Simultaneous Localization and Mapping - a new algorithm for a compass-equipped mobile robot

Tom Duckett
Dept. of Technology
University of Örebro
S-70182 Örebro
Sweden
Tom.Duckett@aass.oru.se

Stephen Marsland and Jonathan Shapiro
Dept. of Computer Science
University of Manchester
Manchester M13 9PL
England
smarsland,jls@cs.man.ac.uk

Abstract

In contrast to several other types of sensor, compasses are used relatively rarely on mobile robots. We show that this sensor offers some distinct practical advantages for solving the problem of simultaneous localization and mapping (SLAM), provided that some reasonable assumptions are made in modelling and reasoning with the uncertainty in the compass readings. In this paper, we derive a new maximum likelihood map learning algorithm for a compass-equipped mobile robot based on a statistical model of the noise. Experiments are presented in which a complex, real world environment was mapped by a real robot equipped with an uncalibrated flux-gate compass. The paper places special emphasis on understanding the assumptions underlying our representations and algorithms, and the impact of these assumptions on the performance of the robot in successfully solving the SLAM problem in semi-structured office environments.

1 Introduction

The problem of simultaneous localization and mapping for mobile robots has attracted considerable interest in the past few years; see, in particular, the paper in this workshop by [Frese and Hirzinger, 2001]. We agree with many of the proposals in that paper, especially that it is a good idea to separate the sub-problems of mapping and landmark identification. In addition, we believe that it is a good idea to separate the problem of self-orientation, since a compass can reduce the complexity of both the mapping and self-localization problems by an order of magnitude. Examples of self-orientation using a geomagnetic compass sense can be found in nature, e.g., in pigeons [O'Keefe and Nadel, 1977], and magnetic compasses have been used for navigation on steel ships for many years.

However, magnetic compasses are rarely used on mobile robots due to a popular conception that "this is known not to work properly in buildings which contain large amounts of steel" [Frese and Hirzinger, 2001]. While this may be true in some extreme cases, we have found that, for practical purposes, a compass can be a very useful extra source of sensory information for navigation in indoor environments, provided that some reasonable assumptions are made in modelling and reasoning with the uncertainty in the compass readings. In this paper, we show that this sensor can be used together with other sensors to obtain an efficient solution to the SLAM problem in semi-structured office environments.

The paper is structured as follows: First, we describe the map representation used in our approach (Sec. 2). Then we derive a new maximum likelihood map learning algorithm for a compass-equipped mobile robot, based on a statistical model of the noise (Sec. 3). This algorithm is proved to converge to a globally optimal solution and can be used to update the map at a computational cost that is linear in the number of places stored in the map. Sec. 4 presents experiments on a Nomad 200 robot equipped with an uncalibrated flux-gate compass, in which large office environments were mapped in real-time, followed by related work (Sec. 5) and a discussion (Sec. 6).

2 Environment Model

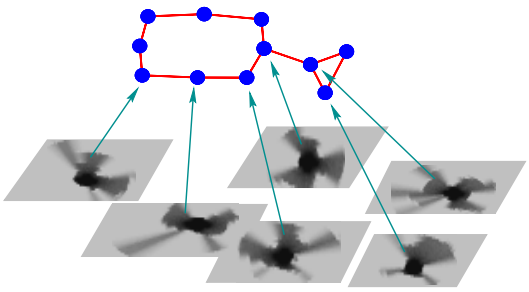

Figure 1: The map representation.

We use a graph-based model of the environment, in which the nodes correspond to places and the links to traversable paths between places (see Fig. 1). Each node in the graph is associated with a local occupancy grid, constructed using the robot's ultrasonic range-finder sensors, which is used as a place signature. For place recog-

nition, we use the self-localization algorithm described in [Duckett and Nehmzow, 1999], which applies an occupancy grid-matching technique to identify both the robot's current place in the map and the relative displacement of the robot from the centre of that particular place.

For the following analysis, we define our maps as follows:

- The topological component of the map consists of a set of N place nodes and a set of links that connect pairs of places.
- Each place i is associated with a pair of Cartesian coordinates $\mathbf{r}_i = (x_i, y_i)^T$ that are initially unassigned, as the true coordinates are unknown to the robot.
- Each link connects two places i and j, and is associated with an estimated metric relation (d_{ij}, θ_{ij}), measured by the robot, that describes the relative distance d_{ij} and angle θ_{ij} between the two places. The angle θ_{ij} is an absolute measurement obtained from the compass sense. In this paper, the links were constrained to be bi-directional, that is, $d_{ij} = d_{ji}$ and $\theta_{ij} = \theta_{ji} + \pi$.

The algorithm derived in the next section aims to assign a globally consistent set of Cartesian coordinates $\{\mathbf{r}_1, ..., \mathbf{r}_N\}$ to the places in the map using the noisy measurements of the distances and angles between the places. This noise means that a perfect map cannot be generated. Instead, we derive a statistically optimal algorithm to deal with this problem in a principled way.

3 Maximum Likelihood Map Learning Algorithm

When maps are generated from the estimates of distances and angles measured by the robot, the geometry of the space will be non-Euclidean. For example, the angles inside a triangle may not add up to 180°. Because the robot is exploring in Euclidean space, this is a problem. Our map building algorithm aims to find an evidence-based way of fixing this problem so that the maps are geometrically consistent. In previous work, we introduced a naive algorithm to solve this problem based on the assumption of circular Gaussian noise in the robot's odometry [Duckett *et al.*, 2000]. In this paper, we derive a new improved algorithm from a noise model, based on a much more detailed consideration of the assumptions underlying our approach.

3.1 Estimating the Noise

When the robot travels between two map places, a large number of small distances and angles are measured as the robot continuously updates its heading. Let δ_t be the displacement and α_t the heading at the t^{th} step. Then

$$d_{ji} = \sqrt{\left(\sum_t^{T_{ji}} \delta_t \cos\alpha_t\right)^2 + \left(\sum_t^{T_{ji}} \delta_t \sin\alpha_t\right)^2}, \quad (1)$$

$$\theta_{ji} = \tan^{-1}\left(\frac{\sum_t^{T_{ji}} \delta_t \sin\alpha_t}{\sum_t^{T_{ji}} \delta_t \cos\alpha_t}\right), \quad (2)$$

where T_{ji} is the number of steps, i.e., the number of updates the robot performs along a path.

Assuming that the number of updates made while travelling between two map places is large and that the statistical properties of each step are independent, the Central Limit Theorem [Reif, 1982] implies that measurements of d_{ji} and θ_{ji} will be normally distributed around their true values.

Suppose that the noise properties are the same along a path between two map places (but not necessarily between paths) and that successive measurements are independent. That is, the measurement of the path from place i to place j is independent of the measurement from place j to place k (this means that the model cannot deal with cumulative phenomena such as battery drain). Then we can write the estimates of the distance travelled and the heading of the robot at each small step as

$$\delta_t = \delta_t^* + \Delta\delta_t \quad (3)$$

$$\alpha_t = \alpha_t^* + \Delta\alpha_t \quad (4)$$

where $\Delta\delta_t$ is the noise in the estimate of the true distance δ_t^* and $\Delta\alpha_t$ is the noise in the estimate of the true angle α_t^*.

Assuming that the noise measurements are small compared to the distances travelled, the covariance matrix C_{ji} of the link measurements (d_{ji}, θ_{ji}) can be calculated using small deviation expansions as

$$C_{ji} = T_{ji} R(\phi_{ji})^{-1} \begin{pmatrix} \Delta_{\delta\delta} & \delta_{ji}\Delta_{\alpha\delta} \\ \delta_{ji}\Delta_{\alpha\delta} & \delta_{ji}^2\Delta_{\alpha\alpha} \end{pmatrix} R(\phi_{ji}), \quad (5)$$

where $\delta_{ji} = (\sum_t^{T_{ji}} \delta_t)/T_{ji}$, that is the average distance between updates, the $\Delta_{\delta\delta}$, etc. are global noise estimates of the subscripted variables, and ϕ_{ji} is the rotational error between the two nodes, where $R(\phi)$ is the rotation matrix

$$R(\phi) \equiv \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix}. \quad (6)$$

In our model, it is assumed that $\phi_{ji} = 0$ because of the compass, so the rotation matrix $R(\phi_{ji}) = I$. This model is invariant to global rotations and depends only on the direction of travel. The noise parameters $\Delta_{\delta\delta}$, etc. are calculated by maximum likelihood estimation as part of the new algorithm (eqns. 12–14).

3.2 The Algorithm

The map can be considered as a set of free nodes that are held together by springs, where each spring connects two adjacent places i and j (this analogy can also be found in [Lu and Milios, 1997a; Golfarelli *et al.*, 1998; Shatkay, 1998]). Each spring reaches minimum energy when the relative displacement between the coordinates of node i and node j is equal to the vector $(d_{ij},\ \theta_{ij})$ measured by the robot. Equilibrium is reached in the whole map when the total energy over all of the springs reaches a global minimum. Thus, global consistency is maintained in the map by minimising the following energy function, which corresponds to the log likelihood:

$$E = \sum_i \sum_j{}' \left(\mathbf{r}_i - \mathbf{r}_j - \mathbf{D}_{ji}\right)^T C_{ji}^{-1} \left(\mathbf{r}_i - \mathbf{r}_j - \mathbf{D}_{ji}\right), \quad (7)$$

where $\sum_j'$ refers to the sum over the neighbours of node i, $\mathbf{r}_i = (x_i, y_i)^T$, and $\mathbf{D}_{ji} = d_{ji} \begin{pmatrix} \cos\theta_{ji} \\ \sin\theta_{ji} \end{pmatrix}$.

Since each C_{ji} is symmetric,

$$\nabla E_i = 2 \sum_j{}' C_{ji}^{-1} (\mathbf{r}_i - \mathbf{r}_j - \mathbf{D}_{ji}), \quad (8)$$

when the position of node i is updated. Therefore, the algorithm that finds the maximum likelihood solution is the one that finds the $\mathbf{r}_i'$ that minimises E, that is,

$$\mathbf{r}_i' = \left(\sum_j{}' C_{ji}^{-1} \right)^{-1} \sum_j{}' C_{ji}^{-1} (\mathbf{r}_j + \mathbf{D}_{ji}). \quad (9)$$

This can be considered to be a form of Gibbs sampling at zero temperature [Reif, 1982]. One component is optimised by keeping all of the others fixed.

The algorithm derived from equation 9 is given in Fig. 2. The basic principle of the algorithm is to "move each node to where its neighbours think it should be". By iteration, the coordinates in the map converge towards a global minimum in the energy function (equation 7). For on-line map learning, steps 2 and 3 of the algorithm are inter-leaved with the rest of the navigation control software so that the map is adapted continuously during exploration.

3.3 Proof of Convergence

To prove convergence of the algorithm, we show here that the algorithm always minimises the energy function (equation 7). When a node i with position $\mathbf{r}_i$ is updated, its new position $\mathbf{r}_i'$ will be given by equation 9. That is, the new position is picked to minimise the value of $\sum_j (\mathbf{r}_i' - \mathbf{r}_j - \mathbf{D}_{ji})$. Hence, the change in energy will be

$$\begin{aligned} \Delta E &= \sum_j{}' \left\{ (\mathbf{r}_i' - \mathbf{r}_j - \mathbf{D}_{ji})^T C_{ji}^{-1} (\mathbf{r}_i' - \mathbf{r}_j - \mathbf{D}_{ji}) \right. \\ &- \left. (\mathbf{r}_i - \mathbf{r}_j - \mathbf{D}_{ji})^T C_{ji}^{-1} (\mathbf{r}_i - \mathbf{r}_j - \mathbf{D}_{ji}) \right\} \quad (15) \\ &\leq 0. \quad (16) \end{aligned}$$

Since E is bounded below, the algorithm must converge and any legitimate stopping criterion must be reached. The energy function is quadratic and therefore has a unique minimum, so the algorithm can only converge to a global minimum. The fact that the energy function is quadratic also means that the updates could be computed in a single step by inverting the $(N \times N)$ matrix showing connections between all of the nodes in the map. Our system performs an iterative update instead, which is computationally simple and fast, and therefore more suitable for robotics applications.

3.4 Complexity Analysis

The computational cost of the new algorithm is linear in the number of places in the map. Because the algorithm makes an iterative refinement to the existing solution, rather than recalculating the entire coordinate system from scratch, only one iteration is typically required whenever new information is added to the map. The complexity of the algorithm is thus bounded by $O(NM)$, where M is the maximum number of neighbours per node. For a topological map, the number of links per node will not grow with the size of the map, so M is constant and the overall complexity is approximately $O(N)$. This compares favourably with the worst case $O(N^3)$ complexity of matrix inversion methods such as [Lu and Milios, 1997a; Golfarelli *et al.*, 1998]. In our current implementation, the robot attempts to space the nodes at equal intervals of 1 metre, so the maximum connectivity per node is 6 (if the nodes are spaced at equal intervals, then for any node, the minimum angle of incidence between any two of its neighbours will be 60°, where the three nodes form an equilateral triangle, so $M = 6$.)

4 Experiments

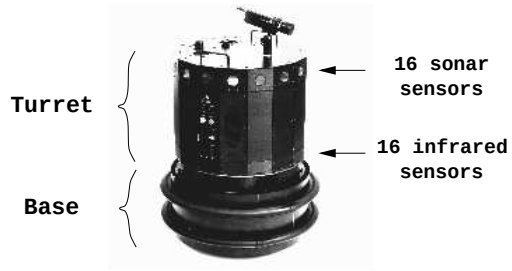

Figure 3: The Nomad 200 mobile robot. The compass is mounted on a turret, which can be rotated independently of the direction of travel.

The new map learning algorithm was tested on a Nomad 200 robot (Fig. 3). To pre-process the compass

1. Initialise the covariance matrix C_{ji} for all links. This could be taken to be the identity matrix, or the covariance matrix computed during a previous run of the algorithm (see step 3).
2. For each node i, do:
   (a) For each of the neighbours j of node i, i.e., the places that are topologically connected to i, obtain an estimate $\mathbf{r}'_{ji}$ of the coordinates of node i using

$$\mathbf{r}'_{ji} = \mathbf{r}_j + \mathbf{D}_{ji}, \tag{10}$$

   where $\mathbf{r}_j = (x_j, y_j)^T$ refers to the coordinates of node j, and $\mathbf{D}_{ji} = d_{ji} \begin{pmatrix} \cos\theta_{ji} \\ \sin\theta_{ji} \end{pmatrix}$.

   (b) Combine the position estimates $\mathbf{r}'_{ji}$ for all j to produce new coordinates $\mathbf{r}'_i$ for node i using

$$\mathbf{r}'_i = \left(\sum_j{}' C_{ji}^{-1} \right)^{-1} \sum_j{}' \left(C_{ji}^{-1} \mathbf{r}'_{ji} \right), \tag{11}$$

   where $\sum'_j$ refers to the sum over the neighbours of node i.
3. Re-estimate the noise parameters in the C_{ji} (equation 5) by maximum likelihood estimation using

$$\Delta_{\delta\delta} = \sum_i \sum_j{}' \frac{1}{NN_iT_{ji}} \left[(R(\phi_{ji})(\mathbf{r}'_i - \mathbf{r}'_{ji}))^T \right]_1 \left[R(\phi_{ji})(\mathbf{r}'_i - \mathbf{r}'_{ji}) \right]_1 \tag{12}$$

$$\Delta_{\alpha\delta} = \sum_i \sum_j{}' \frac{1}{NN_iT_{ji}} \left[(R(\phi_{ji})(\mathbf{r}'_i - \mathbf{r}'_{ji}))^T \right]_1 \left[R(\phi_{ji})(\mathbf{r}'_i - \mathbf{r}'_{ji}) \right]_2 \tag{13}$$

$$\Delta_{\alpha\alpha} = \sum_i \sum_j{}' \frac{1}{NN_iT_{ji}} \left[(R(\phi_{ji})(\mathbf{r}'_i - \mathbf{r}'_{ji}))^T \right]_2 \left[R(\phi_{ji})(\mathbf{r}'_i - \mathbf{r}'_{ji}) \right]_2 \tag{14}$$

   where N is the number of nodes in the network, N_i is the number of neighbours of node i, and $[\cdot]_i$ denotes the ith component of a vector.
4. Repeat from step 2 using the new values of the noise parameters until the change in energy falls below some pre-defined threshold, or some other stopping criterion is reached.

Figure 2: The Maximum Likelihood Map Learning Algorithm.

readings, we used the behaviour-based filtering method described in [Duckett and Nehmzow, 1999]. In this approach, a separate behaviour is used to rotate the robot's turret at small speeds in the direction of an arbitrary 'North', indicated in our experiments by an uncalibrated KVH C100 flux-gate compass[1]. Then the angular estimates are obtained by measuring the relative displacement of the turret against the direction of travel. The effect of this behaviour is to smooth out local fluctuations in the magnetic field of the environment, maintaining a constant orientation around the average value of 'North'. We have found that the performance of this method degrades gracefully with respect to the magnetic variations in the environment. It also has the advantage of keeping the robot's range-finder sensors at a steady orientation, which can help to reduce the noise on the ultrasonic sensor readings.

The new algorithm was tested in the environment shown in Fig. 4, consisting of a relatively large office area (46 m × 12 m). Topological map building was performed using an incremental exploration strategy, in which the robot continually tries to expand the territory that has already been charted (see [Duckett, 2000] for full details). An extra heuristic was added to this strategy to force the robot to close loops, described as follows:

Whenever a new place was added to the map, a search was carried out on each of the adjacent nodes to determine the shortest path actually traversed by the robot from that particular node. If the length of that path exceeded a pre-specified threshold of 3 metres, then the robot was forced to travel directly to that node in order to obtain a measurement of the metric relation for the connecting link. In Fig. 5, the links actually traversed by the robot are shown in bold; the other links were inferred from the sensor readings and the geometric information already contained in the map. Only the physically traversed links were used by the new algorithm in the calculation of the node coordinates (the inferred links were used only for path planning).

The map acquired by the robot in Fig. 5 shows the position of the places in global coordinates calculated by the new algorithm. To illustrate the accuracy of the acquired map, we have also combined the relaxed coordinates with the recorded sonar data to produce a global occupancy grid model of the environment (as reported in [Duckett and Saffiotti, 2000]). Here, we used a Bayesian occupancy grid, as developed by [Moravec and Elfes, 1985]. The derived gridmap is shown in Fig. 6. The map has a resolution of 0.15 metres, and should be accurate enough for safe navigation and planning. This can be compared to the gridmap constructed from the combination of compass and odometry in Fig. 7.

The entire process requires minimal computational resources. Maximum likelihood estimation with the new algorithm was performed on-line as part of the map building process. One iteration of the new algorithm on the full map, consisting of 104 places and 288 physically traversed links, required 17 msec. on a 200 MHz Pentium II processor.

[1] the compass had never been recalibrated following the installation of a pan-tilt camera unit on top of the robot, because this function is no longer supported by the manufacturers of the robot

5 Related Work

[Lu and Milios, 1997a] considered the problem of enforcing geometric consistency in a metric map constructed using laser range-finder sensors without a compass. Their approach maintained a history of all the local frames of sensor data used to construct the map and the network of spatial relations between the frames. The spatial relations were obtained either by odometry or pairwise matching of the range-finder data in adjacent frames, using the scan matching algorithm described in [Lu and Milios, 1997b]. A maximum likelihood algorithm was then used to derive a position estimate for each of the frames, by minimising the Mahalanobis distance between the actual and derived relations over the whole network of frames. A drawback of this method is that it requires the inversion of a $3n \times 3n$ matrix, where n is the number of frames, so the approach is likely to be computationally expensive in large environments. This approach was extended by [Gutmann and Konolige, 1999] to build maps in environments containing large cycles.

A similar approach is described by [Golfarelli *et al.*, 1998], using a graph-based model of the robot's environment. Their algorithm operates under the assumption of no topological errors, so that the robot recognises a place when it visits it for a second time. Their system was based on the analogy of a mechanical spring system, in which each link in the graph is modelled by a pair of springs, a linear axial spring and a rotational one. The elasticity parameters of the springs were used to represent the uncertainty in the robot's odometry measurements, and the equilibrium position for the whole structure was then calculated, generating a $4n \times 4n$ matrix that requires inverting. The algorithm can be applied with or without a compass, although all of their experiments appear to use a compass. Their results seem qualitatively similar to ours (at least by visual inspection) despite the added complexity of the mechanical spring system model.

[Shatkay and Kaelbling, 1997; Shatkay, 1998] addressed the problem of incorporating metric information from odometry into robot maps based on Partially Observable Markov Decision Process (POMDP) models and enforcing geometric consistency in these maps, both with and without a compass. The sensor data from which the models were acquired were first collected by the robot under manual control, then an expectation-maximisation (EM) algorithm was used to find the map which best fitted the recorded data. While this algorithm does not explicitly make the assumption of an initial topological map, in practice it depends heavily on a sufficiently good initial model (obtained in [Shatkay,

1998] from the recorded odometer data) to avoid local maxima and hence build topologically correct maps. The approach would not scale well to larger environments due to the large amount of data needed and the high computational cost of the EM algorithm.

A similar approach is described by [Thrun *et al.*, 1998], where an EM algorithm was used to learn an occupancy grid model of a large environment (90 m × 90 m). Again, this method is extremely expensive, requiring up to two hours of computation to generate a grid map with a spatial resolution of 1 metre, and it depends on a set of manually labelled landmarks. This approach was later extended by [Burgard *et al.*, 1999] to use a network of local gridmaps constructed from sonar data, as in our approach, instead of the pre-defined landmarks.

6 Discussion

In this paper, we presented a new maximum likelihood map learning algorithm for a compass-equipped mobile robot. The algorithm is computationally cheap, enabling fast on-line learning of globally consistent maps. It is particularly efficient because it does not throw useful information away — instead of recalculating the entire map from scratch every time, the existing solution is refined. As a result, only small changes to the map are typically required when new information is added. In the experiments conducted, we found it was only necessary to run the algorithm for a single iteration during each cycle of the map acquisition process. Our results show that a compass enables a very efficient solution to the SLAM problem in semi-structured office environments.

The method works by minimising an energy function in lots of small steps, rather like a Hopfield network [Hopfield, 1982]. Because this energy function corresponds to the log likelihood, minimising this function provides us with the maximum likelihood solution to the map learning problem, provided that the map is topologically correct. We have proved that the algorithm always converges to a globally optimal solution, in contrast to EM algorithms, which are subject to local optima. We should, of course, point out that our method assumes a place recognition system (in other words, no topological errors) before maximum likelihood estimation takes place, an assumption that is not made by [Shatkay and Kaelbling, 1997; Burgard *et al.*, 1999]. However, all of our experiments were conducted on a real, self-navigating robot using a real self-localization system [Duckett and Nehmzow, 1999], thus demonstrating the efficacy of our approach.

The new map learning algorithm is specific to a robot equipped with a compass. Without a compass, we would need to extend the algorithm to deal with the rotation of the robot between nodes, using a 3×3 covariance matrix with estimates of the noise in both θ and ϕ. Obviously, this would require the storage of more angles, and there would be noise in both angular measurements. This is why using a compass, even in indoor environments where there are a lot of fluctuations, is advantageous.

Our complete solution to the SLAM problem harmonizes favourably with the requirements outlined in the companion paper by [Frese and Hirzinger, 2001], as follows:

R1: The hybrid metric-topological representation preserves the "certainty of relations despite the uncertainty of positions".

R2: This representation also means that the memory required for storage grows only linearly with the number of nodes in the map.

R3: The new map learning algorithm is computationally cheap, having a cost that is linear in the number of places stored in the map.

A further advantage of the new algorithm is that all of the parameters in the model are continually re-estimated by the algorithm itself, so we do not need to compute the value of any noise parameters ourselves. This is a distinct advantage for building autonomous robots, since reducing the number of pre-installed parameters reduces the dependence of the robot on a priori knowledge provided by the system designer.

Future work will investigate more intelligent strategies for selecting the landmarks or place signatures in the topological map. The current strategy of adding new places at 1 metre intervals is clearly inefficient, and could be greatly improved. We will also consider building maps in environments containing large cycles.

References

[Burgard *et al.*, 1999] W. Burgard, D. Fox, H. Jans, C. Matenar, and S. Thrun. Sonar-based mapping with mobile robots using EM. In *Proc. of the International Conference on Machine Learning*, 1999.

[Duckett and Nehmzow, 1999] Tom Duckett and Ulrich Nehmzow. Knowing your place in real world environments. In *Proceedings of EUROBOT '99, 3rd European Workshop on Advanced Mobile Robots*, pages 135–142. IEEE Computer Press, 1999.

[Duckett and Saffiotti, 2000] Tom Duckett and Alessandro Saffiotti. Building globally consistent gridmaps from topologies. In *Proc. SYROCO'00, 6th Int. IFAC Symposium on Robot Control*. Elsevier, 2000.

[Duckett *et al.*, 2000] Tom Duckett, Stephen Marsland, and Jonathan Shapiro. Learning globally consistent maps by relaxation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'2000)*, San Francisco, CA, 2000.

[Duckett, 2000] Tom Duckett. *Concurrent Map Building and Self-Localisation for Mobile Robot Navigation*. PhD thesis, Department of Computer Science, University of Manchester, 2000.

[Frese and Hirzinger, 2001] U. Frese and G. Hirzinger. Simultaneous localization and mapping - a discussion. In D. Fox and A. Saffiotti, editors, *Proceedings of the IJCAI-2001 Workshop on Reasoning with Uncertainty in Robotics (RUR'01)*, 2001.

[Golfarelli *et al.*, 1998] Matteo Golfarelli, Dario Maio, and Stefano Rizzi. Elastic correction of dead-reckoning errors in map building. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'98)*, pages 905–911, Victoria, Canada, 1998.

[Gutmann and Konolige, 1999] Jens-Steffen Gutmann and Kurt Konolige. Incremental mapping of large cyclic environments. In *Proceedings of the 1999 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Monterey, CA, 1999.

[Hopfield, 1982] J.J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. In *Proceedings of the National Academy of Sciences*, volume 79, pages 2554–2558, USA, 1982.

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

[Lu and Milios, 1997b] F. Lu and E.E. Milios. Robot pose estimation in unknown environments by matching 2D range scans. *Intelligent and Robotics Systems*, 18:249–275, 1997.

[Moravec and Elfes, 1985] Hans Moravec and Alberto Elfes. High resolution maps from wide angle sonar. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'85)*, pages 116–121, St. Louis, Missouri, 1985. IEEE Computer Society Press.

[O'Keefe and Nadel, 1977] John O'Keefe and Lynn Nadel. *The Hippocampus as a Cognitive Map*. Oxford University Press, Oxford, England, 1977.

[Reif, 1982] F. Reif. *Fundamentals of Statistical and Thermal Physics*. McGraw-Hill, New York, 1982.

[Shatkay and Kaelbling, 1997] Hagit Shatkay and Leslie Pack Kaelbling. Learning topological maps with weak local odometric information. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence*, 1997.

[Shatkay, 1998] Hagit Shatkay. *Learning Models for Robot Navigation*. PhD thesis, Department of Computer Science, Brown University, 1998.

[Thrun *et al.*, 1998] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. A probabilistic approach to concurrent mapping and localisation for mobile robots. *Machine Learning*, 31(5):1–25, 1998.

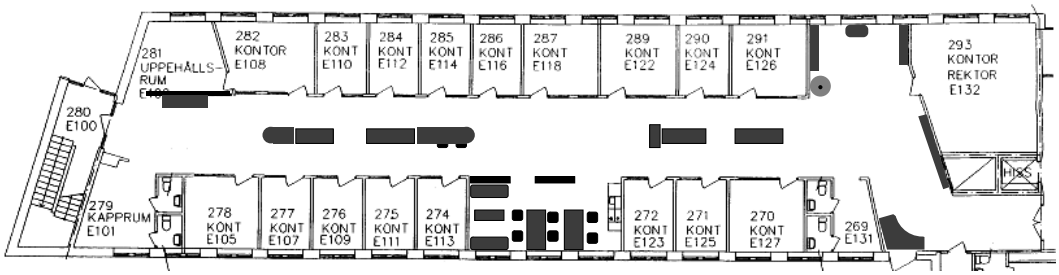

Figure 4: A semi-structured office environment at Örebro University, with an approximate size of 46 × 12 metres.

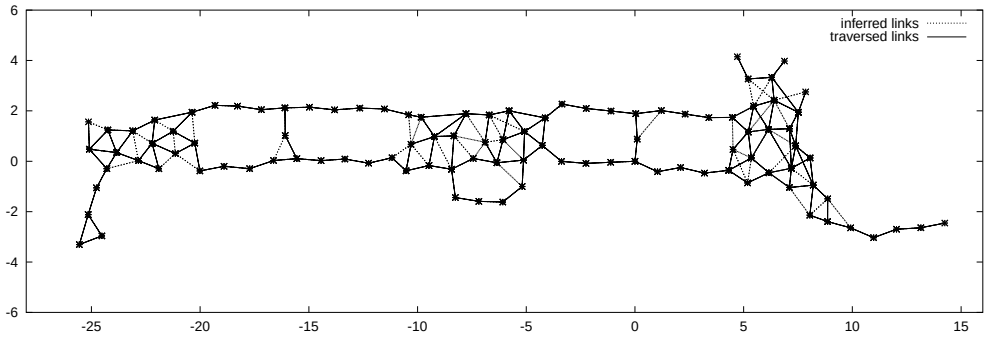

Figure 5: The map acquired with the new map learning algorithm. The links actually traversed by the robot are shown in bold, and the inferred links are shown by dotted lines (only the traversed links were used to estimate the node coordinates).

Figure 6: The global gridmap constructed with the new algorithm, by combining the recorded sonar data with the global coordinates generated by the algorithm.

Figure 7: The global gridmap constructed without the new algorithm, by combining the recorded sonar data with the global coordinates calculated from the compass and odometry. Without the algorithm, the robot fails to build a globally consistent map.