Acquiring Route Graphs from Human Instructions

Andrea Bauer and Martin Buss

Institute of Automatic Control Engineering
Technische Universität München
80290 Munich, Germany
ab@tum.de,mb@tum.de
http://www.lsr.ei.tum.de

Abstract. A mobile robot that is given the task to get to a certain unknown location needs to acquire the missing route information in order to achieve the task. It is the goal of this work to use human-robot communication to gain the necessary route information. The robot asks humans for directions and builds a topological route graph from their answers. The robot then follows this route adding metrical coordinates while moving, thus validating the route graph and transforming it into world coordinates.

1 Introduction

An autonomous robot designed to operate in a highly complex human populated environment should be able to find its way to a given goal without previous route knowledge. The robot could get the missing route information from other robots which have already achieved this task or from web based information services or through exploration (which is very time-consuming in a complex environment). Another option is to ask humans for directions. Bugmann et. al. [1] use instruction based learning for a wayfinding problem where the robot translates the instructions into a sequence of motor commands that will lead it to reach a certain goal. When the robot is supposed to really learn a route to be able to follow it again, the human instructions should be included in a route graph (for modelling of navigational knowledge as route graphs, see [2]).

The aim of the Autonomous City Explorer project (ACE) [3] is to have an outdoor robot navigate autonomously to a given goal in the city of Munich without any prior map knowledge or GPS. The robot needs to acquire the missing route knowledge through human-robot communication. Therefore a system is envisioned that uses human instructions to build a topological route graph, verifies it while following it, and transforms it into a metrical route graph.

2 Overview of the Route Graph Acquisition

The robot (with a system as shown in Fig. 1) asks a passer-by for directions to a certain goal. It uses the human answers to build a topological route graph that includes uncertainties (taking into account the sometimes inaccurate human knowledge and possible misunderstanding). The robot will keep asking other humans as long as the route knowledge is not completed and the goal is not reached. When the robot has got a topological route graph, it will follow the route transforming it into a metrical route graph by including the real world coordinates of the nodes (i.e. the crossings).

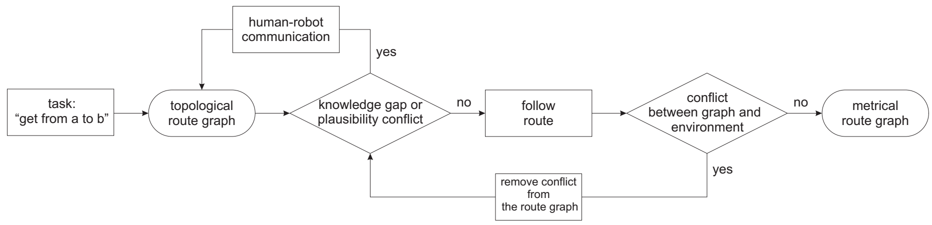

Fig. 1. Overview of the route graph acqusition system

2.1 The Route Graph

The route information is stored as a graph, where the nodes $N_i(x_i, y_i, c_i)$ denote crossings with the coordinates (x_i, y_i), and an additional certainty value c_i. The certainty value ranges from 0 (the coordinates of the crossing are random values, there is no knowledge of the real location) to 1 (the coordinates are correct). Certainty values between 0 and 1 represent information given by humans, the coordinates give relative directions, no absolute coordinates.

The edges denote actions that connect the crossings (e.g. follow the road until the next crossing is reached). The adjacency matrix has values between 0 (no connection) and 1 (certain connection). Values in between 0 and 1 are used to represent uncertainty due to possible misunderstanding or other inconsistencies, such as obstacles the human does not know about.

2.2 From Human Instructions to a Topological Route Graph

The robot starts only with the knowledge of the current position $N_{start}(0, 0, 1)$ and a given goal $N_{goal}(x_{goal}, y_{goal}, 0)$ (the goal coordinates are assigned randomly). The adjacency matrix is a 2x2 zero matrix. The robot will ask a human for a way from the current position to the goal and wait for answers that describe a route that is based on crossings as waypoints and the directions to crossings. The route description given by the human will be translated by the robot into the topological route graph, denoting the directions of crossings relative to the previous ones (i.e. left: x=0, y=-1; right: x=0, y=1; straight: x=1, y=0), the certainty value is set to 0.5. The adjacency matrix is expanded by one row and one column per crossing and the value of the connection between the crossings is set to 0.5.

2.3 From Topological to Metrical Route Graph

The robot has now got topological route knowledge, so it can start following the route towards the goal. If the certainty of the next crossing coordinates c_{next} is 1, it can move exactly to the associated coordinates (x_{next}, y_{next}). If the certainty c_{next} lies between 0 and 1, it follows the street along the given direction until it reaches a crossing. When a crossing is reached, the coordinates are changed from the relative directions to absolute coordinates, the certainty is set to 1 (although a small uncertainty remains due to imperfect sensors). If at some point the robot cannot follow the route in the given direction, e.g. when it gets to a dead end, it will delete the connection to the next crossing from the adjacency matrix and ask another human for directions.

3 Simulation

As a first implementation the robot is simulated in Player/Stage [4]. The robot operates in a maze-like environment (dimensions: 30m x 30m) and is equipped with a laser range finder. The Stage simulation is shown in Fig. 2 (left), the robot (red circle) starts in the upper left, the goal (red square) is in the lower right. The human is shown this map including the robot and is asked to explain the route to the robot. For example:
Robot: Excuse me, how can I get to the red square?
Human: Go straight, then turn left. Take the second turn to the right.
The robot interprets the crossings as nodes $N_2(1, 0, 0.5)$, $N_3(0, -1, 0.5)$, $N_4(1, 0, 0.5)$,

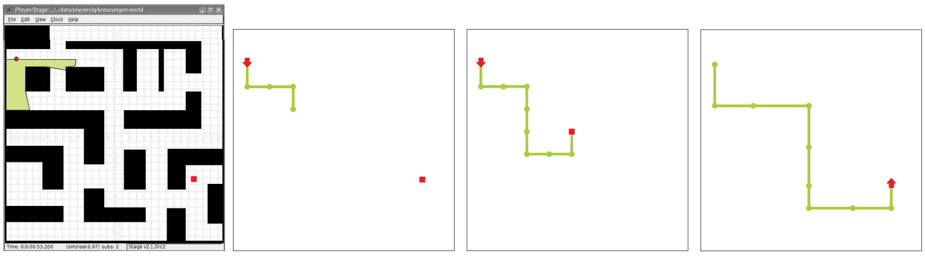

Fig. 2. Left: The environment simulated in Stage with the robot (red circle, laser depicted in green). Center: topological route graph with route knowledge, the edges have unit length. Right: metrical route graph acquired by following the route.

and $N_5(0, 1, 0.5)$ and inserts the connections between the nodes into the adjacency matrix. The route graph is now topological as it includes the directions of the single nodes relative to their neighbors, it contains no real world coordinates. The topological route graph is displayed to the human as shown in Fig. 2 (left center). As the graph is not complete the robot will ask the human again for more information:

Robot: How can I get from there (the last crossing) to the red square?
Human: Follow the street to the second crossing, turn left, go straight until you reach the second crossing, turn left again, and you are there.
The robot includes this new route information analogously into the topological route graph, see Fig. 2 (right center). When the human concludes with the reaching of the goal, the goal coordinates will be adapted to the route graph relative to the previous crossing and the certainty c_{last} is changed from 0 to 0.5.

When the robot has completed the communication with the human, it starts following the route. (It could have as well done this before it had known the entire route. Then it would have had to get back to asking a human at a later point, which is in fact the technique humans use to find their goal in an unknown city.) It turns in the direction given by the next node and moves in that direction until it reaches a crossing, there it sets the coordinates of the node to the measured real world coordinates and the certainty to 1, thus changing the information from topological to metrical. It repeats this behaviour until it reaches the goal where consequently it has got a complete metrical route graph, as shown in Fig. 2 (right).

4 Conclusions

A system is presented that asks humans for directions to a certain goal, includes this information into a topological route graph including uncertainties, and finally compares this graph to the real environment while following the route and transforms it into a metrical route graph. The system was implemented and simulated on Player/Stage.

The system will be implemented on an autonomous outdoor robot (the Autonomous City Explorer) and validated later this year. Also a plausibility check of the different human's answers will be included.

Acknowledgments. This work is supported in part within the DFG excellence initiative research cluster *Cognition for Technical Systems – CoTeSys*, see also `www.cotesys.org`.

References

1. Bugmann, G., Lauria, S., Kyriacou, T., Klein, E., Bos, J., Coventry, K.: Using Verbal Instructions for Route Learning: Instruction Analysis. In Proceedings Towards Intelligent Mobile Robots, Manchester. (2001)
2. Werner, S., Krieg-Brückner, B., Herrmann, T.: Modelling Navigational Knowledge by Route Graphs. In Spatial Cognition II, Integrating Abstract Theories, Empirical Studies, Formal Methods, and Practical Applications, pp 295–316. Springer, London. (2000)
3. Lidoris, G., Klasing, K.,Bauer, A., Xu, T., Kuehnlenz, K., Wollherr, D., Buss, M.: The Autonomous City Explorer Project: Aims and System Overview, In Proceedings of the Int. Conference on Intelligent Robots and Systems, pp. 560–565. (2007)
4. The Player Project, `http://playerstage.sourceforge.net/`