Automatic Map Acquisition for Navigation in Domestic Environments

Philipp Althaus and Henrik I. Christensen
Centre for Autonomous Systems, Numerical Analysis and Computer Science.
Royal Institute of Technology (KTH), S-10044 Stockholm, Sweden
[philipp, hic]@nada.kth.se

Abstract— **An autonomous robot navigating in its environment needs a map representing the large-scale structure of the setting. There are a variety of different types of maps used in mobile robotics: for example, grid maps, maps containing geometrical beacons, or topological maps. Each of those poses its own constraints on the problem of acquiring the map, where its complexity is a major issue. We present a system that is able to build a topological map of large-scale indoor environments in a semi-autonomous way. This means that a person shows the robot around, interacts with it via a basic interface, the platform is following autonomously, and simultaneously creating the map. The acquisition mechanisms are rather simple and computationally inexpensive, due to the low complexity of the topological map. However, experiments show that this map is sufficient to navigate between arbitrary positions in the environment.**

I. Introduction

Mobile robots that perform their tasks in a real-world environment have one common problem to deal with: navigation. This usually includes two types of assignments. On the one hand, for local navigation the robot has to react to immediate sensory events. This is necessary, for example, to drive through narrow passages and, in general, to avoid collision with any kind of obstruction. On the other hand, for global navigation the system needs some kind of map in order to reach a goal, which is too far away to be detected initially by the sensors. For indoor environments, there are several successful navigation systems using various types of maps, depending on the tasks to be achieved and the sensors used by the robot. For example, grid maps [1], [2] are a common approach in sonar based navigation. Other type of maps consist of geometric beacons such as lines [3]. A very different approach than the ones above is the use of a topological map [4]. This type of map contains only information about the connectivity of places in the environment. Together with modules for place recognition, it can be successfully used to navigate in large scale settings.

Whatever kind of map is used, it first has to be acquired. Topological maps are usually created by a human that determines the important places in the setting and, then, creates the map by measuring distances between them. Grid maps and maps consisting of geometric features, however, have to be extracted from sensory data (mostly sonar or laser). This is commonly done by driving the robot with a joystick through the environment, collecting the data, and then processing them offline due to a high computational complexity. This offline creation of a map can be done using methods like Extended Kalman Filters [3] or Bayesian probability theory [5]. The goal of our research was map acquisition for indoor settings in a semi-autonomous way. The scenario is such that a person is guiding through the environment, the robot is tracking this person, following it, and at the same time creating a map. We had two main requirements on this map: 1) the algorithms to acquire it should be computationally inexpensive, and 2) it finally should be good enough to use it for navigation in a large-scale setting.

Our representation of choice was a topological map. Since this type of map is rather simple and contains usually little information. Hence, the necessary processes to acquire it turn out to be inexpensive. The use of such a map for navigation has been demonstrated in our earlier work [6], [7]. There, we developed a navigation system based on the dynamical systems approach to behaviour based robotics [8], [9]. In addition, we implemented modules for place recognition to localise in the topological map. Then, a set of behaviours has been designed to deal with the small scale structure of the environment. Now, an additional behaviour ("person following") has been added. This controller uses sensory data from a laser range scanner to track and follow a person that guides the robot through the area to be mapped. The guide informs the system about the connectivity of the environment via a wireless interface. Using this information and sensory data from sonars the robot is creating the map. These capabilities have clearly the advantage that a representation of the environment can be created without a human measuring the whole area. These additional features of the system have also been integrated into the existing navigation system, which in our mind constitutes a more complete mobile robot.

The topological map and its use for navigation are introduced in Section II. The mechanisms to acquire the map are presented in Section III. In Section IV the results of this acquisition are presented and some details evaluated. Finally, in Section V, an overall discussion, limitations of our approach, and avenues of future research are presented.

II. The Map and its Use for Navigation

We developed a navigation system for large-scale indoor environments for fetch-and-carry type tasks. Typically, the robot has to drive from its charging station to an arbitrary goal point in our institute. A topological map constitutes the knowledge about the large-scale structure of the environment. This map together with modules for place recognition gives the robot the ability to keep track of its position. The design of our system is based on the paradigm of behaviour based robotics [10]. To design the individual behaviours and their coordination we deployed the dynamical systems approach introduced by Schöner and Dose [8], [9]. This approach allows an analytical design of the behaviours and their coordination based on the theory of nonlinear dynamical systems. Below, we first introduce the topological map in Section II-A. Then, in Section II-B, we describe how this map is used for navigation.

A. The Topological Map

A topological map represents distinct places of the environment, which are important for the navigational task. Further, it has to reflect the connectivity of these places. Thus, a common representation for this type of maps is a connected graph. Such a graph is defined by nodes and edges that connect these node. To express the characteristics of the environment, both nodes and edges have associated properties.

In our implementation the nodes have an x- and a y-coordinate in a world-fixed coordinate system. The edges can be of three different types: "corridor", "room", or "door". Thus, nodes have to be placed at every position of the environment where the edge type would change, which means in front of every door. Further, places where the graph splits up (e.g. corridor crossings) have to be represented by a node. In addition, all the places in the environment that are important for the specific tasks carried out (e.g. charging station and goal points) need to be reflected in the map by a node. Fig. 1 depicts the topological map of the large-scale environment of our institute (60×70 meters). This map has been acquired manually by measuring actual distance between nodes. It shows only the part of the premises that is actually accessible by the robot, which means no rooms with high thresholds at the doorway. Depending on the tasks, an arbitrary amount of nodes can be added to the map to include, for example, a mail slot where something has to be picked up, office desks where it has to be delivered to, or any other points of interest.

Fig. 2 shows more details on the placement of the nodes. Nodes in corridors are in the middle of the two walls. The ones in front of doors are aligned with the centre between the door posts. Nodes in rooms are positioned at places that are important for the navigational task. This placing allows the navigation system to effectively keep track of its position and orientation (see Section II-B). Note that there is a redundancy in placing some of the nodes. The distance of a door node to the actual doorway does not really matter as long as it is not more than about 2 meters. Also the position of a goal point does usually not have to be precise, because the controllers of the tasks to be exe-

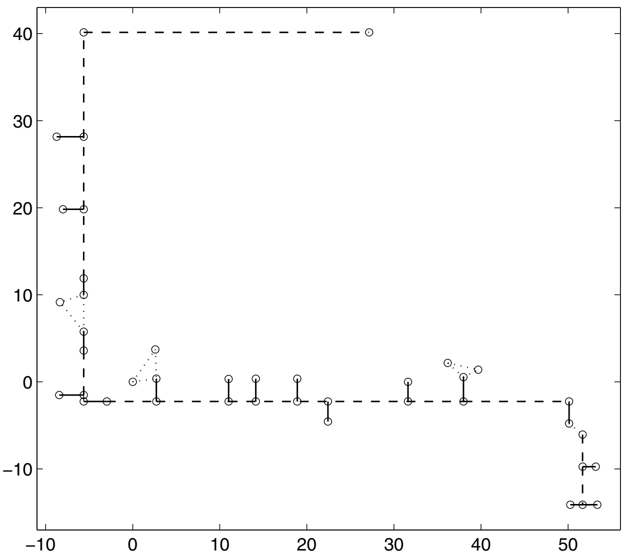

Fig. 1. The topological map of our institute: The circles depict nodes with a position in a fixed coordinate system (units on the axes are in meters). Edges are of three different types: corridor (dashed lines), room (dotted lines), and door (solid lines). The docking station is depicted by the node at the origin. Goal points and other locations of importance for the navigational task can be added arbitrarily in the rooms and the corridors.

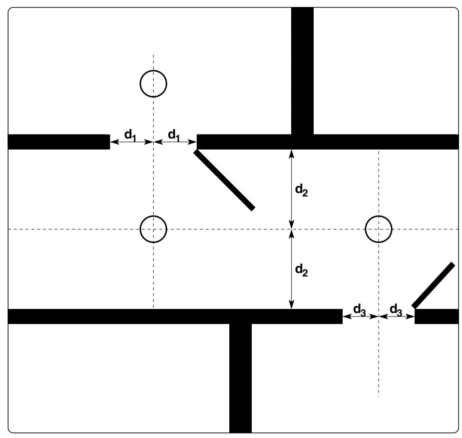

Fig. 2. A schematic drawing of a corridor and two doors leading to rooms. The nodes (depicted by circles) are placed, such that they are centred in the corridor and between the door posts.

Fig. 3. The robot is following a person around the institute.

cuted at this goal point can compensate for that with their sensing capabilities. For example, visual servoing can guide the robot accurately in front of a table in order to pick up an object.

B. Using the Map for Navigation

The platform used in our studies is a Scout robot from Nomadic Technologies (Fig. 3). The platform has a cylindrical shape with a diameter of 38 cm and moves at a speed of up to $1\frac{m}{s}$. Odometric information can be obtained from encoders on the two wheels. The robot is equipped with a ring of 16 evenly spaced ultrasonic sensors. Each sonar has a beam width of 25° and a detection range of 6 to 255 inches. The perception and geometric reconstruction of objects, in order to navigate, is solely based on the information provided by the ultrasonic sensors. A Sick laser scanner has been mounted on top of the robot. This scanner has an angular range of 180° and a resolution of 0.5°. Note that this sensor has only been used for the person following behaviour during map acquisition (Section III). Further, the power system has been extended with two electric contacts at the rear of the robot. This enables the platform to autonomously dock with a power supply in order to recharge its batteries without human interaction. This recharging station can be detected through active IR beacons.

From the sonar data, geometrical representations of the environment are extracted, namely obstacles, walls, corridors, and doors. The properties of these objects are used by a set of 6 reactive behaviours to safely navigate, taking into account the small-scale structure of the environment. The implemented behaviours are: "go to", "obstacle avoidance", "wall avoidance", "corridor following", "door passing", and "docking". The activation of these behaviours is determined by a coordination scheme. It allows switching between different tasks depending on the location of the robot in the topological map. Below, we describe how this location is determined using the robot's odometry and the map. For details on the extraction of the geometrical representations from raw sonar data, the design and implementation of the behaviours, and the coordination scheme, we refer to our earlier work [6], [7].

It is assumed that the initial position and orientation of the robot is known (e.g. charging station). This place is represented as a node in the topological map. The goal position of the navigational task is also reflected by a node. A planner module conducts a depth first search through the graph, which provides a list of nodes and edges that have to be traversed to reach the goal. Depending on the type of edge, a subset of the behaviours is activated to guide the robot to the next node. Each time a new node is reached, the coordination scheme changes the set of active behaviours to drive on to the subsequent node. In other words, the task is split up into a sequence of subtasks consisting of navigating between two nodes (e.g. drive down a corridor or pass a door on the left). Odometric data is used to determine the robot's position and orientation at all times. This introduces errors in the estimation of the exact location of the robot, but is totally sufficient to determine if the system has reached the vicinity of the next node. If this error in the position estimate is not larger than 1 meter, the behaviours themselves are powerful enough to cope with the small-scale structure of the environment in order to accomplish the subtasks.

However, on long trials over a great distance, the error in the robot's position and orientation would grow bigger than desired. To avoid this, the estimates are corrected based on the detected representations of the environment. This correction is done at two different occasions:

1. Each time a corridor is detected:
To detect a corridor the 200 most recent sonar readings are kept in a FIFO buffer. A Hough transform [11] is invoked on the sonar data every five seconds in order to extract the pair of parallel lines (one on either side of the robot) that coincide with the largest number of sonar echos. This process provides an estimate on the relative orientation of the corridor and the robot's distance to the two walls. The actual corridor orientation is the direction of the vector between the two nodes adjacent to the corridor edge (see Fig. 2). This direction is known from the map and can in turn be used to update the estimate of the robot's orientation. Also the position of the centerline is known, which allows to correct the robot's location perpendicular to the walls.

2. Each time a door is passed:
While passing a door the robot keeps track of the narrowest gap, which defines the position of the door and the direction of the goal posts. Also here, the centerline through the door is known from the position of the adjacent nodes (see Fig. 2). This time, the robot's position along the door posts can be updated correctly.

Further, there are other occasion of correcting the robot's pose depending on the tasks. For example, the location of the charging station (reflected as a node in the map) is known. Therefore, after every docking procedure the robot's pose can be determined precisely.

III. Acquisition of the Map

The map is acquired in that a person is showing the robot around. The platform autonomously follows the guide through the environment (Fig. 3). This person carries a laptop to interact with the robot through wireless ethernet. Via this interface, the system is updated about topological changes during their tour. The guide informs the program every time when they leave a room or a corridor, and then again, when a new room or corridor is entered. Also corridor crossings and other points of interest (e.g. docking station) are messaged to the system. During this tour the robot creates a topological map (as introduced in Section II-A), which subsequently can be used for navigation.

To realise this scenario an additional behaviour has been implemented: "person following". The guide is extracted from the laser data taken in the frontal 80° of the field of view. Then, a controller drives the robot towards this person, taking into account its distance and speed. This behaviour is fully integrated into the navigation system. The behaviour coordinator is informed via the interface when "person following" has to be activated and deactivated respectively; in essence when the tour starts and when it ends.

The map acquisition is happening online and begins at the starting location. During a trial the robot has an estimate of its position and orientation using similar processes as in navigation (see Section II-B). First, the system places a node at its initial position. The meaning of the first edge ("corridor", "room", or "door") is entered by the guide. The robot keeps track of its position and orientation through the odometric data. When the system is informed about a new node (e.g. when leaving a room), the robot's position coincides probably not exactly with the nodes correct position as introduced in Fig. 2. Corridors and doorways extracted from the sonar data are used to position the nodes as illustrated in Fig. 4: **A:** The system is informed about leaving the room. While passing the door the narrowest passage (doorway) is extracted from the sonar data. Its position and the orientation

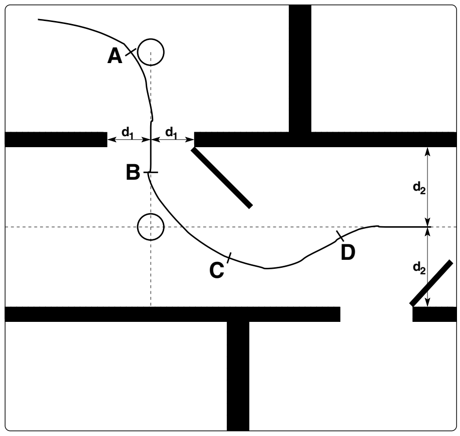

Fig. 4. Correct placement of nodes during the map acquisition process. The black line shows a possible robot trajectory. Nodes are depicted by circles. Situations A-D are explained in the text.

of the door posts are saved. **B:** The system is informed about having entered a corridor. The narrowest passage since A is remembered and the upper node can be placed on the centerline of the door. The robot drives on along the corridor. **C:** The corridor properties (orientation and distance to the walls) are extracted from the sonar data. **D:** The corridor properties are extracted again and averaged with the first guess. This gives a good enough estimate to place the second node centred in the corridor and aligned with the first node. This whole process happens each time the platform moves from a room to a corridor and, similarly, the other way round. Further, nodes of goal points are placed at the location of the robot's position estimate at the time the system is informed about placing a node. Each time the guide tells the robot about a new node, it also announces the type of the next edge.

IV. Results

Fig. 5 shows the topological map acquired by the robot. This is the result of one particular trial. All experiments result in their own maps, which do not look exactly the same. This is generally due to odometry drift and disturbances at door thresholds. However, their difference is in the same order as the discrepancy with the correctly measured map introduced in Section II-A. Hence, we focus our discussion on comparing Figs. 1 and 5. In general, experiments have shown that the maps acquired by the robot function equally well for navigating between two arbitrary nodes as the precise map in Fig. 1.

The most obvious difference between the two maps is that the angles between edges are not the same. This is particularly well visible at the long corridors. However, this does not matter, when the map is used

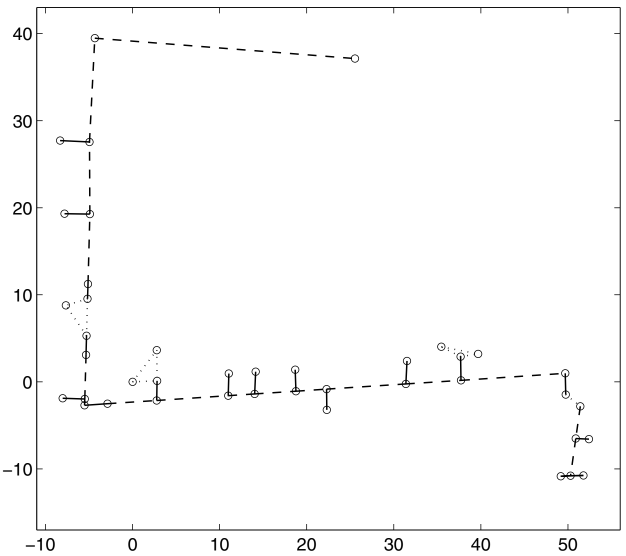

Fig. 5. The map of our institute acquired by the robot (units are in meters): The circles depict nodes. Edges are of three different types: corridor (dashed lines), room (dotted lines), and door (solid lines). The docking station is depicted by the node at the origin.

Fig. 6. A cluttered corridor in our institute. These types of obstructions do not pose any problems for navigation and map building.

later. Remember from Section II-B that the navigational tasks are split up into subtasks that consist only of navigating from one node to a neighbouring one. At the nodes themselves, task switches are invoked and the behaviours take care of the small scale structure of the environment. For example, when the robot has to enter a room from the corridor, the behaviour "door passing" guides the platform through the doorway, where the robot's orientation relative to the map is corrected again. Hence, the tasks do not fail as long as corridors and doors can not be mixed up. However, this would only occur when angular errors were in the order of 45°.

Another discrepancy between the precise map and the one in Fig. 5 is the length of the door edges. This is hard to see in the figures, but they actually vary up to 1 meter. These differences, however, are not errors, since door nodes in rooms can be placed anywhere on the centerline of the door (see Section II-A). This placement does not affect the executions of the navigational tasks.

One aspect of the map that is crucial for the navigation system to succeed is the length of the corridor edges, in order to find the correct door. Comparing these in our initial experiments with the correct map suggests that the relative error is at most 0.5%. The error of the position estimate during navigation is also at most 0.5%. Let's assume that the system is able to find the correct door, if its position relative to the map is less than 1 meter. This suggests that our approach could successfully handle corridors up to a length of 100 meters.

Another aspect that the navigation system has to rely on is the length of the edges inside rooms and the angles between them. The errors in length are equally small as in corridors. However, the errors in angle are of the same magnitude as between individual corridors, because no features of rooms are used to update the estimates of the robot's orientation. This could lead to problems in two cases: 1) if the rooms are very large, and 2) if the robot navigates inside the same room for a long time. Experiments show that rooms of the size of our offices do not pose any problems for navigation to a few goal points. In addition, we believe that a task that demands frequent navigation between nodes in the same room (e.g. fetch-and-carry between two tables) provides additional means to update the robot's orientation (e.g. extracting the tables from sensory data). However, additional studies need to be carried out to verify this believe.

The above mentioned deficiency, of generally not using any features in the rooms, provides an advantage in modelling the whole environment. Since we do not make any assumptions on the structure of a room, everything that does not look like a corridor or a doorway can be modelled as a room. However, this does not mean that cluttered corridors need to be defined as rooms. Fig. 6 shows an example of such a corridor that did not pose any problems to the behaviour "corridor following" to safely navigate. Neither was it a problem to successfully update the robot's orientation estimate in both navigation and map construction.

V. Discussion

A navigation system and its use of a topological map have been introduced and a rather simple mechanism for map construction presented. This new feature enables a robot to follow a person through an environment and simultaneously building a map. Experiments have shown that this map can successfully be applied to navigate through our whole institute. All the mechanisms have been integrated into one complete navigation system for large-scale domestic environments.

To emphasize the simplicity and robustness of our approach, sonars are the only sensors used for navigation and detecting features for map construction. Further, the whole environment is modelled as a set of corridors, rooms, and doorways. The system using this minimalistic approaches, nevertheless, is fully capable of performing the navigational tasks without relying on any geometric maps of features in the setting. This simplification allows to acquire a representation of the entire environment with rather simple mechanisms and low CPU consumption. The latter permits to build the map online without any post-processing. This is clearly an advantage to SLAM approaches deploying Kalman Filters and Bayesian theory together with a complex world model. Recent work shows that also these methods can be applied in real-time (see [12], [13] for two excellent examples). However, in these examples sophisticated algorithms had to be developed to be able to restrain CPU and memory consumption. We have shown that a map that is only used for navigation can be acquired in a much easier fashion.

The use of sonars, as the only sensors for navigation and map acquisition, restricts our system in different ways, which might potentially lead to problems in less structured environments. Future research in this project will be directed towards integration of the laser scanner into place recognition procedures to get a more detailed representation of the small-scale structure of the environment. Also the problem of global localization (neglected in this paper) using just a simple topological map, can only be solved with more sophisticated sensing capabilities. Further, the person tracking, used for following the guide through the environment, runs into troubles when several moving objects are present. The implementation of a more advanced tracking algorithm (e.g. [14]) will improve this part of the system. In addition, the interaction of the guiding person with the robot is not particularly user-friendly. The little information given by the guide (presence of a node and type of the edge) could as well be communicated via a speech interface. This enhancement will make the system much easier to use.

Acknowledgment

This research has been sponsored by the Swedish Foundation for Strategic Research. The support is gratefully acknowledged.

References

[1] A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE Journal of Robotics and Automation*, vol. 3, no. 3, pp. 249–265, 1987.

[2] H. P. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI Magazine*, vol. 9, no. 2, pp. 61–74, 1988.

[3] J. J. Leonard and H. F. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 376–382, 1991.

[4] I. Horswill, "The Polly system," in *Artificial Intelligence and Mobile Robots: Case studies of successful robot systems*, D. Kortenkamp, R. P. Bonasso, and R. Murphy, Eds., chapter 5, pp. 125–139. MIT Press, Cambridge, MA, 1998.

[5] S. Thrun, A. Bücken, W. Burgard, D. Fox, T. Frölinghaus, D. Hennig, T. Hofmann, M. Krell, and T. Schmidt, "Map learning and high-speed navigation in RHINO," in *Artificial Intelligence and Mobile Robots: Case studies of successful robot systems*, D. Kortenkamp, R. P. Bonasso, and R. Murphy, Eds., chapter 1, pp. 21–52. MIT Press, Cambridge, MA, 1998.

[6] P. Althaus, H. I. Christensen, and F. Hoffmann, "Using the dynamical system approach to navigate in realistic real-world environments," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001, pp. 1023–1029.

[7] P. Althaus and H. I. Christensen, "Behaviour coordination for navigation in office environments," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002, pp. 2298–2304.

[8] G. Schöner and M. Dose, "A dynamical systems approach to task-level system integration used to plan and control autonomous vehicle motion," *Robotics and Autonomous Systems*, vol. 10, pp. 253–267, 1992.

[9] G. Schöner, M. Dose, and C. Engels, "Dynamics of behavior: theory and applications for autonomous robot architectures," *Robotics and Autonomous Systems*, vol. 16, no. 2–4, pp. 213–245, 1995.

[10] R. C. Arkin, *Behavior-Based Robotics*, MIT Press, Cambridge, MA, 1998.

[11] J. Forsberg, U. Larsson, and Å. Wernersson, "Mobile robot navigation using the range-weighted hough transform," *IEEE Robotics & Automation Magazine*, vol. 2, no. 1, pp. 18–26, 1995.

[12] J. E. Guivant and E. M. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real-time implementation," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 242–257, 2001.

[13] S. Thrun, W. Burgard, and D. Fox, "A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2000, pp. 321–328.

[14] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers, "Tracking multiple moving objects with a mobile robot using particle filters and statistical data association," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2001, pp. 1665–1670.