

# Department of Electrical and Computer Systems Engineering

## Technical Report MECSE-3-2007

Simultaneous Localisation and Map Building: The Kidnapped  
Way

D. J. Spero and R. A. Jarvis

**MONASH**  
UNIVERSITY

# Simultaneous Localisation and Map Building: The Kidnapped Way

Dorian J. Spero, *Member, IEEE*, and Ray A. Jarvis, *Fellow, IEEE*

Two roads diverged in a wood, and I –  
I took the one less traveled by,  
And that has made all the difference.

Robert Frost, The Road Not Taken

**Abstract**—Most approaches to simultaneous localisation and map building (SLAM) are based on rigorous models of the robot’s locomotive mechanism, sensor errors and environment. This motif inherently translates into a long string of simplifying assumptions and contrivances, which run counter to real-world operation. In this paper, a novel approach to SLAM is proposed that makes the pessimistic assumption that the robot is being continuously kidnapped over time. The robot’s locomotive mechanism and traversal irregularities are therefore irrelevant. There is no vehicle model and associated dead-reckoning information, and no assumption of continuity in the robot’s motion. Thus, SLAM can be implemented as a standalone device in a similar manner to the Global Positioning System (GPS); providing the robot, however complex or unpredictable, anonymity.

The presented approach comprises a landmark detection algorithm for extracting arbitrary environmental features (not necessarily structured); a multiple-hypothesis data association algorithm for recognising landmarks perceived from different viewpoints; and a qualitative error algorithm for representing and handling the positional uncertainties of the robot and landmarks. Practical results were gathered from several outdoor experiments using a vehicle equipped with a scanning laser rangefinder. This paper argues that the kidnapped way, the road only now traveled, facilitates navigation in natural environments.

**Index Terms**—Mobile robots, autonomous navigation, SLAM, position measurement, kidnapped robot problem.

## I. INTRODUCTION

**S**IMULTANEOUS localisation and map building (SLAM) is the dual process of building a feature based map of the environment and using a subset of features in this map, often termed landmarks, to estimate the robot’s absolute pose. SLAM is fundamental to mobile robot autonomy, especially for navigating efficiently and purposefully in an *a priori* unknown environment. However, it has proven to be a complex problem due to three forms of uncertainty: *data association uncertainty*, *navigation error* and *sensor noise* [1].

A number of approaches to the SLAM problem have been proposed, each with relative strengths and weaknesses. Most of these approaches are probabilistic in nature and are effectively simplifications of *Bayes filter* [2]; the most popular being the *estimation-theoretic* or *extended Kalman filter (EKF) based approach* [3]–[5].

The EKF approach provides a theoretically sound solution to SLAM and a means of systematically studying its convergence properties, evolution of the map and propagation of positional uncertainties. In practice, however, the approximation errors caused by linearising the system and measurement functions can lead to filter instability and an inconsistent map [6]. Also, the assumptions of Gaussianity and independence of model errors and perfect data association may not hold true. The risk of the latter assumption being violated increases as environmental clutter or uncertainty in the robot’s estimated state grows, especially when using the *gated nearest-neighbour (NN) algorithm* [7]. Although *multiple hypothesis tracking (MHT)* [8], [9] can provide more robust data association, it increases the computational complexity.

Arguably the biggest problem with the EKF is its reliance on stringent models to support its predictive behaviour. This reliance means that its operational performance is largely dependent on the extent to which the robot and its environment are predisposed to the modeling process. This invariably leads to exclusivity. The types of robots and environments that cannot be easily modeled or manipulated are often avoided, and those that can are tightly bounded with little tolerance for the unknown.

The *expectation maximisation (EM)* approach, proposed in [10], has several advantages over the EKF. Firstly, it provides a solution to the data association problem that does not require the unique identification of landmarks. Data association is performed through gradual reinforcement or degradation of matching probabilities as all the observation data over time is considered. This allows past data association decisions to be revised and possibly corrected. The EM approach also does not assume Gaussian noise. However, a disadvantage is that it does not provide an incremental solution to SLAM where a map is progressively built as observations are made. Another disadvantage is that it is generally only suited to offline processing, though there are exceptions like [11].

Recently, the *FastSLAM* [12] approach, based on the *particle filter* [13], was proposed that has several key strengths. Firstly, data association decisions are robustly made on a per-particle basis, analogous to MHT. That is, instead of just maintaining the data association with the maximum likelihood, the posterior tracks multiple data associations that can be resolved over time. Secondly, it has theoretically a lower computational complexity than the EKF:  $\mathcal{O}(m \log n)$  compared to  $\mathcal{O}(n^2)$  where  $m$  and  $n$  are the number of particles and landmarks in the map, respectively. FastSLAM can also cope with a non-linear vehicle model without the need for linearisation. The downside, inherent to all proactive approaches, is that there

can be an insufficient number of states or particles in the vicinity of the correct one causing the filter to diverge. This is especially a problem in large environments or when the robot is *kidnapped* [14]. Also, the resampling process continually reduces the diversity of the particle set [15], which restricts the size of the *loop* that can be corrected back in time (the concept of *closing the loop* is described in [16]–[18]).

There are other SLAM approaches that use *scan matching* [19], where the idea is to match the overlapping segments of neighbouring sensor scans, e.g., from a laser or sonar scanner, to estimate the robot's change in pose. Many of these approaches are derived from the *Iterative Closest Point (ICP) algorithm* [20], [21]. They iteratively refine an initial robot pose estimate obtained from dead-reckoning information on the assumption that the initial estimate is close enough to the robot's true pose to arrive at the globally optimal match.

Overall, the current approaches are largely reliant on stringent models of the robot's locomotive mechanism, sensor errors and environment (such as the robot's smooth traversal). The roboticist (or engineer) implementing one of these approaches then has the dubious task of modeling these aspects and defining appropriate model boundaries. The intractable real-world, however, is not amenable to such artificial boundaries. So, to achieve some semblance of reliability, SLAM is tailored to operate in only a sliver of the real-world or context-specific situation, and the whole modeling process repeated for another sliver, and so on *ad infinitum*.

This paper sets forth a challenging proposition. A new SLAM approach is proposed that is based on continuously solving the *kidnapped robot problem* over time. The kidnapped robot problem is defined as the problem of re-localising a mobile robot after its undergone an unknown motion, or in figurative terms, been kidnapped and clandestinely placed at an unknown location. It is typically considered in the context of a one-off, unwanted navigation event that needs to be first detected and then resolved. However, in this case, it is assumed to be continuously occurring over time, and by solving such a problem, the specifics of the robot's locomotive mechanism becomes irrelevant. Therefore, whether the robot locomotes via wheels, tracks, limbs, or one of the recent self-reconfiguring designs [22]–[24], has no bearing on the working operation of this approach.

Several other important outcomes are realised. Firstly, odometry and an associated vehicle model are not required, and so this approach is not vulnerable to the large *non-systematic errors* [25] that occur from the physical interaction between the robot and its environment (e.g., outdoor surface irregularities causing a robot's wheels to slip or unpredictable undercurrents acting on a submersible). Another important outcome is that there is no assumption of continuity in the robot's motion. This means that the SLAM process is essentially decoupled from the robot itself and, ergo, has a similar flexibility to the Global Positioning System (GPS) in terms of a standalone device. The difference being that instead of communicating with orbiting satellites like a GPS receiver, the proposed approach functions by sensing the local environment.

However, purposely disregarding odometry and the con-

straints imposed by the robot's locomotive mechanism is an extreme approach to SLAM that may seem illogical, especially if this information is readily available. In terms of the EKF, this is equivalent to making robot pose predictions with infinite uncertainty. The rationale is that the proposed approach only disregards this information in pursuit of its generalised applicability to an arbitrary robot, regardless of the robot's suitability to modeling. But in a context-specific situation, dead-reckoning information from odometry and possibly inertial sensors (gyroscopes and accelerometers) can be added at will, with varying degrees of accuracy, to the SLAM process to improve performance. The benefit is that the proposed approach is still fundamentally based on solving the kidnapped robot problem and so it is only vulnerable to its authoritative reference – the environment – and not the robot itself. However, contrary to other SLAM approaches, the means of using odometry is outside the scope of this paper.

The proposed approach consists of three subsystems: *landmark detection*, *multiple-hypothesis data association*, and *positional error representation and handling*. While the approach from a holistic perspective is novel in itself, these subsystems are also novel in their ideas and methods. The landmark detection subsystem is used to observe arbitrary environmental features (not necessarily structured) and select the most useful amongst them, allowing for redundancy. The multiple-hypothesis data association subsystem is used to match landmarks perceived from different robot locations and, correspondingly, address any uncertainty or ambiguity that results from the matching process, assuming that a gross discontinuity (no previous landmarks available) does not occur. The last subsystem maintains the positional errors of the robot and landmarks with no assumptions regarding the statistical size or shape of sensor noise.

Section II describes the proposed approach and its subsystems in the context of a complete, albeit preliminary package. Section III presents the results gathered from several outdoor experiments in national parks and bushlands<sup>1</sup> with a modified vehicle (1996 Mazda MX-5) acting as a pseudo robot. The vehicle was equipped with a 3D laser scanner and did not use odometry. Finally, Section IV concludes the paper with some of the many directions for future research, including rigorous analyses, comparative studies and extensions.

## II. THE KIDNAPPED WAY

This section provides a blueprint of the proposed SLAM approach. The objective is to estimate the system state  $\mathbf{x}_k$  at discrete time instant  $k$ , given by  $\mathbf{x}_k = [\mathbf{x}_{r_k} \mathbf{x}_1 \dots \mathbf{x}_n]^T$  where  $\mathbf{x}_{r_k}$  is the robot's state and the set  $M = \{\mathbf{x}_i | 1 \leq i \leq n\}$  represents the map of observed landmarks. The robot's state is defined by its 2D pose (position and orientation) in space  $\mathbf{x}_{r_k} = [x_{r_k} \ y_{r_k} \ \theta_{r_k}]^T$  relative to the global reference frame shown in Fig. 1. The landmarks in the map  $M$  are represented as points in space  $\mathbf{x}_i = [x_i \ y_i]^T$ .

<sup>1</sup>A term used to describe the harsh Australian outback.

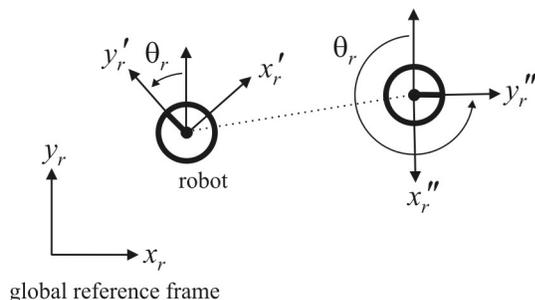


Fig. 1. Robot Coordinate System

### A. Landmark Detection

The landmark detection process involves using a sensor system to perform scans of the environment, and in each scan, observe a batch of landmarks and their relative geometric relationships. Since these relationships tend to remain invariant to the observer's point of view [26], they can be exploited to recognise landmarks whose appearance would otherwise vary according to the robot's changing pose. There is a multitude of *active* and *passive* sensors [27]–[29] that can be used. But because exteroceptive based sensing is such a critical element of this SLAM approach, there are several sensor qualities that are either necessary or, at the very least, highly desirable.

Firstly, the sensor needs to be able to gather a rich, dense data set in each scan that adequately represents a collection of landmarks. The more information that can be gathered to distinguish landmarks, e.g., size, shape, colour, texture, inter-landmark distances and angles, the less combinational matches that exist between the landmarks in a scan and their possible counterparts in the map and, hence, the more efficient the data association process. Secondly, during the scanning interval, the robot is assumed to be either stationary or have a negligible movement with respect to the scanning speed. This is a requirement because without dead-reckoning information, motion compensation cannot be applied to the data set. Also, in the case of 2D SLAM, the sensor may need to either remain in an upright position, e.g., by being suspended on a gimbal, or ascertain its roll and pitch angles through auxiliary sensors such as an inclinometer or gyroscope to track the horizontal plane when the robot traverses over an undulating surface.

An important objective is to maximise the angle of view and range (depth of field) of the sensor in order to ensure a high probability of overlap between the scan and map, and to maximise the overlapping region if one exists. There are also domain considerations that dictate the suitability and capacity of different sensing modalities. In a natural outdoor environment, for instance, a laser scanner can be used to accurately determine the surrounding geometry, whereas changing lighting conditions can make differentiating subtle shades of colour using a digital camera unreliable.

To provide a representative example, the authors obtained the outdoor results in Section III using the 3D laser scanner proposed in [30]. This scanner, shown in Fig. 8, consists of a laser rangefinder (LaserAce IM HR from MDL, UK) mounted on a pan-tilt unit (PTU-46-17.5 from Directed Perception, CA, USA) for 3D angular positioning. The laser rangefinder has

maximum range of 300m; a resolution of 1dm; a typical accuracy of 3dm; and makes range measurements at a rate of 1000Hz. A 3D scan is performed by using the pan-tilt unit to horizontally slue the laser in a back-and-forth manner, while incrementally adjusting its elevation angle.

After each scan, the landmark detection algorithm extracts a batch of landmarks from the collected data set. The algorithm proposed here, first described in [31], extracts a set of 2D point landmarks in the sensor's local frame of reference (refer to Fig. 2) that represents the centroids of environmental features whose spatial extent is orthogonal to the sensor's horizontal plane. This involves searching for arbitrary features (of no particular shape) that are highly visible, laterally compact and not partially occluded by other features. The first two criteria are based on relative measures. That is, highly visible features are those that occupy a large spatial range along the  $z$ -axis relative to their neighbours. Lateral compactness is a measure of how small a feature extends in the horizontal direction. Features that are more laterally compact than others tend to have a lower variability in their perceived centroid when viewed from different positions, and therefore provide more accurate triangulation results. The third criterion, by contrast, removes unstructured features that are partially hidden behind other features, as their true appearance cannot be ascertained.

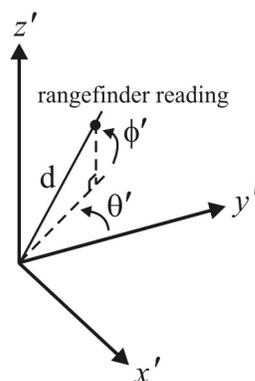


Fig. 2. Sensor Reference System

This algorithm is based on the concept that highly visible landmarks produce more rangefinder readings per unit area on the horizontal plane, projecting the  $z'$  component, than those that are less visible. However, merely counting and comparing the number of readings that fall within each cell of a regular 2D grid presents a number of problems. Since the sensor system operates through the rangefinder pivoting about a central point, the point distribution of a scan decreases radially outward from the local origin with regard to sampling density. Therefore, landmarks positioned closer to the origin are favourably biased. Also, scanning the support surface (i.e., the ground) can produce relatively high point counts and, consequently, false landmarks.

The *spiderweb grid*, shown in Fig. 3, was devised to address these problems. This grid somewhat normalises a scan's irregular point distribution and through a scanline algorithm, described shortly, removes obstructed landmarks and unusable surface characteristics. It is stored as a 2D array in computer memory, with each cell row representing

the circular space swept out by a sequence of angles at a certain distance range from the origin. The parameter  $d_{max}$  represents the rangefinder's maximum range (in this case, 300m). The parameters  $\delta d$  and  $\delta\theta'$  are dependent on the rangefinder's measurement errors and the scan's horizontal angular resolution, respectively. The objective is to assign values to these two parameters that ensures the points of an arbitrary landmark lie within adjacent cells. The larger these parameter values, the more redundancy that is used in fulfilling this objective, however, the coarser the grid. We used the values  $\delta d = 5dm$  and  $\delta\theta' = 0.514^\circ$ , derived from multiplying the rangefinder's typical accuracy by  $1\frac{2}{3}$  and the scan's horizontal resolution by 10, respectively.

Each scan point can be added to its respective grid cell  $(d_{cell}, \theta'_{cell})$  using the following equations:

$$d_{cell} \approx \left\lfloor \frac{d \cos(\phi')}{\delta d} \right\rfloor, \text{ for small } \delta\theta' \quad (1)$$

$$\theta'_{cell} = \left\lfloor \frac{\theta'}{\delta\theta'} \right\rfloor \quad (2)$$

However, before any points are added to the spiderweb, they are processed by the *scanline algorithm*. This algorithm is applied sequentially to each scanline (set of points grouped by a common elevation  $\phi'$ ) as follows.

First, a temporary 1D grid is created with a size of  $\left\lceil \frac{360^\circ}{\delta\theta'} \right\rceil$  cells. The cell sequence of this grid represents the same angular increments as those of a cell row in the spiderweb; however, instead of the individual cells holding a point sum, they hold a  $d_{cell}$  value. Each point in the first scanline is added to the grid by first calculating its  $\theta'_{cell}$  value to index the corresponding cell and then calculating its  $d_{cell}$  value to set the value of this cell, subject to a few provisions. If the indexed cell already has a value set by a previous point, then the  $d_{cell}$  value of the current point replaces the previous one if it is lower, otherwise it is discarded; hence each cell maintains the minimum  $d_{cell}$  value. Also, since the rangefinder hits a stop at either end of its rotational movement, the transitory deceleration, reversal and acceleration period makes the readings at both ends of the scanline unreliable. Therefore, a percentage of the readings is discarded from both ends (we chose 2%). An example of this process is shown in Fig. 4. The symbol ' $\infty$ ' denotes invalid rangefinder readings, possibly caused by the rangefinder pointing at free space, such as the sky, or features that are too close.

After all the points are added to the grid, adjacent cells that have  $d_{cell}$  values within  $\pm 1$  are clustered together (see Fig. 4). The clusters that have  $d_{cell}$  values at both ends (i.e., furthestmost left and right cells) that are lower than those of the adjoining ends of neighbouring clusters are considered to be unobstructed landmark segments. (Note that a ' $\infty$ ' value is treated as being infinitely large and an unknown value, i.e., an empty cell, is treated as being zero for the comparisons; however, cells with either of these two values do not constitute valid clusters.) All cells belonging to these unobstructed landmark segments are added to the spiderweb by incrementing the point counts of the corresponding cells. The process then reiterates for each of the remaining scanlines.

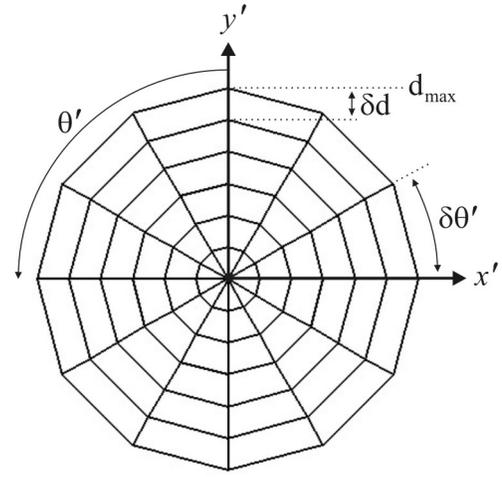


Fig. 3. Spiderweb Grid

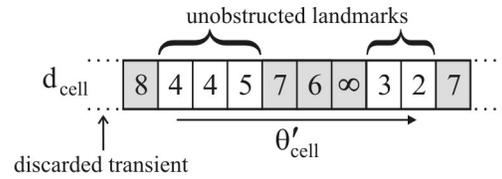


Fig. 4. A Scanline Processing Example

The resulting landmarks in the spiderweb are all the groups of adjacent cells with a point count higher than zero. Each of these landmarks is assigned an importance weighting, calculated by dividing its total point count by the number of cells. Since this weighting is proportional to a landmark's visibility and spatial compactness, the best landmarks are selected by limiting the landmark set to only a certain number of those with the highest weighting (we often limited it to 20). A *convex hull* [32] is then created around each of them to extract their centroid points. This provides the landmark set  $S_k = \{\mathbf{s}_{j,k} \mid 1 \leq j \leq m_k\}$  at time  $k$  where the landmarks, such as trees, bushes, posts, buildings and rock formations, are represented by 2D points in local coordinates  $\mathbf{s}_{j,k} = [x'_{j,k} \ y'_{j,k}]^T$ . The number of landmarks detected  $m_k$  must be at least three to obtain a unique triangulation result in 2D SLAM. However, preferably more than this critical number is detected for the purpose of redundancy.

The stability of the landmarks when viewed from different positions is affected by a number of factors, aside from sensor noise. Firstly, when viewing a large object from different sides, different facets will be visible, and so the centroid will appear to move. Also, the area of each of the cells in the spiderweb, denoted  $A(d_{cell}, \theta'_{cell})$ , varies according to

$$A(d_{cell}, \theta'_{cell}) = \left( d_{cell} + \frac{1}{2} \right) \delta d^2 \sin(\delta\theta') \quad (3)$$

along with the granularity of centroid placement. These factors are taken into account using a Euclidean error distance  $\epsilon$  as a tolerance bound for each centroid. Landmarks that are laterally large, and thus can be occluded from the sensor's field of view to a considerable degree, may be represented by multiple centroids over time. In essence, each of these

centroids represents an individual landmark whose appearance is restricted to when the robot is in a particular viewing area or zone. The number of centroids can be reduced and the zones expanded by increasing  $\epsilon$ , though at the risk of increasing the possible data associations in a cluttered environment.

### B. Multiple-Hypothesis Data Association

The multiple-hypothesis data association algorithm matches the batch of landmarks extracted from the scan  $S_k$  to those in the map  $M$ , while considering any ambiguities that arise from environmental symmetries or sensor limitations. Since at time  $k = 0$  the environment is unknown, the first scan  $S_0$  can be used to provide both the global frame of reference, as shown in Fig. 1, and the initial set of landmarks in  $M$ . The robot's pose is also initialised, somewhat arbitrarily, to  $\mathbf{x}_{r_0} = [0 \ 0 \ 0]^T$ . The proposed data association algorithm is then executed for time  $k \geq 1$ . This algorithm will be first described based on the scan  $S = \{\mathbf{s}_1, \dots, \mathbf{s}_m\}$  (the time subscripts have been left out for brevity) and map  $M = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$  at time  $k = 1$ , and then it will be shown how *multiple hypothesis tracking (MHT)* is incorporated for future times.

Unlike most other SLAM approaches, data association in this approach cannot be performed using the standard *gated nearest-neighbour (NN) algorithm* [7], because without dead-reckoning or motion continuity information, the required pose predictions cannot be made. Instead, we propose a graph matching approach that transforms the scan  $S$  and map  $M$  into two graphs  $G_1 = (V_1, E_1)$  and  $G_2 = (V_2, E_2)$ , respectively, and then finds subgraph isomorphisms between them to determine their commonalities. The vertex sets  $V_1$  and  $V_2$  of the graphs are used to represent the individual landmarks, which in this case are only distinguished by 2D points. The edges that interconnect the vertices, given by the sets  $E_1 \subseteq V_1^2$  and  $E_2 \subseteq V_2^2$ , are used to represent geometric relationships that are invariant to the robot's viewpoint. In doing so, common subgraphs of the graphs  $G_1$  and  $G_2$ , and hence data association hypotheses, can be found independently of the robot's pose. The geometric relationships that are used here are the Euclidean distances between landmark points. While these relationships may not be as discriminant as others, e.g., the relative landmark orientations [33], their detection does not rely on structured features being in the environment.

A data association hypothesis can now be given by the correspondence set  $X \subseteq V_1 \times V_2$ , where the vertices in each of the matched pairs  $\langle \mathbf{s}_a, \mathbf{x}_b \rangle \in X$  share a consistent set of edges. Since the graphs are matched based on their edges, there is a complexity problem that arises if the graphs are complete (i.e., fully connected). While the scan graph  $G_1$  remains relatively small, the map graph  $G_2$  is dynamically expanding over time, causing an exponential growth in the number of edges.

For a complete map graph with  $n$  vertices, the number of edges is given by the binomial coefficient  $\binom{n}{2}$ . Therefore, the number of edges in a graph with 10 vertices is 45; 100 vertices is 4950; or 1000 vertices is 499,500. To reduce this growth rate, an edge filter is proposed that confines the inter-landmark relationships to only those between neighbouring landmarks. This filter is based on the *Delaunay triangulation (DT)* [34],

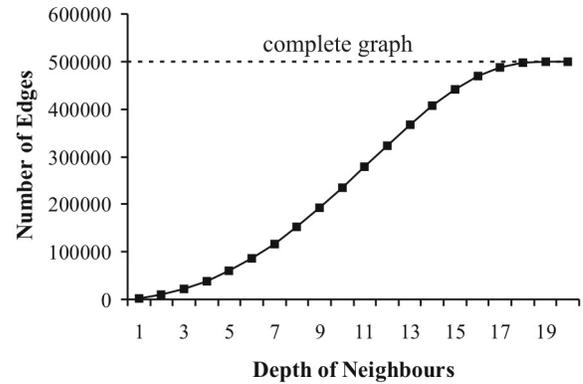


Fig. 5. Edge Count as a Function of Depth Level for a 1000-Vertex Graph

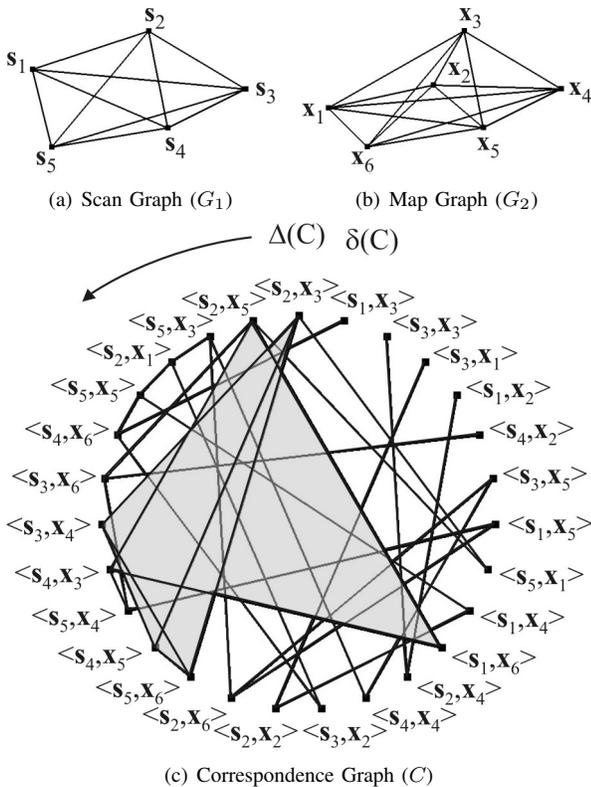
which can optimally generate a triangular mesh from a point set in  $\mathcal{O}(n \log n)$  time.

First, a DT is created for both the landmark sets  $S$  and  $M$ . The resulting mesh edges, indicating the closest landmark neighbours, form the initial edge set of the corresponding graphs  $G_1$  and  $G_2$ , respectively. The vertices of these graphs are now linked to their most immediate neighbours, which is called a depth level of one. To obtain a depth of two, edges are added to the graphs that link landmarks that are separated by a path length of two in the DTs. Similarly, a depth of three is obtained using a path length of three; and so on. The selection of which depth level to use is a tradeoff between graph completeness and computational complexity. The depth can also be changed over time, e.g., decreased to compensate for an expanding map, or different values can be used for each of the vertices, as a measure of their importance or uniqueness, to regulate their contribution to the matching process.

To demonstrate how the number of edges varies according to the chosen depth, a simulation was performed that generated 1000 random points on a 2D plane. The edge filter was then applied, with the depths ranging from one to twenty for all vertices. The resulting edge counts are graphed in Fig. 5. As an example, a depth of two resulted in 9784 edges, which is roughly 2% of the number in a complete graph.

Apart from a selectable reduction in the number of edges, an important property of this filter is that it can adapt to changes in the density and separation of landmarks. That is, a landmark's nearest neighbours is determined by the spatial attributes of its local cluster. Its nearest neighbours is also influenced by the DTs optimality criterion (maximising the minimum angle of the triangles); however, we conjecture that other triangulation schemes can be used with similar results.

A depth of two will be used here for both the scan graph  $G_1$  and map graph  $G_2$ . Hence, the only vertices of these graphs that will be interconnected by edges, representing Euclidean distances, are those that satisfy the edge filter at a depth of two. An example of the resultant graphs  $G_1$  and  $G_2$  is shown in Figs. 6(a) and 6(b), respectively. (Note that the edge distances are to scale.) Although they happen to be complete in this instance, due to their small size, their primary function is to provide a simple example for describing the matching process.


 Fig. 6. Common Subgraphs of the Graphs  $G_1$  and  $G_2$ 

The first step of the proposed matching algorithm involves creating a *correspondence graph* [35], denoted  $C$ , which represents the compatibility between each of the pairs  $\langle s_a, x_b \rangle \in V_1 \times V_2$ . The vertices of graph  $C$  are all the pairs whose two elements share a common property, i.e., a connecting edge of equal distance (within tolerance bounds). The edges of  $C$  represent the consistency between each of these vertices. That is, if  $\langle s_1, x_2 \rangle$  and  $\langle s_3, x_4 \rangle$  are two vertices of  $C$ , they are interconnected by an edge if the distances associated with  $E_1(s_1, s_3)$  and  $E_2(x_2, x_4)$  are equivalent. The method for creating graph  $C$  involves first finding all the edge matches between graphs  $G_1$  and  $G_2$ , and then adding the possible vertex combinations of each edge match to  $C$  as vertices and interconnecting those that are compatible with edges. As an example, the edge  $E_1(s_2, s_3)$  of  $G_1$  in Fig. 6(a) is a match for the edge  $E_2(x_3, x_4)$  of  $G_2$  in Fig. 6(b). The possible vertex combinations, and hence the resulting vertices in  $C$ , include  $\langle s_2, x_3 \rangle$ ,  $\langle s_2, x_4 \rangle$ ,  $\langle s_3, x_3 \rangle$  and  $\langle s_3, x_4 \rangle$ . The compatible edges are  $\{\langle s_2, x_3 \rangle, \langle s_3, x_4 \rangle\}$  and, through symmetry,  $\{\langle s_2, x_4 \rangle, \langle s_3, x_3 \rangle\}$ . The final state of the correspondence graph after all the matched edges between  $G_1$  and  $G_2$  have been incorporated is shown in Fig. 6(c).

It is common practice to then find the *maximum clique* (maximum complete subgraph) in graph  $C$  to obtain the maximum common subgraph (MCS) of the graphs  $G_1$  and  $G_2$  [33], [36]. However, there are several problems with this approach. Firstly, finding the maximum clique of an arbitrary graph is *NP*-complete, and hence computationally complex [37]. Furthermore, the graphs  $G_1$  and  $G_2$  would have to be complete, which as discussed previously, exacerbates the

complexity problem as the map  $M$  expands over time. Another issue is that the MCS, while the best match, may not be the right match. The MCS is merely the best hypothesis at one particular time instant; however, another common subgraph may prove to be a better hypothesis over time. Consequently, a new algorithm is proposed that finds common subgraphs based on the notion that only three consistent vertices in graph  $C$  are required to triangulate the robot's pose.

First, the vertices of graph  $C$  are ordered according to their degree (valence) sequence, i.e., in monotonically non-increasing degrees from the maximum degree  $\Delta C$  to the minimum  $\delta C$  (see Fig. 6(c)). The reason for doing this is that it heuristically places the vertices that are most likely to belong to a large common subgraph early in the order. Next, the vertices are processed in turn, by following each of their edges in search for a triangular subgraph. To show some examples, several triangular subgraphs are highlighted in Fig. 6(c), such as the subgraph  $\{\langle s_2, x_3 \rangle, \langle s_3, x_4 \rangle, \langle s_4, x_5 \rangle\}$ .

When a triangular subgraph is found, the *circle intersection approach* [31] is used to triangulate the robot's pose  $x_{r_k}$  based on the three constituent vertices. A unique pose is found if the three vertices do not lie on a straight line (otherwise a conjugate pair is obtained). The robot's pose is then used to transform the scan  $S$  from local to global coordinates:

$$x_j = x_{r_k} + x'_j \cos(\theta_{r_k}) - y'_j \sin(\theta_{r_k}) \quad (4)$$

$$y_j = y_{r_k} + x'_j \sin(\theta_{r_k}) + y'_j \cos(\theta_{r_k}) \quad (5)$$

With both the scan  $S$  and map  $M$  now being in the same coordinate system, the points in  $S$  are directly compared to those in  $M$  in a similar manner to *RANSAC* [38] to obtain the complete set of vertex matches (inliers) and a match count. This constitutes one hypothesis. The process then reiterates for the next triangular subgraph that is found; however, if it is a subset of the earlier hypothesis, then it can be eliminated.

There are two ways in which this algorithm can terminate. The first way is if every triangular subgraph has been found and processed in a brute force manner. The second is as an *anytime algorithm* [39], which takes advantage of the vertex order to arrive at the best hypotheses within the available time window. After completion, the resultant hypothesis set is given by  $H_k = \{h_k^{(\lambda)} \mid 1 \leq \lambda \leq \Lambda\}$ , where  $\Lambda$  is the number of hypotheses in the set. Note that the superscript  $^{(\lambda)}$  will also be used for other variables to indicate a particular hypothesis.

The hypotheses  $H_k$  are handled by the MHT algorithm, which resolves their associated data association ambiguities by tracking and assessing them over time. It must be emphasised that in a kidnap situation there is a possibility of no overlap between the scan  $S$  and map  $M$ , in which case the set  $H_k$  would be empty or completely erroneous. This situation, which is discussed in Section IV, is not addressed by the MHT algorithm proposed here; hence an overlap is assumed.

The MHT algorithm does not compare hypotheses based on a probabilistic model like in [9]; instead, hypotheses are weighted according to how many data association matches are made over time. Therefore, stronger hypotheses are those whose evolution of the map better corresponds with the

accumulated observation data. Each hypothesis  $h_k^{(\lambda)}$  maintains its own version of the map  $M^{(\lambda)}$ , robot path  $\mathbf{x}_{r_0:k}^{(\lambda)}$  and vertex match count  $w_k^{(\lambda)}$ . The hypotheses are compared based on their accumulated match count  $W_k^{(\lambda)}$  over  $k$  time periods, given by

$$W_k^{(\lambda)} = \sum_{t=0}^k w_t^{(\lambda)} \quad (6)$$

Initially, one hypothesis  $h_0^{(1)}$  is created at time  $k = 0$  to hold the map  $M^{(1)}$  (from scan  $S_0$ ), robot pose  $\mathbf{x}_{r_0}^{(1)}$  and match count  $w_0^{(1)}$  (initialised to zero). For time  $k \geq 1$ , the data association algorithm is executed for each hypothesis in  $H_{k-1}$ , which produces a new set of hypotheses  $H_k$ . These hypotheses are then compared based on their accumulated match count  $W_k^{(\lambda)}$ . The best hypothesis, and hence the SLAM solution, is the one with the highest weighting; however, the best  $\Lambda_{max}$  are kept. The map building algorithm is then called for each hypothesis in turn (described in the next subsection). This process then reiterates for the next time instant.

The maximum set size  $\Lambda_{max}$  is used to maintain computational tractability by limiting the number of hypotheses being spawned during each cycle. Weak hypotheses are eliminated in a "survival of the fittest" manner, leaving strong hypotheses to reign. The potential problem with this, however, is that there may be a point in the past where all hypotheses share the same robot path, which as a result, cannot be revised by this algorithm. This is in fact a similar problem to that found when limiting the number of particles in FastSLAM [15]. The selection of  $\Lambda_{max}$  is thus a tradeoff between the hypothesis diversity and computational complexity.

### C. Positional Error Representation and Handling

This subsection presents a novel approach to the representation and handling of positional errors that is considerably different to any of the current approaches. For instance, there is no use of Gaussian probabilities like in the EKF [3], nor does the multiplicity of samples in one of the Monte Carlo approaches [13] have any discernible similarities. The idea behind this approach stems from a simplification of the *correlation problem* [3], [4], which is generally the problem of maintaining the robot-landmark correlations that are formed when the robot's imprecisely known pose is used to update the landmark positions and *vice versa*. However, since this SLAM approach is based on continuously solving the kidnapped robot problem, the uncertainty in the robot's pose at time  $k$  is entirely attributable to the landmarks used in the triangulation. Therefore, the only correlations that need to be maintained are between the landmarks themselves.

Each landmark is correlated to the specific landmarks used to triangulate its global position. Consequently, a landmark's positional accuracy is dictated by the accuracy of its correlated landmarks, along with the sensor system and landmark detection process (note that odometry does not play a role here). Now, if we forgo any rigorous error models of the sensor system, and the associated contrivances, then the initial landmarks in the map  $M^{(1)}$  (from scan  $S_0$ ) at time  $k = 0$  can be considered the most accurate of all landmarks,

as they define the global origin and are not based on any triangulations. Landmarks that are added to the map from scan  $S_1$  at time  $k = 1$  are triangulated using three of the initial landmarks, and therefore are based on one triangulation. As a result, their positional uncertainty is higher than the initial landmarks because the scanning process introduces new errors, e.g., sensor noise. Similarly, if the landmarks from scan  $S_2$  at time  $k = 2$  are triangulated using three of the landmarks added to the map at time  $k = 1$ , then they will be based on two triangulations, as their positions are derived from a triangulation of a triangulation. Again, these landmarks have a higher positional uncertainty than their correlated landmarks because the scanning process introduces additional errors. From this, we can conclude that the number of triangulations from which a landmark's position is based is associated with an accumulative error. Given that the sensor system has a limited range, landmarks further away from the origin will tend to be based on more triangulations and, consequently, have larger positional errors.

The way in which positional errors are handled is analogous to solving the well-known *traveling salesman problem (TSP)* [40], which is the problem of finding the shortest closed path between  $n$  cities, given their intermediate distances. In this case, the problem is defined as finding the shortest path between each of the  $n$  landmarks and the origin, where the path length is given by the number of triangulations from which a landmark's position is derived. Therefore, the objective is to minimise the number of triangulations used to reach every landmark, and in doing so, minimise the sequential transfer and distortion of primary information (the initial landmarks).

Another analogy is found in a common game known as *Chinese whispers* or the *telephone game*. Players line up and the first person whispers a phrase to his or her neighbour. The neighbour then whispers the message to the next player, and so on down the line. The last player then calls out the message received, which may bear little resemblance to the original message due to the accumulative effect of mistakes along the line. In terms of the proposed approach, if the message represents the initial landmarks and a person represents a triangulation, then the idea is to remove as many people from the line as possible to minimise the chance of error.

The error in each of the landmark positions  $\mathbf{x}_i^{(\lambda)}$  is represented by a single nonnegative integer  $\xi_i^{(\lambda)} \in \mathbb{Z}^*$ . Likewise, the error in the robot pose  $\mathbf{x}_{r_k}^{(\lambda)}$  at time  $k$  is represented by the integer  $\xi_{r_k}^{(\lambda)} \in \mathbb{Z}^*$ . These integers store the path sizes (number of triangulations), and so they each provide a qualitative indication of the magnitude of a positional error. Note that if two landmarks have equal error values, this does not necessarily mean they have the same quantitative errors. It means they both share the same number of time instants where quantitative errors were accumulated.

At time  $k = 0$ , the initial landmarks in the map are each assigned a qualitative error value of zero, as they are based on zero triangulations. For time  $k \geq 1$ , the map of each hypothesis is updated using Algorithm 1 in  $\mathcal{O}(w)$  time. (Note that the hypothesis and time notation has been left out for brevity.) The variable  $\xi_S$  represents the error of each of the

**Algorithm 1** Update\_Map

---

**Require:**  $k \geq 1$

- 1:  $\xi_S \leftarrow \max\{\xi_i\}_{i \in T_S} + 1$
- 2: **for all**  $\langle \mathbf{s}_a, \mathbf{x}_b \rangle \in X$  **do**
- 3:   **if**  $\xi_b \geq \xi_S$  **then**
- 4:      $\mathbf{x}_b \leftarrow \mathbf{s}_a$  {in global coordinates}
- 5:      $\xi_b \leftarrow \xi_S$
- 6:      $T_b \leftarrow T_S$
- 7:   **end if**
- 8: **end for**
- 9: add new landmarks with error  $\xi_S$  and triangulation  $T_S$

---

landmarks in scan  $S$ , along with the robot (i.e.,  $\xi_r \leftarrow \xi_S$ ). Each of the map landmarks  $\mathbf{x}_i$  has an associated index set  $T_i$ , which contains the indices of the landmarks used to triangulate it. For example, if landmark  $\mathbf{x}_{13}$  has the index set  $T_{13} = \{3, 5, 8\}$ , then it was triangulated using landmarks  $\mathbf{x}_3$ ,  $\mathbf{x}_5$  and  $\mathbf{x}_8$ . The index set associated with the initial landmarks at time  $k = 0$  is the null set  $\emptyset$ . The map landmarks used to triangulate the scan  $S$  is given by the index set  $T_S$ . Lastly, the correspondence set  $X$  represents the  $w$  landmark matches between the scan  $S$  and map  $M$ , as hypothesised by the data association process.

If the environment is static, as is commonly assumed, then this algorithm can be extended in several ways. Any changes to a landmark  $\mathbf{x}_i$  can be recursively propagated down to all the landmarks whose position is correlated with it, thereby increasing the convergence rate. Also, without dead-reckoning or motion continuity information, there is essentially no causality between the scans. The scan order can be continually adjusted to maximise data associations or minimise the overall error of the map. The computational cost, however, prohibits the amount of optimisation that can be done in real-time.

There are several properties about the proposed approach that need to be considered. Firstly, sensor noise is only considered in qualitative terms; there are no models or assumptions regarding its statistical size or shape. Although the qualitative errors do not differentiate between different sensors such as a laser or sonar, they tend to be proportional to the quantitative errors which are dependent on the particular sensor used. Secondly, the way in which the map converges is based on recursively triangulating each landmark's position using three of the most accurate landmarks that can be observed in the same scan. As a result, there will be a convergence limit that is determined by the landmark arrangement, including density, and the sensor's range and accuracy. Particularly, the longer the range of the sensor, the more the map converges, and the lower the growth rate of positional errors outward from the origin. Note that the robot's navigational ability is not necessarily hampered when its far away from the origin, as its locally surrounding landmarks can have relative errors that are considerably lower than their global ones.

Finally, addressing the *loop closing problem* [16]–[18], especially for cyclic environments, will be left for future work. Just as a brief comment, the common problem of detecting a previously visited place is merely part of normal operation. There is also no odometric drift, and so errors accumulate

around the loop at a relatively small rate.

*D. Pathological Cases*

There are several cases in which the proposed approach to SLAM can fail. Firstly, since the approach is based entirely on exteroceptive sensing, it cannot function in an environment that has no detectable features. Essentially, the system would become lost. Another failure case is a completely symmetrical environment (no distinguishable features). As an example, if the robot were to navigate down a straight road with only equidistant trees on either side, then the multiple hypotheses that are created cannot be resolved. Consequently, the SLAM process cannot determine if the robot is moving forward or differentiate between the robot standing still or driving at full speed. The unresolvable hypotheses can make it seem that the robot is teleporting between multiple poses, and for this reason, we call it the *teleportation problem*.

Failure can also occur in the case of a very large or cluttered environment. In these situations, the uniqueness of landmark relativities, such as inter-landmark distances, may diminish to a point where the number of possible data associations is too large to process in real-time. If the environment is too cluttered, then the tolerance bounds of the landmarks may permit an inordinate number of false data associations that appear to be symmetries. Also, clutter can significantly impair the sensor's view, limiting the number of previously detected landmarks that can be rediscovered. These problems can be partially counteracted by using a more exclusive landmark recognition technique or bringing dead-reckoning information back into the equation, as discussed in Section IV.

## III. EXPERIMENTAL RESULTS

This section presents the results from four outdoor experiments. In chronological order, the first experiment was performed at the You Yangs Regional Park; the second experiment at Warrandyte State Park; and the third and fourth experiments at a large bushland property, owned by the second author, in Pomonal near the Grampians. These areas are all located in the state of Victoria, Australia: Warrandyte State Park is in Melbourne; You Yangs is 55km south-west of Melbourne; and Pomonal is 239km north-west of Melbourne in country Victoria. They were not scouted beforehand and no artificial landmarks were added.

Fig. 7 shows the robot used in all the experiments. This robot, which we call the Hunter, is a modified vehicle (1996 Mazda MX-5) with a 3D laser scanner attached to the front tow hooks. The laser scanner, described in Section II-A, is shown in Fig. 8. It is controlled by a Linux box on the passenger seat and powered by a 12V car battery in the floor well. The driving controls (steering wheel, brake, accelerator, etc.) are not automated due to the difficulty of dealing with a manual transmission, but more importantly, the Hunter doubles as the first author's primary mode of transportation.

The vehicle was driven along an arbitrary path in each experiment, except for the last two in Pomonal where an online path planner was used (not described in this paper). Since the



Fig. 7. The Hunter in a Natural Environment

SLAM algorithm was always executed online, the results given here were available to the driver during the experiments.

For each of the experiments, labeled *I* through *VI*, the specific parameters that were used will be given first, followed by the results. Note that if certain parameters are omitted from a particular experiment, then their values have not changed from those used in the previous experiment.

#### A. Experiment I

In the first experiment, the scans were performed using a horizontal range of  $\theta' \in [-129^\circ, 129^\circ]$ , horizontal slue rate of  $51.4^\circ/s$  (giving a horizontal resolution of  $0.0514^\circ$ ), vertical range of  $\phi' \in [-18.0^\circ, 7.71^\circ]$  and vertical resolution of  $0.514^\circ$ . This resulted in each scan taking approximately 255,000 readings over a time period of 4.25 minutes. The parameter values used for the spiderweb included:  $d_{max} = 300m$ ,  $\delta d = 5dm$  and  $\delta\theta' = 0.514^\circ$ . The number of landmarks detected  $m_k$  was limited to 30; the number of hypotheses  $\Lambda_{max}$  limited to 3; and the edge tolerance  $\epsilon$  set to  $0.5m$  for all edges. Also, the search space of the correspondence graph was limited to  $\frac{1}{8}$  of the order.

The final results, including the solution, ground truth and landmark errors, are shown in Fig. 9. The ground truth was obtained using a tape measure and compass at each kidnap



Fig. 8. Sensor System

point. There were two pose errors that were caused by a barrier of dense bush that significantly blocked the sensor's view. As a consequence, there was an insufficient overlap between the scans and the map; however, the system was able to recover. The map of the best hypothesis reached a size of 180 landmarks, primarily comprising trees, bushes and plants.

#### B. Experiment II

For Warrandyte State Park, there was only one parameter changed: the edge tolerance  $\epsilon$  was set to  $1m$  for all edges. The trees tended to be laterally large in this environment, often resulting in multiple landmark points per tree. The final results are shown in Fig. 10.

#### C. Experiment III

At the property in Pomonal, the scans were performed using a new vertical range of  $\phi' \in [-17.5^\circ, 8.23^\circ]$  and vertical resolution of  $1.03^\circ$  – the horizontal parameters remained the same. This reduced the scan size to 130,000 readings and the scan period to 2.2 minutes. (We believe these scans are still significantly larger than what is needed; however, minimising the scan time was not one of our primary objectives.) Also, the number of landmarks detected  $m_k$  was limited to 20, and the edge tolerance  $\epsilon$  set to  $1.5m$ .

The final results are shown in Fig. 11. They have been overlaid on an aerial photograph of the area that was captured by a radio-controlled unmanned aerial vehicle (UAV) at an altitude of approximately  $1km$ . Ground truth was collected using a GPS unit (GPSmart from Fortuna, Taiwan).

#### D. Experiment IV

This experiment was performed in a different area of the same property with the same parameters. Fig. 12 shows the excellent results obtained, with each pose being a solution to the kidnapped robot problem.

### IV. CONCLUSIONS AND FUTURE WORK

This paper presented a novel, contrarian approach to SLAM that purposely disregards odometry and the assumption of continuity in the robot's motion to mimic the portability of a GPS receiver. The approach functions by continuously solving the kidnapped robot problem over time through sensing the local environment. This essentially decouples the SLAM process from the robot, and in doing so, there is no model of the robot or its physical interaction with the environment. Therefore, it can be implemented as a standalone device and transferred between robots, regardless of their physical makeup.

At the elemental level, contributions were made in the areas of natural landmark detection, multiple-hypothesis data association, and positional error representation and handling. Experiments were performed in several natural outdoor environments that were quite hostile toward robotic endeavours. The results prove the feasibility of the proposed approach. Similar to a GPS, there was no consideration of the robot's type, capabilities, locomotion mode or control process. The difference is that a GPS functions in open outdoor areas away

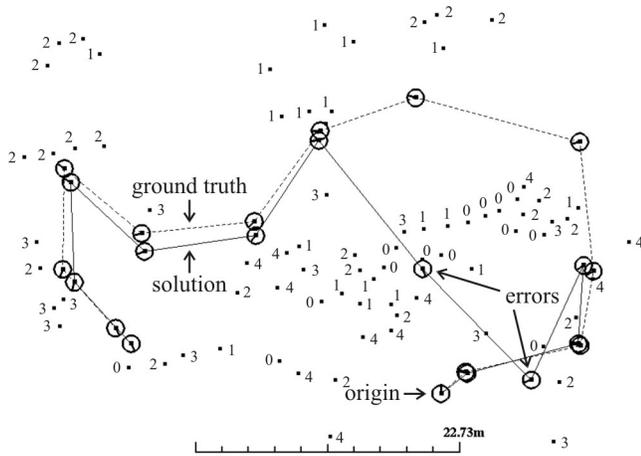


Fig. 9. Experiment I: Results from a Cluttered Environment

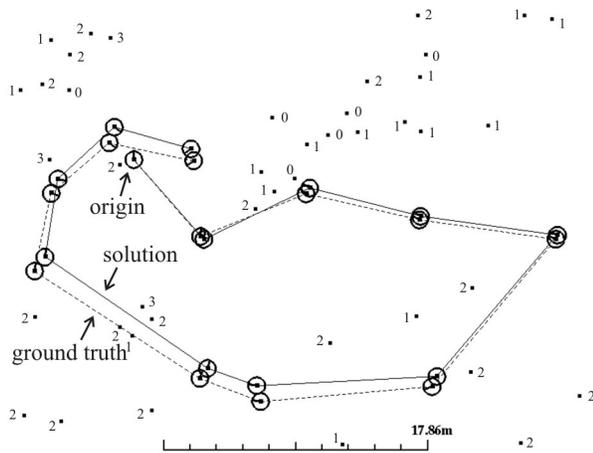


Fig. 10. Experiment II: Results from an Environment with Large Features

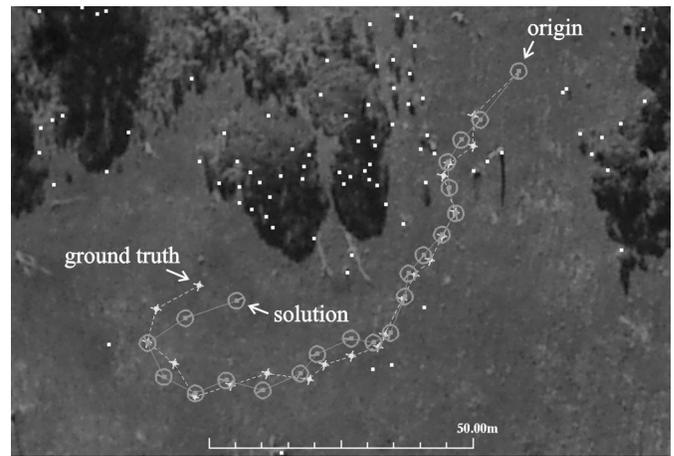


Fig. 11. Experiment III: Results from a Harsh Environment (Area 1)

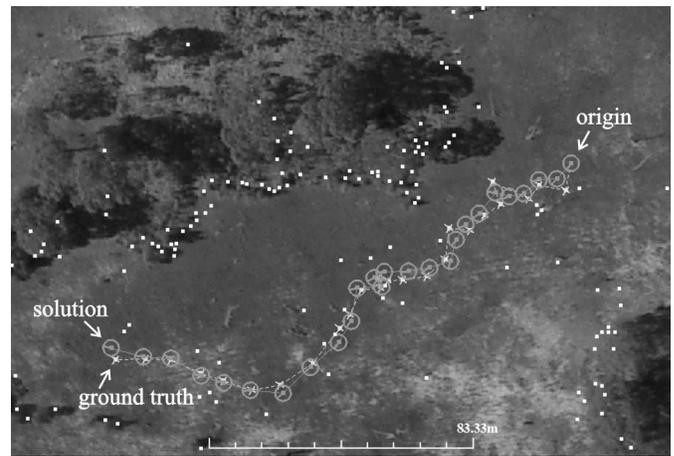


Fig. 12. Experiment IV: Results from a Harsh Environment (Area 2)

from environmental features that obstruct the line of sight to satellites, whereas the proposed approach functions in feature rich environments where it can find overlaps between the scan and map.

The situation where there is no overlap between the scan and map was not addressed here. It can therefore be argued that some form of continuity was invoked; however, this can only be attributed to the map boundaries. If the map expands to cover the robot's workspace, then, without reservation, there is no assumption of continuity. Also, incorporating the information from a scan into the global map requires an overlap to exist or develop over time. In the latter case, if the robot were to make extreme jumps to isolated places, then the local maps of these places could be independently maintained until they overlapped the global map. This process, however, can quickly become computationally intractable, as each map needs to be matched to every other map. Hence, it is a challenging extension.

The universality of this approach and the locomotive freedom it provides to a robot or other agent in its environment warrants further investigation. Some of the many possible avenues for future work include: enhancing the landmark detection system to extract other environmental features and reduce

data association ambiguities; using *negative information* (non-matches) to decrease the confidence in a data association; complexity analysis of the data association method; pruning ineffective landmarks from the map; using dead-reckoning information to improve scalability; performance comparisons with other approaches; developing a version that can operate with moving objects, such as people and vehicles, in the scene; 3D SLAM; and, ultimately, commercialisation of a standalone SLAM sensor.

## REFERENCES

- [1] J. J. Leonard, P. M. Newman, R. J. Rikoski, J. Neira, and J. D. Tardós, "Towards robust data association and feature modeling for concurrent mapping and localization," in *10<sup>th</sup> Int. Symp. Robotics Research*, Lorne, Victoria, Australia, 9-12 Nov. 2001, pp. 7–20.
- [2] S. Thrun, "Probabilistic algorithms in robotics," *AI Magazine*, vol. 21, no. 4, pp. 93–109, 2000.
- [3] R. Smith, M. Self, and P. Cheeseman, "A stochastic map for uncertain spatial relationships," in *4<sup>th</sup> Int. Symp. Robotics Research*. MIT Press, 1987.
- [4] P. Moutarlier and R. Chatila, "Stochastic multisensory data fusion for mobile robot location and environment modelling," in *5<sup>th</sup> Int. Symp. Robotics Research*, Tokyo, Japan, 28-31 Aug. 1989, pp. 85–94.
- [5] J. J. Leonard and H. F. Durrant-Whyte, "Simultaneous map building and localization for an autonomous mobile robot," in *IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, vol. 3, Osaka, Japan, 3-5 Nov. 1991, pp. 1442–1447.

- [6] S. J. Julier and J. K. Uhlmann, "A counter example to the theory of simultaneous localization and map building," in *Proc. IEEE Int. Conf. Robotics and Automation*, Seoul, Korea, 21-26 May 2001, pp. 4238-4243.
- [7] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. Boston, MA: Academic Press, 1988.
- [8] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Automatic Control*, vol. AC-24, no. 6, pp. 843-854, 1979.
- [9] I. J. Cox and J. J. Leonard, "Probabilistic data association for dynamic world modeling: A multiple hypothesis approach," in *Fifth Int. Conf. Advanced Robotics*, Pisa, Italy, 19-22 Jun. 1991, pp. 1287-1294.
- [10] S. Thrun, W. Burgard, and D. Fox, "A probabilistic approach to concurrent mapping and localization for mobile robots," *Machine Learning*, vol. 31, no. 1-3, pp. 29-53, 1998.
- [11] C. Martin and S. Thrun, "Real-time acquisition of compact volumetric 3D maps with mobile robots," in *Proc. IEEE Int. Conf. Robotics and Automation*, Washington, DC, 11-15 May 2002, pp. 311-316.
- [12] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proc. Natl. Conf. Artificial Intelligence*, Alta., Canada, 28 Jul.-1 Aug. 2002, pp. 593-598.
- [13] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag, 2001.
- [14] S. P. Engelson and D. V. McDermott, "Error correction in mobile robot map learning," in *Proc. IEEE Int. Conf. Robotics and Automation*, Nice, France, 12-14 May 1992, pp. 2555-2560.
- [15] M. Montemerlo, "FastSLAM: A factored solution to the simultaneous localization and mapping problem with unknown data association," Ph.D. dissertation, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 2003.
- [16] R. Chatila and J. Laumond, "Position referencing and consistent world modeling for mobile robots," in *IEEE Int. Conf. Robotics and Automation*, St. Louis, MO, 25-28 Mar. 1985, pp. 138-145.
- [17] F. Lu and E. Milius, "Globally consistent range scan alignment for environment mapping," *Autonomous Robots*, vol. 4, no. 4, pp. 333-349, 1997.
- [18] J. S. Gutmann and K. Konolige, "Incremental mapping of large cyclic environments," in *Proc. IEEE Int. Symp. Computational Intelligence in Robotics and Automation*, Monterey, CA, 8-9 Nov. 1999, pp. 318-325.
- [19] F. Lu and E. Milius, "Robot pose estimation in unknown environments by matching 2D range scans," *J. Intelligent and Robotic Systems*, vol. 18, no. 3, pp. 249-275, 1997.
- [20] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," in *Proc. IEEE Int. Conf. Robotics and Automation*, Sacramento, CA, 9-11 Apr. 1991, pp. 2724-2729.
- [21] P. J. Besl and H. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239-256, 1992.
- [22] M. Yim, Y. Zhang, and D. Duff, "Modular robots," *IEEE Spectrum*, vol. 39, no. 2, pp. 30-34, 2002.
- [23] D. Rus, Z. Butler, K. Kotay, and M. Vona, "Self-reconfiguring robots," *Communications of the ACM*, vol. 45, no. 3, pp. 39-45, 2002.
- [24] R. R. Murphy, "Marsupial and shape-shifting robots for urban search and rescue," *IEEE Intelligent Systems*, vol. 15, no. 2, pp. 14-19, 2000.
- [25] J. Borenstein, H. R. Everett, L. Feng, and D. Wehe, "Mobile robot positioning: Sensors and techniques," *J. Robotic Systems*, vol. 14, no. 4, pp. 231-249, 1997.
- [26] L. G. Shapiro and R. M. Haralick, "Structural descriptions and inexact matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-3, no. 5, pp. 504-519, 1981.
- [27] H. R. Everett, *Sensors for Mobile Robots: Theory and Application*. Wellesley, MA: A. K. Peters, 1995.
- [28] R. A. Jarvis, "Range sensing for computer vision," in *Three-Dimensional Object Recognition Systems*, A. K. Jain and P. J. Flynn, Eds. Singapore: Elsevier, 1993.
- [29] M. Hebert, "Active and passive range sensing for robotics," in *Proc. IEEE Int. Conf. Robotics and Automation*, San Francisco, CA, 24-28 Apr. 2000, pp. 102-110.
- [30] D. J. Spero and R. A. Jarvis, "3D vision for large-scale outdoor environments," in *Australasian Conf. Robotics and Automation*, Auckland, New Zealand, 27-28 Nov. 2002, pp. 228-233.
- [31] —, "Towards exteroceptive based localisation," in *IEEE Conf. Robotics, Automation and Mechatronics*, Singapore, 1-3 Dec. 2004, pp. 822-827.
- [32] R. A. Jarvis, "On the identification of the convex hull of a finite set of points in the plane," *Information Processing Letters*, vol. 2, no. 1, pp. 18-21, 1973.
- [33] T. Bailey, E. M. Nebot, J. K. Rosenblatt, and H. F. Durrant-Whyte, "Data association for mobile robot navigation: A graph theoretic approach," in *Proc. IEEE Int. Conf. Robotics and Automation*, vol. 3, San Francisco, CA, 24-28 Apr. 2000, pp. 2512-2517.
- [34] M. Bern and D. Eppstein, "Mesh generation and optimal triangulation," in *Computing in Euclidean Geometry*, D. Z. Du and F. Hwang, Eds. Singapore: World Scientific, 1992, pp. 23-90.
- [35] H. G. Barrow and R. M. Burstall, "Subgraph isomorphism, matching relational structures and maximal cliques," *Information Processing Letters*, vol. 4, no. 4, pp. 83-84, 1976.
- [36] R. C. Bolles and R. A. Cain, "Recognizing and locating partially visible objects: The local-feature-focus method," *Int. J. Robotics Research*, vol. 1, no. 3, pp. 57-82, 1982.
- [37] P. M. Pardalos and J. Xue, "The maximum clique problem," *J. Global Optimization*, vol. 4, pp. 301-328, 1994.
- [38] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381-395, 1981.
- [39] S. Zilberstein, "Resource-bounded sensing and planning in autonomous systems," *Autonomous Robots*, vol. 3, no. 1, pp. 31-48, 1996.
- [40] E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys, Eds., *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. Chichester: Wiley, 1985.



**Dorian J. Spero** received the B.Sc. (1st class honours) and B.E. (1st class honours) degrees in computer science and electronics, respectively, from LaTrobe University in Melbourne, Australia, in 2001, and the Ph.D. degree in robotics from Monash University in Melbourne in 2006. He is currently a researcher at Monash with an interest in mobile robot navigation in natural environments.



**Ray A. Jarvis** received the B.E. (electrical engineering) and Ph.D. degrees from the University of Western Australia in 1962 and 1968, respectively. After two years at Purdue University in West Lafayette, Indiana, he was appointed to the Australian National University (ANU) where he was instrumental in establishing the Department of Computer Science and was its first Head of Department. He then joined Monash University in Melbourne in 1985, and established the Intelligent Robotics Research Centre in 1987. He has been the Center's Director ever

since, and has also recently taken on the role of Director of the ARC Centre for Perceptive and Intelligent Machines in Complex Environments. His research interests are in computer vision and robotics (indoors and out).