Cooperative Localisation and Mapping

Andrew Howard and Les Kitchen
Department of Computer Science and Software Engineering
University of Melbourne, Parkville, 3052, Australia
fax: 61 3 9348 1184, email: a.howard@cs.mu.oz.au

Abstract

Recently, many authors have considered the problem of simultaneous localisation and mapping (SLAM). The paper addresses a somewhat different problem, that of *cooperative* localisation and mapping (CLAM). Basically, the CLAM approach involves two or more robots cooperating to build a map of the environment. This cooperation is not aimed at simply increasing the *speed* with which the map is constructed; rather, it is aimed at increasing the *accuracy* of the resultant maps. This paper describes some early work aimed at validating the CLAM concept.

1 Introduction

Recently, many authors have considered the problem of simultaneous localisation and mapping (SLAM). The paper addresses a somewhat different problem, that of *cooperative* localisation and mapping (CLAM).

The aim of SLAM, as opposed to CLAM, is to build a map of an unknown environment and simultaneously localise the robot with respect to this map. The map might be relational, or it might be defined with respect to some coordinate system (the latter is more common). If the robot has access to some kind of global localisation sensor, such as satellite GPS, this task is not very difficult. The processes of localisation and mapping are effectively de-coupled – the robot can use the GPS sensor to determine its position, and use other sensor readings to construct the map. Unfortunately, the robots used in many applications (such as service robots, mining robots, underwater vehicles, and so on) do not have access to this kind of sensor. They must instead rely on a mixture of odometry (or inertial navigation or dead-reckoning) and landmark-detection. For these robots, the processes of localisation and mapping are strongly coupled – to determine its location, the robot must have a map, but to build the map, the robot must first know its location. This strong coupling is one of the factors that makes SLAM difficult.

When constructing a SLAM system, one must be aware of two key problems.

- Odometry is subject to cumulative drift.
- Landmarks can be ambiguous.

Consider, for example, a robot that starts out near a landmark such as a doorway, and takes an extended journey around the environment. Eventually, it arrives at a doorway again. If there is any significant odometric drift (as there is bound to be, if the environment is large enough), it will be difficult for the robot to tell whether it has arrived at the original doorway, or a different one. Without this information, the robot cannot construct an accurate map.

The concept of CLAM can be used to address both of these problems. Imagine a group of robots moving through an unknown environment. These robots can operate much like a team of surveyors mapping out an area, or a squad of soldiers moving through a hostile environment:

- The robots can reduce odometric drift by 'watching' each other.
- The robots can resolve landmark ambiguity by acting as landmarks themselves.

Consider, for example, a scenario involving two robots. At any given point in time, only one robot is allowed to move (the explorer). The other robot (the observer) watches the explorer and estimates its relative position. This estimate is combined with the explorer's own estimate (based on odometry) to obtain an estimate that is more accurate than that obtained with odometry alone. When the distance between the robots becomes large, or when the robots are about to become occluded, the robots swap roles – the explorer becomes the observer and vice-versa. By proceeding in this fashion, the robots can greatly reduce the uncertainty in their location, and hence produce a much more accurate map. Furthermore, if the robots detect a pair of landmarks that may or may not be the same, they can act cooperatively to resolve the ambiguity. For example, one robot can stay with the first landmark, while the other sets out for the second. If the robots subsequently meet up, they know that there is in fact only one landmark. If they fail two meet, they know that there are indeed two distinct landmarks.

This paper describes some early work aimed at exploring and validating the CLAM concept. This paper outlines the theoretical foundations of the CLAM concept, describes the basic implementation and presents some preliminary experimental results. It considers only the first of the problems described above – how a pair of robots may coordinate their activities to reduce odometric uncertainty whilst exploring an unknown environment. The key claim made in this paper is that the maps produced in this way are more accurate than those obtained using odometry alone.

2 Related Work

While the problem of robot map building has received much attention recently [9, 6, 2, 7], few authors have explored the possibility of employing multiple robots for this task. Notable exceptions are the work of Yamauchi [11], Barth and Isiguro [1], and Yagi *et al* [10], all of whom describe approaches to map building using multiple robots. In each case, however, the emphasis is on increasing the *speed* with which maps are built, not the accuracy. For a more comprehensive review of these techniques (among others), see [5].

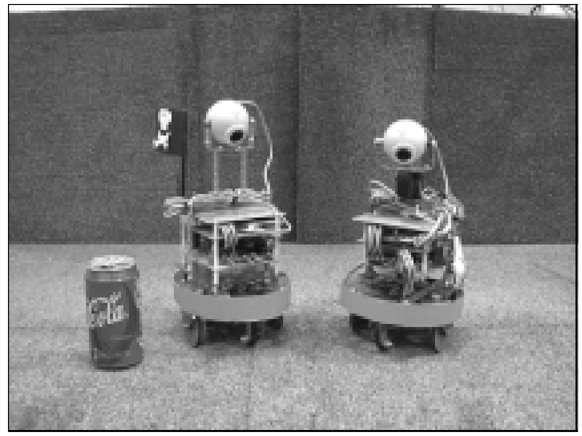

Figure 1: Tigger I and Tigger II. The robots can locate each other using the coloured strips around each robot's base (the strips are bright orange).

3 Theory and Implementation

The CLAM approach has been implemented on a pair of small mobile robots – Tigger I and Tigger II (see Figure 1). Each of these robots is capable of making odometric measurements, and is equipped with a colour camera. The robots can recognise each other using coded tags, and can distinguish between obstacles and the floor on the basis of colour.

The robots are in constant communication with a host PC that coordinates their activities. The host must manage three cooperative processes: localisation, map building and exploration. We will consider each of these processes in the following sections.

3.1 Localisation

Two sources of data are available to help determine the robots' locations:

- Odometric data, which allows each robot to estimate its own position.
- Visual data, which allows the robots to estimate their position relative to each other.

Fortunately, these two sources of data complement each other well: by combining odometry with vision, it is possible to obtain an estimate of the robots' loca-

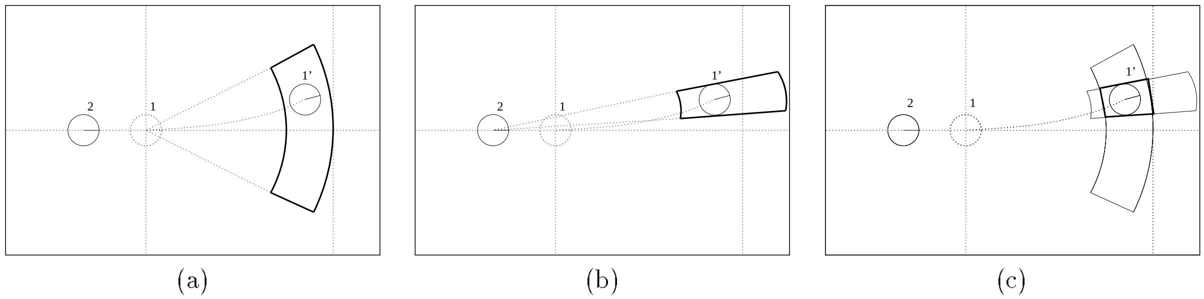

Figure 2: Localisation example. The first robot has moved approximately one meter. The uncertainty in the robot's subsequent location is indicated by the heavy polygon. (a) Uncertainty when using odometric data only. (b) Uncertainty using visual data only. (c) Uncertainty using odometry and vision together.

tion that is better than that obtained using odometry or vision alone.

Consider, for example, the series of images shown in Figure 2. The first image shows the distribution of possible robot locations for a robot that has moved 1m from its starting point. This distribution is based entirely on odometric measurements and reflects the uncertainty associated with these measurements. Note that there is relatively little uncertainty in the *distance* the robot has moved; there is, however, a great deal of uncertainty in the *direction* it has moved. This kind of distribution is typical of twin-drive-wheel robots, such as Tigger I and Tigger II.

The second image shows the distribution of possible robot locations for the same scenario, but this time using visual data obtained from the second robot. This time, there is relatively little uncertainty in the *bearing* of the first robot (relative to the second), but a great deal of uncertainty in its *range*. This uncertainty in arises from the fact that, for a robot with a single camera, the range must be determined from perspective. A small uncertainty in the position of the robot in the image will therefore correspond to a large uncertainty in its range.

Th final image shows the distribution of possible robot locations for combined odometric and visual data. Note how the two forms of data complement each other: the robot must lie at the intersection of the two distributions, which yields a very good estimate of the robot's location.

A detailed description of the theory underlying the cooperative localisation mechanism is, unfortunately, beyond the scope of this paper. A full description can be found in [5].

3.2 Map Building

Maps are built using visual data obtained from the robots' cameras. A simple colour-based segmentation routine is applied to the raw images to distinguish between obstacles and the floor (the floor is assumed to have a constant colour). The ground plane constraint is then used to infer the range-and-bearing of obstacles. In this way, each camera acts as a kind of 'virtual' range sensor [4], whose output is similar to that obtained by a (not very accurate) laser range finder.

The range-and-bearing data is used to form a global occupancy map [3]. This is a simple grid-based map in which cells may be in one of three states: occupied (meaning there is an obstacle at this location), unoccupied and unknown. Bayesian inference is used to determine the occupancy state of each cell, based on the virtual range data provided by each robot. Data from both robots is fused to form a common map. A detailed description of this process can be found in [5].

3.3 Exploration

In choosing an exploration strategy for the robots, one must consider two factors: speed and accuracy. In this paper, we will ignore the issue of speed and instead focus entirely on accuracy. Unfortunately, at this early stage of research, the optimal strategy is far from clear. We will therefore consider two different strategies, and assess their relative merits.

The two strategies in question are called the O1 and O2 strategies. They are defined as follows.

- **O1**: Each of the robots is assigned the role of either *explorer* or *observer*. While the explorer sets out to investigate the environment, the observer sits still, watching the explorer. When the explorer is about to leave the field-of-view of the observer (or is about to become occluded) the explorer stops and waits for the observer to reposition itself. The exploration then commences once again. Ideally, with this strategy, the explorer always remains visible to the observer.

- **O2**: The O2 strategy is identical to the O1 strategy, with the exception that the robots take turns at being observer and explorer. When the explorer is about to leave the field-of-view of the observer, the two robots swap roles: the explorer now becomes the observer and vice-versa. With this strategy, the robots effectively 'leapfrog' their way around the environment.

Note that both of these strategies are examples of what human surveyors call an *open traverse* [8]. An open traverse is one in which there are no fixed reference points, and as a result, open traverses are prone to cumulative errors. For this reason, human surveyors tend to eschew open traverses and instead make use of *closed traverses*. In a closed traverse, all measurements are associated, either directly or indirectly, with one or more fixed reference points. In surveying, closed traverses are the norm, since they minimise the effects of cumulative errors.

Within the context of CLAM, it is possible to design strategies based upon closed traverses. For example, one of the robots can act as the fixed reference point, while the other sets out to explore the environment. The explorer can periodically return to the first robot to perform a visual check of their relative position. Unfortunately, there are a number of difficulties associated with such strategies. First and foremost amongst these is that, unlike human surveyors, the robots do not know the topology of the environment *a priori*. Designing an appropriate sequence of closed traverses is therefore quite difficult. It is for this reason that we have concentrated on strategies based upon open traverses; we believe these strategies can be implemented in a fairly straight-forward fashion.

Note that, to date, neither of the strategies described above has actually been implemented. Instead, experiments have been conducted using simple scripts composed by a human operator. We hope to proceed with actual implementation in the near future.

4 Experiments

This section presents the results of some preliminary experiments aimed at validating the CLAM concept. All of these experiments were conducted in simulation, and for a single environment; consequently, the results should treated with some caution.

The experimental method is as follows. A pair of robots is placed in a (simulated) environment consisting of a simple room with a box in the middle. The robots are allowed to explore this environment, building a global occupancy map in the process. The robots follow a simple script containing a sequence of basic movement commands, such as: 'move forward', 'turn left', 'turn right', 'wait', and so on. The script is composed by a human operator, and different scripts are used for each of the navigation strategies O1 and O2. Since this is a simulation, the effect of different kinds of errors can be investigated. For example, we can evaluate the performance of a each strategy in the presence varying amounts of odometric noise. The results of these experiments are summarised in Figure 3.

Figures 3(a) and 3(b) show the results for the O1 and O2 strategies respectively. The first map in each row was produced using perfect odometry, and hence should be regarded as the ideal or control case. The dark regions of the map correspond to areas that probably contain obstacles (the darker the area, the more likely it is to be occupied). Note that the map is less than perfect – these imperfections arise from the fact that while the simulated odometry may be flawless, the simulated vision is not. The simulation contains a ray-tracing algorithm for generating images as they would appear to the real robot. While the 'virtual' range sensor described in Section 3.2 is very good at locating obstacles in these images, the

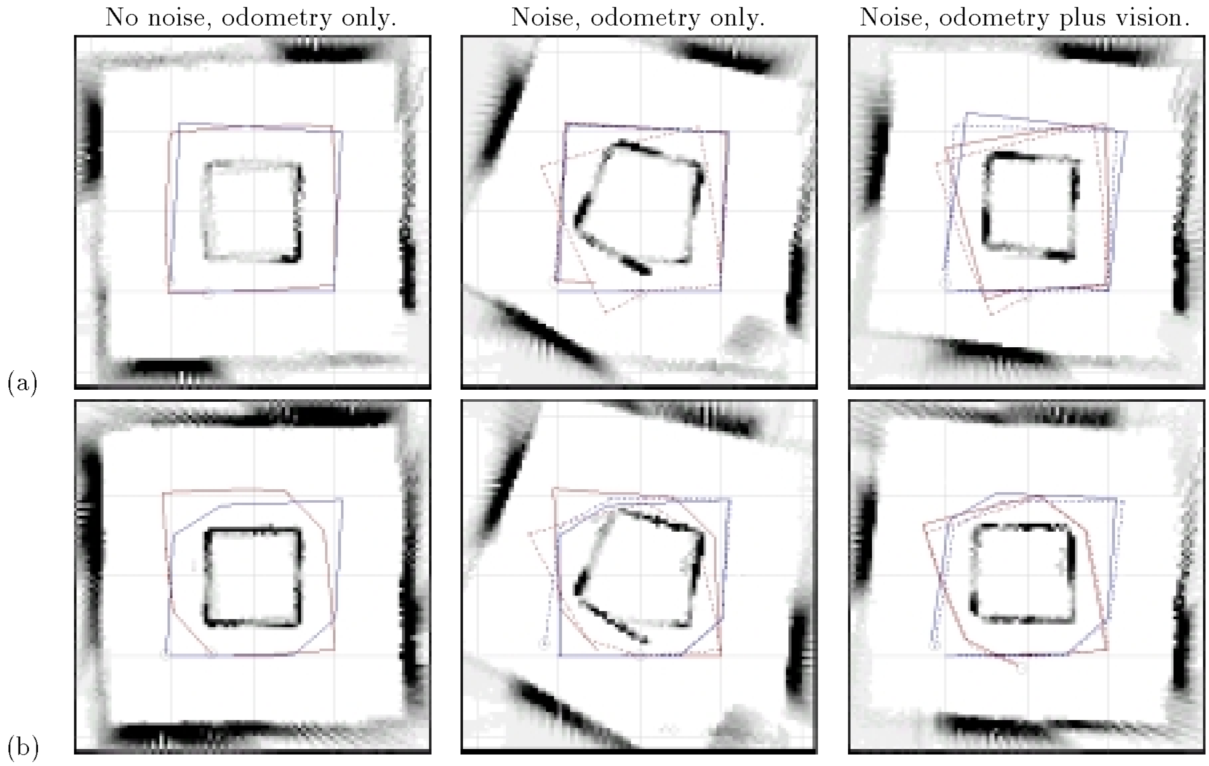

Figure 3: Experimental results. The dotted lines show the actual path of the robots; the solid lines show the estimated paths. (a) Results using strategy O1. (b) Results using strategy O2.

actual range of the corresponding obstacle must be inferred from the ground-plane constraint. There is an inherent uncertainty associated with this step – a single pixel in the image may correspond to quite a large area on the ground.

The second map in each row was produced using less-than-perfect odometry; specifically, a small noise term was added to the odometric measurements. Furthermore, this map was produced using odometry alone; that is, visual measurements where not used for localisation. In this map, the dotted lines show the actual paths followed by the robots; the solid lines show the estimated paths. Note that the noise term used in the simulation has both systematic and stochastic components: the systematic component represents uncertainties associated with the physical dimensions of the robot (such wheel size), while the stochastic component represents uncertainties associated with the physical properties of the environment (such as uneven floors). Empirically, we have found that both terms are required to accurately simulate the properties of real odometry.

The third map in each row was produced under the same conditions are the second, but this time using both odometric and visual data for localisation. That is, these are the maps produced using the full CLAM approach.

Looking at the maps produced using odometry only, the effect of the noise term is readily apparent – there is a substantial disagreement between the actual path followed by the robot and the estimated path. As a result, the maps appear somewhat 'bent'. This is true for both strategies. However, when one includes visual data in the localisation process, as in the final set of maps, one observes a dramatic improvement in the results – the disagreement between the actual and observed paths is greatly reduced, and, as a result, the maps are much closer to the ideal.

Thus, the CLAM approach appears to be working as intended.

Interestingly, the O2 strategy appears to produce better results than the O1 strategy. This is not surprising if one considers the nature of the constraints that are generated by each strategy. When the observer watches the explorer, it generates a strong constraint on the explorer's position (relative to the observer), but has nothing to say about the explorer's orientation. On the other hand, the observer generates a strong constraint on its *own* orientation (relative to the explorer), but only a weak constraint on its own position. In the O1 strategy, where the robots have fixed roles, this means that errors can accumulate in the *orientation* of the explorer and in the *position* of the observer. In the O2 strategy, where the robots swap roles, there exist strong constraints on the position and orientation of *both* robots. Hence, there is less opportunity for errors to accumulate.

5 Conclusion

The key conclusion to be drawn from the experiments presented in the previous section is that, using CLAM, one can generate better maps than are possible using odometry alone. Furthermore, the experiments indicate that the quality of the results is quite sensitive to the exploration strategy used.

Clearly, much work remains to be done. In particular, the system needs to be tested on real robots in a range of environments, and the exploration strategies described in Section 3.3 need to be implemented properly. Also, while the discussion in the paper has considered the case of two robots only, there is no reason why the overall approach could not be generalised to larger numbers of robots. We suspect (without proof), that the quality of the results will improve significantly as the number of robots is increased.

References

[1] M. J. Barth and H. Ishiguro. Distributed panoramic sensing in multiagent robotics. In *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 739–746, 1994.

[2] J. A. Castellanos, J. M. Martinez, J. Neira, and J. D. Tardos. Simultaneuous map building and localization for mobile robots: a multisensor fusion approach. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 1244–1249, 1998.

[3] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6), 1989.

[4] A. Howard and L. Kitchen. Fast visual mapping for mobile robot navigation. In *Proceedings of the IEEE International Conference on Intelligent Processing Systems*, pages 1251–1255, 1997.

[5] A. Howard and L. Kitchen. Cooperative localisation and mapping: Preliminary report. Technical Report TR1999/24, Department of Computer Science and Software Engineering, University of Melbourne, 1999.

[6] J. J. Leonard and H. F. Durrant-Whyte. Dynamic map building for an autonomous robot. *International Journal of Robotics Research*, 11(4):286–298, 1992.

[7] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4(4):333–349, 1997.

[8] F. H. Moffitt and J. D. Bossler. *Surveying*. Addison-Wesley, tenth edition, 1998.

[9] S. Thrun. Learning metri-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1):21–71, 1998.

[10] Y. Yagi, S. Izuhara, and M. Yachida. The integration of an environmental map observed by multiple robots with omnidirectional image sensor COPIS. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2, pages 640–647, 1996.

[11] B. Yamauchi. Frontier-based exploration using multiple robots. In *Proceedings of the International Conference On Autonomous Agents*, pages 47–53. ACM, 1998.