Path Planning for Elastic Plates Under Manipulation Constraints

Florent Lamiraux   Lydia E. Kavraki

Department of Computer Science
Rice University, Houston, TX, USA
{lamiraux,kavraki}@cs.rice.edu

Abstract

This paper addresses the problem of path planning for a thin elastic metal plate under fairly general manipulation constraints. The underlying geometric model for the plate is provided by a Bézier representation. The geometric model is augmented by a realistic mechanical model. We assume that the plate is manipulated in accordance with a set of user-defined grasping constraints that specify the position and orientation of two opposite edges. Our mechanical model permits the computation of the shape of the plate with respect to the grasping constraints by minimizing the energy function of the deformation of the plate. Paths are computed by a planner that is based on the principle of probabilistic roadmaps. The planner builds a roadmap in the configuration space. The nodes of the roadmap are equilibrium configurations of the plate under the grasping constraints, while its edges correspond to quasi-static equilibrium paths. Paths are found by searching the roadmap. Several experimental results illustrate our approach.

1 Introduction

The Problem The problem of planning a collision-free path for a rigid robot has been extensively studied over the last decade [6, 10]. Recent work addressed the case of planning for a flexible robot/part [7, 9]. The paper extends previous work on planning for flexible parts by considering the case of an elastic metal plate that is manipulated by constraining, through grasping, the position and orientation of two opposite edges. The grasping constraints treated in this paper have not been considered in previous work. The paper illustrates many of the issues and difficulties arising when planning for flexible parts.

Motivation and Related Work Recent work on the path planning problem has produced several practical planners for robots that consist of rigid parts [1, 2, 5, 8]. These methods routinely take into account geometric constraints such as joint limits and obstacles, but also constraints arising from kinematics such as nonholonomic velocity constraints [11], or constraints over the radius of curvature of a car-like system [12].

With the exception of the areas of dynamics and control that have guided the design of modern robots, there are few cases where physical constraints and planning have been tightly coupled. The issue of flexibility has been primarily investigated by building and studying flexible robots. Those robots can perform tasks such as hammering a peg into a hole. But research in this field deals mainly with the control of these robots and not motion planning. Examples include the work in [16] which considers, from a control point of view, the motion and deformation of a flexible object grasped by two robot arms, and the work in [13] where the problem of inserting one end of a wire into a hole while holding the other end is solved.

Today several applications require the treatment of flexible parts. Take as an example virtual prototyping applications where path planners are now used to compute removal paths of parts from assemblies given only the CAD model for the assemblies [4]. The existence of such paths ensures that the part can be repaired (or replaced if needed) in the final assembled product. Current work treats the part as a rigid object. However, designers often use flexible parts to produce compact assemblies. For example, in the automotive industry, several assemblies have parts that are elastic metal sheets [14]. These are the kind of applications that are primarily targeted by our work. Other applications may include computer graphics animation, medical surgery with flexible tools, and computer-assisted pharmaceutical drug design.

Although planning with deformable parts has not been addressed in the robotics literature, there is a large amount of work concerning deformable objects in mechanics [18] where elasticity is a well understood issue. Additionally, graphics applications use deformable models [17]. Geometric representations for deformable parts can be found in geometric modeling [3]. In subsequent sections, we discuss the models and representations that we borrow from these domains.

Our Approach A first step in planning for flexible objects and elastic plates was presented in [7, 9]. In that work, we have developed a planner that computes paths for a thin elastic metal plate that can only bend. In this paper we treat more general manipulation constraints: we now also specify the tangents at opposite sides of the plate. We modify our planner to deal with the challenge of planning under these constraints.

According to elasticity theory, grasping via two opposite edges defines limit conditions for the deformation of the surface. Any equilibrium deformation minimizes an energy function over the set of all deformations fitting the limit conditions. However, this set is infinite dimensional and we have to approximate it. Several representations are available for approximating continuous functions. We use Bézier curves [3] because the limit conditions of grasping are easy to translate into constraints on their control points. In our research we have also considered spline models without significant differences in our results. An exhaustive and accurate mechanical study of a plate would require many control points (hundreds) to represent the stress and strain fields of the deformation. However, our goal is only to approximate the shape of the surface and for that far less control points are sufficient.

The configuration space of our system is the Cartesian product of the space of deformations fitting the grasping conditions and the space of rigid body transformations. We assume that the velocity is small enough to neglect the dynamic effects of deformations and we consider that any motion is composed of equilibrium deformations.

Our planner, f-PRM, is an extension of PRM [8]. It generates a large number of random configurations and connects them using a local planner. When the roadmap thus obtained is dense enough, f-PRM tries to connect the initial and goal configurations to the same connected component of the readmap and builds a path by searching for a sequence of edges connecting both configurations. To get through narrow passages of the configuration space, an enhancement step is added. f-PRM incorporates new mechanisms for selecting random configurations and computing paths.

The paper is organized as follows. Section 2 defines the geometric and mechanical model for the deformations of the plate. Section 3 describes implementation issues about f-PRM. Simulation results are provided in Section 4 and a short discussion concludes the paper in Section 5.

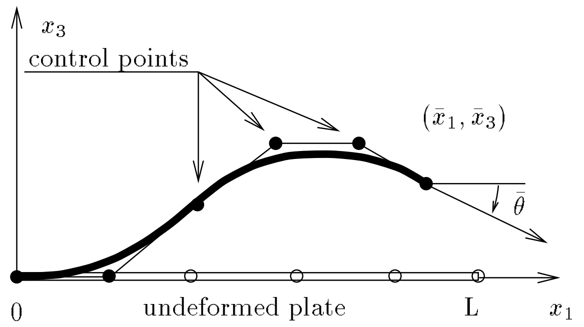

Figure 1: Grasping by two opposite edges

2 Deformation of an Elastic Plate

Before describing the details of our planning approach, we present the geometric and mechanical model used for the elastic plate. We expect that the plate is manipulated by one or two robot arms. In this paper we do not consider the full version of the manipulation problem: we only specify conditions, which we call *grasping constraints* or *limit conditions* that reflect the effect of manipulation on our object. These constraints account for deformations of the plate as manipulation by robotic arms would do. In this respect, our work is in similar spirit with work done in assembly sequencing [19].

2.1 Grasping Constraints

An elastic metal plate can be deformed either by applying forces on it or by constraining the position of a subset of its points (position of two edges for instance). These constraints are defined by grasping and give rise to limit conditions according to which the global shape of the plate has to be computed.

Let us suppose that we control the deformation of the plate by constraining the position and orientation of two opposite edges relatively to one another. Let (x_1, x_2, x_3) be an orthonormal coordinate system in $\mathbf{R}^3$. The first edge of the plate is set on the x_2-axis and the plate is tangent to the plane $x_3 = 0$ at this extremity. The second edge, also parallel to the x_2-axis passes by the point $(\bar{x}_1, 0, \bar{x}_3)$. $\bar{\theta}$ represents the constrained direction of the tangent to the plate at this extremity (see Figure 1). Our parameterization clearly does not limit the set of constraints that can be represented.

Under these hypotheses, the shape of the deformed plate is a cylinder and thus, can be defined by a curve. The main advantage of the type of grasping considered in this paper is that it allows a big range of deformations keeping a simple representation for them.

2.2 Geometric Model of the Plate

Given grasping conditions $(\bar{x}_1, \bar{x}_3, \bar{\theta})$, the space of deformations fitting the corresponding limit conditions is isomorphic to the space of continuous curves connecting two points, with given tangents at the extremities. We need to approximate this infinite-dimensional space by a finite-dimensional one. A good way to do that is to represent deformations by Bézier curves [3]. A Bézier curve of order n is defined by $n+1$ control points $(P_0, ..., P_n)$ according to the following formula:

$$B(u) = \sum_{i=0}^{n} B_n^i(u) P_i,$$

where $u \in [0,1]$ and $B_n^i(u) = \left(\begin{array}{c} n \\ i \end{array}\right) u^i (1-u)^{n-i}$ are the Bernstein polynomials. The main advantage of Bézier curves for our purposes is that the manipulation constraints can be easily translated into constraints on the control points. Indeed, $B(0) = P_0$, $B(1) = P_n$ and the tangents to the curve at its extremities are given by $B'(0) = n(P_1 - P_0)$ and $B'(1) = n(P_n - P_{n-1})$, where B' is the derivative of B.

2.3 Geometric Analysis of Deformations

In its load free configuration, the plate is a rectangle of length L, width W, thickness $h << L$, tangent to the plane $x_3 = 0$ (Figure 1) and the positions of the control points are: $P_i^0 = (iL/n, 0, 0)$. It can be easily verified that the corresponding Bézier curve is $B^0(u) = (Lu, 0, 0)$, the tangent vector $\mathbf{t}^0(u) = B^{0\prime}(u)$ to this curve is thus constant and equal to $(L, 0, 0)$. As explained in Figure 2, the deformation of the plate at each point can be defined by the extension coefficient ε and the curvature χ of the Bézier curve. The expressions of these coefficients are the following:

$$\varepsilon(u) = \frac{1}{L}(||\mathbf{t}(u)|| - L) \quad (1)$$

$$\chi(u) = \frac{||\mathbf{t}(u) \times \mathbf{t}'(u)||}{||\mathbf{t}(u)||^3} \quad (2)$$

where $\mathbf{t}'(u) = B''(u)$ is the derivative of $\mathbf{t}(u)$ in the deformed configuration and $\times$ is the cross product. Let us point out that the coefficient ε represents the extension rate of a small piece of matter in the x_1 direction (Figure 2). From now on, we let $\mathbf{e}(u) = (\varepsilon(u), \chi(u))$ be the *strain vector*.

In the general case, the local deformation of a surface is represented by 6 coefficients (3 for plane strain

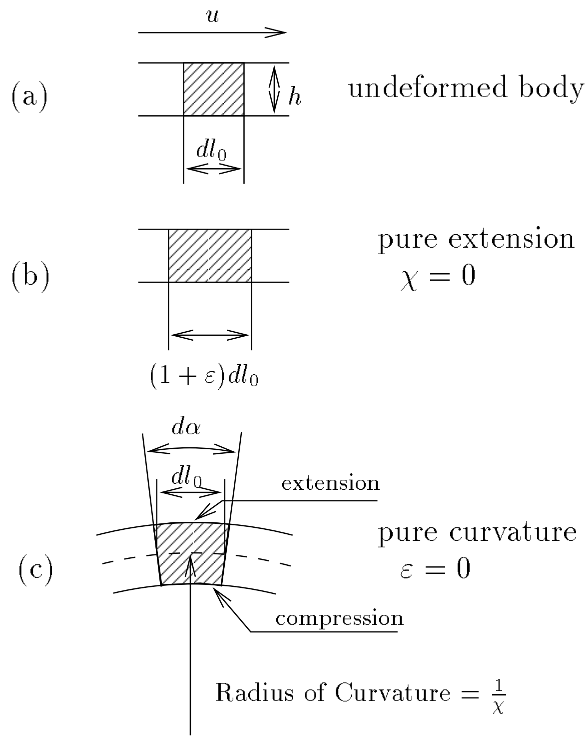

Figure 2: Extension and curvature coefficients

and 3 for curvature). For a discussion of that case see [9]. Our hypothesis of one-dimensional deformation however, enables us to simplify the representation.

2.4 Mechanical Analysis of Deformations

Under the hypothesis of elasticity of the material, and given limit conditions, an equilibrium deformation of the plate locally minimizes a function called *elastic energy*. This function is obtained by integrating the *elastic energy per unit of surface* $\psi(\mathbf{e}(\mathbf{x}))$ over the plate, where $\mathbf{e}(\mathbf{x})$ is the previously defined strain vector at some point $\mathbf{x}$ of the plate. Even if it is thin, a plate is a three dimensional object. The hypothesis of thinness however, enables us to approximate the elastic energy of a deformation w.r.t. the shape (extension and curvature) of the medium surface. We do not give in this paper the definition of elasticity for three dimensional objects, nor do we present the computations leading to the elastic energy function that we will use. These can be found in [18].

The elastic energy per unit of surface w.r.t. the local deformation is given by:

$$\psi(\mathbf{e}) = \psi^{\chi}(\chi) + \psi^{\varepsilon}(\varepsilon), \quad (3)$$

where

$$\psi^{\chi}(\chi) = \frac{Eh^3}{24} \frac{\chi^2}{1-\nu^2}, \quad (4)$$

$$\psi^\varepsilon(\varepsilon) = \frac{Eh}{2}\frac{\varepsilon^2}{1-\nu^2}. \qquad (5)$$

The Young modulus E and the Poisson ratio ν are coefficients characterizing the elasticity of the material[1].

2.5 Equilibrium Deformations

Let us call *isometric* a deformation with null plane extension $(\forall u \in [0,1], ||\mathbf{t}(u)|| = L)$. If the plate is thin with respect to the radius of curvature, that is $h\chi << 1$, equations (3), (4) and (5) state that the curvature energy is far smaller than the plane extension energy. Thus if for given limit conditions $(\bar{x}_1, \bar{x}_3, \bar{\theta})$, the space of isometric deformations is not empty, any equilibrium deformation will be close to this space. On the other hand, the elasticity limit of most metals is very small ($< 10^{-3}$), and thus any deformation has to be isometric, up to numerical approximations. According to this remark, the constraints related to the grasping $(\bar{x}_1, \bar{x}_3, \bar{\theta})$ become:

$$\begin{array}{ll} B(0) = (0,0,0) & B(1) = (\bar{x}_1, 0, \bar{x}_3) \\ \mathbf{t}(0) = (L,0,0) & \mathbf{t}(1) = (L\cos(\bar{\theta}), 0, L\sin(\bar{\theta})), \end{array}$$

that is $P_0 = (0,0,0)$, $P_1 = (L/n, 0, 0)$, $P_{n-1} = (\bar{x}_1 - L/n\cos(\bar{\theta}), 0, \bar{x}_3 - L/n\sin(\bar{\theta}))$ and $P_n = (\bar{x}_1, 0, \bar{x}_3)$.

An equilibrium deformation is then obtained by minimizing the elastic energy over the $n-3$ remaining control points. To perform this task, we use the conjugate gradient method. The gradient of the elastic energy is obtained by integrating over the plate the partial derivative of (3), with respect to the free control points. Integrals are computed using Simpson's formula [15].

3 Path Planning

Below we give a high level overview of the preprocessing and query processing steps of the planner and discuss in details only the differences from previous work [7, 9]. A crucial element in our planner is that it uses deformations which are equilibrium deformations given the grasping conditions and which are computed as explained in Section 2.5. For the computation of equilibrium configurations and equilibrium paths, we decouple deformation and rigid-body transformation.

3.1 Preprocessing

The preprocessing step builds a roadmap according to the following algorithm:

[1] For instance, for aluminum, $E = 74$ MPa and $\nu = 0.34$.

1. Generate a set of values for the grasping parameters at random. Then compute an equilibrium deformation corresponding to the limit conditions induced by the chosen values and test if it fits the elasticity limits of the material (see below). If it does, generate N random rigid body transformations (with the same deformation) and test each corresponding configuration for collision with the obstacles.

2. Update the graph $R = (V, E)$ in which E consists of every pair of configurations that can be connected with a local planner. Consider all existing nodes for connections.

3. Identify "difficult" areas and refine sampling in these areas by generating M more nodes (see discussion below). Connect the new nodes to R and update R.

We now give some details about specific implementation choices for the preprocessing step.

Local Planner The local planner should have good chances of success when the two nodes are close together. We represent a rigid-body transformation by a translation vector and a rotation vector. A local path is computed by linear interpolations in the parameter space $\mathbf{R}^6$ of the rigid-body transformation and in the parameter space of the grasping conditions $(\bar{x}_1, \bar{x}_3, \bar{\theta})$ independently. The rigid body transformation is applied first. If a collision is detected during the rigid-body transformation which is computationally faster, the planner does not need to build the local path in the deformation space. As seen previously, the grasping condition determines the position of some control points of the Bézier curve: P_0, P_1, P_{n-1} and P_n. The path followed by the $n-3$ other control points is a set of configurations minimizing the elastic energy for the given grasping conditions. To approximate the corresponding deformation path, we sample the grasping condition path and we compute minimization at the sample points. However, due to the possible existence of different local minima of the elastic energy for a given grasping condition, the path in the deformation space can be discontinuous, even if the grasping conditions vary continuously. To avoid this, and to make minimization faster, we take as initial deformation of the minimization procedure the equilibrium deformation of the former sampled grasping condition.

Elasticity Limit Our planner should ensure that all used deformations are within elasticity limits of the material of the plate so that the plate is not damaged during manipulation. The linear elastic model is usually valid only for small deformations: $\varepsilon < \varepsilon_{max}$, where ε_{max} is a constant depending on the material. Out of the elasticity domain, the material has an elasto-plastic behavior. That is, the unconstrained shape of the object is different before and after the deformation and the material is irreversibly affected. To avoid this, we have to test each randomly generated deformation to determine whether it lies in the elasticity domain. As seen previously, the deformation is defined by two coefficients:

- **Plane Strain Limit** Given limit conditions, if the corresponding isometric deformation space is empty, the equilibrium deformation found by the minimization procedure has a big plane strain and it is rejected. Otherwise, the minimum will automatically have small plane strain since the term related to plane extension (5) is dominant in the energy formula (3).

- **Curvature Limit** As seen from Figure 2, a curvature χ of the plate gives rise to an extension (on the upper surface): the straight line of undeformed length $dl = Rd\alpha$ is extended to $(R + h/2)d\alpha$, where $R = 1/\chi$ is the radius of curvature. According to its definition (1), the extension coefficient on the upper surface is

$$\bar{\varepsilon} = \frac{(R+\frac{h}{2})^2 d\alpha^2 - R^2 d\alpha^2}{R^2 d\alpha^2} \approx \frac{h}{2R} = \frac{1}{2}\chi h.$$

april Thus, an isometric deformation is in the elastic domain if its curvature is bounded by $2\varepsilon_{max}/h$.

Other Issues For a discussion on the enhancement step of preprocessing see [9]. The RAPID collision-checking library is used: our surface is triangulated and tested for collisions as explained in [9].

3.2 Query Processing

Given a roadmap R an initial and goal configurations s and g, f-PRM uses the local planner to connect s and g to nodes s' and g' of the same connected component of R. If successful, the component is searched for a sequence of edges connecting s' and g'.

4 Experimental Results

We run f-PRM on the example displayed on Figure 3 on an SGI R10000. Our code is written in C++. Our first environment comes from a car assembly [4]. We treat the description of preprocessing in Section 3.1 as a basic step that is repeated a number of times. The parameters for the basic step of f-PRM are $N = 200$,

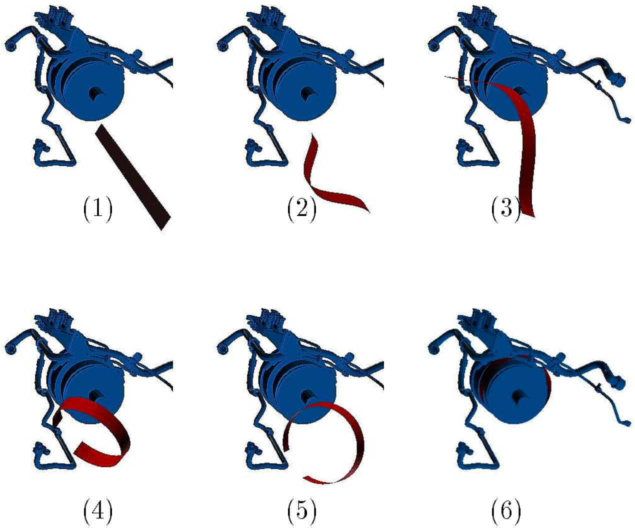

Figure 3: Positioning a metallic belt in a pipe assembly of a car.

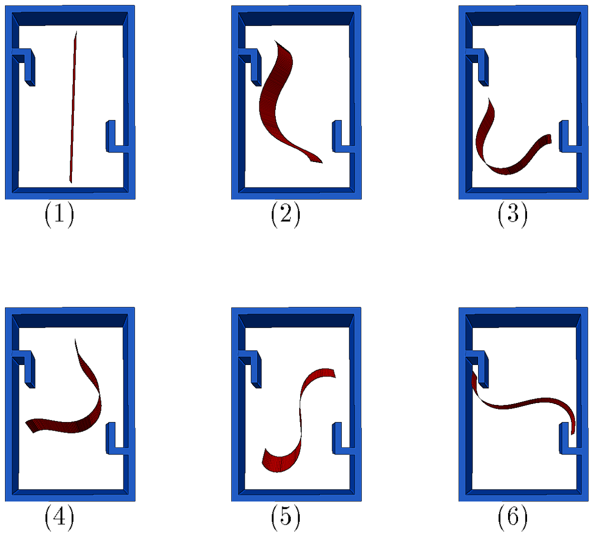

Figure 4: Motion of an elastic plate in a tight environment.

$M = 100$, and $K = 20$. K is the number of neighbors considered for connection. During enhancement the random walk consists of a maximum of 10 reflections, each of which can be 100 steps long (see also [9]). The surface is represented by an 8 control point Bézier curve. This problem, which is a 9 DOF problem, was solved in 7h 7min with 25200 nodes. A computationally expensive operation of the planner is the construction of local paths. The reason is two-fold. Firstly, for far away configurations, interpolating the grasping conditions can lead to very large deformations that make the local planner fail. For this reason, our planner works well for neighboring configurations but paths tend to be short and many of them need to be computed. Secondly, intermediate configurations along the path need to be minimized and this is an expensive operation.

Another challenging 9 DOF problem is shown in Figure 4. In this example the surface is manipulated inside a box. Note that our snapshots do not show the upper and lower lid of the box for clarity purposes. The planner was run with the same parameters. The surface consists of 10 control points. It took a roadmap of 26000 nodes to solve the problem.

Remark: To determine precisely the stress and strain in the material, we would need appropriate models with high dimensions (finite elements with hundreds of points). However, simulations carried out with increasing numbers of control points for the same grasping conditions showed that 8 to 10 control points give a fairly accurate shape for the surface. In our simulations we considered curves with up to 50 control points.

5 Discussion

This paper described a planner for an elastic plate that is manipulated by constraining the position and orientation of two opposite edges. A realistic mechanical model simulates the behavior of the object under manipulation. The underlying technique for solving a planning problem consists in the construction of a random roadmap in the configuration space. Our experiments show that our new planner f-PRM can deal with highly constrained realistic models.

Our work raises several interesting issues. These include finding better strategies for the local planner in order to improve its efficiency, more efficient energy minimization procedures, allowing more complex manipulation constraints, developing geometric representations that support them, and smoothing the resulting paths to avoid unnecessary deformations.

Acknowledgments Lydia Kavraki and Florent Lamiraux are partially supported by NSF CAREER Award IRI-970228. The authors would like to thank Rusty Holleman for his invaluable help in the implementation of the planner, and Ron Goldman, Leo Guibas, Jean-Claude Latombe, and Joe Warren 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. Artific. Intelligence*, pages 671–675, London, England, 1992.

[2] J. Barraquand and J. Latombe. Robot motion planning: A distributed representation approach. *Int. J. of Robotics Research*, 10:628–649, 1991.

[3] R. H. Bartels, J. C. Beatty, and B. A. Barsky. *An introduction to Splines for use in Computer Graphics and Geometric Modeling.* Morgan Kaufmann Publishers, Los Altos, California, 1987.

[4] H. Chang and T. Li. Assembly maintainability study with motion planning. In *Proc. IEEE Int. Conf. on Rob. and Autom.*, pages 1012–1019, 1995.

[5] P. Chen and Y. Hwang. Sandros: A motion planner with performance proportional to task difficulty. In *Proc. of IEEE Int. Conf. Robotics and Automation*, pages 2346–2353, Nice, France, 1992.

[6] D. Halperin, L. 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.

[7] C. Holleman, L. Kavraki, and J. Warren. Planning paths for a flexible surface patch. In *Proc. IEEE Int. Conf. Robotics and Automation*, 1998.

[8] L. Kavraki, P. Švestka, J. 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.

[9] L. E. Kavraki, F. Lamiraux, and C. Holleman. Towards planning for elastic objects. In *Workshop on Algorithmic Foundations of Robotics*. 1998. To appear.

[10] J. Latombe. *Robot Motion Planning.* Kluwer, Boston, MA, 1991.

[11] J. Laumond. Feasible trajectories for mobile robots with kinematic and environment constraints. In *Proc. Intelligent Autonomous Systems*, pages 346–354, 1987.

[12] J. Laumond, P. Jacobs, M. Taix, and R. Murray. A motion planner for nonholonomic mobile robots. *IEEE Tr. on Rob. and Autom.*, 10:577–593, 1994.

[13] 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.

[14] 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.

[15] W. H. Press, S. A. Teutolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C.* Cambridge University Press, 1995.

[16] 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.

[17] D. Terzopoulos and A. Witkin. Physically based models with rigid and deformable components. *IEEE Computer Graphics and Applications*, pages 41–51, November 1988.

[18] G. Wempner. *Mechanics of Solids with applications to thin bodies.* McGraw-Hill, NY, 1991.

[19] R. Wilson and J. Latombe. Geometric reasoning about mechanical assembly. *Artificial Intelligence*, 71:371–396, 1995.