Developing a Benchmarking Framework for Map Building Paradigms

J.J. Collins, Malachy Eaton, Mark Mansfield, and David Haskett.
Dept. of Computer Science and Information Systems
University of Limerick, Limerick
Limerick, Ireland.
Email:[j.j.collins/mark.mansfield/david.haskett/malachy.eaton]@ul.ie

Shane O'Sullivan
IBM Software Group
Santry,
Dublin 9, Ireland.
Email:osushane@ie.ibm.com

Abstract

This paper presents a benchmarking suite to empirically evaluate map-building algorithms in the field of intelligent autonomous mobile robots. We suggest that one of the more crucial tasks currently facing researchers is the provision of a common task, or set of tasks, as a means of evaluating different approaches to robot design and architecture, and the generation of a common set of experimental frameworks to facilitate these different approaches. Map building is seen as a key component in the evolution of robot control software architecture, and forms the basis upon which higher order behaviours such as pursuit and evasion can be designed. This paper outlines a set of benchmarks for map-building and briefly evaluates their usefulness across a sample set of paradigms.

1. Introduction

In the field of mobile robots, a variety of simultaneous localisation and map-building (SLAM) approaches can be adopted for model construction. The maps themselves generally fall into two categories [1]:

- Metric maps, in which the environment is decomposed into a 2D or 3D lattice of grids or cells, with each cell representing the probability of occupancy, as depicted in figure 2 for an environment shown in figure 1.
- Topological maps are graph-oriented models with nodes representing structural features of the environment, and vertices capturing adjacency and ordering information, as depicted in figure 3. Topological maps are computable from a metric map.

A considerable range of map building paradigms have been documented in the literature, ranging from probabilistic and Bayesian frameworks, to artificial neural networks, to the Cartesian symbolic-oriented approach. While empirical results of individual paradigms have been presented consistently, very little work has been done on developing a benchmarking framework to cross compare and contrast sets of map-building algorithms.

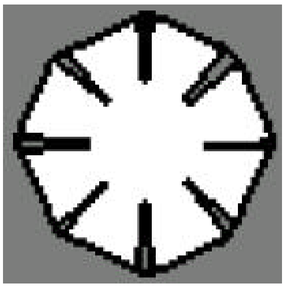
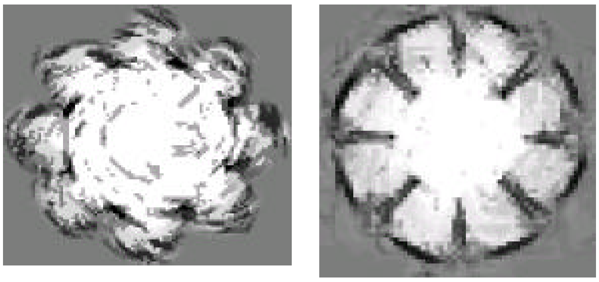


Figure 1. The ideal map of a star environment.



(a) (b)

Figure 2. Occupancy grid maps generated for figure 1.

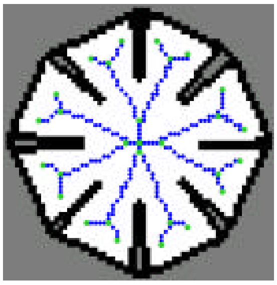


Fig 3. Star ideal map with inscribed Voronoi graph of all possible paths in the environment.

Map-building in static environments is difficult due to the following:

1. Specular reflections from sonar range sensors introduce and propagate noise into the model.

2. Map-building relies on localisation. Dependence on odometry is insufficient in itself to guarantee self-localisation.
3. The selection of the sonar model itself impacts significantly upon the paradigm used to compute occupancy values.
4. Independence assumptions introduced into some of the paradigms for reasons of computational tractability are only approximations of the true theoretical model.

These difficulties are depicted in figure 2, illustrating the difference in quality of the maps generated by alternative paradigms when a robot is deployed in an environment that corresponds to that shown in figure 1.

The evaluation of ideal versus generated maps based on correlation techniques from the image processing domain focus primarily on a cell-by-cell comparison. These metrics fail to take into account that [2]:

1. Maps are artefacts primarily used for navigation
2. Discrepancies between occupancy values in ideal and generated maps will have varying degrees of significance.

This paper presents a benchmarking framework for map-building based on:

1. Baron's cross correlation coefficient [3].
2. Martin and Moravec's map score [4].
3. An extension of Thrun's [5] loss metric based on the computation of Voroni diagrams from the ideal map, estimation of the universe of unique paths, and determining their feasibility in the generated maps.

2. In Benchmarking

One of the more important tasks currently facing researchers in the fields of artificial life and embodied intelligence is the provision of common benchmarks for performance evaluation. Current benchmarks, while useful, have their problems. We advocate the generation, from the bottom up of a common set of experimental frameworks, for performance evaluation and benchmarking of bio-inspired robots.

A current *de facto* standard in this field is RoboCup annual challenge. RoboCup operates in four categories: simulated teams, a small size league, a middle size league, and legged robots. An example small size robot is Khephera, a typical middle sized robot is Pioneer (1/2) and The Sony artificial dog fits in the third category. There is also a humanoid league. RoboCup tests skills at the individual, the inter-individual, the collective and at the competition level. [6]

Individual skills to be mastered include navigation and localisation on the field of play, including the selection of optimal paths. Inter-individual skills include the coordination of movements with playing partners in order to pass accurately. At the top level the tasks of strategy generation and recognition of opponents' strategies are crucial.

Criticisms of RoboCup stem from the controlled environment in which the robots operate, and the fact that soccer playing skills are quite specific and may lead to the development of highly focused robots of little use for any other task. Also, the self-localisation problem is somewhat constrained by the used of landmarks that completely reduce dependence on dense sensor matching-oriented paradigms. Also, considering the complexity of the task facing the robot teams, how does one assign credit to individual components within the software architecture, given that there is only one reinforcement signal per episode – win or loss.

So while RoboCup may currently be a useful testing bed for approaches to Artificial Intelligence and Artificial Life, problems exist. One potential approach, which we espouse, involves the provision of a set of specifically designed experimental frameworks, and involving tasks of increasing complexity, rigorously defined to facilitate experimental reproducibility and verification.

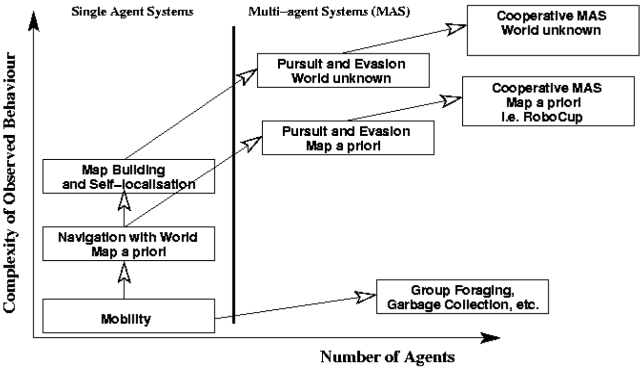

Single Agent Systems
Multi–agent Systems (MAS)
Cooperative MAS World unknown
Pursuit and Evasion World unknown
Cooperative MAS Map a priori i.e. RoboCup
Pursuit and Evasion Map a priori
Map Building and Self–localisation
Navigation with World Map a priori
Mobility
Group Foraging, Garbage Collection, etc.
Complexity of Observed Behaviour
Number of Agents

Figure 4 Benchmarking framework.

The authors advocate, and shown in figure 4, that issues of map-building and self-localisation are fundamental to the progressing robotic-based research [7]. It is also argued that that pursuit and evasion should be undertaken prior to RoboCup type applications, primarily because the modelling of behaviour and interaction can be constrained to just two players. Secondly, pursuit and evasion have strong biological and game theoretic foundations, and thus provide a framework in which scientific modelling of the system can be performed, and later validated through experimentation [8]. Figure 4 strongly hints that current robot competitions should be focused on map building and, pursuit and evasion with a map a priori.

3. Metrics for Map Building Paradigms

3.1 Cross Correlation Between Maps

One possible fitness measure of the map generated by the system is calculated using Baron's *cross correlation coefficient* [1]:

$$C_N(y) = \frac{\langle I_T T\rangle - \langle I_T\rangle\langle T\rangle}{s(I_T)s(T)}$$

This is based on *template matching*, where $C_N(y)$ is the cross correlation coefficient, I_T is the map to be matched, T is the original map being matched against, and <> is the average operator. There are two fundamental problems with this approach:

1. Maps must be anchored precisely along the same coordinate system.
2. This metric makes no allowance for freespace readings in the ideal map that are occupied in the generated, and vice-versa.

3.2 Map Score

Martin and Moravec [39] developed a map comparison measure called *Map Score* in order to facilitate the automatic learning of sensor models. Unlike correlation, map scoring compares two maps on a cell-by-cell basis. Given two maps, M and N, the score between them is calculated as the sum of the squared differences between corresponding cells:

$$Match = \sum_{m_{X,Y}\in M, n_{X,Y}\in Y} \left(m_{X,Y} - n_{X,Y}\right)^2$$

where $m_{X,Y}$ is the value of the cell at position (x,y) in map M and likewise for $n_{X,Y}$.. This formula is equivalent to the well-known sum of squared error metric used widely in machine learning paradigms. This formula works well with the application for which it was designed, learning a sonar model, since the only requirement is to minimise the score/fitness on a single map.

In order effectively use the map score metric as a benchmark, it is necessary to normalise it. The ideal map only has three possible values, 0, 0.5, and 1, or empty, unknown and occupied respectively. A naïve method of finding the worst map would be to set each empty cell to 1, each occupied cell to 0, and each unknown cell to either 0 or 1. One could then compute the map score between the ideal and worst possible map, and use this as a normalising factor. However, this still does not take cognizance of the fact that there are places in the map that the robot cannot go, for example there may be a large expanse of unknown area with no access to it. It therefore is logical to create the worst possible map of an environment based on where the robot *can* go. This is done by setting the empty cells to 1, as above, but only change the value of an unknown or occupied cell if it is proximal to an empty cell and is detectable from that empty cell.

Another weakness of the *Map Scoring* technique is that mapping algorithms that overestimate the empty regions of space. To redress the balance, a second *score* test is used that only compares the occupied cells in both maps. For any two maps M and N, if either the value $m_{X,Y} > 0.5$ or $n_{X,Y} > 0.5$, then the squared difference between those two cells is added to the final score. Otherwise they are ignored.

3.3 Path Analysis

Most quantitive analysis presented on the grid maps focus on techniques similar to those discussed in the earlier part of this chapter – namely comparing the values in the cells to each other. When training a sonar model in a small area this is quite a good approach to take, but to properly evaluate the worth of a map, one must get away from the idea of a map being an isolated The purpose of a map is, after all, to enable a robot to get from point *A* to point *B* as quickly and as safely as possible. However, many researchers ignore the fact that a map need not *perfectly match* the ideal map of an environment to be *perfectly usable* in that environment. It is not necessary for a map to be an exact replica of the surrounding environment, it is just necessary for it to be an abstraction of that environment which, when combined with an appropriate path planning algorithm, generates a true real world path for the robot to follow. It is the quality of these paths that truly give the value of the map, a value based on the use to which the map will be put rather than a metric based on techniques that can be seen as an extension of human vision such as map matching and image correlation. To get the true worth of a map, two elements of the map must be tested:

1. The degree to which the robot should be able to plan a path from one position to another using the generated map, but cannot – false negatives.
2. The degree to which paths calculated in the generated map would cause the robot to collide with an obstacle, and are therefore invalid – false positives.

Both of these are tested using only the map to guide the robot's movement.

There are four steps required to calculate the two above items of information from a generated map.

- **Step 1**: Calculate all possible paths, in the ideal map P^I, by generating a Voronoi graph. Record the endpoints of each path, E^I. For each pair of endpoints in E^I that have a path between them in the ideal map, attempt to create a path in the generated map between those two endpoints using a path-planning algorithm.

- **Step 2**: Count the number of paths between endpoints in E^I that could not be completed in the generated map due to obstacles existing in that map where they do not exist in the ideal map – false negatives. The percentage of false negatives is then:

$$\frac{\sum \text{incomplete paths in generated map}}{\sum \text{edges in the Ideal map}}$$

- **Step 3**: Calculate all possible paths in the generated map, once again using a Voronoi diagram. Superimpose each Voronoi edge from the generated map onto the ideal map, and count the percentage of edges that pass through occupied spaces, and would therefore cause the robot to crash – false positives.

4. Results

The following three occupancy-grid map-building paradigms and extensions were evaluated using the benchmarking suite:

- Moravec and Elfes probability-based framework with a Gaussian sonar model [9].
- Mathias and Elves Bayesian-update framework with a Gaussian sonar model [10].
- Konolige's Bayesian update framework with Normal-distribution sonar model, pose buckets and dynamic mixture models [11].

The extension also evaluated is O'Sullivan's linear feature predictor [2] used to identify specular reflections. The results demonstrate that path analysis approach is a more intuitive and reliable metric that captures the accuracy and reliability of the maps generated. Path analysis clearly demonstrated that a framework that incorporates a Normal distribution sonar model, pose buckets, and linear feature prediction clearly outperforms all other combinations.

5. Discussion and Conclusion

The benchmarking suite is now being extended with metrics that capture the difficulty posed by an environment for map-building paradigms. These metrics are based on cycles in the environment, total path distance, number and acuteness of corners responsible for specular reflections, etc.

.

In addition to this, our current focus is on extending the benchmarking suite to robot pursuit and evasion. Typical metrics captured are tag rates, and some recent work has focused on the informational aspect to PE based on a measure of entropy. These metrics are behavioural in that results are based on observation of external behaviour. It is difficult in PE contests there to relate behaviour to the underlying mechanics responsible for the agent's actions. This is further exacerbated by virtue of the fact that an agent is deployed in a world characterised by uncertainty. Current work is focused on introspective metrics for PE that facilitates the assignment of credit to paradigms responsible for the generation of behaviour.

[1] Thrun, S. (2002). Robotic Mapping: A Survey. CMU Technical Report CMU-CS-02-111, Carnegie Mellon University, Pittsburgh, Pennsylvania.

[2] O'Sullivan, S. 2003. An Empirical Evaluation Of Map Building Methodologies in Mobile Robotics Using The Feature Prediction Sonar Noise Filter And Metric Grid Map Benchmarking Suite. MSc Thesis, University of Limerick, Ireland.

[3] Baron, R. J. (1981). Mechanisms of human facial recognition." International Journal of Man Machine Studies 15: 137-178.

[4] Martin, M. C., Moravec, H.P. (1996). Robot Evidence Grids. CMU Technical Report CMU-RI-TR-96-06, Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania.

[5] Thrun, S. (1998). Learning Metric-Topological Maps for Indoor Mobile Robots. Artificial Intelligence 99(1):21-71, February 1998.

[6] Kitano H, Asada M, Kuniyoshi Y., et al (1995) RoboCup: The Robot World Cup Initiative, IJCAI-95 Workshop on Entertainment and AI/Alife, 1995.

[7] Eaton, M. Collins, J., and Sheehan, L (2000). Towards a benchmarking framework for research into bio-inspired hardware-software artefacts. In Proc. of the 5th Int. Symp. On Artificial Life and Robotics, Oita, Japan.

[8] Mansfield, M, Collins, J.J., Eaton, M., Haskett, D. (2004). Developing a Statistical Baseline for Robot Pursuit and Evasion using a Real-World Control Architecture. To appear in Procs. Of the 9th Int. Sym. On Artificial Life and Robotics, Oita, Japan.

[9] Moravec, H. P., Elfes, A. (1985). High Resolution Maps from Wide Angle Sonar. In Procs. of the 1985 IEEE Int. Conf. on Robotics and Automation.

[10] Matthies, L., Elfes, A. (1988). Integration of Sonar and Stereo Range Data Using a Grid-Based Representation. In Procs. of the 1988 IEEE Int. Conf. on Robotics and Automation.

[11] Konolige, K. (1997). Improved Occupancy Grids for Map Building. Autonomous Robots 4:351-367.