Optimizing Schedules for Prioritized Path Planning of Multi-Robot Systems

Maren Bennewitz† Wolfram Burgard† Sebastian Thrun‡

†Department of Computer Science, University of Freiburg, 79110 Freiburg, Germany
‡School of Computer Science, Carnegie Mellon University, Pittsburgh PA, USA

Abstract

The coordination of the motions of the robots is one of the fundamental problems for multi-robot systems. A popular approach to avoid planning in the high-dimensional composite configuration space are prioritized and decoupled techniques. While these methods are very efficient, they have two major drawbacks. First, they are incomplete, i.e. they sometimes fail to find a solution even if one exists, and second, the resulting solutions are often not optimal. They furthermore leave open how to assign the priorities to the individual robots. In this paper we present a method for optimizing priority schemes for such prioritized and decoupled planning techniques. Our approach performs a randomized search with hill-climbing to find solutions and to minimize the overall path lengths. The technique has been implemented and tested on real robots and in extensive simulation runs. The experimental results demonstrate that our method is able to seriously reduce the number of failures and to significantly reduce the overall path length for different prioritized and decoupled path planning techniques and even for large teams of robots.

1 Introduction

Path planning is one of the fundamental problems in mobile robotics. As mentioned by Latombe [9], the capability of effectively planning its motions is "eminently necessary since, by definition, a robot accomplishes tasks by moving in the real world."

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 [9]. 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 In contrast, the *decoupled* approach first computes separate paths for the individual robots and then resolves 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.

Many methods use potential field techniques [2, 3, 17] to guide the search. These techniques apply different approaches to deal with the problem of local minima in the potential function. Other methods restrict the motions of the robots to reduce the size of the search space. For example, [16, 8, 10] restrict the trajectories of the robots to lie on independent roadmaps. The coordination is achieved by searching the Cartesian product of the separate roadmaps.

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 [6] 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 the order implied by these priorities. The method presented in [18] uses a fixed order and applies potential field techniques in the configuration time-space to avoid collisions. The approach described in [7] also uses a single priority scheme and chooses random detours for the robots with lower priority.

Another approach to decoupled planning is the path coordination method which was first introduced in [14]. 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 [4]). To reduce the complexity in the case of huge teams of robots [12] recently presented a technique to separate the overall coordination problem into sub-

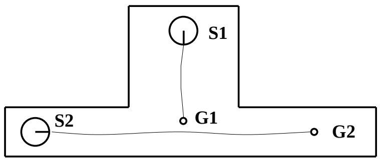

Figure 1: Situation in which no solution can be found if robot 1 has higher priority than robot 2

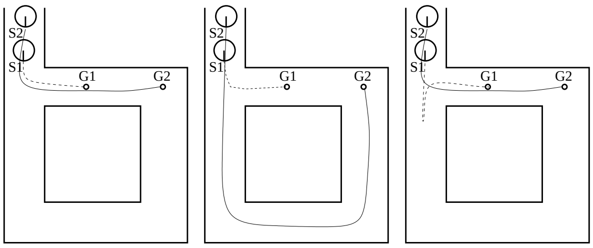

Figure 2: Independently planned optimal paths for two robots (left), suboptimal solution if robot 1 has higher priority (center), and solution resulting if the path for robot 2 is planned first (right).

problems. This approach, however, assumes that the overall problem can be divided into very small sub-problems, a serious assumption which, as our experiments described below demonstrate, is often not justified. In general, therefore, a prioritized variant has to be applied.

The methods described above leave open how to assign the priorities to the individual robots. In the past, different techniques for selecting priorities have been used. [5] applied a heuristic which assigns higher priority to robots which can move on a straight line from the starting point to their target location. In [1] all possible assignments are considered. Due to its complexity this approach has only been applied to groups of up to three robots.

For decoupled and prioritized methods the order in which the paths are planned has a serious influence on whether at all a solution can be found and if so, how long the resulting paths are. Figure 1 shows a situation in which no solution can be found if robot 1 has a higher priority than robot 2. Since the path of robot 1 is planned without considering robot 2, it arrives at its target location marked G1 before robot 2 has passed the t-junction. Thus, it blocks the way of robot 2 which can no longer reach its designated target point G2. However, if we change the priorities and plan the trajectory of robot 2 before that of robot 1, then robot 1 considers the trajectory of robot 2 during path planning and this way waits until robot 2 has passed by.

Another example is shown in Figure 2 (left). If we start with robot 1 then every planner has to choose a large detour for robot 2 (see Figure 2 (center)), because robot 1 blocks the corridor. However, if the path of robot 2 is planned first, then we can obtain a much more efficient solution (see Figure 2 (right)).

These two examples illustrate that the priority scheme has a serious influence on whether a solution can be found and on how long the resulting paths are. Unfortunately the problem of finding the optimal schedule is NP-hard for most of the decoupled approaches. For example, the Job-Shop Scheduling Problem with the goal to minimize maximum completion time with unit processing time for each job [11] can be regarded as a special instance of the path coordination method.

In this paper we present a randomized and hill-climbing technique which starts with an initial priority scheme and optimizes this by swapping two randomly chosen robots. This way it can significantly increase the number of problems for which a solution can be found. Additionally it is able to reduce the overall path length. Furthermore, our approach has any-time characteristic which means that it can return the best solution found so far at any point in time and whenever it is interrupted. Our technique has been implemented and tested on real robots. In extensive experiments it has been proven to be very effective even for large teams of robots and using two different decoupled path planning techniques.

The paper is organized as follows. The following section describes the prioritized and decoupled path planning techniques we apply our algorithm presented in Section 3 to. Section 4 contains experimental results illustrating the capabilities of our approach.

2 Prioritized A^*-based Path Planning and Path Coordination

The basic algorithm to compute optimal paths for single robots applied throughout this paper is the well-known A^* search procedure. The next section briefly describes the variant we are using. To represent the environment of the robots we apply occupancy grids [13] which separate the environment into a grid of equally spaced cells and store in each cell $\langle x, y\rangle$ the probability $P(occ_{x,y})$ that it is occupied. In the remainder of this section we then present the key ideas of decoupled prioritized path planning and discuss how the A^* procedure can be utilized to plan the motions of teams of robots by this approach.

2.1 A^*-based Path Planning

The A^* procedure simultaneously takes into account the accumulated cost of reaching a certain location $\langle x, y\rangle$ 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 $P(occ_{x,y})$. Furthermore, the estimated cost for reaching the target location is approximated by $c \cdot ||\langle x, y\rangle - \langle x^*, y^*\rangle||$ where c is chosen as the minimum occupancy probability $P(occ_{x,y})$ in the map and $||\langle x, y\rangle - \langle x^*, y^*\rangle||$ is the straight-line distance between $\langle x, y\rangle$ and $\langle x^*, y^*\rangle$. Since this heuristic is admissible, A^* determines the cost-optimal path from the starting position to the target location.

2.2 Decoupled Path Planning for Teams of Robots

In this paper we consider decoupled and prioritized path planning approaches which plan the paths in the configuration time-space. Such approaches proceed as follows. First, one computes for each robot the path without considering the paths of the other robots. Then one checks for possible conflicts in the trajectories of the robots (we regard it as a conflict between two robots if their distance is less than δ where $\delta = 1.2m$ in our current system). Conflicts between robots are resolved by introducing a priority scheme. A priority scheme determines the order in which the paths for the robots are planned. The path of a robot is planned in its configuration time-space computed based on the map of the environment and the paths of the robots with higher priority.

Our system applies the A^* procedure to compute the cost-optimal paths for the individual robots, in the remainder denoted as the independently planned optimal paths for the individual robots. We also apply A^* search to plan the motions of the robots in the configuration time-space. In this case the cost of traversing a location $\langle x, y\rangle$ at time t is determined by the occupancy probability $P(occ_{x,y})$ plus the probability that one of the other robots with higher priority covers $\langle x, y\rangle$ at that time.

In this paper we consider two different strategies: A^*-based planning in the configuration time-space as well as a restricted version of this approach denoted as the path coordination technique [12]. It differs from the general A^*-search in that it only explores a subset of the configuration time-space given by those states which lie on the initially optimal paths for the individual robots. The path coordination technique thus forces the robots to stay on their initial trajectories. The overall complexity of both approaches is $O(n \cdot m \cdot \log(m))$ 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).

Due to the restriction during the search, the path coordination method is more efficient than the general A^* search. Its major disadvantage, however, lies in the fact that it fails more often. A typical example is shown in Figure 2. Whereas the path coordination method fails independently of the planning order, the general A^* procedure is able to find a solution in both cases.

3 Optimizing Priority Schemes

As already mentioned above, prioritized and decoupled approaches to multi-robot path planning are incomplete and sub-optimal. However, as the examples given in Figures 1 and 2 illustrate, the order in which the paths are planned has a significant influence on whether a solution can be found and on how long the resulting paths are. This raises the question of how to find a priority scheme for which the decoupled approach does not fail and how to find the order of the robots leading to the shortest paths.

Recently, randomized search techniques have been used with great success to solve constraint satisfaction problems or to solve satisfiability problems [15]. Our algorithm presented here is a variant which performs a randomized and hill-climbing search in order to optimize the planning order for decoupled and prioritized path planning of teams of mobile robots. It starts with an arbitrary initial priority scheme Π and randomly exchanges the priorities of two robots in this scheme. If the new order Π' results in a solution with shorter paths than the best one found so far, we continue with this new order. Since hill-climbing approaches like this frequently get stuck in local minima, we perform random restarts with different initial orders of the robots. The complete algorithm is listed in Table 1.

Table 1: The algorithm to optimize priority schemes.

```
FOR tries := 1 TO maxTries BEGIN
   select random order Π
   if (tries = 1)
      Π* := Π
   FOR flips := 1 TO maxFlips BEGIN
      choose random i, j with i<j
      Π' := swap(i, j, Π)
      if moveCosts(Π') < moveCosts(Π)
         Π := Π'
   END FOR
   if moveCosts(Π) < moveCosts(Π*)
      Π* := Π
END FOR
return Π*
```

Please note that an additional advantage of our randomized optimization approach lies in its any-time character. The procedure can be terminated at any point in time and return the currently best priority order whenever it is interrupted.

Figure 4 shows a typical application example carried out with our robots Albert and Ludwig shown in Figure 3 in our office environment. In this example we used the general $A*$ procedure in the configuration time-space. 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. If the path of Ludwig is planned before that of Albert, the system fails because Albert cannot reach

Figure 3: The mobile robots Albert (left) and Ludwig (right).

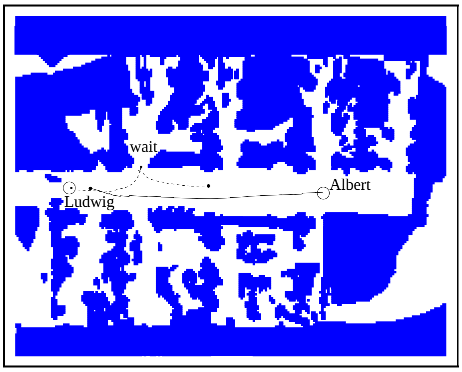

Figure 4: Real world application of A^*-based planning in the configuration time-space.

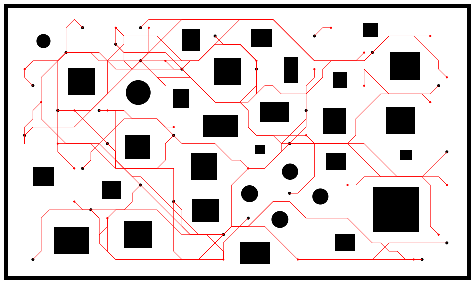

Figure 6: Paths resulting after priority optimization.

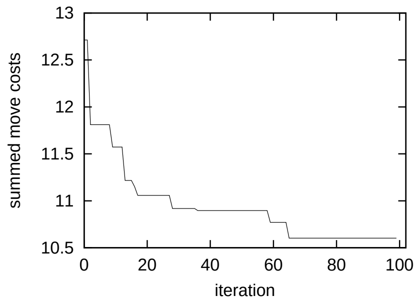

Figure 7: Summed move costs plotted over time.

its target point if Ludwig stays on its optimal trajectory. If we alter the planning order, our system is able to find a solution. In this case, Ludwig is moved into a doorway in order to let Albert pass by. Please note, that no solution can be found in this situation if the path coordination technique would be used. The resulting trajectories are shown in Figure 4 including the position where Ludwig waited to let Albert pass by.

Figure 5 shows a simulated situation with 30 robots. By applying our algorithm using the general A^* procedure we obtain the paths depicted in Figure 6. In this and all experiments described below we used a value of 10 for `maxFlips` and `maxTries`. Figure 7 plots the evolution of the summed move costs of the best solution found so far during these 100 trials. Obviously, compared to the initial solution shown in Figure 5 with summed move costs of 12.7, the final solution illustrated in Figure 6 has move costs of 10.9 which corresponds to a reduction of 15%. Please note, that there is a huge number of conflicts between the robots in this example. As a result, the whole trajectory graph is a single connected component. Accordingly, the decomposition technique presented in [12] cannot be applied.

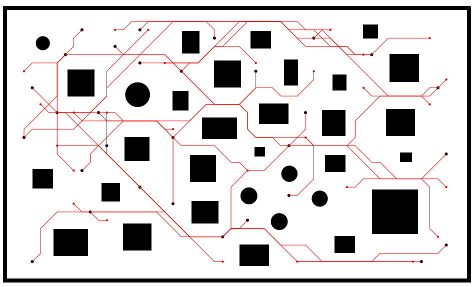

Figure 5: Independently planned paths for 30 robots.

4 Experimental Results

The algorithm described above has been tested thoroughly in extensive simulation runs. To evaluate the general applicability, we applied our method to the two decoupled and

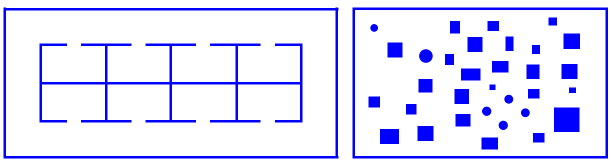

Figure 8: Environments used for the simulation runs.

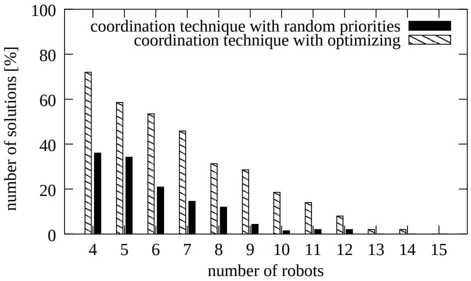

Figure 9: Reducing the number of failures for the path coordination technique by optimizing priority schemes.

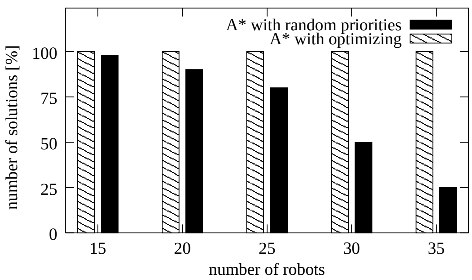

Figure 10: Reducing the number of failures for A^*-based planning in the configuration time-space by optimizing priority schemes.

prioritized path planning techniques described above. The current implementation is highly efficient. It requires less than 0.2 seconds to plan a collision-free path for one robot in all environments described below. Throughout the experiments we used the two different environments depicted in Figure 8. Whereas the map on the left of Figure 8 is a typical corridor environment, the map on the right is corresponds to an unstructured environment.

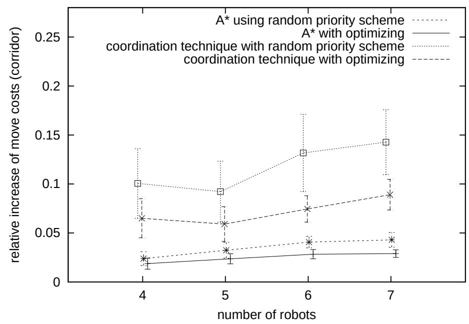

Figure 11: Relative increase of the move costs compared to the sum of the optimal move costs for the independently planned paths in the corridor environment.

4.1 Reducing the Number of Failures

The first set of experiments is designed to illustrate that the overall number of failures can be reduced significantly using our optimization technique. Figure 9 summarizes the results we obtained using the path coordination technique for different numbers of robots in over 600 runs in the unstructured environment. In each experiment we randomly chose the starting and target locations of the robots and determined whether the coordination technique is able to find a solution given a randomly chosen initial priority scheme. Then we optimized this scheme using our algorithm described above. For example, if 4 robots are used then the path coordination technique fails in more than 60% of the cases. Our approach, in contrast, yields a solution in more than 70% of all situations. For more than 15 robots, however, it is nearly impossible to find configurations for which the path coordination method can find a solution. One reason is that starting or goal locations often lie too close to the trajectories of other robots so that they cannot pass by any more.

Additionally, we applied our approach to 100 randomly chosen situations to which we used the A^*-based planning in the configuration time-space. The result of this experiment is shown in Figure 10. As this figure shows, A^* has a significantly higher success rate even for larger numbers of robots. However, even using this approach and a single priority scheme, the number of solutions decreases monotonously. Our optimization technique, in contrast, was always able to find a solution.

4.2 Minimizing the Overall Path Lengths

The second set of experiments is designed to demonstrate that our optimization technique is able to significantly reduce the overall path length. For different numbers of robots we performed a series of experiments in both environments shown in Figure 8. Again we randomly chose starting and target locations and then computed the paths for the robots using the two decoupled and prioritized path planning techniques with and without our optimization technique. We then measured the average path length and compared it to the average length of the optimal paths for the individual robots[1].

Figure 11 shows the relative increase of the move costs for four to seven robots in the corridor environment (in contrast to the other experiments, the starting and target locations were chosen from a given set of hand-selected positions in the map). As can be seen, our approach reduces the overall path length for the path coordination technique and for A^*-based planning in the configuration time-space. Additionally, it illustrates that the latter approach results in more

[1]Please note that throughout this paper we take the optimal paths for single robot planning problems as reference, since computing the optimal solution for the whole problem is not tractable in practice at least for larger numbers of robots.

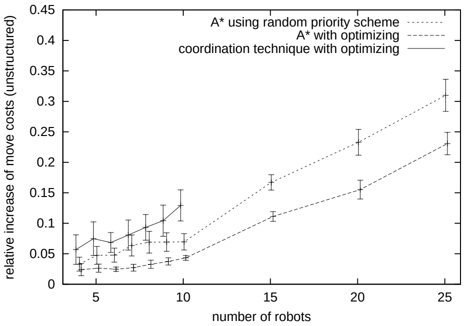

Figure 12: Increased move costs compared to the sum of the optimal move costs for the independently planned paths in the unstructured environment.

efficient paths than the coordination technique.

We performed similar experiments for the unstructured environment. Figure 12 summarizes the results we obtained for over 300 runs. It shows the relative increase of the move costs for different numbers of robots. Since the path coordination technique using the initially chosen priority scheme failed in most of the cases, we omit the corresponding results here. As the figure shows, our optimization technique applied to A^*-based planning in the configuration time-space yields the best results.

5 Conclusions

In this paper we presented an approach to optimize the priorities for decoupled and prioritized path planning methods for groups of mobile robots. Our approach is a randomized method which repeatedly reorders the robots to find a sequence for which a plan can be computed and to minimize the overall path lengths. It is an any-time algorithm since it can be stopped at any point in time and can always return its currently best estimate. The approach has been implemented and tested on real robots and in extensive simulation runs for two different decoupled path planning techniques and for large numbers of robots. The experiments demonstrate that our technique significantly decreases the number of failures in which no solution is found for a given planning problem. Additionally, its application leads to a significant reduction of the overall path length.

References

[1] K. Azarm and G. 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, B. Langois, and J. C. Latombe. Numerical potential field techniques for robot path planning. *IEEE Transactions on Robotics and Automation, Man and Cybernetics*, 22(2):224–241, 1992.

[3] 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.

[4] 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.

[5] S. J. Buckley. Fast motion planning for multiple moving robots. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 1989.

[6] M. Erdmann and T. Lozano-Perez. On multiple moving objects. *Algorithmica*, 2:477–521, 1987.

[7] C. Ferrari, E. Pagello, J. Ota, and T. Arai. Multirobot motion coordination in space and time. *Robotics and Autonomous Systems*, 25:219–229, 1998.

[8] L. Kavraki, P. Svestka, J. C. Latombe, and M. Overmars. Probabilistic road maps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, pages 566–580, 1996.

[9] J.C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Boston, MA, 1991. ISBN 0-7923-9206-X.

[10] S. M. LaValle and S. A. Hutchinson. Optimal motion planning for multiple robots having independent goals. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 1996.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] B. Selman, H. Levesque, and D. Mitchell. A new method for solving hard instances of satisfiability. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 1992.

[16] 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.

[17] 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.

[18] 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.