MAP EXPLORATION USING A LINE-BASED FORMATION OF MOBILE ROBOTS

Bart Wyns, Jens Boeykens and Luc Boullart

Dept. of Electrical Energy, Systems and Automation, Ghent University, Technologiepark 913, Zwijnaarde, Belgium

Keywords: Map exploration, Formation strategies, Mobile robots, Concave obstacles.

Abstract: Exploration of an unknown environment is a well-studied problem for single robot systems. However, using just a single robot limits the speed in which a map can be fully explored. Using a multi-robot approach, a noticeable performance gain can be achieved. In this article a line-based formation strategy to explore a static area is introduced, without making any assumptions about the shape of the obstacles within. A software simulator including this line-based formation strategy was built to evaluate performance in different environments. Results show that the robot formation can easily get around both convex and concave obstacles whilst constructing a map that is both complete and correct.

1 INTRODUCTION

Map exploration is one of the most widespread tasks performed by mobile robots, due to many practical applications. Mine field clearance, planetary exploration and rescue missions are just three out of many examples. It is therefore considered as one of the fundamental problems in mobile robotics. Since the potential of a single robot is quite restricted, an increasing amount of research is focused on multi-robot exploration (A. U. Irturk, 2006; K. Easton and J. Burdick, 2005; W. Burgard et al., 2005; W. Kerr et al., 2005; S. Weihua et al., 2006; Simmons et al., 2000). An increased performance, distributed sensing and higher fault tolerance are evident motives. However, moving from one robot towards a multi-robot setup also strongly increases complexity.

Exploring unknown territory can be categorized in complete coverage (I. Rekleitis et al., 2004) and no complete coverage (as in (Rogge and Aeyels, 2007) for example). The latter use a zigzag formation with two leader robots. Whenever an obstacle is found, the formation beaks up in two groups, always between the middle robots. The major drawback of their strategy is the lack of support for concave obstacles. Furthermore recursive splitting is bounded by the number of available robots, which in turn limits the density of

the obstacles spread across the area. Coordinating the robots with a formation obviously implies establishing and maintaining that formation. In (J. Fredslund and M. J. Mataric, 2002) this is achieved with local sensing and minimal communication. Multiple robots almost always need to communicate in order to avoid collisions and minimize repeat coverage. It is often assumed that communication is limited (W. Burgard et al., 2005), sometimes with the restriction of line-of-sight (I. Rekleitis et al., 2004; J. Ota, 2006). In this article we shall not adopt this assumption, considering it as future work. If communication is indeed limited, a message can always be passed on from one robot towards another until its destination is reached, just as if the robots were forming a chain. A more complete and general overview of exploration and coverage strategies with multiple robots is found in (A. U. Irturk, 2006; H. Choset, 2001).

The main goal of this article is to design a strategy which guarantees correct exploration with multiple robots, resulting in a complete map of the area. We only require free space to be covered by the robot's sensors, not by the robot itself. A formation is used to coordinate the robots which explore a room one slice at a time. Furthermore we don't make any assumptions about the shape of the obstacles in the environment, unlike many related work. Both concave and

convex obstacles are thus allowed. At the same time we try to obtain maximum benefit from the number of robots that are available for exploration.

The remainder of this paper is organized as follows. Firstly, we describe the exploration strategy for a general type of robot using a theoretical approach in section 2. More details about the implementation within the simulator can be found in section 3. Finally, we discuss the results in section 4, followed by some conclusions in section 5.

2 ALGORITHM DESCRIPTION

In our approach, robots are ordered on a straight line. An important advantage of a straight line is its width, which is the largest of all possible formations. In this way a large part of the map can be explored at once. In addition it's a very simple formation, which allows for easy establishment via a chain of friendships (J. Fredslund and M. J. Mataric, 2002). As for the exploration itself we use a complete sensor coverage approach as in (Rogge and Aeyels, 2007).

Like many previous work, the robots explore the environment slice per slice. The deliberative approach differs a lot from traditional algorithms (such as behavior-based systems) as we don't simply split the robot team in two groups, each wall-following a different side of the obstacles. Instead, two separate phases are executed when obstacles are found. If one robots detects an obstacle, this is communicated to the team and all robots stop. Next, they start the first phase in which all new obstacles are added to the map (fully explored via wall-following). This is called the exploration phase and is illustrated in figure 1. When no new obstacles can be found this phase is terminated and the robots initiate the second phase. In this phase the robots move round the obstacles until each robot has passed it and the formation is regrouped. If during the second phase a new obstacle is found or an obstacle is preventing the formation to regroup, the robots go back to the exploration phase. This process continues until the formation is reestablished and no obstacles are blocking their path. The formation can then explore the remainder of the slice.

Once no new obstacles can be found near the robots, they start moving round each obstacle blocking their path. Since each of those obstacles is already added to the map, they can locate their correct goal position. Each goal position is chosen so that all areas are explored, in particular the cavities of concave obstacles. To do so an imaginary line is drawn, starting from the robot's position and going through the obstacle blocking its path. As soon as enough free space is found, a goal point is located.
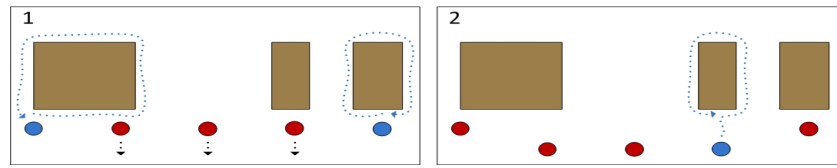
3 EXPERIMENTAL SETUP

We implemented our strategy on the EyeBot-platform within a simulator called EyeSim (A. Koestler and T. Bräunl, 2004). In addition we built a generic test suite providing easy testing, configuration and evaluation. The setup comprises several robots and one computer acting as a server. The robots are fully autonomous and can communicate both directly with each other and with the server (wireless).
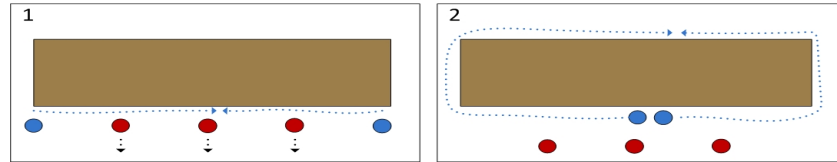
3.1 EyeBotServer

The server has several tasks. Firstly, it is responsible for the configuration of each experiment. It reads in a configuration-file containing parameters like the shape of the formation, driving speed, width of slices, etc. These parameters are sent to all robots which means configuration is centralized and the need to recompile the robots program is eliminated. It also allows for easy extending the suite with other formations and strategies resulting in a very generic platform. The second task of the server originates from the exploration of an environment. It is responsible for storing the global map of the area, constructed by the robots. We do this to adjust for the limited memory capacity of the robots and to provide a safer storage of the map. Since a computer has a large amount of memory, we use a grid to model the environment. A grid makes it easy to calculate goal points and the shortest route towards them. The last task of the server comprises exporting the results of the exploration. A visualization of the map via a PNG image is generated from the grid structure and stored on disk.

3.2 EyeXplore

EyeXplore is the main program run by each EyeBot and is an implementation of the deliberative strategy described in section 2. We emphasize that the robots running this program are fully autonomous in the sense that they don't get instructions from the server. For them the server acts as an extended memory in which the map can be stored. During exploration, the robots are responsible for determining the edges of obstacles which are then send to the server. The robots don't store these edges permanently in local memory in order to save space. There is thus no distributed map among the robots. In order to explore

(a) Exploring robots in front of different obstacles.

(b) In case of larger obstacles, the two exploring robots might explore the same obstacle. When they actually meet each other during wall-following, they know there is only one obstacle. Both robots then only explore half of the obstacle's border to gain speed.

Figure 1: Illustration of exploration phase. Two explorer robots are selected and perform a wall-following.
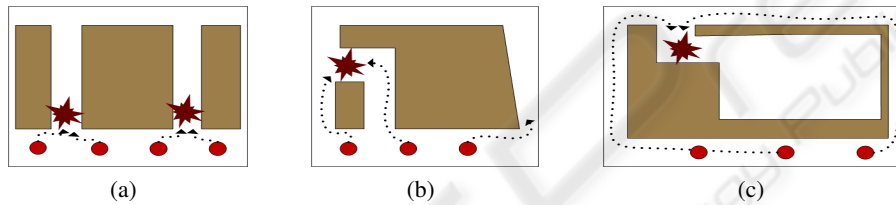

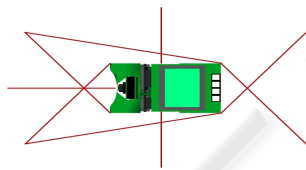Figure 2: Robot collisions during wall-following to reach goal-point.

Figure 3: PSD sensors used during the experiments.

the edges of obstacles and to avoid collisions, we use distance sensors as shown in figure 3.

4 RESULTS

In this section we discuss some results of the deliberative strategy. We examine the exploration time for five different environments (shown in figure 4). Due to page restrictions only a single map is shown.

In order to examine the influence of the number of robots on the exploration time, each environment was explored five times. For each test the number of robots was increased with one, starting from a single robot until a maximum of five robots. This gives us five durations in total for each environment (third column of figure 4). From the results it is clear that introducing a second robot always leads to a faster exploration with respect to a single robot. This is caused by a higher parallelization and a bisection of the number of slices required to explore the area. Bringing in even more robots doesn't always have the same positive effect. This observation has several causes. Firstly, remark that the extra decrease in the number of required slices declines as the number of robots increases. Another cause is that within our strategy, at most two robots are exploring new obstacles at the same time. The other robots are stalled during that time, which can lead to relatively long delays depending on the obstacles within the environment. In figure 4 the large S-shaped obstacle is explored by all robots when the formation contains only two robots. Adding a third robot causes this obstacle to be explored by just one robot, introducing delays for the remaining two robots. This explains the large increase of exploration time. Also note that the obstacles in figure 4 are placed such that the formation cannot be maintained as soon as there are more than two robots, since there is not enough space to form a line. This also means the robots will hamper each other more often which has a negative effect on exploration time.

We also examined the influence of the width of the formation and thus the width of the slices. For this purpose each environment was explored five times once more. Now each test is done by a formation comprising a constant number of robots, namely
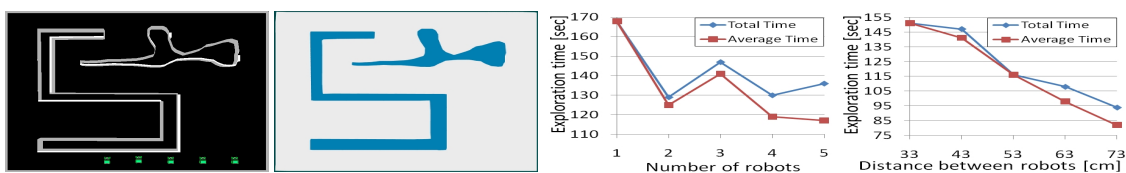
Figure 4: Results for a difficult map (4 other maps are not shown due to page restrictions). Four columns are shown for each environment. From left to right: the simulator, the resulting PNG image of the map, exploration time with increasing number of robots and finally with an increasing slice width.

three. Instead, we vary the horizontal distance between each two neighboring robots standing in a line formation. The smallest distance is 33 cm and this is increased with 10 cm in each step, corresponding to the width of a single EyeBot. The largest distance between two robots considered here is thus 73 cm.

The exploration times for each environment are shown in the fourth column in figure 4. Note that the graphs now show a more clear descending tendency and exploration times are lower than when we increased the number of robots. This is caused by the decreasing number of slices as we increase slice width. Fewer delays are introduced as well because for most of the test-environments two robots were exploring at the same time, implying more parallelization. The robots also don't hinder each other as often as with a formation comprising four or five robots. Therefore we can assume it is better to first increase slice width in order to lower exploration times. Of course the range of the sensors and communication hardware will limit the maximum width and eventually additional robots should be used as well.

5 CONCLUSIONS AND FUTURE WORK

We developed a deliberative strategy to explore an unknown environment with a formation of robots. Both concave and convex obstacles are supported, which is the primary innovation of our research. Moreover, the strategy is very scalable as it can be applied from a single robot to any number of robots (as long there is enough space in the environment). The results from our tests are very promising, yet further research is still needed to exploit the number of robots even more.

REFERENCES

A. Koestler and T. Bräunl (2004). Mobile Robot Simulation with Realistic Error Models. *International Conference on Autonomous Robots and Agents*, 6:46–51.

A. U. Irturk (2006). Distributed Multi-robot Coordination For Area Exploration and Mapping.

H. Choset (2001). Coverage for robotics - A survey of recent results. *Annals of mathematics and artificial intelligence*, 31(1-4):113–126.

I. Rekleitis, V. Lee-Shue, Ai Peng New, and H. Choset (2004). Limited communication, multi-robot team based coverage. *Proc. 2004 IEEE International Conference on Robotics and Automation*, 1-5:3462–3468.

J. Fredslund and M. J. Mataric (2002). A General Algorithm for Robot Formations Using Local Sensing and Minimal Communication. *IEEE transactions on robotics and automation*, 18(5):837–846.

J. Ota (2006). Multi-agent robot systems as distributed autonomous systems. *Advanced engineering informatics*, 20(1):59–70.

K. Easton and J. Burdick (2005). A coverage algorithm for multi-robot boundary inspection. *IEEE International Conference on Robotics and Automation (ICRA)*, 1-4:727–734.

Rogge, J. and Aeyels, D. (2007). A novel strategy for exploration with multiple robots. *Icinco 2007: proceedings of the fourth international conference on informatics in control, automation and robotics, vol ra-1- robotics and automation*, 1:76–83.

S. Weihua, Y. Qingyan, T. Jindong, and X. Ning (2006). Distributed multi-robot coordination in area exploration. *Robotics and autonomous systems*, 54(12):945–955.

Simmons, R., Apfelbaum, D., Burgard, W., Fox, D., Moors, M., Thrun, S., and Younes, H. (2000). Coordination for multi-robot exploration and mapping. *Proceedings Seventeenth National Conference on Artificial Intelligence (AAAI-2000). Twelfth Innovative Applications of Artificial Intelligence Conference (IAAI-2000)*, pages 852–8.

W. Burgard, M. Moors, C. Stachniss, and F. Schneider (2005). Coordinated Multi-robot exploration. *IEEE transactions on robotics*, 21(3):376–386.

W. Kerr, D. Spears, W. Spears, and D. Thayer (2005). Two formal gas models for multi-agent sweeping and obstacle avoidance. *Formal Approaches to Agent-Based Systems*, 3228:111–130.