

A Probabilistic Roadmap Planner for Flexible Objects with a Workspace Medial-Axis-Based Sampling Approach

Leonidas J. Guibas
Department of Computer Science
Stanford University
Stanford, CA 95305
guibas@cs.stanford.edu

Christopher Holleman Lydia E. Kavraki
Department of Computer Science
Rice University
6100 Main Street, Houston, TX 77005
{rusty,kavraki}@cs.rice.edu

Abstract

Probabilistic roadmap planners have been used with success to plan paths for flexible objects such as metallic plates or plastic flexible pipes. This paper improves the performance of these planners by using the medial axis of the workspace to guide the random sampling. At a preprocessing stage, the medial axis of the workspace is computed using a recent efficient algorithm. Then the flexible object is fitted at random points along the medial axis. The energy of all generated configurations is minimized and the planner proceeds to connect them with low-energy quasi-static paths in a roadmap that captures the connectivity of the free space. Given an initial and a final configuration, the planner connects these to the roadmap and searches the roadmap for a path. Our experimental results show that the new sampling scheme is successful in identifying critical deformations of the object along solution paths which results in a significant reduction of the computation time. Our work on planning for flexible objects has applications in industrial settings, virtual reality environments, and medicine.

1 Introduction

Robot path planning is important in a wide array of applications from industrial automation to computer-aided animation and computational pharmacology [6, 9, 11, 20]. The problem considered in this paper is an extension of the traditional path planning problem of [23]: given a flexible object/robot with known physical characteristics (e.g. an elastic object) and a set of ways that the object can be manipulated (called manipulation constraints or limit conditions hereafter), find a quasi-static path for the object from an initial to a goal configuration. The deformations of the object are prescribed by the manipulation constraints in combination with a physical model of the object. Figure 1 offers an illustration. The flexible object is an elastic plate and the manipulation constraints restrict the tangents of two opposite edges of the plate. The snapshot shown is from a path that has been computed by the planner described in this paper.

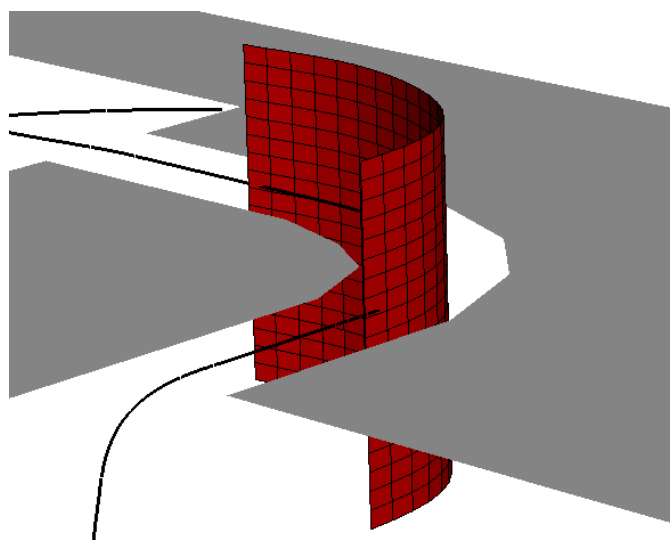


Figure 1: Snapshot along a manipulation path for an elastic plate (Three-quarter view).

Several important applications motivate our research on planning for flexible objects: in industrial settings there is a need to manipulate sheets of metal [26], pipes that can bend [30], and cables [25]. In assembly maintainability studies done with virtual prototyping, planning is used to compute a removal path for a part from an assembly, given only the CAD model of the assembly [6]. The flexibility of the part needs to be considered as engineers use deformable parts to produce compact assemblies [5]. In medical and surgical procedures, flexible catheters are inserted into human vessels [4, 31]. Accurate planning studies may help in choosing the size and properties of the catheter used. In computer-assisted pharmaceutical drug design, path planning techniques are used to compute paths for drug molecules to their docking sites [29]. In that context the rigorous treatment of the physical properties of the drug molecule is crucial for obtaining sequences that are of low-energy and can thus be encountered in nature. Last but not least, our work may have applications in domains like computer generated animation and virtual en-

vironments where the physical properties of objects need to be considered for the creation of realistic motion.

The probabilistic roadmap approach to planning (PRM) [16, 18, 19, 24, 27] has been adapted to flexible objects and a new planner (f-PRM) is described in [14, 17, 22]. f-PRM follows the principle of the PRM framework. Initially PRM planners generate a large number of nodes through probabilistic sampling and then create a roadmap by making local connections between nodes. Initial and goal configuration can be connected to this roadmap which is then searched for a feasible path. Unfortunately, f-PRM is computationally expensive. A general deformation is infinite-dimensional. Let us consider as an example the case of an elastic surface. We are forced to approximate with available geometric representations. We can model it using a Bézier surface which constrains us to a finite-dimensional subspace of the deformation space. The dimension of this subspace must still be large in order to accurately model the elastic properties. The set of usable deformations is an even smaller subset of this deformation space — deformations that are within the elasticity domain of the material and can be produced practically in real life. To obtain deformations from this subset we introduce manipulation constraints that succinctly describe practical manipulation situations and rely on a minimization procedure to map a set of manipulation constraints (a limit condition) to a geometric representation. This allows us to plan in a suitably low-dimensional space. Minimization, however, is costly. It is desirable to choose only deformations that are likely to contribute to a path which we accomplish by matching the manipulation constraints to sections of the medial axis of the workspace. The geometry and topology of the medial axis dictate the creation of deformations. This is illustrated in Figure 1 where the elastic surface ‘follows’ the medial axis to go through a narrow passage. While fitting a flexible object is more straightforward and more intuitive, the rigid 6D robot case is also an interesting issue and is investigated in a companion paper [13]. We are also investigating the use of the GVG [7] instead of the medial axis.

This paper is organized as follows. Section 2 discusses related work and Section 3 contains all necessary notation as well as a brief description of the f-PRM scheme. Section 4 describes our new sampling scheme. For the computation of the medial axis we use a well-known recent algorithm developed by Amenta et al [3] that was originally devised for the reconstruction of surfaces. Section 4 also presents an example of our fitting scheme for a simple case. Experiments in two dimensions are given in Section 5. We conclude with future work in Section 6.

2 Related Work

Planning for robots with many dof has been extensively treated in recent literature ([1, 2, 11, 12, 21, 23]). The proba-

bilistic roadmap approach to planning (PRM) [16, 18, 19, 24, 27] has gained wide acceptance because the method is easy to implement and use and provides good performance results. The work in [14, 17, 22] has produced f-PRM, a planner for flexible objects. When flexible objects are manipulated one needs to model the geometry of the objects and their mechanical properties and employ powerful minimization procedures to compute their shape as a local minimum of their internal energy. Hence the work in this paper draws from research in geometric modeling [8], mechanics [32], optimization theory [28], and graphics [10].

An important issue in PRM planners is the method for choosing the random configurations for the construction of the roadmaps. Recent work has considered several alternatives to a uniform random distribution of configurations for rigid or articulated robots. A resampling step, creating additional nodes in the vicinity of nodes that are connected with few others, is shown in [18, 19]. Nodes close to the surface of obstacles are added in [2]. A dilation of the configuration space has been suggested in [15]. In [33] a procedure for retracting configurations onto the freespace medial axis is presented. In this paper, we treat deformable objects and we attempt to generate configurations close to the medial axis of the workspace. The same idea is explored in [13] for rigid objects.

As far as the computation of the medial axis is concerned we use recent work that is especially suited for our problem. The algorithm in [3] reconstructs surfaces from unorganized data in \mathcal{R}^3 . It can be adapted to approximate the medial axis of the shape described by the data — this is described in the current paper. Since general workspaces come as polygon soups, our intention is to sample these polygons, efficiently construct a sufficient approximation of the medial axis, and generate configurations for our planner based on the medial axis. This is further explained in Section 4.

3 Background and Notation

Let W be our workspace of dimension d and let \mathcal{C} be the configuration space of our robot. \mathcal{C} is the Cartesian product of the deformation space \mathcal{D} and $\mathcal{M}_c = \mathcal{R}^d \times \text{SO}(d)$, the space of rigid-body transformations. Deformability can be caused by different types of mechanical behavior, the case of elasticity being the more common of them. For an object made of an elastic material, elasticity theory states the existence of a scalar function ψ , called the *density of elastic energy*, at each point x . The latter depends only on the local deformation at x . By integrating this local energy function over the domain of the object, we obtain a functional over the space of deformations. The value of this functional for any deformation $d \in \mathcal{D}$ is called *elastic energy* of the deformation. When an object is manipulated, the manipulation constraints restrict the position of certain parts of the object and the valid deformation is one

that minimizes the elastic energy of the object. In our work we always compute the local minimum of the elastic energy for a given manipulation constraint (both for guessed configurations and for configurations along a path) and hence all the paths that we produce are quasi-static and can be expected in nature. An upper bound on the elastic energy of allowable deformations safeguards our program from using deformations that violate the elasticity limit of the object and hence can cause a permanent deformation (plastic deformations).

4 f-PRM based on Workspace Medial-Axis Sampling

The traditional f-PRM generates random configurations of the flexible object. Given the decoupling of position and deformation described above, the generation of a configuration consists of (a) choosing uniformly at random the manipulation constraints, (b) minimizing the energy to obtain a valid deformation (c) generating a random translation and rotation, (d) checking the final configuration for collision with the obstacles. The generation of a configuration is expensive, mainly due to the high cost of minimization. Reusing deformations, as in [17], cuts down costs but also limits exploration of the deformation space.

In this paper, we form a heuristic for finding useful deformations by employing the following scheme. First we compute the medial axis of the workspace. In the sampling step we fit our object along the medial axis, or, to be more specific, we fit our manipulation constraints along the medial axis and minimize the energy of the object to obtain its correct shape. Details of our planner are given below. In this paper we limit our discussion to two dimensional workspaces. As an example, we treat the case where an elastic plate is manipulated by its opposite ends and we require that all rotations and translations are in the plane.

4.1 Computation of the Medial Axis

For the purposes of the planner, the medial axis is approximated by a subset of the Voronoi diagram of points sampled along the boundary of the freespace. The algorithm used is derived largely from [3]. An informal description of the two dimensional algorithm is given below; for a more rigorous examination see [3]. Note that for our purposes only an approximation of the medial axis is required. We do not rely on its precise attributes, but rather on capturing the topology of the freespace and maximizing clearance from obstacles. The medial axis is prone to having small “twigs” where there are perturbations in the obstacle boundaries. These are of no use during the planning stage and are discarded if their length is less than a specified threshold.

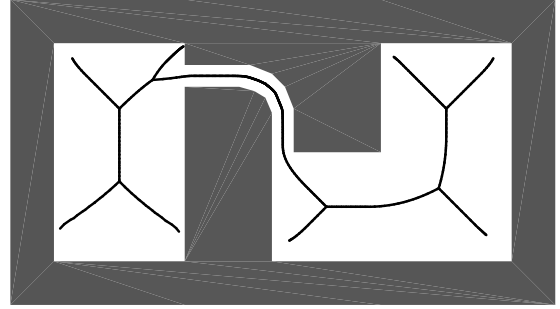


Figure 2: Medial axis approximation.

2D Medial Axis Approximation Let F be a set of closed curves in \mathcal{R}^2 that describe the boundary of the obstacles in the workspace, and let S be a set of points sampled along these curves (The algorithm presupposes sufficient sampling of the curves. The generation of S is discussed below).

The medial axis computation proceeds as follows. Let V_1 be the Voronoi diagram of S . D is then computed as the Delaunay triangulation of $S \cup V_1$. All edges in D for which both endpoints are in S are part of the *crust*, an approximation to F ([3]). For each edge in V_1 , there exists a corresponding (and perpendicular) edge in the dual Delaunay triangulation, D_1 . Define P as the set of edges that exist in both D_1 and D , and Q as the corresponding edges in V_1 . The approximation to the medial axis, R , is then given by $V_1 - Q$. An example is shown in Figure 2.

Sampling the Boundary of the Obstacles The algorithm for computing an approximation to the medial axis requires strict conditions on the ratio of the spacing between samples and the curvature of the sampled curve. We are most interested in polygonal workspaces, i.e. F is a set of simple polygons. Barring modifications to the above algorithm, vertices in a polygonal obstacle (having infinite curvature) will not be handled correctly. Since precisely determining the best approximation to the medial axis is not an absolute requirement in our application, there is a tradeoff between accuracy of the medial axis approximation and time spent computing the approximation. To this end, our approach to sampling F is to take a constant number of points along each edge, filtered by a simple moving average of length W . That is to say that for each closed polygon we create a sequence of samples by traversing the edges, then for each window of W consecutive samples, we create a new sample that is the average of those samples. Areas of F with high detail have more edges and will thus be sampled more heavily. The moving average effectively rounds corners, resulting in feasible ratios between the curvature and sample spacing.

The problem of dealing with the vertices of a polygonal

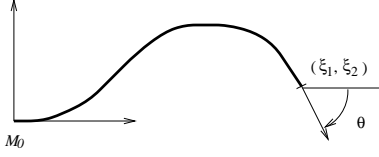


Figure 3: Manipulation constraints for a 2D curve.

representation can be handled more efficiently in the medial axis algorithm by keeping adjacency information with the samples when they are first created. In the algorithm outlined above, D is needed only to glean adjacency information for S — this step can be skipped if that information is maintained from the start. We choose not to take this approach because it places undue limitations on the representation of the obstacles. Triangle soups are the common denominator of 3D representations, and often the only way that two incompatible systems can exchange data. For this reason we felt that it was important to maintain the ability to use this type of data. We aim to extend our work to three dimensional cases and to simplify this step we avoided modifications that were too specific to the two dimensional case.

4.2 Sampling Along the Medial Axis

Once the preprocessing step of computing the medial is complete, we sample configurations along the medial axis. To describe our sampling in detail, let us first clarify how manipulation constraints are specified. For the 2D curves in this paper, we assume that we grasp the curves from two opposite edges and that we control the position and orientation of these edges. Figure 3 provides an illustration. In the local frame M_0 of the curve the first edge is at $(0, 0)$ tangent to the x-axis, while the other edge is at (ξ_1, ξ_2) with orientation θ .

First a vertex, v_0 , of the medial axis R is chosen at random. (Note that the approximation to the medial axis is a set of vertices and adjacency information for each, while the true medial axis is a set of continuous curves). A direction on the medial axis is chosen randomly and the distances between vertices are added up as we walk down the medial axis. At intersections we randomly choose which way to go, until the sum of the lengths of the segments, ending with v_n is greater than the stable length of the flexible surface. A rigid transformation T is calculated that aligns the reference point of the surface with v_0 such that the surface at the reference point is tangent to the segment between v_0 and v_1 (the first vertex encountered walking along the medial axis). We interpolate between v_{n-1} and v_n to get v' such that the distance walked is exactly the stable length of the surface. ξ_1 and ξ_2 of the limit condition are then calculated from the coordinates of v' in the frame defined by T and θ is calculated from the angle of $v_n - v_{n-1}$ (also in the frame defined by T). The surface is then minimized and we check for collisions with the obsta-

cles. This process may fail in several ways — we may run out of medial axis, the deformation may be unacceptable due to elasticity constraints, or the deformation may be in collision with the obstacles. In any of these cases we simply discard the configuration and try again, choosing a new v_0 .

4.3 Building the Roadmap of C_{free}

f-PRM now proceeds by employing an iterative procedure by which a roadmap is augmented until it is able to satisfy the query at hand. While we only investigate this “one-shot” application, the graph can be used to satisfy multiple requests and the graph can be augmented as needed for these additional queries.

The goal of the roadmap is to capture the connectivity of the freespace with as few nodes as possible. f-PRM as described in [17] samples across \mathcal{C} evenly. By sampling only in the neighborhood of the medial axis we avoid filling large empty spaces with configurations and place more configurations in tight areas.

The iterative step of the planner proceeds as follows:

1. Generate M nodes (along the medial axis or in the neighborhood of the medial axis).
2. For each node, find the K closest neighbors and attempt to connect to each of these K nodes by using a simple and fast local planner (A planner that does a linear interpolation on the limit conditions and the rigid transformation will work here).
3. Choose N nodes for enhancement. A probability distribution is created over all recently created nodes such that the weight of a configuration q_i is defined as

$$w(q_i) = \frac{|\{j : q_j \in N_k(q_i) \wedge \text{Component}(q_j) \neq \text{Component}(q_i)\}|}{|N_k(q_i)|}$$

and the probability that q_i is selected for enhancement is given by

$$P(q_i) = \frac{w(q_i)}{\sum_{q \in G} w(q)}$$

In the above equations $N_k(q_i)$ are the nodes in the neighborhood of q_i and G is the set of roadmap nodes. We compute the neighborhood of each node by considering the K closest other nodes for connections. Hence, we use a heuristic based on a node’s proximity to nodes in other components of the graph to choose N nodes to enhance. For each chosen node, we execute a random walk and attempt to make connections from the configuration at the end of the random walk.

4.4 Extensions

Several improvements of the above scheme are under consideration. Currently we sample uniformly all of the points of the medial axis. We could use the local characteristics of the medial axis to weight the distribution of points and guess more configurations in regions that may be more difficult, i.e. regions with intersections or narrower corridors. Currently, our sampling is based only on the medial axis. Some key deformations may not be guessed by the medial axis fitting. We could augment the medial axis fitting with a random generation of deformations as in the earlier version of f-PRM. Another possible improvement concerns the connection of neighboring configurations on the medial axis. We could use the medial axis as a guide for the generation of the local path reducing the chances for collision of the object with the workspace along the path. A similar approach could be used to strengthen the enhancement routine by heuristically choosing directions based on the closest portion of the medial axis.

5 Experimental Results

We show below experimental results planning for elastic curves in two dimensions. We use Bézier curves to represent our objects. The manipulation constraints are specified as in Figure 3. The mechanical model used corresponds to a homogeneous isotropic and linearly elastic material. The elastic energy is computed as in [17]. We perform the minimization of the local energy by using the conjugate gradient method [28]. The local planner interpolates linearly between the rigid transformations of the two configurations and then between the limit conditions.

Our code is written in C++ and executed on an SGI R10000. We show in this paper two environments (Figure 4 and Figure 5), both with narrow corridors that require the object to maneuver from the open area on the left to the open area on the right. The majority of the running time is spent in the minimization routines.

Snapshots along the path generated by our planner are given in Figures 4 and 5. We show the full surface in these pictures but the problem solved is essentially a 2D problem as explained earlier. The running times average 33 minutes for the case in Figure 4 and 43 minutes for the case in Figure 5. A uniformly random sampling planner, without the benefit of the medial axis, generally terminated without finding a path. For a simpler case, where the medial-axis planner spent approximately 3 minutes, the planner that guessed deformations randomly took over 4 hours.

We also compared our planner with the most efficient previous implementation of f-PRM available so far (described in [17]). The implementation in [17] makes use of several optimizations that are not possible in the context of our work

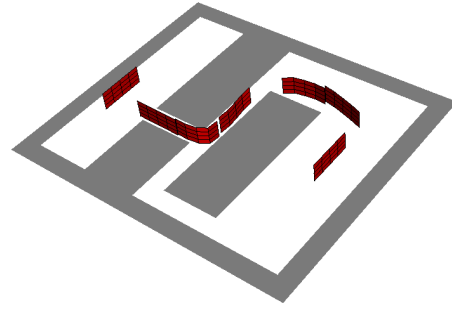


Figure 4: Snapshots for a path in Workspace 1.

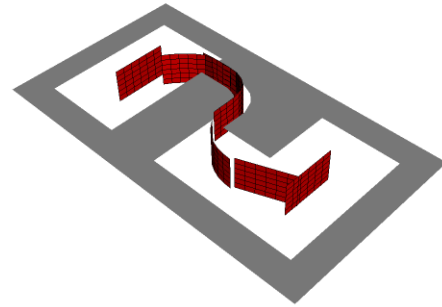


Figure 5: Snapshots for a path in Workspace 2.

here. For the workspace in 4 it averages 50 minutes, compared to 43 minutes for our implementation using the medial axis. For environments where more drastic deformations are required [17] rarely succeeds. For environments that are simply tight but do not necessitate a large deformation, f-PRM of [17] is faster as it can concentrate more on random walks and is more efficient at making connections. It achieves this efficiency by caching copies of minimized deformations that are reused at multiple positions. In our implementation position is not independent of deformation, and this optimization is not possible.

6 Discussion

We presented a new sampling scheme for f-PRM, a planner that computes paths for flexible objects. The 2D case was treated in this paper. We first obtained an approximation to the medial axis by adapting the algorithm in [3]. Then we created configurations of our objects on or near the medial axis by fitting the manipulation constraints to the medial axis. Our experimental results showed that the new sampling scheme is fairly successful in generating deformations that are critical for solving the path planning problem which results in significant savings in running times.

Our future plans involve the extension of the planner to three dimensions. We hope that with our algorithms, the approximation of the medial axis will be feasible for polygon soups and this will allow us to attack real-world problems. Furthermore, we are working on exploring the ideas of medial axis sampling for rigid parts in the context of the traditional PRM planner.

Acknowledgments

Lydia Kavraki is partially supported by NSF IRI-970228 and NSF CISE SA1728-21122N. Rusty Holleman is partially supported by an REU grant related to IRI-970228. The authors would like to thank Tim Danner, Ron Goldman, Florent Lamiroux, Jean-Claude Latombe, and Scott Owens for their comments.

References

- [1] J. M. Ahuactzin, E.-G. Talbi, P. Bessière, and E. Mazer. Using genetic algorithms for robot motion planning. In *10th Europ. Conf. Artif. Intelligence*, pages 671–675, London, England, 1992.
- [2] N. Amato, B. Bayazit, L. Dale, C. Jones, and D. Vallejo. Obprm: An obstacle-based prm for 3d workspaces. In P. K. Agarwal, L. E. Kavraki, and M. Mason, editors, *Robotics: The Algorithmic Perspective*. AK Peters, 1998.
- [3] N. Amenta, M. Bern, and M. Kamvyselis. A new voronoi-based reconstruction algorithm. In *SIGGRAPH*, Orlando, FL, 1998.
- [4] F. Arai, M. Tanimoto, T. Fukuda, K. Shimojima, and M. Negoro. Multimedia tele-surgery using high-speed optical fiber network. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 878–883, Minneapolis, MN, 1996.
- [5] H. Chang. Personal Communication.
- [6] H. Chang and T. Li. Assembly maintainability study with motion planning. In *Proc. IEEE Int. Conf. on Rob. and Autom.*, pages 1012–1019, 1995.
- [7] H. Choset. Incremental construction of the generalized voronoi diagram, the generalized voronoi graph, and the hierarchical generalized voronoi graph. In *Proc. of 1st CGC Workshop on Comp. Geom.*, Baltimore, MD, 1997.
- [8] G. Farin. *Curves and Surfaces for Computer-Aided Geometric Design*. Academic Press, San Diego, 1988.
- [9] P. W. Finn and L. E. Kavraki. Computational approaches to drug design. To appear in *Algorithmica*.
- [10] S. Gibson and B. Mirtich. A survey of deformable modeling in computer graphics. Technical Report TR-97-19, MERL, 1997.
- [11] K. Gupta and A. P. del Pobil. *Practical Motion Planning in Robotics*. John Wiley, West Sussex, England, 1998.
- [12] D. Halperin, L. E. Kavraki, and J.-C. Latombe. Robotics. In J. Goodman and J. O'Rourke, editors, *Discrete and Computational Geometry*, pages 755–778. CRC Press, NY, 1997.
- [13] C. Holleman and L. E. Kavraki. Sampling on the workspace medial axis for probabilistic roadmap planners. In preparation.
- [14] C. Holleman, L. E. Kavraki, and J. Warren. Planning paths for a flexible surface patch. In *Proc. IEEE Int. Conf. on Robotics and Automation*, 1998.
- [15] D. Hsu, L. E. Kavraki, R. Motwani, and S. Sorkin. On finding narrow passages with probabilistic roadmap planners. In P. Agarwal, L. E. Kavraki, and M. Mason, editors, *Robotics: The Algorithmic Perspective*, pages 141–154. AK Peters, Natick, MA, 1998.
- [16] L. Kavraki and J.-C. Latombe. Randomized preprocessing of configuration space for fast path planning. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 2138–2145, San Diego, CA, 1994.
- [17] L. E. Kavraki, F. Lamiroux, and C. Holleman. Towards planning for elastic objects. In P. Agarwal, L. E. Kavraki, and M. Mason, editors, *Robotics: The Algorithmic Perspective*, pages 313–326. AK Peters, Natick, MA, 1998.
- [18] L. E. Kavraki and J.-C. Latombe. Probabilistic roadmaps for robot path planning. In K. G. nd A. P. del Pobil, editor, *Practical Motion Planning in Robotics: Current Approaches and Future Challenges*, pages 33–53. John Wiley, West Sussex, England, 1998.
- [19] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. Overmars. Probabilistic roadmaps for fast path planning in high dimensional configuration spaces. *IEEE Tr. on Rob. and Autom.*, 12:566–580, 1996.
- [20] Y. Koga, K. Kondo, J. Kuffner, and J.-C. Latombe. Planning motions with intentions. *Computer Graphics (SIGGRAPH'94)*, pages 395–408, 1994.
- [21] K. Kondo. Motion planning with six degrees of freedom by multistrategic bidirectional heuristic free-space enumeration. *IEEE Tr. on Robotics and Automation*, 7(3):267–277, 1991.
- [22] F. Lamiroux and L. E. Kavraki. Path planning for elastic plates under manipulation constraints. In *Int. Conf. on Robotics and Automation*, Detroit, MI, 1999.
- [23] J.-C. Latombe. *Robot Motion Planning*. Kluwer, Boston, MA, 1991.
- [24] S. M. LaValle, J. H. Yakey, and L. E. Kavraki. A probabilistic roadmap approach for systems with closed kinematic chains. In *Proc. IEEE Int. Conf. on Rob. and Autom.*, 1999.
- [25] H. Nakagaki and K. Kitagaki. Study of deformation tasks of a flexible wire. In *Proc. IEEE Int. Conf. on Rob. and Autom.*, Albuquerque, NM, 1997.
- [26] W. Ngugen and J. Mills. Multi-robot control for flexible fixtureless assembly of flexible sheet metal auto body parts. In *Proc. IEEE Int. Conf. on Rob. and Autom.*, pages 2340–2345, Minneapolis, MN, 1996.
- [27] M. Overmars and P. Svestka. A probabilistic learning approach to motion planning. pages 19–37. A K Peters, Ltd., 1995.
- [28] D. A. Pierre. *Optimization Theory with Applications*. Dover, New York, 1986.
- [29] A. Singh, J.-C. Latombe, and D. Brutlag. Path planning for molecular docking. Personal Communication.
- [30] D. Sun, X. Shi, and Y. Liu. Modeling and cooperation of two-arm robotic system manipulating a deformable object. In *Proc. IEEE Int. Conf. on Rob. and Autom.*, pages 2346–2351, Albuquerque, NM, 1996.
- [31] T. van Walsum and M. Viergever. Deformable b-splines for catheter simulation. Technical report, Image Science Institute, Utrecht University, 1998.
- [32] G. Wempner. *Mechanics of Solids with applications to thin bodies*. McGraw-Hill, NY, 1991.
- [33] S. A. Wilmarth, N. M. Amato, and P. F. Stiller. Mapprm: A probabilistic roadmap planner with sampling on the medial axis of the freespace. In *Proc. IEEE Int. Conf. on Rob. and Autom.*, Detroit, MI, 1999.