An Experimental Comparison of Path Planning Techniques for Teams of Mobile Robots

Maren Bennewitz Wolfram Burgard

Department of Computer Science, University of Freiburg, 79110 Freiburg, Germany
{maren,burgard}@informatik.uni-freiburg.de

Abstract. This paper considers the problem of path planning for teams of mobile robots. It investigates two decoupled and prioritized approaches to coordinate the movements of the mobile robots in their environment. The first approach considered is the coordination technique. The second approach is an A^*-based path planning technique which computes the paths for the individual robots in the configuration time-space. Thereby it trades off the distance to both to static objects as well as to other robots and the length of the path to be traveled. In different experiments carried out with real robots and in simulations we demonstrate that the A^*-based approach is well suited to control the motions of a team of robots in various environments and illustrate its advantages over the coordination technique.

1 Introduction

Path planning is one of the fundamental problems in mobile robotics. As mentioned by Latombe [5], the capability of effectively planning its motions is "eminently necessary since, by definition, a robot accomplishes tasks by moving in the real world." Especially in the context of autonomous mobile robots, path planning techniques have to simultaneously solve two complementary tasks. On one hand, their task is to minimize the length of the trajectory from the starting position to the target location, and on the other hand they should maximize the distance to obstacles in order to minimize the risk of colliding with an object.

In this paper we consider the problem of motion planning for multiple mobile robots. This problem is significantly harder than the path planning problem for single robot systems, since the size of the joint state space of the robots grows exponentially in the number of robots. Therefore, the solutions known for single robot systems cannot directly be transferred to multi-robot systems.

The existing methods for solving the problem of motion planning for multiple robots can be divided into two categories [5]. In the *centralized* approach the configuration spaces of the individual robots are combined into one composite configuration space which is then searched for a path for the whole composite system (see for example potential field techniques [2, 11] or roadmap methods [10]). In contrast to that, the *decoupled* approach first computes separate paths for the individual robots and then tries to resolve possible conflicts of the generated paths.

While centralized approaches (at least theoretically) are able to find the optimal solution to any planning problem for which a solution exists, their time complexity is exponential in the dimension of the composite configuration space. In practice one is therefore forced to use heuristics for the exploration of the huge joint state space.

Decoupled planners determine the paths of the individual robots independently and then employ different strategies to resolve possible conflicts. According to that, decoupled techniques are incomplete, i.e. they may fail to find a solution even if there is one. A popular decoupled approach is planning in the configuration time-space [4] which can be constructed for each robot given the positions and orientations of all other robots at every point in time. Techniques of this type assign priorities to the individual robots and compute the paths of the robots based on these priorities. The method presented in [12], for example, applies potential field techniques in the configuration time-space to avoid collisions. Obviously, an important question is how to assign the priorities to the individual robots. In [1] all possible assignments are considered. However, due to the exponential complexity their approach can only be applied to groups of up to three robots.

Another approach to decoupled planning is the path coordination method which was first introduced in [9]. This method computes the paths of the individual robots independently and then applies scheduling techniques to deal with possible conflicts. The key idea of this technique is to keep the robots on their individual paths and let the robots stop, move forward, or even move backward on their trajectories in order to avoid collisions (see also [3]). Unfortunately, the problem of finding the optimal schedule for the robots is NP-hard, since the Job-Shop Scheduling Problem with the goal to minimize maximum completion time with unit processing time for each job [6] can be regarded as a special instance of the path coordination method. To deal with the enormous complexity in the case of huge teams of robots [7] recently presented a technique to separate the overall coordination problem into sub-problems. Thereby they assume that the overall problem can be divided accordingly.

In this paper we compare the path coordination method to another decoupled and prioritized approach to coordinated path-planning for multiple robots. This approach applies the well-known A^* procedure to compute cost-optimal paths in the configuration time-space and uses the optimal move costs of the individual robots to determine their priorities. The experiments carried out with real robots and in simulation runs and in different environments illustrate that the A^*-based procedure significantly outperforms the path coordination approach described in [7] with respect to the number of solutions found and the average overall path length.

2 A^*-based Path Planning for Multiple Robots

The goal of path planning is to determine a trajectory with the optimal trade-off between the overall length and the distance to obstacles in the environment. To effectively plan the path of a mobile robot, path planning systems need a model of the environment. In this paper, the map of the environment is given by an occupancy grid map [8]. The key idea of occupancy maps is to separate the environment into a grid of equally spaced cells. Each cell of such a grid contains the probability that this cell is occupied.

Given such a map one can use the well-known A^* procedure to determine the path from the current location to the target point. For each location $\langle x, y \rangle$ the A^* procedure simultaneously takes into account the cost of reaching $\langle x, y \rangle$ from the starting position as well as the estimated cost of reaching the target location $\langle x^*, y^* \rangle$ from $\langle x, y \rangle$. In our case the cost for traversing a cell $\langle x, y \rangle$ is proportional to its occupancy probability $P(occ_{x,y})$. The estimated cost for reaching the target location is approximated by the straight-line distance $||\langle x, y \rangle - \langle x^*, y^* \rangle||$ between $\langle x, y \rangle$ and $\langle x^*, y^* \rangle$.

To plan the trajectories of teams of robots, one can easily extend this approach to a decoupled method. First, one computes for each robot the cost-optimal path using the A^* procedure mentioned above. Then one checks for possible conflicts in the trajectories of the robots (in our current system a conflict occurs whenever two robots get closer than 1.2m). To resolve possible conflicts, a common approach is to use a priority scheme and to determine new paths for the robots with lower priority. Obviously, the overall complexity of a prioritized A^*-based path planning technique for teams of robots is $O(nm \cdot \log(m))$ where n is the number of robots and m is the maximum number of states expanded by A^* during planning in the configuration time-space (i.e. the maximum length of the OPEN-list).

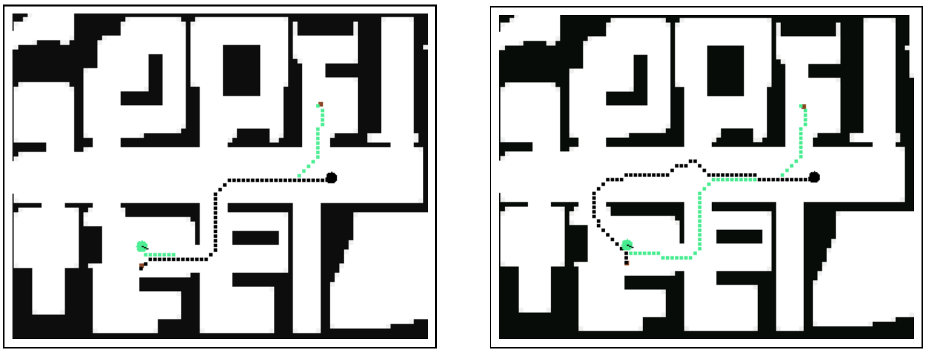

Fig. 1. A conflict situation for two robots (left) and the resolved conflict by choosing a detour for the second robot (right).

A typical application example of A^*-based path planning for a pair of robots is illustrated in Figure 1. In the situation depicted on the left side of Figure 1 the robot depicted in light grey is supposed to move to the fourth room in the north. The second robot depicted in black starts in the corridor and has its target location close to the starting point of the first robot. Since both paths are planned independently, they impose a conflict between the two robots. After applying the A^* procedure in the configuration time-space for the second robot, the conflict is resolved. The planner decides that the black robot has to avoid the conflict with the grey robot by moving to the north just at the door where the first robot enters the corridor. After this collision avoidance action, the path through the next doorway appears to have less costs, so that it takes a completely different trajectory. The resulting trajectories are depicted in the right image of Figure 1.

Please note, that the coordination technique performs significantly worse in this case. Since it forces the robots to stay on their individually optimal trajectories, one of the robots has to wait until the other robot has passed by. This is independent of the priorities with which the robots travel along their paths.

3 Experimental Results

To evaluate the different approaches, we implemented the A^*-based approach in order to demonstrate its performance on real robots. The current implementation is quite efficient, although there still is a potential for improvements. For the 19×15 m^2 large environment in which we carried out the experiments described here, our system is able to plan a collision-free path in the configuration time-space using a spatial resolution of 20×20 cm^2 for the grid map in less than 6 seconds and in less than 1.5 seconds for a cell size of 40×40 cm^2. Please note that this computation time will not increase in the number of robots, since one can use lookup-tables to store the costs introduced by the previously planned robots. Thus, the time needed for single robot path planning in the two-dimensional configuration space is generally less than 0.01 seconds. These performance measures were taken on a 500MHz Intel Pentium III running Linux.

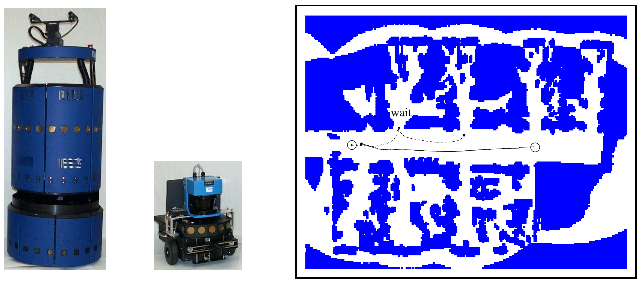

Fig. 2. The robots Albert (left) and Ludwig (center) used in the experiments as well as a situation in which Ludwig moves away in order to let Albert pass by (right).

3.1 Application Example with Real Robots

The first experiment described here has been carried out using our robots Albert and Ludwig which are depicted in Figure 2. Whereas Albert is an RWI B21 robot, Ludwig is a Pioneer I system. Both robots are equipped with a laser-range finder to reactively avoid obstacles. Figure 2 (right) shows one situation, in which both robots have a conflict. While Ludwig starts at the left end of the corridor of our lab and has to move to right end, Albert has to traverse the corridor in the opposite direction. Because of the space required by Albert, Ludwig decides to move into a doorway in order to let Albert pass by. Whereas the trajectory of Ludwig is depicted by a dashed line, Albert's trajectory is indicated by a solid line. The position where Ludwig waited for Albert is indicated by the label "wait". Please note, that the coordination technique is not able to find a solution in this situation since the robots do not have the chance to pass by if they are forced to stay on their trajectories.

3.2 Comparisons for Larger Numbers of Robots

The overall goal of our experiments is to demonstrate that A^*-based path planning in the configuration time-space significantly outperforms the coordination technique described in [7] with respect to the average time needed to reach the target location as well as with respect to the number of cases in which a solution can be found. To deal with larger groups of robots we use the optimal move costs of the individual robots to determine the priorities. Since the complexity of the coordination technique grows exponentially in the number of robots, we apply the same priority scheme for this approach.

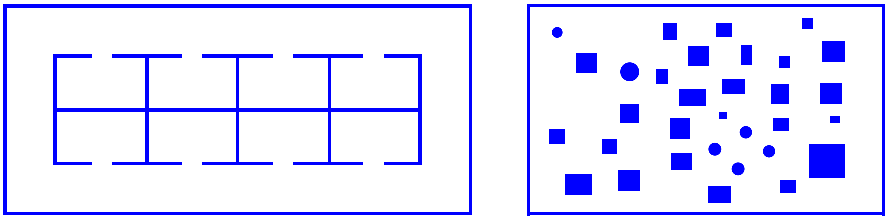

Fig. 3. Two different environments used for simulation runs.

To get a quantitative assessment of the performance of the two prioritized techniques we performed extensive simulation runs for different numbers of robots using two different environments. Figure 3 depicts the two environments used in the experiments. The first environment shown on the left side of Figure 3 is a typical corridor environment. The second situation is a rather unstructured environment (see right image of Figure 3) which offers many possibilities for the robots to change their routes. In 9000 experiments we evaluated the path planning techniques for 2 to 6 robots in both environments. The corresponding start and goal positions were randomly chosen from a set of predefined positions. In each experiment we measured the sum of the move costs generated by the A^*-based technique in the configuration time-space and computed by the prioritized variant of the coordination technique. Since the optimal solutions were not known (and cannot be computed in a reasonable amount of time for more than two robots) we compared the results of the planning techniques with the sum of the optimal move costs for the individual robots if the paths are computed independently, i.e. in independent single robot problems.

Figure 4 (left) shows for both environments the average number of conflicts each robot is involved in. Please note that we only evaluated situations in which there was at least one conflict between the robots. As can be seen this number is significantly higher in the corridor environment than in the unstructured environment because all robots have to travel along the corridor whereas they have a lot more possibilities to choose alternative routes in the unstructured world.

Additionally, we evaluated different priority schemes and analyzed their influence on the quality of the solution for the A^*-based technique. We sorted the robots according

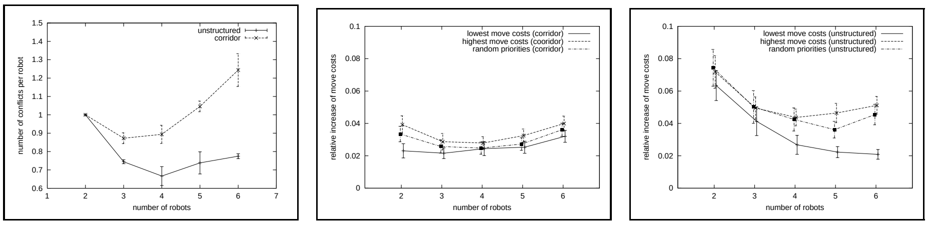

Fig. 4. Average number of conflicts between the robots (left) and performance of the different priority schemes in the corridor environment (center) and in the unstructured environment (right).

to the optimal move costs between their initial and their goal position in decreasing as well as in increasing order and compared this to a random order. Figure 4 (center and right) shows the performance of these schemes in the two environments considered here. Obviously, the approach to sort the robots according to their optimal move costs yields the best results.

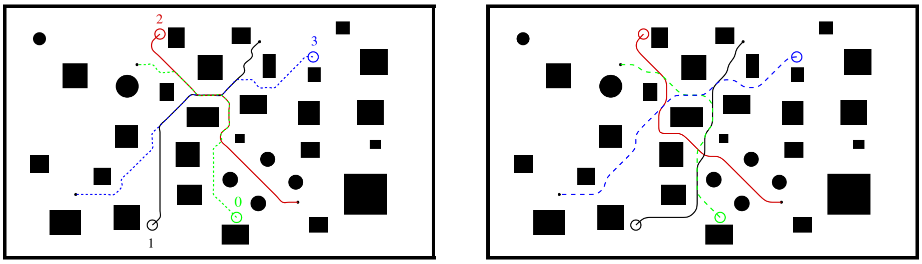

Fig. 5. Typical situation with four robots including their independently planned optimal trajectories and their priorities (left) as well as paths computed by the A^*-based method (right).

To compare the performance of the A^*-based technique in the configuration time-space and the coordination method we evaluated 50 situations in the corridor and 100 experiments in the unstructured environment. Thereby we only considered such situations in which there was a conflict between the robots and in which both techniques were able to compute a solution. A typical example with four robots is shown in Figure 5 (left). The trajectories computed with the A^*-based planning technique according to the priority scheme are shown in Figure 5 (right).

To compare the two techniques we compared the move costs of the robots (shown in Figure 5 (right)) with the corresponding costs obtained with the coordination technique both relative to the move costs of the optimal paths in Figure 5 (left). As can be seen in Figure 6 (left) A^*-based planning significantly outperforms the coordination technique in both environments. Especially in the corridor environment the coordination technique frequently forces the robots to wait in a room for longer periods of time until another robot passed by. Since A^*-based planning allows to robots to choose detours in the corridor, the reduction in the average move costs is much higher.

Another interesting aspect is the number of situations in which the different approaches were able to generate a solution. Figure 6 (right) shows for both methods the

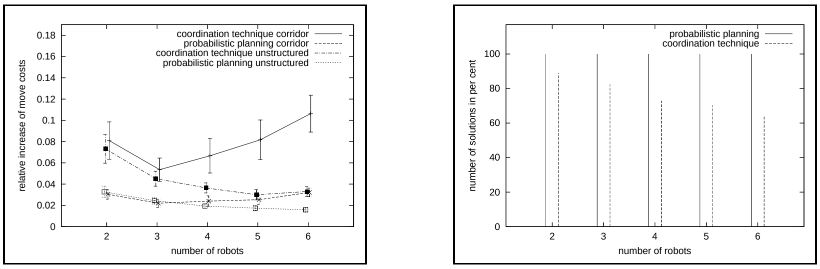

Fig. 6. Comparison of the relative increase of move costs of the A^*-based technique and the coordination technique (left) as well as the number of cases in percent where a solution could be found in the unstructured environment (right).

number of cases in percent in which a solution could be found in the unstructured environment. Obviously, the coordination technique quite often cannot find a solution as the number of robots grows. For example, for 6 robots only about 60% of the planning problems could be solved by the coordination technique whereas the A^*-based technique was able to find a solution in all situations.

4 Conclusions

In this paper we compared two approaches to decoupled and prioritized path planning for groups of mobile robots. The first approach is the coordination-technique [7]. This approach first determines the optimal paths of the individual robots and then computes a schedule how the robots have to traverse these trajectories. The second approach uses A^* to compute the paths of the robots in their configuration time-spaces. It uses a priority scheme to plan the paths of the robots and takes into account the paths of the robots with higher priority during the planning process.

The A^*-based technique has been implemented and tested on real robots. The independent planning of the paths for the individual robots is highly efficient and requires not more than 0.01 seconds. Additionally, the system can rather quickly resolve conflicts. For the examples in the map of our department the computation of a collision-free path in the configuration time-space generally requires less than 6 seconds using a spatial resolution of 20×20 cm^2 and less than 1.5 seconds for a cell size of 40×40 cm^2. Please note that this computation time will not significantly increase in the number of robots, since one can use lookup-tables to store the costs introduced by the previously planned robots.

We performed a series of experiments with up to 6 robots to compare the A^*-based approach to the coordination method. These experiments demonstrate that the A^* technique significantly outperforms the coordination technique with respect to the number of situations for which a solution can be found as well as with respect to the average length of the paths.

Apart from the promising results presented in this paper, there are different aspects for future research. First, in the experiments carried out here we assumed equal constant velocities of the robots. In practice, teams often are inhomogeneous and contain different types of robots with different average velocities which has to be taken into account. Furthermore, we only used a fixed priority scheme. More flexible assignments of priorities to the individual robots will with high likelihood result in more efficient solutions. The techniques considered here provide no means to react to possible deviations of the robots from their planned trajectories during the plan execution. For example, if one robot is delayed because unforeseen objects block its path, alternative plans for the robots might be more efficient. In such situations it would be important to have means for detecting such opportunities and to re-plan dynamically. On the other hand, the delay of a single robot may result in a dead-lock during the plan execution. In this context, path planning systems require techniques to detect dead-locks while the robots are moving and to resolve such dead-locks appropriately.

References

1. K. Azarm and Günther Schmidt. A decentralized approach for the conflict-free motion of multiple mobile robots. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1667–1674, 1996.
2. J. Barraquand and J. C. Latombe. A monte-carlo algorithm for path planning with many degrees of freedom. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 1990.
3. Z. Bien and J. Lee. A minimum-time trajectory planning method for two robots. *IEEE Transactions on Robotics and Automation*, 8(3):414–418, 1992.
4. M. Erdmann and T. Lozano-Perez. On multiple moving objects. *Algorithmica*, 2:477–521, 1987.
5. J.C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Boston, MA, 1991. ISBN 0-7923-9206-X.
6. E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys. Sequencing and scheduling: Algorithms and complexity. Technical report, Centre for Mathematics and Computer Science, 1989.
7. S. Leroy, J. P. Laumond, and T. Simeon. Multiple path coordination for mobile robots: A geometric algorithm. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 1999.
8. H.P. Moravec and A.E. Elfes. High resolution maps from wide angle sonar. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 116–121, 1985.
9. P. A. O'Donnell and T. Lozano-Perez. Deadlock-free and collision-free coordination of two robot manipulators. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 1989.
10. P. Sveska and M. Overmars. Coordinated motion planning for multiple car-like robots using probabilistic roadmaps. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 1995.
11. P. Tournassoud. A strategy for obstacle avoidence and its application to multi-robot systems. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, pages 1224–1229, 1986.
12. C. Warren. Multiple robot path coordination using artificial potential fields. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, pages 500–505, 1990.