Hybrid Probabilistic RoadMap - Monte Carlo Motion Planning for Closed Chain Systems with Spherical Joints

Li Han
Dept. of Mathematics and Computer Science
Clark University
Worcester, MA 01610
Email: lhan@clarku.edu

Abstract— **In this paper we propose a hybrid Probabilistic RoadMap - Monte Carlo (PRM-MC) motion planner developed under the general methodology of PRM. For a given robot, PRM planners generally need to sample and connect a large number of robot configurations in order to build a roadmap that reflects the properties (such as the connectivity or energy landscape) of the robot configuration space. The proposed PRM-MC planner uses Monte Carlo simulation to generate and connect neighboring robot configurations and uses PRM local planners to connect the connected components generated from MC simulation. This strategy follows the random sampling principle of PRM that leads to the probabilistic completeness of the PRM-type randomized planners, while exploring the continuity property of motion planning constraints to improve the computation efficiency and roadmap quality.**

We apply the PRM-MC approach to closed chain motion planning in this paper. Our current planner uses rotation pivots as attempted Monte Carlo moves for 3D closed chains with spherical joints. Pivot motions are developed as an efficient way to deform closed chains without violating the closure constraints, which have proved problematical for randomized approaches. We will discuss how to identify feasible rotation pivots of kinematic chains and utilize them in PRM-MC planning. Our simulation results show that the PRM-MC closed chain planner can build roadmaps with good connectivity and efficiently generate self-collision-free closure configurations for closed chain systems with many links and multiple loops.

I. Introduction

Motion planning problem [1] has been one of very actively studied problems in robotics community during the past decades. Given a robot, its work environment with obstacles, and start and goal configurations (positions and orientations), the motion planning problem amounts to finding a valid transition path (a sequence of intermediate configurations) between the two specified configurations. A valid path satisfies physical laws and system limits associated with robot motion, such as path continuity, collision avoidance and robot joint limits. Some major robotics motivations for the study of the path planning problem are the paramount importance of efficient motion planners in the realization of highly autonomous robots and in the applications of robots in manufacturing, space exploration and environment hazard cleaningup. Research interests in this problem have been further fueled by the insight that the robot motion planning problem shares much similarity with and can serve as a model of diverse physical geometry problems such as mechanical system disassembly, computer animation, protein folding, ligand docking and surgery planning.

Motion planning is a very challenging problem that involves complicated physical constraints and high-dimensional configuration spaces. The fastest existing complete (deterministic) planner [2] takes time exponential in the number of degrees of freedom of the robot. On the other hand, a class of randomized planners proposed during the last decade have successfully solved many previously unsolved problems. In particular, *Probabilistic RoadMap(*PRM*)* methods [3], [4], [5], [6], [7], [8], [9], [10] have been used successfully in high-dimensional configuration spaces. The general methodology of PRMs is to construct a graph (the roadmap) during pre-processing to capture the connectivity of the valid subset of the robot's configuration space and then to query the roadmap to find a path for a given motion planning task. Roadmap vertices are randomly sampled configurations which satisfy feasibility requirements (*e.g.*, collision free), and roadmap edges correspond to connections between 'nearby' vertices found with simple local planning methods.

To get a roadmap with connectivity reflecting the topological structure of the robot configuration space $\mathcal{C}$, PRM planners normally need to sample $\mathcal{C}$ quite extensively. This extensive sampling also facilitates connection between roadmap nodes and specified query configurations. Traditionally, the $\mathcal{C}$ space is sampled randomly, and only valid (depending on considered constraints) configurations and transitions between nearby configuration pairs are used as roadmap nodes and edges. The randomized approaches treat each node generation independently and can be proved to be probabilistic complete. Treating each node generation separately, however, does not take advantage of the *continuity* property existing in most constraints involved in robot motion planning problems. When a robot configuration is collision free, stays within joint limits, and/or has low energy, there is higher probability to find more configurations with similar desirable traits in its neighborhoods than in the neighborhoods of a configuration without these properties.

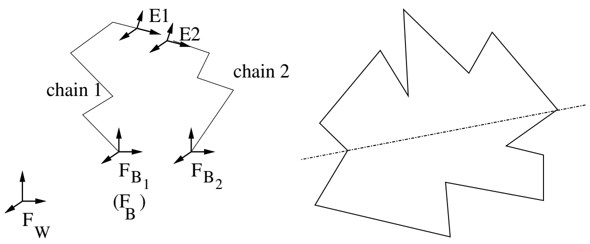

Fig. 1. Closure Configuration Generation Approaches

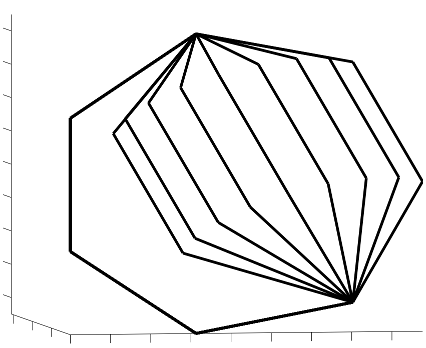

Fig. 2. A Sample Pivot Motion Generated by Our Planner

The PRM-MC planner proposed in this paper is designed to explicitly take advantage of the continuous property and generate nodes and edges in the neighborhoods of existing valid configurations. First the planner generates a small number of seed configurations using random sampling or some other more deliberate approaches. Then the planner uses Monte Carlo simulation to generate and connect nodes in the neighborhoods of the seed configurations to create graphs reflecting the connectivity of the neighborhoods. Finally the planner uses PRM type connection strategies [11] to try to connect these different connected components generated from Monte Carlo exploration.

In this paper we will show the application of this strategy in closed chain motion planning to illustrate the effectiveness of this approach. Closed chain systems arise in many practical problems such as Stewart Platform [12], molecular rings [13], reconfigurable robots [14], [15], and the closed chain system formed by multiple robots grasping an object [16], [17].While closed chains can offer advantages over open chains in terms of the rigidity of the mechanism, motion planning and control of closed chains is complicated by the need to maintain the loops existing in closed chain structures, the so-called *closure constraints*. Prior attempts in solving closed chain motion planning problem have mainly used the strategy of breaking a closed chain to at least two open chains and then try to satisfy the closure constrains by making the end points of the open chains meet. For example, in Figure 1, chain 1 and chain 2 are two open sub-chains of a closed chain and the frames E_1 and E_2 attached to the breakpoint (the "end effector") must coincide to satisfy the closure constraints.

While this approach can be applied in the Monte Carlo fashion, we propose to use rotation pivots as one efficient way to deform 3D closed chain systems with spherical joints to improve the successful rate of generating and connecting closure configurations. Take the single loop structure shown in the right half of Figure 1 as an example. One way to deform this closure structure is to rotate the links on one side of the axis, shown as a broken line in the figure, without moving links on the other side. Figure 2 shows the rotation pivot of a 7-link chain generated by our planer. The illustrated motion is an example of the rotation pivots where different subsets of a mechanism can rotate about an axis in different directions and/or with different angles without breaking the loops . The axes used in rotation pivots are called pivot axes or just pivots for short. As rotations are rigid body motion preserving distances and angles among points, rotation pivots can be proved [18] to preserve the closure constraints while simultaneously inducing changes to the closure configurations of the mechanism. We will discuss later how to identify feasible rotation pivots of kinematic mechanisms and use them in hybrid PRM-MC planning for closed chains.

Since closure constraints have been a major hurdle for motion planning of closed chain systems and are intrinsic properties of robot kinematic structures, we will consider motion planning for closed chains without environment obstacles. We will also fix one chain link to avoid the *trivial* rigid body motion that does not cause deformation of closed chain closure configurations. A roadmap generated under these conditions can serve as a kinematic roadmap used in a two-stage PRM closed chain planner [19], which populates copies of the kinematic map to the environment and takes environment obstacles into account at the second stage. Also as in paper[19], we do consider the robot self-collision (collision among robot links) avoidance constraints and only use self-collision-free configurations and paths as roadmap nodes and edges.

Our major motivation for this PRM-MC approach is to integrate the strengths of both probabilistic roadmap and Monte Carlo simulation methods. We use PRM to capture the global structure of $\mathcal{C}$ and use MC to explore configuration neighborhoods so as to improve the computation efficiency in roadmap generation. Our simulation results indicate that the PRM-MC planner can generate well-connected roadmaps for closed chains with many links and multiple loops.

II. Related Work on PRMs and Closed Chain Motion Planning

As a randomized planning approach, PRM has been proved to be probabilistic complete. The theoretical analysis results indicate that roadmaps generated by PRM planners can be consistent with the topological structure of the configuration space as long as PRMs has enough samples. In practice, many PRM planners randomly generate a large number of robot configurations in order to have a good coverage of the configuration space. A large number of well-distributed roadmap nodes can facilitate the connections among the nodes, improve the connectivity of the roadmap, and make the connections of the start and goal configurations to the roadmap easier. While this approach works well for robots with relatively simple constraints such as rigid body robots and open chain (serial chain) robots, a purely randomized approach is not computational efficient for other systems such as flexible objects[20] and proteins where low energy conformation are favored and for closed chains[21] where loops existing in kinematic structures need to be maintained, the so-called closure constraints.

Conceptually, a closed chain system can be viewed as a linkage system consisting of a collection of open chains, created from 'breaking' each closed chain, and then satisfy the closure constraints, if any, by forcing the break points to coincide. It has been proved [22], [23], [24] that the set of all robot configurations satisfying closure constraints, denoted by $\mathcal{C}_{closure}$, forms an algebraic variety and a compact manifold (except for robots with link lengths belonging to a finite set) embedded in the higher-dimensional configuration space $\mathcal{C}$. (Note that $\mathcal{C}_{closure}$ is defined by the closure constraints only, without any regard for other constraints such as collision avoidance and joint limits.) This is roughly analogous to embedding a two-dimensional surface or a one-dimensional curve in a three dimensional space. The fact that the volume measure of a low-dimensional entity in a high-dimensional ambient space is zero is why the probability that a random configuration in $\mathcal{C}$ satisfies the closure constraints is zero.

To the best of our knowledge, most motion planning algorithms for closed chains follow the strategy of breaking a closed chain to open sub-chains and then trying to make the end points meet. The first work of adapting PRMs to closed chains was reported in paper [21], where random configurations in the ambient $\mathcal{C}$ space , which normally would be in violation of closure constraints, were first generated and then tried to be pushed onto $\mathcal{C}_{closure}$ through a sequence of randomized gradient descent. Only configurations satisfying closure constraints (within certain tolerance) and avoiding robot self-collision(collision among robot links) and collision with environment obstacles were used in their roadmaps. A two-stage kinematics-based PRM planner for closed chains was reported in [19] where forward and inverse kinematics of links were used to make the end points meet and a kinematic map, with one robot linked fixed and environment obstacles ignored, was first generated and then populated in the environment. This work was motivated by the observation that robot closure constraints are intrinsic to the robot mechanism. By generating and populating kinematic maps, the planner could explore $\mathcal{C}_{closure}$ more thoroughly and generate and connect closure configurations in the environment more efficiently. To connect two different closure configurations, the above two approaches follow similar strategies: generate a sequence of intermediate configuration using some simple local planners such as random walk and straight-line planners and then try to make each intermediate configuration satisfy closure constraints via random descent or kinematics computation. As reflected in the simulation results in their paper, this connection approach has not been very efficient, especially for closed chains with many links.

The closed chain motion planning problems as described above are closely related to the *polygon reconfiguration* problems that have been extensively studied in the geometry community. Briefly, the polygon reconfiguration problem studies how to deform one 2D polygonal configuration to another while maintaining the distance constraints between joints and allowing 3D intermediate configurations. Clearly the polygon reconfiguration problem can be viewed as one special type of closed chain motion planning with both end configurations being 2D polygonal configurations.

One type of polygon reconfiguration problems is the polygon convexification problem addressing how to reconfigure a polygon to a convex one through a sequence of motions. Erdos[25] first proposed to use *flip* moves, which are rotations about lines of support defined by the convex hull of the polygon, to solve this problem, which was proved to work by Nagy[26]. Since then, various problems related to polygon convexification have been studied and different moves [22], [24]have been identified. In particular, *rotation pivots* have been regarded as one type of moves effective for polygon reconfiguration. In brief, rotation pivots are rotations about an axis, defined by a pair of non-adjacent vertices, by an open subchain with the two vertices as the endpoints. Figure 2 shows the pivot motion of a 7-link chain generated by our planer. See [27], [28], [29] for more information on polygon reconfiguration.

Most of the prior work on polygonal reconfiguration allows collision. When collision is allowed, it has been proved[22] that any two closure configurations of any 3D closed chain with fully rotatable spherical joints can be deformed from one to the other. In other words, $\mathcal{C}_{closure}$ *of 3D closed chains with spherical joints is connected if it is not empty*. Several complete planners such as the line tracking planner [22] have been developed for deforming polygons. Recently, with their new results on the manifold properties of the $\mathcal{C}_{closure}$ (corresponding to $\mathcal{C}_{kin}$ in their papers), Trinkle and Milgram [23], [24] have developed a complete path planner on $\mathcal{C}_{closure}$ for 2D closed chains with revolute joints and 3D closed chains with spherical joints. Their planner uses accordion moves to bring link by link to their goal configurations and is guaranteed to find a transition path between two closure configurations for any 3D closed chain system with spherical joints. Note that as a planner on $\mathcal{C}_{closure}$, the accordion planner only considers closure constraints without any regard for collision avoidance and may generate paths involving robot self collision. It was noted in their papers that the complete planner could be used as a local planner for PRM in conjunction with collision checking package to discard paths involving collisions. In addition, the computational strategies in their planner can also be used for the generation of closed chain configurations. For example, after randomly generating a configuration for one of the open chains created from breaking a closed chain, we can try to make other chains to reach the same distance between the base and the end-effectors and then use rigid motions to bring the end points to a same point, a same strategy as used in the accordion path planner. This approach, however, can be computational intensive since the planner iteratively uses Newton-type algorithm in solving closure constraints.

Closed chain deformations have also been studied by other approaches such as mechanism singularity analysis, where most of the work does not consider self-collision problems. Please refer to papers[23], [24] for a more detailed discussion of the prior work on closed chain path planning.

When collision is prohibited, researchers have identified polygons that cannot be reconfigured to planar convex polygons. This means that in general, the set of all polygonal configurations consists of multiple connected components with collision disallowed. It is recently proved[30] that the polygon reconfiguration problem under collision-free constraints is P-space hard.

This paper proposes a hybrid Probabilistic RoadMap - Monte Carlo (PRM-MC) planner for closed chain systems. As mentioned in section I, we will only consider motion planning for closed chains with fixed bases and without environment obstacles. This setting is also similar to what was used in the development of the accordion planner. But as in the two stage PRM closed chain planner [19], our planner considers the self-collision avoidance constraints and only uses self-collision free nodes and edges in roadmaps. The PRM-MC planner has been designed to generate a good sampling of configuration space while improving the node generate and connection efficiency, two critical components in the construction of roadmaps. Our basic idea is to take advantage of the continuity property in the robot $\mathcal{C}$ space and use deliberate techniques as part of the Monte Carlo exploration of configuration neighborhoods.

Some previously developed planners can be interpreted with implicit usage of the neighborhood exploring concepts, one trademark of Monte Carlo simulation. For example, more sampling in the neighborhood of configurations in difficult regions aimed at improving roadmap connectivity [8], [31] and populating kinematic roadmaps in environments with obstacles taken into account at the second stage of the closed chain planner can be considered in such a fashion. Rapidly-expanding Random Trees (RRT) [32] is one PRM variant that grows a tree from each seed configuration with the expansion of the tree biased toward under-sampled regions. RRT can be viewed as implicitly following Monte Carlo principles: each attempted expansion from existing trees can be viewed as a trial move. One distinguishing feature of the planner presented in this paper is that it explicitly incorporate Monte Carlo simulation in the generation of the whole roadmap and systematically uses the information of existing nodes to increase the probability of obtaining valid configurations and configuration transitions that can serve as roadmap nodes and edges.

III. Hybrid PRM-MC Motion Planning

The major steps of our hybrid PRM-MC planner can be described as follows.

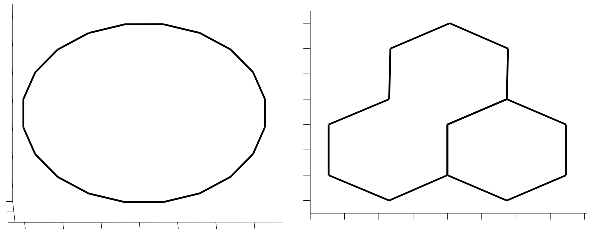

Fig. 3. Seed Cfgs for a single loop with 20 links and a two-loop structure

PROTOTYPE PRM-MC PLANNER FOR A ROBOT

I. PREPROCESS THE KINEMATIC STRUCTURE AND THE ENVIRONMENT OF THE ROBOT
   Identify Monte Carlo moves such as rotation pivots

II. GENERATE SEED CONFIGURATIONS
   Generate configurations via random sampling or other methods and retain those with "desired" properties as seed cfgs

III. CONDUCT MONTE CARLO SIMULATION IN THE NEIGHBORHOODS OF THE SEED CFGS
   Use Monte Carlo simulation to generate neighboring nodes retain the valid nodes as roadmap nodes and the valid Monte Carlo moves between nodes as roadmap edges

IV. USE PRM-TYPE CONNECTION STRATEGY TO CONNECT DIFFERENT CONNECTED COMPONENTS OF THE ROADMAP GENERATED FROM STEP III

Step I analyzes the kinematic structure and the environment of the robot and identifies Monte Carlo moves that are expected to have high probability of leading to valid configurations from existing configuration with desired properties. In the case of closed chain motion planning, we want to find some deformation schemes that can generate new closure configurations from existing ones based on the analysis of the kinematic structure of the robot. For some other planners such as obstacle-based PRM[3] and media-axis PRM [9], [33] where environment obstacles are considered, the information of the environment and the robot structure will be used in deriving desirable Monte Carlo moves that can make robot configurations stay in contact with obstacles or on the media axis of the environment. We will discuss in next section how to identity rotation pivots as one type of favored Monte Carlo moves for closed chains.

Step II generates some seed configurations to be used as starting points for the Monte Carlo simulation in step III. These seed configurations can be generated using random sampling and other approaches. For example, in the case of protein folding, we can include known native folds of the studied protein stored in PDB as (a subset of) seed configurations. Also for a loop formed by links of equal length, we can generate regular n-gon configuration for the loop, where n is the number of links in the loop. We can also easily generate some other special seed configurations such as rectangular configurations for loops with even number of equal-length links. This type of heuristics can be generalized to some extent to multi-loop mechanisms. Figure 3 shows one seed configuration for a 20-link chain and a two-loop structure, respectively.

Step III uses Monte Carlo simulation to generate and connect nodes in the neighborhoods of the seed configurations with favored Monte Carlo moves identified in Step I. Purely randomized walk can be used be in MC simulation for some appropriate problems. Please note that the neighborhoods of the seed configurations explored by MC are not necessarily limited to "local" neighborhoods. Some Monte Carlo moves such as rotation pivots of closed chains can induce significant deformation to closed chains, which correspond to new configurations in $\mathcal{C}_{closure}$ that are far away from the original one.

Our current planner uses two parameters to control the MC exploration of the neighborhood of each seed configuration: one is the number of MC runs and the other is the number of successful MC moves per run. Different values of these two parameters encode different preferences for MC exploration. For example, small number of MC runs per seed and large number of MC moves per run per seed prefer expanded exploration of the configuration space while large number of MC runs per seed and small number of MC moves per run per seed causes dense sampling in the neighborhoods of seed configurations.

Step IV uses PRM-type connection strategies to try to connect the different connected components of the roadmap graph generated in step III . Our current closed chain planner is built upon the OBPRM code developed by Amato's group [34] and has access to a diverse set of connection strategies and local planners implemented in the OBPRM code. In our preliminary simulation study, we used the straight line planner to try to connect close nodes in different connected components. We are in the process of implementing more deliberate planners such as the local Jacobian planner and the complete accordion planner[24], [23] as local planners. We are also study the feasibility of using rotation pivots in local planners.

IV. Deformations of Closed Chain Systems

In order to use rotation pivots to deform a kinematic chain system, we need to identify rotation axes about which different subsets of the joint systems can rotate in different directions and/or angles without breaking the loop structures in the chain. It is well known that any non-adjacent vertex pairs of a single-loop closed chain can define a pivot axis. To facilitate rotation pivots in general closed chain structures, we have identified the connection between the joints that can form a rotation axis and the articulation sets of the graph underlying the closed chain structure. An articulation set of a connected graph is a subset of the vertices of the graph whose removal will disconnect the graph. In other words, the graph resulted from removing the vertices of the articulation set and the edges incident on these vertices will have more than one connected components. It is the joints corresponding to these different connected components that can potentially have different rotations about the rotation axis formed by the vertices corresponding to the articulation set while satisfying the closure constraints. This intuitive description can be formalized in the following theorem that is proved in our technical report.

Theorem *1:* For a 3D kinematic chain system with spherical joints, when the system joints that correspond to the vertices of an articulation set of the underlying graph are on one straight line, this line can serve as a rotation axis. Subsets of system joints belonging to different connected components in the graph resulted from cutting out the articulation set can rotate about the pivot axis in different directions and with different angles without breaking any kinematic constraints

Based one the above theorem, it is easy to reach the following conclusions regarding articulation sets of different cardinality (different numbers of vertices in articulation sets).

Corollary *1:* For a 3D kinematic chain system with spherical joints, any articulation set of cardinality less than or equal to two can define a pivot axis.

This is because a line can always be defined with one point or two points to pass by the line. Since n points, when $n > 2$, generally do not fall on one line, we have the following result for articulation sets of cardinality greater than 2.

Corollary *2:* For a 3D kinematic chain system with spherical joints, an articulation set of cardinality greater than two can define a pivot axis only when the system joints corresponding to the articulation set are collinear.

Articulation sets are a general concept with no restriction on applicable graph structures, and there exist many graph algorithms analyzing graph connectivity and identifying articulation sets [35]. Similar to closure constraints, pivot axes are also intrinsic properties of closed chains. So in step I of our algorithm, the PRM-MC planner analyzes the closed chain structure and identifies the set of all possible pivots and corresponding connected components. Then in step III, the planner randomly picks a pivot axis and generates different rotation pivots (different rotation directions and angles) for different connected components. For a given connected component and its rotation direction and angle about the pivot axis, the planner computes the rigid transformation matrix in $SE(3)$ corresponding to the motion and uses it to update the configurations of the links in this connected component. After updating configuration for each moved link, the planner checks for collision among links. For a self-collision free configuration, the planner updates the joint coordinates affected by the rotation pivot.

V. Simulation Results

We have developed PRM-MC planners for closed chains and proteins and have obtained promising simulation results in our preliminary study [18]. Our current planner is written in C++ and built upon the OBPRM code, developed by Amato's research group[34]. All simulation results reported in this section were performed on a 1.2GHz Pentium III laptop and used the RAPID [36] package for 3D collision detection.

This section includes results for a few closed chain systems that are composed of links of equal lengths. In particular, we consider four closed chain structures in this paper: the first three have only one loop with the number of links being 7, 20, and 100 respectively; and the last one has two loops, as shown in Figure 3, with one loop having 6 links and the other having 10 of which two links are shared between the loops.

Table I shows the running times (seconds) and statistics of roadmaps constructed with and without the Monte Carlo

TABLE I

ROADMAP CONSTRUCTION TIMES (SECONDS) AND STATISTICS

Roadmap Construction							
	Pivot MC			Generation		Connection	
Chains	sec	cfg	CC	sec	cfg	sec	CC
7(0)	2.90	101	9	2.90	101	0.95	1
7(1)	8.06	202	9	8.06	202	1.81	1
7	–	–	–	0.13	100	9.58	4
20(0)	5.44	101	1	5.44	101	0.0	1
20(1)	14.98	202	7	15.28	202	1.78	4
20	–	–	–	36.55	100	0.91	97
100(0)	15.09	101	1	15.09	101	0.0	1
100(1)	15.56	101	1	31.73	101	0.0	1
100	–	–	–	172.20	0	–	–
2loops(0)	6.76	101	4	6.76	101	1.19	1
2loops(1)	15.33	202	7	15.47	202	2.34	4
2loops	–	–	–	6.98	100	7.17	64

explorations that used pivot deformations as attempted MC moves. In the table, cfg and CC denote the number of nodes and connected components of roadmaps generated from each step. For the results presented in table I, we used our kinematics-based PRM closed chain planner to try to generate and connect 100 random nodes. (Recalled that our PRM closed chain planner breaks the loops to open chains and then tries to use forward and inverse kinematics to close the loops.) We used our PRM-MC planner to conduct ten Monte Carlo simulation runs, each with ten successful MC moves, for each seed configuration. We included the regular n-gon configuration of each structure as a seed configuration. We also tried to use our PRM closed chain planner to generate one random configuration as an additional seed configuration.

In the "chains" column in table I, the numbers in the parentheses after the link numbers are the numbers of random seeds used in our PRM-MC planner. So the results in the first row of each structure were generated with the regular polygon as the only seed configuration, while the results in the second row were generated with one additional random seed, if one was found by our PRM closed-chain planner within certain number of tries. The results in the last row of each structure were generated by the PRM closed chain planner. The generation times for PRM-MC planners include times for both random seed generation and rotation pivot. The differences between the generation times and the corresponding rotation pivot times were time spent by the PRM planner to generate a random self-collision-free closure configuration. It should be noted that the PRM-MC node generation steps, namely seed configuration generation and rotation pivots, also generate edges in the rotation pivot step. As shown in the table, the pivot MC successfully connected all nodes for some systems and generated a roadmap of one connected component. For these cases, the PRM-type connection step was not executed.

The table shows that our kinematics-based PRM planner performed reasonable well only for the 7-link chain. It failed to find any self-collision-free closure configurations for the 100-link chain after spending 172.2 seconds on 10^5 random tries. So the third raw for the 100-link chain in table I contains few numerical data. For the results shown in the second raw for the 100-link chain, our PRM-MC planner stopped the unsuccessful generation of a random seed after 10^4 tries. As for the 20-link chain and the two-loop chain, the PRM planner made few connections and generated roadmaps with many connected components. On the other hand, our PRM-MC planner generated roadmaps with a small number of connected components for all four closed chain structures. The good connectivity of the PRM-MC roadmap shows the effectiveness of using pivot deformation as MC moves to sample and connect closure configurations. We emphasize, however, that the connectivity of the PRM-MC roadmaps as given in table I is not sufficient for drawing conclusions about the topological connectivity of the self-collision-free configuration spaces of closed chains, although the results probably do shed some lights on the issue.

VI. CONCLUSION

Probabilistic roadmap methods have been very successful in solving complex problems in high-dimensional configuration spaces. Numerous trajectories between pairs of configurations can be extracted from the generated roadmap graph. This is one distinct feature of PRM as compared to Monte Carlo simulation, where each successful simulation run generates one trajectory and many simulation runs simply fail. The success of PRM can be partly attributed to its efficient reuse of nodes and edges in the roadmap graph. Conventionally, roadmap nodes are generated independently. In this paper, we propose a hybrid PRM-MC planner where MC simulation is used to generate and connect roadmap configurations in neighborhoods of existing configurations. This approach is designed for taking advantage of the continuity property of most constraints involved in motion planning to improve the successful rate of node generation and connection.

This paper also presents a PRM-MC motion planner for 3D closed chains with spherical joints to demonstrate the effectiveness of this hybrid approach. In particular, we identify rotation pivots as one efficient way to generate and connect closure configurations while maintaining the closure constraints. Our preliminary simulation results shows that our planner can efficiently generate well-connected roadmaps for closed chain with many links and multiple loops.

We are currently working on various improvements on our closed chain planner. The current planner does not consider joint limits. Incorporating joint limit constraints will restrict feasible pivot deformation to stay within the joint limits. A related interesting problem would be to introduce different penalties for different types of structure changes and try to generate optimal rotation pivot under the penalty definition. This problem is motivated by the models of protein flexibility where torsion angles are generalized modeled to be flexible and bond angles are normally modeled with fixed value or a small range of allowed changes. Pivot motion as discussed in this paper is not feasible for closed chain with revolute joints. For this situation, the local and accordion planners[24], [23] as well as the line tracking planner[37] can be adapted to serve as MC moves. For systems that have feasible pivot deformation, we are working on a local planner that uses rotation pivots to connect two closure configurations. However, there are other deformation modes such as sheer motion available to closed chain mechanism. So we need to identify conditions under which it is feasible to use a sequence of rotation pivot to connect two closure configurations. If two closure configurations cannot be connected by rotation pivots (and other deformation methods), we can use the accordion complete planner to generate a transition path between them.

ACKNOWLEDGMENT

The work presented in this paper is partly inspired by the important theoretical results developed by Drs. J.C. Trinkle and R.J. Milgram. I am grateful to Dr. Trinkle for giving me his matlab code for the accordion path planner for 2D closed chains with revolute joints. I would also like to thank Dr. N.M. Amato and her robotics group for letting me build upon their motion planning code and providing support for their ever-expanding OBPRM code. My thanks also go to Dr. H. Servatius who gave a series of talks to our department on combinatorial rigidity.

REFERENCES

[1] J. C. Latombe, *Robot Motion Planning*. Boston, MA: Kluwer Academic Publishers, 1991.

[2] J. F. Canny, "On computability of fine motion plans," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 1989, pp. 177–182.

[3] N. M. Amato, O. B. Bayazit, L. K. Dale, C. V. Jones, and D. Vallejo, "OBPRM: An obstacle-based PRM for 3D workspaces," in *Proc. Int. Workshop on Algorithmic Foundations of Robotics (WAFR)*, 1998, pp. 155–168.

[4] N. M. Amato and L. K. Dale, "Probabilistic roadmap methods are embarrassingly parallel," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 1999, pp. 688–694.

[5] N. M. Amato and Y. Wu, "A randomized roadmap method for path and manipulation planning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 1996, pp. 113–120.

[6] V. Boor, M. H. Overmars, and A. F. van der Stappen, "The Gaussian sampling strategy for probabilistic roadmap planners," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 1999, pp. 1018–1023.

[7] L. Kavraki and J. C. Latombe, "Randomized preprocessing of configuration space for fast path planning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 1994, pp. 2138–2145.

[8] L. Kavraki, P. Svestka, J. C. Latombe, and M. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans. Robot. Automat.*, vol. 12, no. 4, pp. 566–580, August 1996.

[9] S. A. Wilmarth, N. M. Amato, and P. F. Stiller, "Motion planning for a rigid body using random networks on the medial axis of the free space," in *Proc. ACM Symp. on Computational Geometry (SoCG)*, 1999, pp. 173–180.

[10] M. Apaydin, D. Brutlag, C. Guestrin, D. Hsu, and J.-C. Latombe, "Stochastic roadmap simulation: An efficient representation and algorithm for analyzing molecular motion," in *Proc. Int. Conf. Comput. Molecular Biology (RECOMB)*, 2002, pp. 12–21.

[11] G. Song, S. L. Thomas, and N. M. Amato, "A general framework for prm motion planning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2003.

[12] D. Stewart, "A platform with six degrees of freedom," *Proc. of the Institute of Mechanical Engineering*, vol. 180, no. part I(5), pp. 171–186, 1954.

[13] A. Singh, J. Latombe, and D. Brutlag, "A motion planning approach to flexible ligand binding," in *7th Int. Conf. on Intelligent Systems for Molecular Biology (ISMB)*, 1999, pp. 252–261.

[14] K. Kotay, D. Rus, M. Vona, and C. McGray, "The self-reconfiguring robotic molecule: Design and control algorithms," in *Proc. Int. Workshop on Algorithmic Foundations of Robotics (WAFR)*, 1998, pp. 375–386.

[15] A. Nguyen, L. J. Guibas, and M. Yim, "Controlled module density helps reconfiguration planning," in *Proc. Int. Workshop on Algorithmic Foundations of Robotics (WAFR)*, 2000.

[16] J. Trinkle, A.O.Farahat, and P. Stiller, "First-order stability cells of active multi-rigid-body systems," *IEEE Trans. Robot. Automat.*, vol. 11, no. 4, pp. 545–557, 1995.

[17] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal, "Vehicle/arm coordination and multiple mobile manipulator decentralized cooperation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 1996, pp. 546–553.

[18] L. Han, "A hybrid probabilistic roadmap – monte carlo motion planner for articulated robots and proteins," Clark University, Tech. Rep., 2003.

[19] L. Han and N. M. Amato, "A kinematics-based probabilistic roadmap method for closed chain systems," in *Algorithmic and Computational Robotics – New Directions (WAFR 2000)*, 2000, pp. 233–246.

[20] L. Kavraki, F. Lamiraux, and C. Holleman, "Towards planning for elastic objects," in *Proc. Int. Workshop on Algorithmic Foundations of Robotics (WAFR)*, 1998.

[21] S. LaValle, J. Yakey, and L. Kavraki, "A probabilistic roadmap approach for systems with closed kinematic chains," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 1999.

[22] W. Lenhart and S. Whitesides, "Reconfiguring closed polygon chains in euclidean d-space," *Discrete and Computational Geometry*, vol. 13, pp. 123–140, 1995.

[23] R. Milgram and J. Trinkle, "The geometry of configuration spaces for closed chains in two and three dimensions," *Homology Homotopy and Applications*, 2002, to appear.

[24] J. Trinkle and R. Milgram, "Complete path planning for closed kinematic chains with spherical joints," *Int. J. Robot. Res.*, vol. 21, no. 9, pp. 773–789, 2002.

[25] P. Erdős, "Problem 3763," *Amer. Math. Monthly*, vol. 42, p. 627, 1935.

[26] B. de. Sz.Nagy, "Solution to problem 3763," *Amer. Math. Monthly*, vol. 46, pp. 176–177, 1939.

[27] G. T. Toussaint, "The Erdos-Nagy theorem and its ramifications," *Proc. ACM Symp. on Computational Geometry (SoCG)*, 1999.

[28] T. C. Biedl, E. D. Demaine, M. L. Demaine, S. Lazard, A. Lubiw, J. OŔourke, M. H. Overmas, S. Robbins, I. Streinu, G. T. Toussaint, and S. Whitesides, "Locked and unlocked polygonal chains in three dimensions," *Discrete and Computational Geometry*, vol. 26, pp. 269–281, 2001, one of the papers that showed the existence of multiple components in the collision-free space of polygonal chains.

[29] E. Demaine, S. Langerman, J. OŔourke, and J. Snoeyink, "Interlocked open linkages with few joints," in *Proc. ACM Symp. on Computational Geometry (SoCG)*, 2002, pp. 189–198.

[30] H. Alt, C. Knauer, G. Rote, and S. Whitesides, "The complexity of (un)folding," in *Proc. ACM Symp. on Computational Geometry (SoCG)*, 2003, pp. 164–170.

[31] D. Hsu, L. Kavraki, J.-C. Latombe, R. Motwani, and S. Sorkin, "On finding narrow passages with probabilistic roadmap planners," in *Proc. Int. Workshop on Algorithmic Foundations of Robotics (WAFR)*, 1998.

[32] S. M. LaValle and J. J. Kuffner, "Rapidly-Exploring Random Trees: Progress and Prospects," in *Proc. Int. Workshop on Algorithmic Foundations of Robotics (WAFR)*, 2000, pp. SA45–SA59.

[33] C. Holleman and L. E. Kavraki, "A framework for using the workspace medial axis in prm planners," in *Proceedings of the 2000 International Conference on Robotics and Automation (ICRA 2000)*. San Fransisco, CA: IEEE Press, 2000, pp. 1408–1413.

[34] N. M. Amato, "Motion planning group webpage," `http://parasol-www.cs.tamu.edu/people/amato/`.

[35] R. Sedgewick, *Algorithms in C++, Part V: Graph Algorithms*. Addison-Wesley, 2001.

[36] S. Gottschalk, M. Lin, and D. Manocha, "Obb-tree: A hierarchical structure for rapid interference detection," University of N. Carolina, Chapel Hill, CA, Technical Report TR96-013, 1996.

[37] W. Lenhart and S. Whitesides, "Reconfiguring closed polygon chains in euclidean d-space," *Discrete and Computational Geometry*, vol. 13, pp. 123–140, 1995.