

# Stratified Motion Planning on Non-Smooth Domains with Application to Robotic Legged Locomotion and Manipulation

Yejun Wei                      Bill Goodwine  
ywei@nd.edu    goodwine@controls.ame.nd.edu

Aerospace and Mechanical Engineering  
University of Notre Dame  
Notre Dame, Indiana 46556

*Abstract*— This paper presents an extension of the authors' previous 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 includes motion planning for legged robots over non-smooth (but known) terrain and manipulation of non-smooth objects with multiple robotic manipulators.

## 1 Introduction

This paper presents an extension of a novel 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 a means for motion planning for systems with switching dynamics [1], [2], [3], [4], [5], [6], [7], [8]. 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 non-smooth.

Specifically, the previous work of the authors assumed that the configuration manifold for the system under consideration was *smooth* and that the discontinuous nature of the dynamics of the system resulted only from the intermittent physical contact among various elements of the overall system. A consequence of this assumption was that a particular critical element (the “bottom stratum,” described subsequently) was smooth. Previous results did not consider the case where the bottom stratum was not smooth, which is a case that includes legged locomotion over nonsmooth terrain and manipulation of nonsmooth objects, which is the focus of this paper.

The main difficulty with such systems, and stratified systems in general, is to determine a method to analytically incorporate, either in an analysis tool or control synthesis algorithm, the discontinuous nature

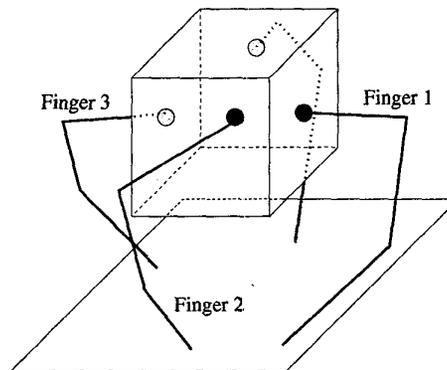
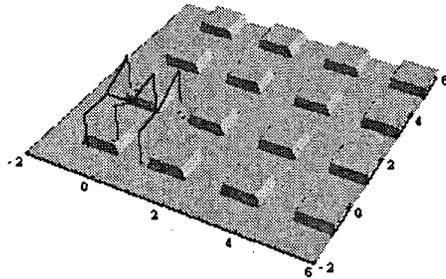


Figure 1. Non-smooth object manipulation.

of the equations of motion for the system. Incorporating the discontinuities of the equations of motion of a system into a general motion planning algorithm is difficult because almost all motion planning methods assume that the equations of motion are smooth.

An overall goal of the approach is to formulate the results in a mathematically general way so that they apply to the broadest possible class of problems. This allows, for example, the implementation of the general approach via a software toolkit where a particular user would only need to input the particular kinematics of the system under consideration and a description of the terrain. The software toolkit would neither be limited to any particular kinematic design, nor would it depend on the number of legs and/or fingers of a robotic platform. Indeed, both the manipulation results, schematically illustrated in Figure 1 and presented in Section 4.1 as well as the legged locomotion results, schematically illustrated in Figure 2 and presented in Section 4.2, use nearly identical software to plan the motions for the systems. One aspect of our current efforts is the development of a “production quality” software toolkit for general stratified systems.

Prior research efforts concerning legged locomotion have typically focused either on a particular morphology such as in [9], [10], [11], [12] or a particular locomotion assumption such as in [12], [13]. Some more



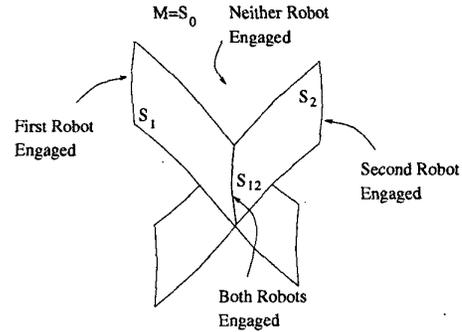
**Figure 2.** Hexapod locomotor on non-smooth terrain.

general results exist, such as in [14], [15], [13]. 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 [16]. Many efforts considered the analysis of grasp stability and force closure [17], [18], [19], motion planning assuming continuous contact [20], [21], [22] and haptic interfaces [23], [24], [25]. Finger gaiting has been implemented in certain instances [26], [27], [28] and also partially considered theoretically [5], [29], [30], [31]. Perhaps the approach which most closely mirrors that of the subject of this proposal is in [16] where notions of controllability and observability from “standard” control theory are applied to grasping (however, these results are limited to the linear case and do not allow for fingers to intermittently contact the object). In contrast 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.

## 2 Smooth Stratified Systems

This work is an extension of previous work by the authors; therefore, a short review of previous results is necessary. This section outlines the stratified motion planning method for smooth systems, which forms the basis for the extension to non-smooth stratified case. Many details are necessarily omitted, and the interested reader is referred to [1], [2], [3], [4], [5], [6], [7], [8] for a complete, detailed exposition.

A simple example will provide an intuitive understanding of the geometry inherent in stratified systems. Consider the simplistic example two fingers intermittently engaging the a smooth object, such as a sphere. The set of configurations corresponding to one of the robots engaging the object is a smooth codimension one submanifold contained in the configuration space.



**Figure 3.** Configuration manifold structure for two cooperating robots.

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*. The structure of the configuration manifold for such a system is abstractly illustrated in Figure 3.

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  $i$ th robot engages the object. Denote, the intersection of  $S_i$  and  $S_j$ , by  $S_{ij} = S_i \cap S_j$ . The set  $S_{ij}$  physically corresponds to states where both the  $i$ th and  $j$ th robots engage the object. Further intersections can be similarly 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*. 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.

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

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

where the first subscript,  $I$ , indexes the stratum upon which the equations are defined. The motion planning algorithm for smooth stratified systems is based upon the method presented in [32]. The approach is to construct an *extended system* 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 is trivial since it is constructed so that the span of all the vector field is full rank. Formal algebraic computations utilizing indeterminates,  $b_i$ , formal exponential expansions of the form  $e^{b_i} = 1 + b_i + \frac{b_i^2}{2!} + \dots$  which can

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

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

where  $\phi_\epsilon^{g_1}(x_0)$  represents the solution of the differential equation  $\dot{x} = g_1(x)$  at time  $\epsilon$  starting from  $x_0$  provide a mechanism to determine the real control inputs.

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 (called ‘‘Lie bracketed decoupling,’’ which will always be satisfied with holonomic manipulators. See [4] for a complete discussion.), then it is straight forward to show that vector fields defined on *multiple strata* can be considered simultaneously in the motion planning algorithm (a detailed explanation can be found in [7]). An outline of the algorithm is as follows.

1. Check that the Lie bracket decoupling assumption holds.
2. Check that the stratified system is controllable (see [1], [3]).
3. Determine a *nominal trajectory* in the bottom stratum.
4. Construct the *extended stratified system* on the bottom strata. This is of the form

$$\begin{aligned} \dot{x} = & g_1(x)v_1 + \cdots + g_m(x)v_m \\ & + \underbrace{g_{m+1}v_{m+1} + \cdots + g_nv_n}_{\text{from higher strata}} \\ & + \underbrace{g_{n+1}v_{n+1} + \cdots + g_pv_p}_{\text{any Lie brackets}} \end{aligned} \quad (3)$$

where the  $\{g_1, \dots, g_p\}$  span  $T_x S_0$  and are the control vector fields from multiple strata, the inputs  $v_1, \dots, v_n$  are real, and the inputs  $v_{n+1}, \dots, v_p$  are *fictitious*.

5. Construct the *formal equation*, which is simply Equation 3 written in indeterminates,  $\dot{S}(t) = S(t)(b_1v_1 + \cdots + b_s v_s)$ , where the  $S(t)$  are polynomial Lie series (see [32] for details).
6. 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*,  $h_i$ .
7. Solve the o.d.e.'s 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.

8. 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.

Unfortunately, most of these steps are rather involved, but space limitations prevent the inclusion of most detail. Again, references [32], [33] provide a good overview of the nonstratified version of the algorithm, and references [1], [2], [3], [4], [5], [6], [7], [8] present the *smooth* stratified version.

### 3 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 2 can be extended to the non smooth case. Consider the case of the four fingers manipulating the cube illustrated in Figure 4. If each finger has, say, three revolute joints, then the overall configuration space for the system is  $S_0 = \text{SE}(3) \times S^{3 \times 4}$ , where  $\text{SE}(3)$  describes the configuration of the cube and  $S^{3 \times 4}$  represents the configuration of the joints. As described in Section 2, 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 6 disconnected smooth manifolds *with boundary*.

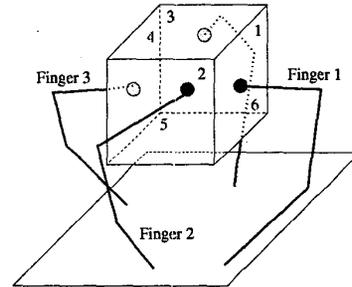
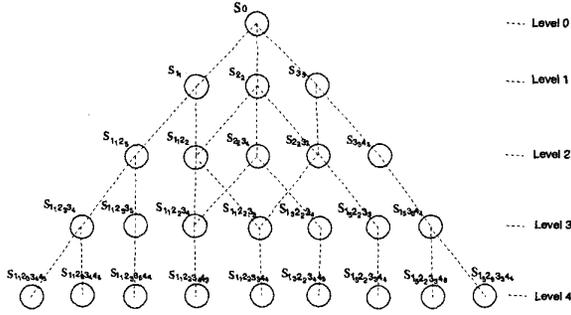


Figure 4. A polygonal object.

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 referred to as its codimension. Thus, the bottom stratum for the structure our system is on the 4th level and contains all the strata with codimension 4 representing all the four fingers are in contact with the cube. And, 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 5, where the nodes of 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.



**Figure 5.** The Structure of the Non-Smooth Stratified System.

Comparing the structure of this stratified system with the system of robot fingers engaging a smooth object described in Section 2, we can find that for the stratified system with multiple fingers to manipulate a smooth object, there is just one bottom stratum in the configuration structure. However, for our stratified system for four finger to manipulate a cube object, the bottom stratum is the union of  $P_6^4 = 360$  manifolds with boundary. In the rest of the development, we will consider each of these manifolds with boundary as a separate stratum.

On the bottom level, Figure 5 shows that the system cannot move from one stratum to another since there are no edges between them. But, 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 example, the system can be moved along a series of strata  $S_{1,2,3,4,5} \leftrightarrow S_{1,2,3,4} \leftrightarrow S_{1,2,3} \leftrightarrow S_{1,2,3,5} \leftrightarrow S_{1,1,2,3,5,4,4}$  to move between strata  $S_{1,1,2,3,4,5}$  and  $S_{1,1,2,3,5,4,4}$  by going through the strata in the levels 2,3 and 4, or the system can be moved between strata  $S_{1,5,2,3,4,6}$  and  $S_{1,5,2,6,3,3,4,4}$  by going through the strata just in levels 3 and 4  $S_{1,5,2,6,3,3,4,6} \leftrightarrow S_{1,5,3,3,4,4} \leftrightarrow S_{1,5,2,6,3,3,4,4}$ .

Although we have presented the structure of a non-smooth stratified system by way of a particular example, namely, four fingers manipulating a cube, formulating the generic structure is straight forward. In particular, a stratum is simply a smooth manifold *with boundary* and a single, smooth bottom stratum does not necessarily exist, but is, in fact, the union of mul-

iple strata.

Having developed the structure of a non-smooth stratified system, we consider motion planning for non-smooth systems, which only involves one additional complication relative to the smooth stratified system case. In particular, since the bottom stratum is not simply a smooth manifold, then the nominal trajectory (step 3 for the smooth case) will not be contained within a single stratum. To see this, consider the same example of four fingers manipulating a cube and consider an ending configuration wherein the four fingers are contacting a different face of the cube than upon which they started. Since the nominal trajectory is computed in the bottom stratum (all the fingers in contact with the cube), the nominal trajectory will need to switch among the various strata from which the bottom stratum is comprised.

## 4 Examples

### 4.1 Robotic Manipulation

To verify the motion planning method for the manipulation of the cube, we apply the method by simulating the motion when four spherical fingers manipulate a cube with width  $2b$  ( $b = 3$  inch). We number the fingers and the surfaces of the cube the manner illustrated in Figure 4. The parameterization of the surface of the cube is assumed to be

$$\begin{aligned} \text{on surface } S_1: \quad & co_1(u_{o_1}, v_{o_1}) = \{-b, v_{o_1}, u_{o_1}\}^T \\ \text{on surface } S_2: \quad & co_2(u_{o_2}, v_{o_2}) = \{u_{o_2}, v_{o_2}, b\}^T \\ \text{on surface } S_3: \quad & co_3(u_{o_3}, v_{o_3}) = \{b, u_{o_3}, v_{o_3}\}^T \quad (4) \\ \text{on surface } S_4: \quad & co_4(u_{o_4}, v_{o_4}) = \{v_{o_4}, b, u_{o_4}\}^T \\ \text{on surface } S_5: \quad & co_5(u_{o_5}, v_{o_5}) = \{v_{o_5}, b, u_{o_5}\}^T \\ \text{on surface } S_6: \quad & co_6(u_{o_6}, v_{o_6}) = \{u_{o_6}, -b, v_{o_6}\}^T, \end{aligned}$$

where in each case,  $u_{o_i}, v_{o_i} \in (-3, 3)$ . The surface parameterization of the finger is:

$$c_f(u_f, v_f) = \begin{bmatrix} 1.1075 \cos(u_f) \cos(v_f) \\ -1.1075 \cos(u_f) \sin(v_f) \\ 1.1075 \sin(u_f) \end{bmatrix},$$

where  $u_f \in (-\frac{\pi}{2}, \frac{\pi}{2})$  and  $v_f \in (-\pi, \pi)$ . Space limitations prevent including the details, but the kinematics of the manipulators are taken to be identical to a PUMA 560.

We would like to rotate the cube along axis  $\omega = [1, 1, 1]^T$  for an angle of  $\theta$  up to  $\pi$  using the four spherical fingers. The construction of all the fingers and the object at the initial state is shown in Figure 6, a close picture of both the fingers and the object during the motion procedure are shown in Figure 7 to 11.

The theoretical and simulation results have also been verified experimentally. The experimental system is shown in Figure 12, which consists of four PUMA 560 robots mounted on a common platform, where each

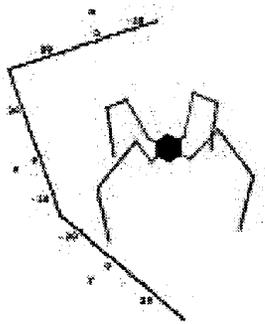


Figure 6. Manipulation at  $\theta = 0$

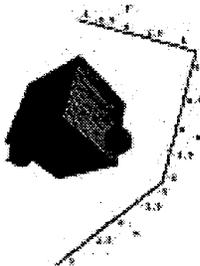


Figure 7. Manipulation at  $\theta = 0.6$

robot has and identical spherical end effector. Three 8-axis Galil 1880 motion control boards are installed on one PC running Linux operating system to control the motion of the robots. An arbitrary motion can be specified for the object besides the motion we showed in the simulation session. Experiments show that algorithmically the method works as expected. Due to the highly nonlinear nature of the problem and the fundamentally open loop nature of the algorithm, only limited manipulation is possible before the cube is dropped. A future paper will detail current efforts to implement a vision-based system which will provide feedback information to the manipulation system, leading to more robust manipulation.

#### 4.2 Legged Locomotion

The algorithm was also verified via simulation on a hexapod robot model as illustrated in Figure 2. The surface upon which the robot is locomoting is parameterized by

$$h(x, y) = \frac{\text{floor}(\text{mod}(x, 2)) \text{floor}(\text{mod}(y, 2))}{4}$$

which produces the partial "checker-board" height pattern in Figure 2. The equations of motion for the hexapod are taken to be

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

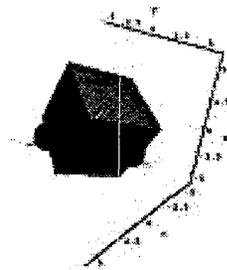


Figure 8. Manipulation at  $\theta = 1.2$

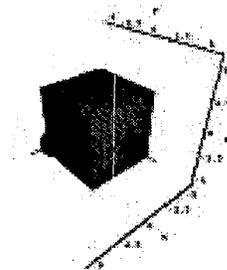


Figure 9. Manipulation at  $\theta = 1.8$

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

where  $(x, y, \theta)$  represents the planar position of the center of mass,  $\phi_1$  is the front to back angular deflection of legs 1-4-5,  $\phi_2$  is the angular deflection of legs 2-3-6,  $l$  is the leg length and  $h_i$  is the height of the legs off the ground. The functions  $\alpha(h_1)$  and  $\beta(h_2)$  are defined by

$$\alpha(h_1) = \begin{cases} 1 & \text{if } h_1 = 0 \\ 0 & \text{if } h_1 > 0 \end{cases} \quad \beta(h_2) = \begin{cases} 1 & \text{if } h_2 = 0 \\ 0 & \text{if } h_2 > 0 \end{cases}$$

Note that these simplistic equations literally require some foot slippage in order to describe the motion of a robot like the one illustrated in Figure 2; however, we utilize this very simple model in order to clarify the presentation of the theory. A real robot could be designed to conform to such a kinematic model if the front and rear leg front to back deflections were appropriately modified and if the "knee" joint were kinematically linked to the front to back deflections in such a manner as to prevent foot slippage.

Figure 13 illustrates the motion of the hexapod as it traverses the nonsmooth terrain following a nominal trajectory of

$$\begin{aligned} x(s) &= s \\ y(s) &= s \\ \theta(s) &= 2\pi s, \end{aligned}$$

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 com-

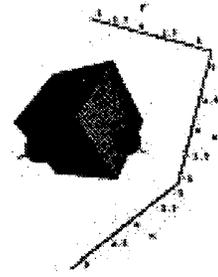


Figure 10. Manipulation at  $\theta = 2.4$

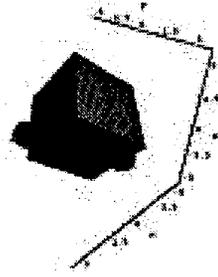


Figure 11. Manipulation at  $\theta = 3.0$

pletes one unit diagonally in the  $x$  and  $y$  directions. 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. We will denote the strata composing the bottom stratum by  $S_{000000}$  (all feet on the lower level) through  $S_{111111}$ . 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

A plot of the motion of the robot near the end of its motion is illustrated in Figure 13. The black



Figure 12. Manipulation of a Cube Object.

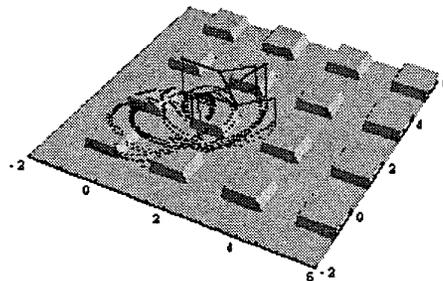


Figure 13. Hexapod motion.

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.

## 5 Conclusions and Future Work

This paper presented an extension of the stratified motion planning algorithm to the case where the domain upon which the robot is evolving is non-smooth, 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, which is relatively simple to do. The theory was illustrated with very simple examples.

Avenues of related current and future work include

1. supplementing the algorithm with a means for vi-

sual sensing;

2. extending motion planning algorithms other than the one from [32] to the smooth and non-smooth stratified case; and,
3. development and dissemination of a general stratified motion planning Mathematica toolkit.

**Acknowledgments:** 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*, (3):358–368, 2001.
- [2] Bill Goodwine and Joel Burdick. Trajectory generation for kinematic legged robots. In *IEEE International Conference on Robotics and Automation*, pages 2689–2696, Albuquerque, NM, 1997.
- [3] 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.
- [4] J. William Goodwine. *Control of Stratified Systems with Robotic Applications*. PhD thesis, California Institute of Technology, 1998.
- [5] Bill Goodwine and Joel Burdick. Stratified motion planning with application to robotic finger gaiting. In *Proceedings of the 1999 IFAC World Congress*, 1999.
- [6] Bill Goodwine. Stratified motion planning with application to robotic finger gaiting. To appear in the 10th World Congress on the Theory of Machines and Mechanisms, 1999.
- [7] Bill Goodwine and Joel Burdick. A general method for motion planning for quasi-static legged robotic locomotion and finger gaiting. To appear in the *IEEE Transactions on Robotics and Automation*, available electronically at <http://controls.ame.nd.edu/~bill/papers.html>, 2000.
- [8] Bill Goodwine and Yejun Wei. Theoretical and experimental investigation of stratified robotic finger gaiting. In *Proceedings of the 2001 IEEE International Conference on Robotics and Automation*. IEEE.
- [9] S. Kajita and K. Tani. Study of dynamic biped locomotion on rugged terrain. In *IEEE International Conference on Robotics and Automation*, pages 1405–1411, Sacramento, CA, 1991.
- [10] 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.
- [11] Matt Berkemeier. Modeling the dynamics of quadrupedal running. Accepted for publication in the *International Journal of Robotics Research* (September, 1998), 1998.
- [12] S.M. Song and K.J. Waldron. *Machines that walk: the Adaptive Suspension Vehicle*. MIT Press, 1989.
- [13] M.H. Raibert. *Legged Robots that Balance*. MIT Press, 1986.
- [14] J. J. Collins and Ian Stewart. Hexapodal gaits and coupled nonlinear oscillator models. *Biological Cybernetics*, 68:287–298, 1993.
- [15] 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.
- [16] 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.
- [17] E. Rimon and J.W. Burdick. Configuration space analysis of bodies in contact – i. *Mechanism and Machine Theory*, 30(6):897–912, August 1995.
- [18] E. Rimon and J.W. Burdick. Configuration space analysis of bodies in contact – ii. *Mechanism and Machine Theory*, 30(6):913–928, August 1995.
- [19] K.B. Shimoga. Robot grasp synthesis algorithms: A survey. *The International Journal of Robotics Research*, 15(3):230–266, 1996.
- [20] D. J. Montana. The kinematics of contact and grasp. *The International Journal of Robotics Research*, 7(3):17–25, 1988.
- [21] 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.
- [22] L. Han, Y.S. Guan, Z.X. Li, Q. Shi, and J.C. Trinkle. Dexterous manipulation with rolling contacts. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, pages 992–997. IEEE, 1997.
- [23] 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.
- [24] 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.
- [25] K. Salisbury, D. Brock, T. Massie, N. Swarup, and C. Zilles. Haptic rendering: Programming touch interaction with virtual objects. In *Proceedings of the Symposium on Interactive 3D Graphics*, pages 123–130, 1995.
- [26] T. Okada. Object handling system for manual industry. *IEEE Transactions on Systems, Man and Cybernetics*, 9(2):79–89, 1979.
- [27] Maw Kae Hor. *Control and Task Planning of the Four Finger Manipulator*. PhD thesis, NYU, 1987.
- [28] 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.
- [29] J. Hong and G. Lafferriere. Fine manipulation with multifinger hands. In *IEEE International Conference on Robotics and Automation*, pages 1568–1573, Cincinnati, OH, May 1990.
- [30] 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.
- [31] 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.
- [32] 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.
- [33] Richard M. Murray, Zexiang Li, and S. Shankar Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Inc., 1994.