Stratified Motion Planning on Nonsmooth Domains with Robotic Applications

Yejun Wei† and Bill Goodwine‡

Abstract— **This paper presents an extension of stratified motion planning results to the case where the base manifold upon which the motion planning occurs is not smooth. Robotic applications of this work include motion planning for legged robots over known, nonsmooth terrain and manipulation of nonsmooth objects with multiple robotic manipulators.**

Keywords— **Stratified motion planning, legged locomotion, robotic manipulation.**

I. Introduction

This paper presents an extension of a control strategy which considers motion planning for robotic systems which are characterized by switching dynamics. Previous work by the authors developed a "stratified motion planning" algorithm which provided an analytical means for motion planning for systems with switching dynamics [1], [2], [3], [4]. One application of this previous work is legged locomotion over smooth terrain where the switching dynamics occur when various feet make and break contact with the ground. Another application is robotic manipulation of smooth objects where the switching dynamics occur when the robotic fingers make and break contact with the manipulated object. This paper presents an extension of this algorithm to handle the case where the terrain or object is nonsmooth, as is schematically illustrated in Figure 1. The analytical nature of the algorithm guarantees that as long as the robot is controllable, a solution will be determined.

The main difficulty with stratified systems in general, is to determine a method to analytically incorporate, either in an analysis tool or control synthesis algorithm, the discontinuous nature of the equations of motion for the system. Incorporating the discontinuities of the equations of motion into a general motion planning algorithm is difficult because almost all general motion planning methods assume that the equations of motion are smooth.

Prior research efforts concerning legged locomotion have typically focused either on a particular morphology such as in [5], [6], [7], [8] or a particular locomotion assumption such as in [8], [9]. Some more general results exist, such as in [9], [10], [11]. In contrast to robotic legged locomotion, many results in robotic grasping and manipulation are formulated in a manner that is independent of the morphology of the gripper, such as in [12]. Many efforts considered the analysis of grasp stability and force closure [13], [14], [15], motion planning assuming continuous contact [16], [17], [18] and haptic interfaces [19], [20], [21].

† Email: ywei@nd.edu

‡ Email: bill@controls.ame.nd.edu. Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN 46556

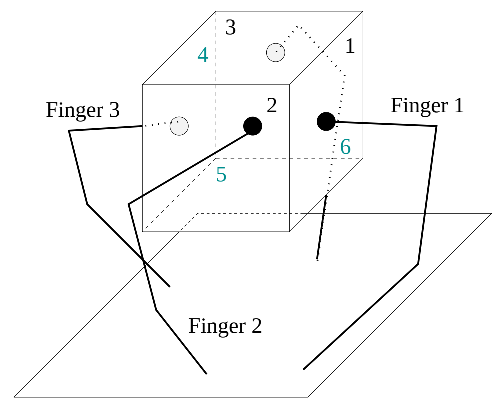

Fig. 1. Nonsmooth object manipulation.

Finger gaiting has been implemented in certain instances [22], [23], [24] and also partially considered theoretically [3], [25], [26], [27]. In contrast with the current work, none of these methods directly use the inherent geometry of stratified configuration spaces to formulate results which span many different morphologies and assumptions.

II. Smooth Stratified Systems

This section outlines the stratified motion planning method for smooth systems, which forms the basis for the extension to nonsmooth stratified case. Many details are necessarily omitted, and the interested reader is referred to [1], [2], [3], [4] for a complete exposition.

A simple example will provide an intuitive understanding of the geometry of stratified systems. Consider two robotic fingers intermittently engaging a smooth object. The set of configurations corresponding to one of the robots engaging the object is a smooth codimension one submanifold (one less dimension than the entire configuration space) of the configuration space. The same is true when the other robot engages the object. Similarly, when both robots engage the object, the system is on a smooth codimension two submanifold of the configuration space formed by the intersection of the single contact submanifolds. Each submanifold is referred to as a *stratum*.

More generally, let S_0 denote the system's entire configuration manifold and $S_i \subset S_0$ denote the smooth codimension one submanifold of S_0 that corresponds to all configurations where only the ith robot engages the object. Denote, the intersection of S_i and S_j, by $S_{ij} = S_i \cap S_j$. Stratum S_{ij} physically corresponds to the set of states where both the ith and jth robots engage the object. Further intersections can be be recursively defined: $S_{ijk} =$

$S_i \cap S_j \cap S_k = S_i \cap S_{jk}$, *etc.* The lowest-dimensional stratum will be called the *bottom stratum.* In the case of the grasping problem in Figure 1, the bottom stratum is when all four fingers are in contact with the object. All our previous efforts have assumed that all the strata are smooth manifolds, which, as illustrated subsequently, will not be true in the case of nonsmooth objects or terrain.

Definition 1: Let S_0 be a manifold, and n functions $\Phi_i : S_0 \mapsto \mathbb{R}$, $i = 1, \ldots, n$ be such that the level sets $S_i = \Phi_i^{-1}(0) \subset S_0$ are regular submanifolds of S_0, for each i, and the intersection of any number of level sets, $S_{i_1 i_2 \cdots i_m} = \Phi_{i_1}^{-1}(0) \cap \Phi_{i_2}^{-1}(0) \cap \cdots \cap \Phi_{i_m}^{-1}(0)$, $m \leq n$, is also a regular submanifold of S_0. Then S_0 and the functions Φ_i define a *stratified configuration space.*

We assume that the equations of motion on each stratum are of the form

$$\dot{x} = g_{I,1}(x)u_{I,1} + \cdots + g_{I,n_I}(x)u_{I,n_I}, \tag{1}$$

where the first subscript, $I = i_1 i_2 \ldots i_m$, indexes the stratum upon which the equations are defined. We assume that the robot is able to control switches among strata. The motion planning algorithm for smooth stratified systems is based upon the method presented in [28]. The approach is to construct an *extended system* on the bottom stratum in which the original set of equations of motion is appended with Lie bracket vector fields associated with which are fictitious inputs. For the extended system, motion planning along a nominal trajectory is trivial since it is constructed so that the span of all the vector fields is full rank. Formal algebraic computations utilizing indeterminates, b_i, formal exponential expansions of the form

$$e^{b_i} = 1 + b_i + \frac{b_i^2}{2!} + \cdots ,$$

which can be related to solutions of the original equations 1 and approximations to Lie brackets of the form

$$\phi_\epsilon^{-g_2} \circ \phi_\epsilon^{-g_1} \circ \phi_\epsilon^{g_2} \circ \phi_\epsilon^{g_1}(x) = \phi_{\epsilon^2}^{[g_1,g_2]}(x) + \mathcal{O}(\epsilon^3), \tag{2}$$

where $\phi_\epsilon^{g_1}(x_0)$ represents the solution of the differential equation $\dot{x} = g_1(x)$ at time ϵ starting from x_0, provide the mechanism to determine the real control inputs. The exact form of the nominal trajectory is not critical for the performance of the algorithm, it is only important that it be a path from the starting configuration to the desired ending configuration. Typically, a straight line is utilized. In fact, the nominal trajectory is not exactly followed since approximations of the form of Equation 2 are used to move in a Lie bracket direction.

For stratified system, if it is the case that the Lie bracket between the vector fields which switch the system among strata and any other vector fields is zero, then it is straight forward to show that vector fields defined on *multiple strata* can be considered simultaneously in the motion planning algorithm on the bottom stratum (a detailed explanation can be found in [4]). This is useful because it effectively increases the control authority of the robot by being able to utilize vector fields defined on multiple strata. For holonomic manipulators or robotic legs, this Lie bracket will always be zero (see [2] for a complete discussion). A description of the algorithm extended to handle the nonsmooth case, is presented in Section III.

III. Nonsmooth Stratified Systems

In this section we consider the geometry of a nonsmooth stratified system and the manner in which the motion planning algorithm outlined in Section II can be extended to the nonsmooth case. Consider the case of the four fingers manipulating the cube illustrated in Figure 1. If each finger has, say, three revolute joints, then the overall configuration space for the system is $S_0 = \mathrm{SE}(3) \times S^{3\times 4}$, where SE(3) describes the configuration of the cube and $S^{3\times 4}$ represents the configuration of the joints. As described in Section II, if the object were smooth, then the set of all configurations where one finger contacts the object defines a *smooth* codimension one submanifold of S_0. However, since the object is not smooth, the set of configurations where the finger contacts the cube will be the *union* of six smooth manifolds *with boundary.* The six manifolds correspond to each face of the cube, and their boundaries correspond to the edges of the cube. (See [29] for a complete development of manifolds with boundary).

In general, we use stratum $S_{I_m J_n K_p L_q}$, where $1 \leq I < J < K < L \leq 4$ and m, n, p and q are different integers between 1 and 6, to represent the configuration when four fingers I, J, K and L are in contact with surfaces m, n, p and q respectively. Similarly, stratum $S_{I_m J_n K_p}$ represents the configuration when three of the four fingers, I, J and K, are in contact with the surfaces m, n and p of the object respectively. The level of the stratum is the codimension of the strata. The *bottom stratum* is the union of all the lowest dimensional, bottom strata. Thus, the bottom stratum for the structure our system is on the 4th level and is the union of all the strata with codimension 4 representing that all the four fingers are in contact with the cube. Similarly, all the strata representing three of the four fingers are in contact with the cube are in level 3. Part of the combinatorial structure of the stratified system is shown in Figure 2, where the nodes in the figure represents different strata, the edges connecting the nodes indicate that it is possible for the system to move from one stratum to another. Thus, if the nodes are connected by an edge, the system can move between the strata, if there is no edges between two nodes, the system cannot move between them directly. Whether or not an edge connects two nodes is, of course, problem dependent.

For the example stratified system of four fingers manipulating a cube object, the bottom stratum is the union of $P_6^4 = 360$ manifolds with boundary (this assumes that it is desirable to exclude the possibility of two fingers contacting a single face of the cube simultaneously). The combinatorial size of the stratified graph could affect the planning efficiency in that many strata could impact the problem; however, it would also be beneficial in that the increased number of vector fields that are available from multiple

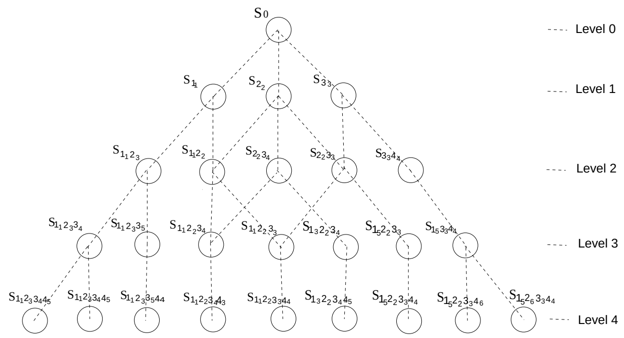

Fig. 2. The structure of the nonsmooth stratified system.

strata would make planning actually easier.

On the bottom level, Figure 2 shows that the system cannot move from one stratum to another since there are no edges between them. While this figure is not meant to be a precise representation of the four finger/cube manipulation problem (due to the fact that the bottom stratum is comprised of 360 strata), the lack of edges between the strata in the bottom stratum is physically reasonable for such a system because it would imply that a finger can switch from one of the bottom strata to another without disengaging the object. Clearly, however, the system actually can be moved from one stratum to another on the bottom level by moving up to the strata in the upper levels and then move down to the stratum in the bottom level.

For motion planning, since the bottom stratum is not simply a smooth manifold, the nominal trajectory will not necessarily be contained within a single stratum. Consider the case of the cube when all four fingers start and finish on different faces of the cube. Since the nominal trajectory is computed in the bottom stratum the nominal trajectory will need to switch among the various strata from which the bottom stratum is comprised.

The algorithm is as follows:

1. Check that the Lie bracket decoupling assumption holds and that the system is controllable (see [1], [30]).
2. Determine a nominal trajectory in the bottom stratum. Consistent with smooth stratified motion planning, gait stability considerations may necessitate that the overall trajectory be divided into multiple subtrajectories (see [4]). Essentially, for a legged robot, the size of the subtrajectory will dictate the step size. A subtrajectory that is too long will require large step sizes, which may destabilize the robot gait. In [4], explicit, but conservative, bounds on the size of the subtrajectory are computed which guarantee stability.
3. Construct the *extended stratified system* on the bottom strata. This is of the form

$$\begin{aligned}\dot{x} &= g_1(x)v_1 + \cdots g_m(x)v_m \\ &+ \underbrace{g_{m+1}v_{m+1} \cdots + g_n v_n}_{\text{from higher strata}} \\ &+ \underbrace{g_{n+1}v_{n+1} + \cdots + g_p v_p}_{\text{any Lie brackets}}, \end{aligned} \tag{3}$$

where the $\{g_1, \ldots, g_p\}$ span $T_x S_0$ and are the control vector fields from multiple strata, the inputs $v_1, \ldots, v_n$ are real, and the inputs $v_{n+1}, \ldots, v_p$ are *fictitious*. This equation may be different for each of the strata that comprise the bottom stratum.
4. Construct the *formal equation*, which is simply Equation 3 on each bottom stratum written in indeterminates,

$$\dot{S}(t) = S(t)(b_1 v_1 + \cdots + b_s v_s),$$

where the $S(t)$ are polynomial Lie series (see [28]).
5. Construct the Chen-Fleiss series, namely,

$$S(t) = e^{h_s(t)b_s} e^{h_{s-1}(t)b_{s-1}} \cdots e^{h_1(t)b_1},$$

differentiate it with respect to time and equate the coefficients of the b_i's in the resulting equation with the coefficients of the corresponding b_i's in the equation in the previous step, to construct ordinary differential equations for the *backward Philip Hall coordinates*, h_i.
6. Solve the differential equations from the previous step to determine the h_i's to determine how long the system should flow along each basis element, b_i, to reach the goal point. If the b_i represents a Lie bracket, then an approximation of the form of Equation 2 should be used.
7. If two sequential b_i's belong to different strata, then the decoupled vector field (checked in step 1) must be actuated to switch strata.

IV. Example

The algorithm was also verified via simulation on a very simple hexapod robot model as illustrated in Figure 3. The surface upon which the robot is locomoting is parameterized by

$$h(x, y) = \frac{\lfloor \mathrm{mod}(x, 2) \rfloor \lfloor \mathrm{mod}(y, 2) \rfloor}{4}$$

which produces the partial "checker-board" height pattern in Figure 3. The equations of motion for the hexapod are taken to be

$$\begin{aligned}\dot{x} &= \cos\theta\,(\alpha(h_1)u_1 + \beta(h_2)u_2) \\ \dot{y} &= \sin\theta\,(\alpha(h_1)u_1 + \beta(h_2)u_2) \\ \dot{\theta} &= l\alpha(h_1)u_1 - l\beta(h_2)u_2\end{aligned}$$

$$\begin{aligned}\dot{\phi}_1 &= u_1; & \dot{\phi}_2 &= u_2 \\ \dot{d}_1 &= u_3; & \dot{d}_2 &= u_4\end{aligned}$$

where (x, y, θ) represents the planar position of the center of the body of the robot, ϕ_1 is the front to back angular deflection of legs 1–4–5, ϕ_2 is the angular deflection of legs 2–3–6, l is the leg length and d_i is the distance of the legs off the ground. The functions α and β are defined by

$$\alpha(d_1) = \begin{cases} 1 & \text{if } d_1 = 0 \\ 0 & \text{if } d_1 > 0 \end{cases} \qquad \beta(d_2) = \begin{cases} 1 & \text{if } d_2 = 0 \\ 0 & \text{if } d_2 > 0 \end{cases}.$$

Figure 3 illustrates the motion of the hexapod as it traverses the terrain based upon the nominal trajectory

$$(x(s), y(s), \theta(s)) = (s, s, 2\pi s) \tag{4}$$

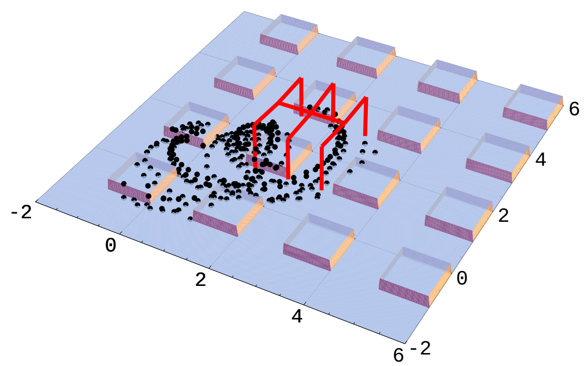

Fig. 3. Hexapod motion.

where $s \in (0,1)$ parameterizes the path it follows. In such a case, the robot walks diagonally across the floor while "spinning" one complete revolution as it completes one unit diagonally in the x and y directions. To maintain gate stability, the trajectory is divided into 30 identical subtrajectories.

To make the presentation a manageable length, we assume that the robot's equations of motion are the same regardless of which combination of feet are on the lower level or up on the square bumps; although, we emphasize that this is not at all required by the theory previously presented. Furthermore, since all the bumps have the same height, then the bottom stratum is composed of all $2^6 = 64$ possible different combinations of the various feet being either on the lower plane or higher bumps.

Following the steps in the algorithm:
1. The inputs u_3 and u_4 move the feet in and out of contact with the ground. Clearly, when the feet are out of contact with the ground, all the other variables $(x, y, \theta, \phi_1$ and $\phi_2)$ are independent of the foot height. Thus, the Lie bracket decoupling assumption holds.
2. The nominal trajectory is given in Equation 4. We will denote the strata composing the bottom stratum by S_{000000} (all feet on the lower level) through S_{111111} (all feet on the upper level). An easy numerical computation shows that the system traverses 45 members of the bottom strata as s goes from 0 to 2. In particular, the first and last 5 strata and associated s–values are:

S_{000010}	0.0
S_{001001}	0.0001
S_{001000}	0.0816
S_{000000}	0.1051
S_{000100}	0.1318
$\vdots$	$\vdots$
S_{010100}	1.8682
S_{010110}	1.8792
S_{010010}	1.8950
S_{000010}	1.9185
S_{000110}	2.0

3. Denoting $(x, y, \theta, \phi_1, \phi_2, d_1, d_2)$ by $(x_1, \ldots, x_7)$, the stratified extended system is

$$
\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} = \begin{bmatrix} l\cos t \\ l\sin t \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} v_1 + \begin{bmatrix} l\cos t \\ l\sin t \\ -1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} v_2
+ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} v_3 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} v_4 + \begin{bmatrix} -2l\sin t \\ 2l\cos t \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} v_5.
$$

The first two vector fields correspond to motion when all feet are in contact with the surface, the third and fourth vector fields correspond to motion when the feet are out of contact with the surface and the seventh is the Lie bracket between the first two vector fields.
4. The formal equation is simply

$$\dot{S}(t) = S(t)\left(b_1 v_1 + \cdots + b_5 v_5\right),$$

where b_1 through b_5 represent the five vector fields in the stratified extended system.
5. Differentiating the Chen-Fliess series and equating the coefficients of the b_i in the formal equation gives the following set of differential equations:

$$
\begin{aligned}
\dot{h}_1 &= v_1 & \dot{h}_4 &= v_4 \\
\dot{h}_2 &= v_2 & \dot{h}_5 &= h_1 v_2 + v_5 \\
\dot{h}_3 &= v_3.
\end{aligned}
$$

6. The v_i's are determined by differentiating the nominal trajectory with respect to time, equating it with the stratified extended system and solving for the v_i's. From these fictitious inputs, the backward Philip Hall coordinates can be determined from the set of differential equations in the previous step. For the nominal trajectory in Equation 4 divided into 30 subtrajectories, on the 25th step, at which point the state of the system is $x = 1.6$, $y = 1.6$ and $\theta = 5.02655$ the values for the fictitious inputs and the backward Philip Hall coordinates are:

$$
\begin{aligned}
v_1 &= 0.080931 & h_1 &= 0.080931 \\
v_2 &= -0.109469 & h_2 &= -0.109469 \\
v_3 &= 0.0599014 & h_3 &= -0.080931 \\
v_4 &= 0.109469 & h_4 &= 0.109469 \\
v_5 &= -0.080931 & h_5 &= 0.051042.
\end{aligned}
$$

A plot of the motion of the robot near the end of its motion is illustrated in Figure 3. The black dots represent the foot placement locations, illustrating the complex pattern of foot placements necessary to achieve the motion.

We emphasize that this was a greatly simplified example in that all the bumps had the same height (so that it was only necessary to check if a foot was on *a* bump, rather than differentiate among the bumps; furthermore, the kinematics of the robot were assumed to be very simple and unchanged regardless of on which of the strata of which the bottom stratum is composed the robot is. Introducing more realistic complexity is not theoretically problematic; however, it would result in a much more cluttered and hard to follow presentation.

The algorithm has also been experimentally verified using a robotic platform with four PUMA 560 manipulators and a vision based control strategy to supplement the open loop method presented here. The details of the vision system as well as a complete description of the performance of the system can be found in [31].

V. Conclusions and Future Work

This paper presented an extension of the stratified motion planning algorithm to the case where the domain is nonsmooth, but known. The extension was rather straight forward in that, while the structure of the stratified space increased in complexity significantly due to the fact that the bottom stratum is actually a set of multiple strata, the only necessary modification to the algorithm is the need to compute the nominal trajectory through multiple bottom strata. The theory was illustrated with a simple example.

Avenues of related current and future work include supplementing the algorithm with a means for visual sensing, extending motion planning algorithms other than the one from [28] to the smooth and nonsmooth stratified case; and, the development and dissemination of a general stratified motion planning Mathematica toolkit.

The authors gratefully acknowledge the support of this research by the National Science Foundation via grants ISS99-84107-CAREER and IIS99-10602-SGER.

References

[1] Bill Goodwine and Joel Burdick. Controllability of kinematic systems on stratified configuration spaces. *IEEE Transactions on Automatic Control*, 46(3):358–368, 2001.

[2] J. William Goodwine. *Control of Stratified Systems with Robotic Applications.* PhD thesis, California Institute of Technology, 1998.

[3] Bill Goodwine and Joel Burdick. Stratified motion planning with application to robotic finger gaiting. In *Proceedings of the 1999 IFAC World Congress*, 1999.

[4] Bill Goodwine and Joel Burdick. A general method for motion planning for quasi–static legged robotic locomotion and finger gaiting. *IEEE Transactions on Robotics and Automation*, 18(2):209–222, April 2002.

[5] S. Kajita and K. Tani. Study of dynamic biped locomotion on rugged terrain. In *IEEE International Conference Robotics and Automation*, pages 1405–1411, Sacramento, CA, 1991.

[6] J.K. Lee and S.M. Song. Path planning and gait of walking machine in an obstacle-strewn environment. *J. Robotics Systems*, 8:801–827, 1991.

[7] Matt Berkemeier. Modeling the dynamics of quadrupedal running. Accepted for publication in the International Journal of Robotics Research (September, 1998), 1998.

[8] S.M. Song and K.J. Waldron. *Machines that walk: the Adaptive Suspension Vehicle.* MIT Press, 1989.

[9] M.H. Raibert. *Legged Robots that Balance.* MIT Press, 1986.

[10] J. J. Collins and Ian Stewart. Hexapodal gaits and coupled nonlinear oscillator models. *Biological Cybernetics*, 68:287–298, 1993.

[11] C.H. Chen, K. Mirza, and D.E. Orin. Force control of planar power grasp in the digits system. In *Fourth International Symposium on Robotics and Manufacturing*, pages 189–194, 1992.

[12] Domenico Prattichizzo and Antonio Bicchi. Dynamic analysis of mobility and graspability of general manipulation systems. *IEEE Transactions on Robotics and Automation*, 14(2):241–258, April 1998.

[13] E. Rimon and J.W. Burdick. Configuration space analysis of bodies in contact – i. *Mechanism and Machine Theory*, 30(6):897–912, August 1995.

[14] E. Rimon and J.W. Burdick. Configuration space analysis of bodies in contact – ii. *Mechanism and Machine Theory*, 30(6):913–928, August 1995.

[15] K.B. Shimoga. Robot grasp synthesis algorithms: A survey. *The International Journal of Robotics Research*, 15(3):230–266, 1996.

[16] D. J. Montana. The kinematics of contact and grasp. *The International Journal of Robotics Research*, 7(3):17–25, 1988.

[17] J.C. Trinkle and R.P. Paul. Planning for dexterous manipulation with sliding contacts. *The International Journal of Robotics Research*, 9(3):24–48, 1990.

[18] L. Han, Y.S. Guan, Z.X. Li, Q. Shi, and J.C. Trinkle. Dextrous manipulation with rolling contacts. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, pages 992–997. IEEE, 1997.

[19] Ronald S. Fearing and T.O. Binford. Using a cylindrical tactile sensor for determining curvature. *IEEE Transactions on Robotics and Automation*, 7(6):806–817, 1991.

[20] Kenneth Salisbury and Christopher Tarr. Haptic rendering of surfaces defined by implicit functions. In *Proceedings of the 1997 ASME International Mechanical Engineering Congress and Exposition*, pages 61–67, 1997.

[21] K. Salisbury, D. Brock, T. Massie, N. Swarup, and C. Zilles. Haptic rendering: Programming touch interation with virtual objects. In *Proceedings of the Symposium on Interactive 3D Graphics*, pages 123–130, 1995.

[22] T. Okada. Object handling system for manual industry. *IEEE Transactions on Systems, Man and Cybernetics*, 9(2):79–89, 1979.

[23] Maw Kae Hor. *Control and Task Planning of the Four Finger Manipulator.* PhD thesis, NYU, 1987.

[24] R.S. Fearing. Implementing a force strategy for object reorientation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 96–102, 1986.

[25] J. Hong, G. Lafferriere, B. Mishra, and X. Tan. Fine manipulation with multifinger hands. In *IEEE International Conference on Robotics and Automation*, pages 1568–1573, Cincinnati, OH, May 1990.

[26] I-M Chen and J.W. Burdick. A qualitative test for n–finger force–closure grasps on planar objects with applications to manipulation and finger gaits. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 814–820, 1993.

[27] Moëz Cherif and Kamal K. Gupta. Panning quasi-static fingertip manipulations for reconfiguring objects. *IEEE Transactions on Robotics and Automation*, 15(5):837–848, 1999.

[28] G. Lafferriere and Hector J. Sussmann. A differential geometric approach to motion planning. In X. Li and J. F. Canny, editors, *Nonholonomic Motion Planning*, pages 235–270. Kluwer, 1993.

[29] R. Abraham, J. E. Marsden, and T. Ratiu. *Manifolds, Tensor Analysis, and Applications.* Springer–Verlag, second edition, 1988.

[30] Bill Goodwine and Joel Burdick. Gait controllability for legged robots. In *Proceedings of the 1998 International Conference on Robotics and Automation*, Leuven, Belgium, 1998. IEEE.

[31] Yejun Wei, Steven B. Skaar, and Bill Goodwine. Vision-based stratified robotic manipulation. In *To appear in the Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems.*

List of Captions