Multi-Robot Motion Planning by Incremental Coordination

Mitul Saha
Artificial Intelligence Lab
Stanford University
Stanford, CA 94305
Email: mitul@ai.stanford.edu

Pekka Isto
Research Institute for Technology
University of Vaasa
Vaasa, Finland
Email: isto@uwasa.fi

***Abstract*— A recent study showed that high failure rates in many instances of decoupled multi-robot motion planning makes the decoupled planning approach an unsuitable choice in comparison to the centralized approach. Motivated by this study we devised a new multi-robot motion planning strategy in which we partially merge the two phases of the basic decoupled planning approach. Our experimental results show that this new strategy significantly improves upon the reliability of the decoupled planning approach. Overall in our tests, the new strategy outperforms both the centralized and the decoupled planning approach, indicating that in practice our new strategy can be a better choice than both these approaches for solving multi-robot motion planning problems.**

I. Introduction

Considerable amount of research has been dedicated to the task of devising efficient motion planning algorithms for different scenarios. The predominant approach is to search the configuration space of the robotic system with a suitable algorithm. The family of sampling-based algorithms has proved to be effective and efficient in solving a variety of practical problems [1]. However, the basic problem is intractable [2], making any algorithm susceptible to failures.

If an important and distinct class of motion planning problems can be identified or defined along with the invariant characterizing properties, it is often possible to develop tailored algorithms that exploit those properties to gain an advantage. For instance, practical multi-robot motion planning problems are often decoupled in the sense that the robots have pairwise interactions only in some limited portions of the configuration space. This fact is exploited in the two-phase decoupled approach of multi-robot motion planning [3]. In the first phase of the basic decoupled approach, a path for each robot is computed individually ignoring other robots. Then in the second phase robot motions are coordinated such that inter-robot collisions are avoided. The second phase, the key phase in the decoupled planning approach, can also be seen as coordinating the individual robot motions so that only one robot at a time may enter the area of potential inter-robot interference. Though the two-phase decoupled planning approach is an incomplete approach, it is usually faster (when it succeeds) than planning in the joint configuration space of all the robots (centralized approach, [3]). This is because a search space explored by the decoupled planner has a lower dimensionality than the joint configuration space explored by the centralized planner.

A recent study [4] has shown that even though the decoupled approach is faster (when it succeeds) than the centralized approach, it is still an inferior choice for many multi-robot motion planning problems, as it is quite unreliable due to its high failure rate in many practical examples. Following up on this work, our goal has been to improve the reliability of the decoupled approach. This has led to a new multi-robot motion planning strategy, the main contribution of this paper, which works by planning and coordinating robot motions in an incremental fashion. The key idea in our new strategy is to partially merge the two phases of the decoupled planning approach. Like the decoupled approach, our incremental coordination strategy is also an incomplete strategy. But in practice our strategy is much more reliable. We have tested our new planning strategy, against both the decoupled and the centralized approach, in various multi-robot motion planning problems involving tight robot coordinations. Results from the tests demonstrate that our new approach is much more reliable than the decoupled approach. It is also usually faster than both the centralized and the decoupled approach. Overall, our results indicate that our new strategy can be a better choice than both these approaches for solving multi-robot motion planning problems.

The rest of the paper is organized as follows. Section II describes the previous work in multi-robot motion planning. Section III describes our new multi-robot motion planning strategy. Section IV empirically compares an implementation of our new strategy with a decoupled and a centralized planner. Section V concludes the paper.

II. Previous Work in Multi-Robot Motion-Planning

The authors of [5] point out that several taxonomies for multi-robot planning approaches exists. Various planners are described as centralized, decentralized, distributed, or decoupled. In the centralized planning approach, all the robots are grouped together as a single composite robot. Thereafter, the problem reduces to a single-robot motion planning problem [3]. The issue with this approach is that usually the resulting composite robot has many degrees of freedom

(DOF), which is quite undesirable given the fact that the time complexity of general path planning methods grows exponentially with number of DOF [6]. Centralized planning algorithms with lower time complexity are available for special cases, e.g., for pairs or triples of robots operating in low-density workspaces [7]. General centralized multi-robot planners can also take advantage of the problem properties (for instance, flocking behaviour in [5]), to improve their observed performances in those problems. The exponential space complexity of motion planning [2] further motivates development of tailored algorithms. It is often the space complexity that becomes an obstacle to practical deployment of modern motion planning algorithms.

The complexity of the centralized approach led to the emergence of the decoupled approach to multi-robot motion planning ([8], [9]), where completeness is sacrificed in the favor of complexity. In the basic decoupled approach, planning is done in two essentially decoupled phases. In the first phase, for each robot, a path is computed which is collision-free with respect to the obstacles in the environment excluding the other robots. Collisions between the robots are resolved in the second phase by *velocity tuning*, i.e., velocities for robots along their paths, computed in the first phase, are selected in such a way that the robots avoid collisions with each other along their respective paths. In essence, velocity tuning involves assigning time along the paths planned in the first stage so that inter-robot collisions are avoided. A robot path planned in the first phase can be parametrized in an abstract sense by some abstract parameter instead of time. The parameters corresponding to all the robot paths span a space called path coordination space, and the assignment of parameter values along the robots paths defines a coordination path in the coordination space.

The two phase decoupled planning strategy is inherently an incomplete strategy, because a path may not exist in the coordination space for the paths computed in the first phase, even though there may exist a path in the joint configuration space of all the robots. One may mitigate the situation by tuning the velocities over sets of alternative paths (roadmaps) for the robots [10] rather than single paths.

A variant of the basic decoupled planning approach is the *prioritized* planning approach [11]. In this approach, motions for the robots are computed sequentially in the order determined by some prioritization. During each iteration, motion is determined for a robot such that it is collision free with respect to obstacles in the environment, including the moving robots whose motions have been computed in previous iterations and excluding any other robot. The assignment of the priorities for the robots is critical for the success of this method. One may use fixed priorities [12], heuristic priorities [13], or search the space of prioritization schemes [14]. One may also constrain the planning of the robot's motion among the obstacles to a roadmap [13] or paths with given properties [15].

A *decentralized* planner distributes the planning to the individual robots which plan their motions based on the local knowledge and limited communication with other robots (e.g., [16]). The robots may plan their cooperative motions with centralized algorithms considering only partial knowledge but taking advantage of the problem characteristics including their decoupled multi-robot nature (e.g., [17]).

A recent study [4] compared the decoupled and the centralized approach by implementing their basic versions in the Probabilistic Roadmap (PRM) planning framework. The PRM framework allows one to plan motions for robots with many DOF, operating in environments with complex structure (for instance, geometries modeled by several thousand triangles) ([18], [19], [20], [21], [22]). The study concluded that decoupled planners are very unreliable in practice due to their inherent incomplete nature. Hence they suggest that a centralized planner should usually be the prefered choice (especially in cases requiring tight robot coordination), even though they are slower than a decoupled planner when the latter succeeds. Other experiments demonstrate that exploiting the decoupled nature of multi-robot problems can yield an order of magnitude improvement in performance [23]. These results have been the prime source of our motivation for improving the reliability of the decoupled planning approach.

III. THE METHOD OF INCREMENTAL COORDINATION

A. Problem Statement

In this section, we state the multi-robot motion planning problem that we intend to solve. Let a workspace W be shared by a set of n possibly articulated robots R_1, R_2, $\ldots R_n$ and populated by a set $\mathbf{B}$ of stationary obstacles. The geometries of the robots and obstacles are known, and the robots are holonomic. Let $\mathbf{q}_i$ be a configuration of robot R_i in some parametrization describing the position of every point of the robot. The configuration space C_i is the set of all possible configurations $\mathbf{q}_i$ for robot R_i. The number of DOF of robot R_i is the dimensionality d_i of C_i. The system configuration space C is the joint configuration space of all the n robots: $C = C_1 \times C_2 \times \ldots \times C_n$. The portion of the configuration space C in which each robot is collision-free with respect to itself, stationary obstacles, and other robots is called the free space F. The multi-robot planning problem that we intend to solve is the following: Given a start configuration $\mathbf{s} \in F$ and a goal configuration $\mathbf{g} \in F$, compute a path $\tau \subset F$ from $\mathbf{s}$ to $\mathbf{g}$, if there exists one.

B. Incremental Coordination

Due to the exponential growth of the size of joint configuration space C with the number of robots, few motion planning algorithms can solve problems with more than a few robots by directly searching C if the robots have moderate numbers of DOF ($d_i \geq 6$). The decoupled planning approach is to first decompose the problem into planning tasks for each robot individually and then, in the subsequent coordination phase, attempt to coordinate motions along the individual paths while avoiding collisions among the robots. More precisely the two-phase basic decoupled approach [8] works as follows. Let $\mathbf{s}_i$ be the start configuration and $\mathbf{g}_i$ the goal configuration for the robot R_i. Let the curve τ_i be a collision-free path

Algorithm MRP-IC ($\{R_1, R_2, \ldots, R_n\}$, $\mathbf{B}$, $\mathbf{s}$, $\mathbf{g}$)
1. τ_1 = SEARCH(C_1, $\mathbf{s}_1$, $\mathbf{g}_1$)
2. If $\tau_1 = nil$, then exit with failure
3. i=2
4. While ($i \leq n$)
   a. (π_i, τ_i) = SEARCH($P^{i-1} \times C_i, \tau_1, \ldots, \tau_{i-1}, (\mathbf{0}, \mathbf{s}_i), (\mathbf{1}, \mathbf{g}_i)$)
   b. if $\tau_i = nil$, then exit with failure
   c. $i = i + 1$
5. τ = JOIN($\tau_1, \tau_2, \ldots, \tau_n, \pi_n$)
6. return τ

Fig. 1. Our algorithm for multi-robot motion planning based on incremental coordination

from $\mathbf{s}_i$ to $\mathbf{g}_i$, planned without considering collisions with the other robots. Let $P = [0, 1]$ and $P^k = [0, 1]^k$ be the coordination space for the first k robots. Paths for individual robots can be parametrized as $\tau_i : p_i \in P \mapsto \tau_i(p_i)$. Now in the first phase, the basic decoupled approach would search the individual robot paths τ_i's so that the robots do not collide with the obstacles in $\mathbf{B}$. In the second phase, it would search the coordination space P^n for a coordination path $\pi : p' \in P \mapsto \pi(p')$ connecting configurations $\mathbf{0}$ and $\mathbf{1}$ in P^n, such that the multi-robot path $\tau : p \in P \mapsto (\tau_1(\pi(p)_1), \tau_2(\pi(p)_2), \ldots, \tau_n(\pi(p)_n))$, where $\pi(p)_j$ is the j-th element of the vector $\pi(p)$, is free of any inter-robot collision.

The core idea in our new multi-robot motion planning approach is to partially merge the two phases of the basic decoupled planning approach described in the previous paragraph. Our approach searches simultaneously for motions for a robot and coordination of motions of robots along paths already planned while ignoring the robots whose motions have not been planned. This involves iteratively searching a composite search space of a robot's configuration space and coordination space of the robots whose paths have already been planned. The motivation is that the robots which already have paths can re-tune their velocities along those paths to make it easier for the robot to maneuver among the obstacles and the other robots.

Figure 1 gives the pseudo-code for our multi-robot motion planner based on incremental coordination, which we call MRP-IC. MRP-IC searches C_1 and a sequence of composite spaces $P^1 \times C_2, P^2 \times C_3, \ldots, P^{n-1} \times C_n$. The dimensionality of the composite spaces is at most $n + d_i - 1$. The algorithm requires a suitable search algorithm for first searching the pure configuration space C_1 for the first robot (Step 1) and, subsequently, the composite spaces for the remaining robots (Step 4.a). In addition to standard search algorithms, many motion planning algorithms (for instance sampling-based algorithms described in [1], Chapter 7) are available for searching C_1. These algorithms can also be adopted for searching the composite spaces by considering the coordination space as auxiliary DOF. $\mathbf{0}$ and $\mathbf{1}$ in Step 4.a are i-dimensional. Finally, if all the search iterations succeed, MRP-IC will use the procedure JOIN to compute a single coordinated motion path from the individual paths τ_i's and the final coordination path π_n (Step 5). JOIN yields a multi-robot path $\tau : p \in P \mapsto (\tau_1(\pi_n(p)_1), \tau_2(\pi_n(p)_2), \ldots, \tau_n(\pi_n(p)_n))$, where $\pi_n(p)_j$ is the j-th element of the vector $\pi_n(p)$.

As a matter of fact, MRP-IC is more similar to the prioritized planning approach of [11] than the basic decoupled approach. Like the prioritized approach, MRP-IC considers the $i - 1$ robots, whose paths have already been planned, as moving obstacles while planning a path for the i-th robot. But unlike the prioritized approach, MRP-IC allows those moving obstacles to re-tune their respective velocities. As a result, while searching for a path for the i-th robot, MRP-IC explores a space of dimensionality $i + d_i - 1$ compared to $d_i + 1$ in the case of the prioritized approach. Even though both, the prioritized approach and MRP-IC, are incomplete, the latter is expected to fail less often. The dimensionality of a composite space searched by MRP-IC can be higher than that of the individual configuration space of a robot or a coordination space searched in the basic decoupled planning or a space-time configuration space searched in the prioritized planning. But it will still be much lower than that of the joint configuration space of all the robots searched in the centralized planning for most practical scenarios.

IV. Experimental Results

We have implemented the three multi-robot motion planners - MRP-IC, DP (a decoupled planner), and CP (a centralized planner) - in a sampling-based planning framework ([1], Chapter 7), because it allows one to plan motions for robots with many DOF operating in complex environments. Specifically, we used the single-query Probabilistic Roadmap (PRM) planner SBL [21], available at http://ai.stanford.edu/∼mitul/mpk, for implementing the procedure SEARCH in Figure 1. We could have used other sampling-based single-query planners also, such as RRT [20] and PRT [22], with comparable effectiveness.

Figure 2 shows the four planning environments (Examples 1-4, with the respective start and goal configurations) that we used for comparing the multi-robot planners. We had two versions each for Examples 1, 3, and 4, with different numbers of robots. All of the planning examples required tight robot coordination for collision-free motions between the respective start and goal configurations. Table I lists the total numbers of robots, DOF, and geometric primitives (triangles) in each of the planning examples, giving some sense of the environment complexity. A collision-free multi-robot path between the start and the goal configuration exists in each of the planning examples, because at least one of the planners was able to find it.

We ran the multi-robot planners in these environments on a 1GHz Intel Pentium III PC with 1GB RAM. Table II compares the average running time in seconds (denoted by T) and percentage success (denoted by S), computed over several runs (including unsuccessful runs), for the implemented planners,

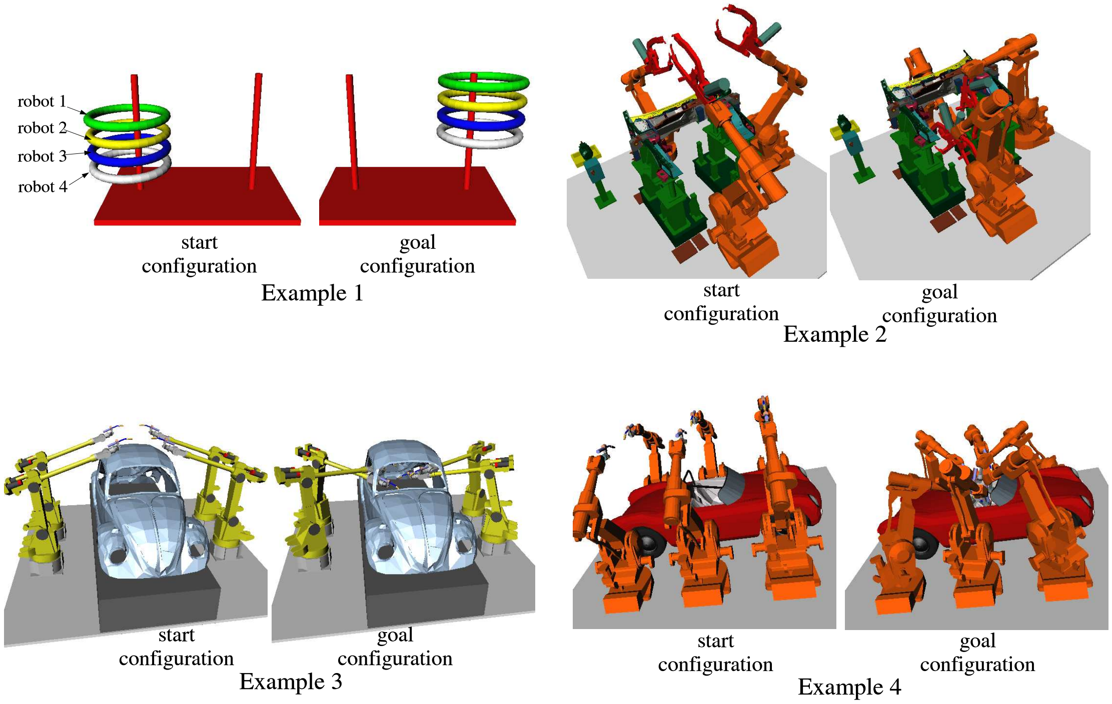

Fig. 2. The four planning environments, with robots in start and goal configurations, used in our experiments

in each of the planning examples. Each run of a planner was allowed to use at most 35% of the available RAM. The limit was set low enough to instigate failures of CP on the examples. The space complexity of the problem guarantees that any limited amount of memory can be made insufficient by increasing the difficulty of the examples.

Since a path exists for all of the examples, for CP a failure implies that more computational resources (available RAM in our case) are required to solve the respective planning problem. Additionally, for DP and MRP-IC, a failure may also be due to their respective incomplete natures.

According to the data in Table II, MRP-IC is the fastest planner and the only one having a non-zero success rate on all the examples. In particular, our new planner greatly improves upon the reliability of the basic decoupled planning approach. For instance, in Example 1 (4 robots), MRP-IC is successful more than 90% of the time, while DP is successful less than 7% of the time. Overall, MRP-IC has a success rate of 100% in 3 out of 7 problems and a success rate of more than 85% in 5 out of 7 problems. On the other hand, DP has a success rate of less than 35% in all the cases. MRP-IC is also faster than CP in all the cases. Even though CP is probabilistically complete (as SBL is probabilistically complete [21]), it has a worse success rate than MRP-IC. This implies that centralized planning requires significantly more computational resources.

Also the performance of DP (especially the low success rate in all cases) in comparison to CP is in agreement with the comparative evaluation of the decoupled and the centralized planning approaches in [4].

MRP-IC essentially addreses the reasons for failures of CP and DP: dimensionality and deadlocking. For many relevant problems the composite spaces searched by MRP-IC are of lower dimensionality than the joint configuration space of the problem, and thus MRP-IC is much less likely to succumb to RAM exhaustation than CP. On the other hand, by re-tuning the velocities iteratively, as more robots are included in the motion plan, MRP-IC deadlocks less often than DP.

TABLE I

TOTAL ROBOTS, DOF, AND GEOMETRIC PRIMITIVES (TRIANGLES) IN EACH OF THE PLANNING ENVIRONMENT

	DOF	**triangles**
Ex. 1 (2 robots)	12	8452
Ex. 1 (4 robots)	24	16644
Ex. 2 (3 robots)	18	14048
Ex. 3 (2 robots)	12	9044
Ex. 3 (4 robots)	24	14048
Ex. 4 (3 robots)	18	30462
Ex. 4 (6 robots)	36	41244

V. CONCLUSION

In this paper we have addressed a very critical challenge in multi-robot motion planning: improving the reliability of the decoupled planners. This challenge was brought into light by a recent study [4], which demonstrated that the centralized planning approach is a better choice in many cases than the decoupled approach, due to latter's frequent failing nature. Responding to this challenge, we proposed a new multi-robot motion planning strategy based on incremental coordination, which we call MRP-IC.

The key idea in MRP-IC is to partially merge the two phases of the classical decoupled planning approach. Our

TABLE II

COMPARISON OF CENTRALIZED, DECOUPLED, AND OUR NEW PLANNER

	Ex. 1 (2 Robs)		Ex. 1 (4 Robs)		Ex. 2		Ex. 3 (2 Robs)		Ex. 3 (4 Robs)		Ex. 4 (3 Robs)		Ex. 4 (6 Robs)	
	T	S	T	S	T	S	T	S	T	S	T	S	T	S
MRP-IC	0.65	100	146	91	89	100	160	86	2030	42	1.02	100	1719	63
DP	1693	29	1199	6.2	1213	0	1512	0	4212	0	1401	35	1989	15
CP	2.4	100	463	79	259	100	230	98	3811	0	16	100	5010	62

experimental results show that MRP-IC greatly improves upon the reliability of the basic decoupled planning approach.

In our tests, the MRP-IC planner outperformed the decoupled and the centralized planner as well. In many cases, it improved upon the success rate of the decoupled planner by many folds. In all cases, it was much faster than both. This indicates that MRP-IC can be a better choice than the decoupled as well as the centralized approach for solving multi-robot motion planning problems.

Like the basic decoupled planning approach, our new multi-robot motion planning approach is also inherently incomplete. In the future, we would like to characterize the class of problems in which either a centralized or a basic decoupled planner outperforms our new approach. The algorithm presented in this paper plans and coordinates the robots in an arbitrary order. It would be interesting to investigate if a priority scheme improves the performance of the approach. MRP-IC is a fast but incomplete algorithm, and as such it could be a candidate for being used as a local planner in a PRM motion planner. Such a planner would exploit the multi-robot nature of the problem but would be probabilistically complete.

REFERENCES

[1] H. Choset, et al.. "Principles of Robot Motion: Theory, Algorithms, and Implementations", MIT Press, 2005.

[2] J.H. Reif. "Complexity of the mover's problem and generalizations". Proc. the 20th IEEE Symp. on Found. Comp. Science, pages 421-427, 1979.

[3] J.-C. Latombe. "Robot Motion Planning". Kluwer Academic Publishers, Boston, MA, 1991.

[4] G. Sanchez and J.-C. Latombe. "Using a PRM planner to compare centralized and decoupled planning for multi-robot systems". Proc. IEEE Int. Conf. on Robotics and Automation, pages 2112-2119, 2002.

[5] T.-Y. Li and H.-C. Chou. "Motion planning for a crowd of robots". Proc. IEEE Int. Conf. Robotics and Automation, pages 4215-4221, 2003.

[6] S. Basu, R. Pollack, and M.-F. Roy. "Computing roadmaps of semi-algebraic sets on a variety", J. of the AMS, 3(1):55-82, 1999.

[7] B. Aronov, M. de Berg, A.F.Van der Stappen, P. Svestka, and J. Vleugels. "Motion planning for multiple robots". Discrete and Computational Geometry, 22:505-525, 1999.

[8] K.G. Kant and S.W. Zucker. "Towards efficient trajectory planning: path velocity decomposition". Int. J. of Robotics Research, 5:72-89, 1986.

[9] P.A. O'Donnell and T. Lozano-Perez. "Deadlock-free and collision-free coordination of two robot manipulators". Proc. IEEE Int. Conf. Robotics and Automation, pages 484-489, 1989.

[10] S.M. LaValle and S.A. Hutchinson. "Optimal motion planning for multiple robots having independent goals". IEEE Trans. on Robotics and Automation, 14(6):912-925, 1998.

[11] M. Erdmann and R. Lozano Perez. "On multiple moving objects". Proc. IEEE Int. Conf. Robotics and Automation, pages 1419-1424, 1986.

[12] C.W. Warren. "Multiple robot path coordination using artificial potential fields". Proc. IEEE Int. Conf. Robot. and Autom., pages 500–505, 1990.

[13] J.P. van der Berg and M.H. Overmars. "Prioritized motion planning for multiple robots". Proc. IEEE/RSJ Int. Conf. Int. Rob. and Sys., pages 2217-2222, 2005.

[14] M. Bennewitz, W. Burgard, and S. Thrun. "Finding and optimizing solvable priority schemes for decoupled path planning techniques for teams of mobile robots". Robotics and Autonomous Systems, 41(2):89-99, 2002

[15] Th. Fraichard. "Trajectory planning in a dynamic workspace: a 'state-time space' approach". Advanced Robotics, 13(1):75-94, 1999.

[16] G.A.S. Pereira, A.K. Das, V. Kumar, and M.F.M. Campos. "Decentralized motion planning for multiple robots subject to sensing and communication constraints". Multi-Robot Systems: From Swarms to Intelligent Automata, Volume II, Proc. International Workshop on Multi-Robot Systems, pages 267-278, 2003.

[17] C. Clark, S. Rock, and J.-C. Latombe. "Dynamic networks for motion planning in multi-robot space systems". Proc. Int. Symp. of Artificial Intelligence, Robotics and Automation in Space, 2003.

[18] L.E. Kavraki, P. Svestka, J.-C. Latombe, and M.H. Overmars. "Probabilistic roadmaps for path planning in high-dimensional configuration spaces", IEEE Trans. on Robotics and Automation, 12(4):566-580, 1996.

[19] N.M. Amato, O.B. Bayazit, L.K. Dale, C. Jones, and D. Vallejo. "OBPRM: An obstacle-based PRM for 3D workspace". Robotics: The Algorithmic Perspective, editors: P.K. Agarwal and et al., pages 155-168, 1998.

[20] J.J. Kuffner and S.M. LaValle. "RRT-connect: An efficient approach to single-query path planning". Proc. IEEE Int. Conf. on Robotics and Automation, pages 995-1001, 2000.

[21] G. Sanchez and J.-C. Latombe. "A single-query bi-directional probabilistic roadmap planner with lazy collision checking". Proc. Int. Symp. Rob. Res., pages 404-417, 2003.

[22] E. Plaku, K.E. Bekris, B.Y. Chen, A.M. Ladd, and L.E. Kavraki. "Sampling-based roadmap of trees for parallel motion planning". IEEE Trans. on Robotics, 21(4):597-608, 2005.

[23] P. Isto and M. Saha. "A slicing connection strategy for constructing PRMs in high-dimensional cspaces". Proc. IEEE Int. Conf. on Robotics and Automation, 2006.