Robot Learning for Moving between Subgoals

Katsu MORIZANE* Seiji YAMADA†
Jun'ichi TOYODA*

*ISIR, Osaka University
8-1 Mihogaoka, Ibaraki, Osaka 567, JAPAN
E-mail: morizane@ymd.ssie.titech.ac.jp
E-mail: toyoda@ai.sanken.osaka-u.ac.jp
†Department of Computational Intelligence and Systems Science
Interdisciplinary Graduate School of Science and Engineering
Tokyo Institute of Technology
4259 Nagatsuta, Midori-ku, Yokohama 226, JAPAN
E-mail: yamada@ssie.titech.ac.jp

Keywords: Robot Learning, Distinctive place, Navigation

Abstract

In this research, our goal is that a mobile robot learns to move between subgoals in a real environment without interaction with humans. The robot does not have knowledge on the environment: positions of subgoals and obstacles. There are affects of various errors in experiments of a robot learning in a real world, and thus the robot learning is unavoidably influenced by the errors. We pay attention this point, and suggest how to learn to move efficiently between subgoals with resetting errors. In our system, we define the distinctive place on which the movement control of a robot changes from sensor-based to coordinates-based. The distinctive place is identified only with local information, and the cumulative errors through the movement between two subgoals are reset on distinctive places. First, the robot moves randomly and tries to find subgoals. Next, it moves between subgoals repeatedly, and learns to move efficiently. We made experiments in a real environment, and found out that the robot's movement was gradually improved by learning.

1 Introduction

In almost previous works in the area of robot motion planning [Latombe 91], the task is finished when the robot arrived at its destination at once, and there is no improvement by experience. In the view of practical applications, the task is insufficient. Because a robot may move around an environment constantly, and is hoped to be able to move more efficiently through the exploration. Furthermore, in a real environment, there is no helpful teacher for the learning. Thus we attempt to build a robot that learns to move efficiently between subgoals without interaction with humans. In a real environment, the sensor data include cumulative errors. Hence we consider the learning to move efficiently is done by resetting the errors sometimes. This work is concerned with a learning procedure to come and go to subgoals freely for mobile robots operating in an unknown real environment with obstacles. Subgoals are defined as the places that mobile robots recognize and identify only if they are on. The positions of each ones, however, are unknown in advance.

A mobile robot used for our experiments is called as *Micromouse* (Fig.2). Micromouse is a small robot equipped with eight IR proximity sensors, a rotation angle detector and a distance meter. The estimation of the current location of the robot is done by dead reckoning.

In recent years, a number of sensor-based path-planning algorithms that deal with errors of the position estimation have been developed in robotics.

In LOGnet [Malkin 90], a distinctive place is formed by an omni-directional sonar pattern. The system identifies a distinctive place by pattern matching with rotation, and experiments were made in a real environment. Unfortunately, the computation of the pattern matching is expensive. Moreover, LOGnet does not deal with dead reckoning error, and the robot does not improve the behavior by learning.

Noborio has described deadlock-free path planning algorithms in the presence of a dead reckoning error [Noborio 95]. These algorithms can deal with obstacles in arbitrary shape. These assume, however, that the robot can recognize revisiting to the special points in spite of a dead reckoning error. In general, a robot with dead reckoning can not recognize revisiting to a point without additional guidance system.

In a real world, there are effects of several estimation errors on mobile robots. Some of these, for example, distance estimation errors and orientation estimation errors, affect on position estimations by dead reckoning seriously. Therefore, for estimating the robot location, additional means such as beacons [Skewis 92] are necessary. We develop a learning procedure that does not need such an additional guidance system.

The outline of learning process is following. First, for finding subgoals, the robot moves around the real environment using initial behavior rules. Then, the robot attempts to move from some subgoal to another one. However, because the coordinates of subgoals got in the first step contain dead reckoning errors, the robot can not move to the destination subgoal correctly. In our procedure, we reduce this cumulative error at distinctive places [Kuipers 87].

We explain a distinctive place in §2, a mobile robot used for our experiment in §3 and our learning procedure in §4. In §5, we show an experiment with Micromouse. In §6, we discuss some limitation of our procedure. Finally, we conclude this research and present some future work in §7.

2 Distinctive Place

There are two control methods for mobile robots in a real world; a *position-based* and a *sensor-based method.*

In position-based control, the motion direction of mobile robots depends on the current position and the destination position. When a robot has no sensor data of obstacles in the goal direction, it is controlled in the position-based way.

In sensor-based control, the motion direction is decided not to collide obstacles by taking advantage of sensor data. When there is sensor data of obstacles in the goal direction and the distance to the obstacle is less than a threshold, the robot becomes to be controlled in the sensor-based way. The sensor-based control is used for avoiding obstacles by *wall-following* [Ando 95]. In general, a sensor-based control is known robust. Our robot uses the two approaches depending on situations.

In this work, we define a *distinctive place* [Kuipers 87] as the place on which the control method for a mobile robot changes from sensor-based one to position-based one. In other words, at a distinctive place, a mobile robot stops avoiding obstacles by sensor-based wall-following and goes straight at the destination subgoal. The distinctive place is usually a convex corner of a obstacle and identified only with local information. A corner of an obstacle is detected by missing the sensor echoes in the wall direction. Thus it can be found robustly. In Fig.1 , A and

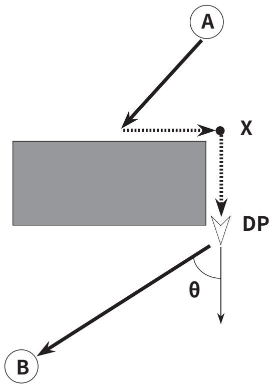

Figure 1: Distinctive place

B are subgoals, and DP is a distinctive place. Because the direction toward B is obstructed at corner X, X is not a distinctive place.

If a robot reaches a corner and the direction toward the destination subgoal is open, it will register the place as a distinctive place by recording the following data.

(a) The position and the orientation of the robot.

(b) Degrees of the *dw-angle* (θ in Fig.1) : an angle between the direction of the destination subgoal and the direction of a wall have been followed just now.

(c) The number of corners of an obstacle that are passed until the robot reaches the distinctive place.

(c) is used to identify the distinctive place afterwards. We make use of visiting a distinctive place to reset the coordinates and the orientation data with cumulative error to ones the robot had when it visited that place previously.

3 Micromouse : the robot used for our experiments

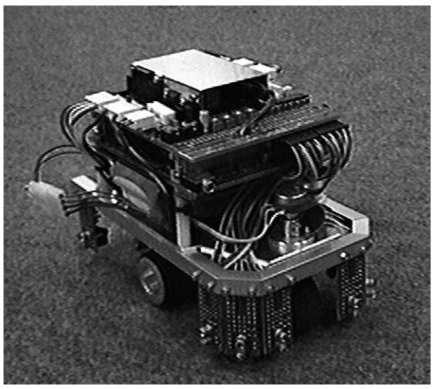

Figure 2: The robot used for our experiments

A mobile robot used for our experiment is called as *Micromouse* (Fig.2). Micromouse is a very compact robot equipped with eight IR proximity sensors, a rotation angle detector and a

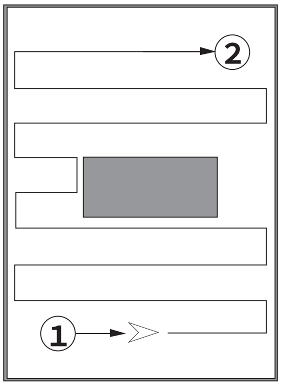

Figure 3: Initial movement - Search for Subgoals

distance meter. The orientation of the robot is calculated by accumulating each rotation angle from the beginning, and naturally contains some errors. The estimation of the current location of the robot is done by dead reckoning. The location estimation error is largely due to the orientation estimation error, and an estimate of distance traveled is relatively accurate.

The robot is as large as 10×18×13 centimeters. The robot has two independently driven wheels on both sides. Two wheels placed in front and back are steered in the reverse direction respectively and thus the robot can rotate.

4 Learning Procedure

A learning process is mainly divided into the three steps. The first step is a initial movement to search for subgoals. The second step is a discovery and a registration of distinctive places. The third step is a path refinement using distinctive places. In the followings, for simplicity, we explain the case that there are only two subgoals in an environment.

4.1 Initial movement - Search for Subgoals

Because locations of subgoals are not given initially, the robot must move around its environment to get information about the locations. Note that the estimations of movements of the robot contain errors, and thus acquired information is inaccurate.

Step1 : Initial movement

(1-1) If a robot reaches two subgoals, go to Step2 in the next section.

(1-2) The robot goes straight in an arbitrary direction.

(1-3) If the robot reaches a subgoal, it records the position coordinates of the subgoal and go to (1-1).

(1-4) If its movement direction is obstructed by an obstacle, it avoids the obstacle by wall-following in a given follow direction, left or right, until its travel distance is up to a given *following distance*. Then the robot turns to the direction that is perpendicular to the direction of the wall and goes straight. Set the follow direction to the reverse one, and go to (1-3).

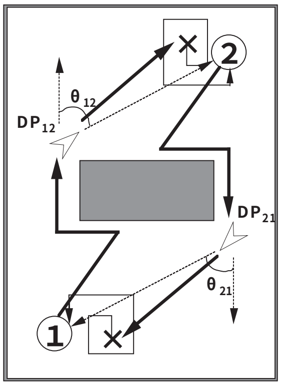

Figure 4: Discovery and Registration of distinctive places

The following distance is given to a robot. In our experiments, 20 centimeters are used . Fig.3 shows the initial movement.

4.2 Discovery and registration of distinctive places

The robot will go straight at the destination subgoal with the information obtained in the initial movement if its direction toward the subgoal is not obstructed by any obstacle. Otherwise, it will avoid the obstacle by wall-following in a given local direction, left or right, until its direction toward the subgoal becomes to be open. Fig.4 shows the discovery and the registration of distinctive places. In Fig.4 , the local direction is left. The detail procedures are followings.

Step2 : Finding distinctive places

(2-1) If the robot return to the origin subgoal after it reached the other one, go to Step3 in the next section.

(2-2) The robot goes straight at the destination subgoal.

(2-3) If the robot encounters an obstacle, it will follow the obstacle boundary in the local direction, until one of the following occurs:

(2-3-a) If the robot reaches the destination subgoal, go to (2-5).

(2-3-b) If the robot reaches a corner and the direction toward the destination subgoal is open, it will register this place as a distinctive place. Go to (2-2).

(2-4) If coordinates of the robot are equal to ones of the destination subgoal and the robot is not on that subgoal in fact (such places are marked $\times$ in Fig.4), it will search around for the subgoal by a spiral movement. This is called as *local search*. Go to (2-5).

(2-5) If the robot reaches a subgoal, degrees of turn angle at the registered distinctive places (θ in Fig.4) are recomputed and are updated. Then the robot goes straight at the other one and go to (2-1).

4.3 Path refinement

In this step, the processes are almost the same as in Step2. Step3 is different from Step2 in that the processes continue until the local search becomes unnecessary and that the robot uses the data of registered distinctive places for the path refinement (Fig.5).

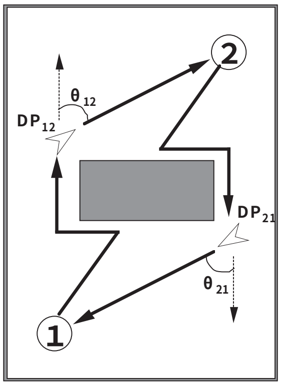

Figure 5: Path refinement

Step3 : Path refinement

(3-1) If the local search is not necessary, finish.

(3-2) The robot goes straight at the destination subgoal.

(3-3) If the robot encounters an obstacle, it will follow the obstacle boundary in the local direction, until one of the following occurs:

(3-3-a) If the robot reaches the destination subgoal, go to (3-5).

(3-3-b) If the robot reaches a corner and its direction toward the destination subgoal is open there, using the information of corresponding registered distinctive place, it will reset the position and the orientation data to previously recorded one. Furthermore, the robot rotates with the dw-angle. Go to (3-2).

(3-4) If coordinates of the robot are equal to ones of the destination subgoal and if the robot is not on that subgoal in fact, it will search around for the subgoal by the local search. Go to (3-5).

(3-5) If the robot reaches a subgoal, degrees of turn angle at the registered distinctive places, θ in Fig.5 , are recomputed and are updated. Then the robot will go straight at the other one and go to (3-1).

5 Experiment

We made an experiment in an environment with two subgoals and one obstacle (Fig.6). The environment size is 180 by 108 centimeters, and the obstacle it contains is a rectangle, 36 by 18 centimeters. The subgoal is a circular plate whose radius is 18 centimeters. In this experiment, the robot recognizes that it is on a subgoal by a signal from a human observer.

Fig.7 shows a top view of the environment and trajectories of the robot. For easy understanding, trajectories of the initial movement and the spiral movements are not drawn. The robot robustly reached the distinctive place where it had reached earlier, and also could gradually converge the dw-angles to the true value. Eventually, the robot could come and go to the subgoals successfully.

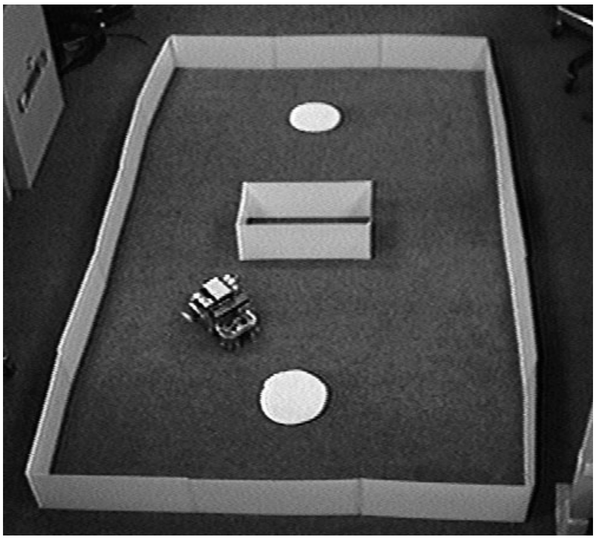

Figure 6: Experimental environment

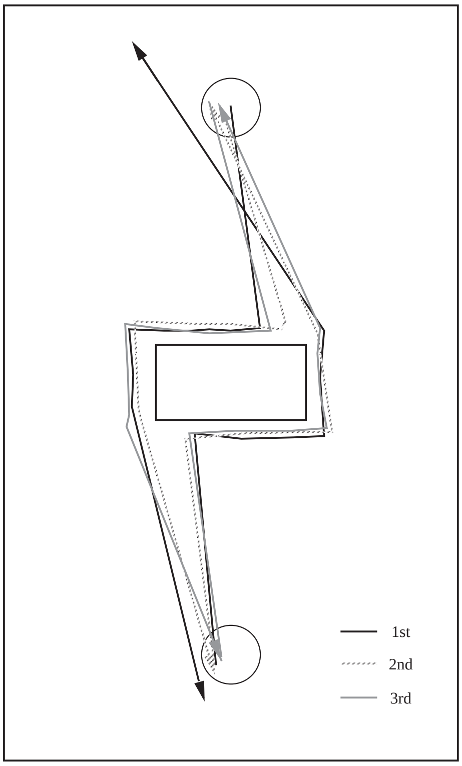

Figure 7: Experiment

6 Discussion

There are two problems about the initial movement. First, because the initial movement procedure is a heuristic approach, the robot can not always get to subgoals successfully. Second, the environmental data acquired in the initial movement is only about positions of subgoals, not obstacles. We are developing a framework to make use of the other environmental data. In detail, we are attempting to generate a potential-field that reflects information of obstacles followed by wall-following partially in the initial movement and that guides the robot to the shorter way to the destination subgoal when it avoids the obstacles afterwards.

In this work, the distinctive place is on a corner of an obstacle. The corner of the obstacle is detected by missing the sensor echoes in the wall direction. In order to find the corner of the obstacle, that is, the distinctive place robustly, the obstacle must be a polygon.

Basically, we assume that the robot visits the same obstacles at each time when it moves between the same subgoals. If a positioning error is largely and the robot visits a different obstacle from the obstacle visited earlier, the robot will neither reach the correct distinctive place nor reach the destination subgoal. This is our open problem.

7 Conclusion

In this paper, we suggested a learning procedure to come and go to subgoals freely for a mobile robot in an unknown real environment with obstacles. We made an experiment on the situation that there are not a few distance, orientation and location errors. The experimental result showed that the robot became to move between subgoals successfully without adding any guidance systems like beacons.

In future, we will improve the procedure of initial movement and make several experiments in more complex environments with many obstacles and many subgoals.

References

[Latombe 91] Latombe, J.C., Robot Motion Planning, *Kluwer Academic Publishers*, 1991

[Kuipers 87] Kuipers, B.J., Byun, Y.T., A Qualitative Approach to Robot Exploration and Map-learning, *In Proc. of Workshop on Spatial Reasoning and Multi-Sensor Fusion*, pp.390-404, 1987

[Malkin 90] Malkin, P.K., Addanki, S., LOGnets : A Hybrid Graph Spatial Representation for Robot Navigation, *In Proc. of the eighth AAAI-90*, pp.1045-1050, 1990

[Skewis 92] Skewis, T. and Lumelsky, V., Experiments with a Mobile Robot Operating in a Cluttered Unknown Environment, *In Proc. of the 1992 IEEE Int. Conf. on Robotics and Automation*, pp.1482-1487, May 1992

[Noborio 95] Noborio, H., Yoshioka, T., and Hamaguchi, T., On-Line Deadlock-Free Path-Planning Algorithms in the Presence of a Dead Reckoning Error, *In Proc. of the 1995 IEEE Int. Conf. on System, Man, and Cybernetics*, pp.483-488, 1993

[Ando 95] Ando, Y. and Yuta, S., Following a Wall by an Autonomous Mobile Robot with a Sonar-Ring, *In Proc. of the 1995 IEEE Int. Conf. on Robotics and Automation*, pp.2599-2606, 1995