several triangles. The first four steps are shown in the figure. In each graph, circled nodes correspond to robots that are to move during the step; nodes enclosed in squares refer to robots that are constrained to be immobile during the step.

4 Conversion to a Cartesian map

Naturally, there exist applications for which a global, metric map is required. For example, if the robots are to report positions to another system that uses absolute position information, such positions must be reconstructed from our representation.

For this to be possible, two reference points are necessary to establish position and orientation with respect to the global coordinate system. If we assume that the positions of any two neighbouring robots are specified in the global coordinate frame we can derive (x, y) coordinates for all robots by iteratively propagating information outwards from the known positions, by repeatedly trangulating for a third, unknown, robot position using two known-position neighbours.

The errors in the coordinates will typically grow as the computation steps further from the fixed positions. If additional accuracy is required, more points may be specified to the algorithm in areas far from the initial points. The use of our representation does not preclude the use of other, environment-dependent position-fixing techniques.

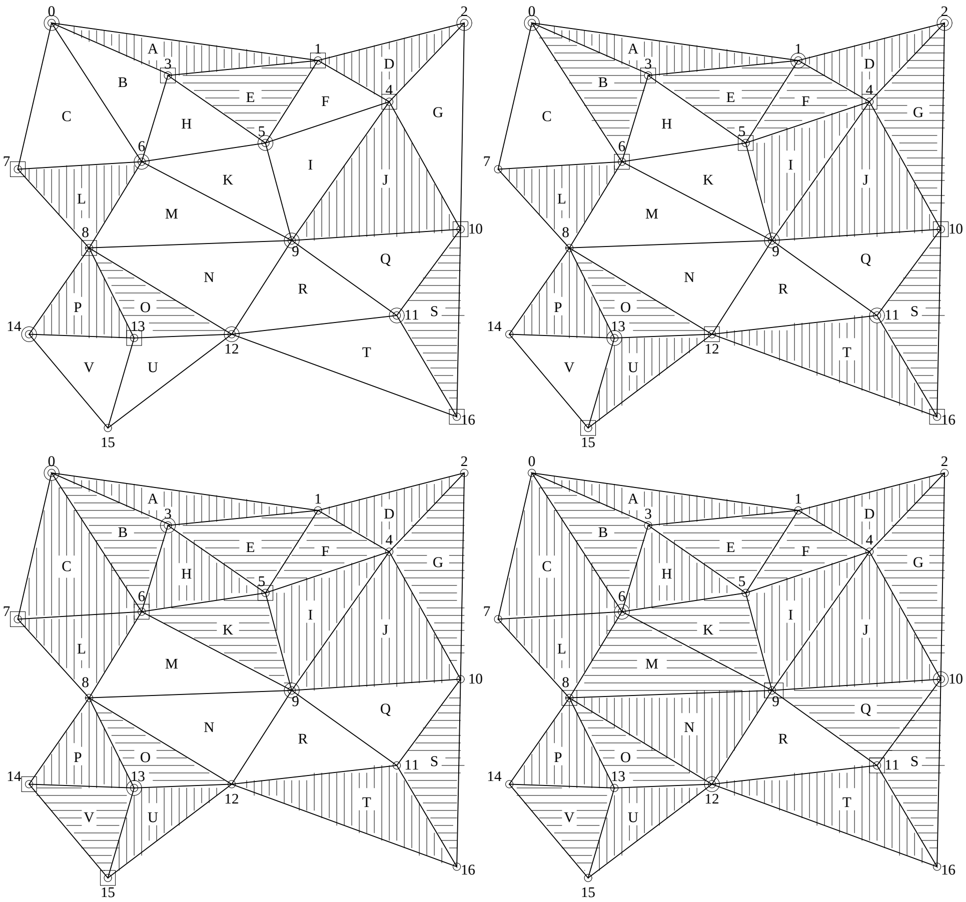


Figure 4: A traversal example

5 Conclusions and Future Work

We have presented an approach to spatial representation using multiple robots that removes any dependence on environmental landmarks. In many domains a major obstacle to the introduction of autonomous robots has been the need for sufficiently robust sensing to permit robust self-location. This often requires special-purpose designs. The approach we suggest here suggests that such difficulties may be alleviated in a general-purpose way by using multiple robots.

Several experimental issues remain to be explored. One significant issue we have not addressed is how mobility on the part of the individual robots could be exploited to obtain a more uniform covering of the environment. As well, there may exist better parallel traversal algorithms within this framework.

References

[1] Dudek, G., Jenkin, M., Milios, E. and Wilkes, D., "A Taxonomy for Swarm Robotics", *Proc. IROS*, to appear, 1993.

[2] Dudek, G., Jenkin, M., Milios, E., Wilkes, D., "On the utility of multi-agent autonomous robot systems," submitted to this workshop.

[3] Tanenbaum, A.S., *Computer Networks*, Prentice-Hall, Englewood Cliffs, New Jersey, 1981.