Stratified Motion Planning with Application to Robotic Finger Gaiting

Bill Goodwine
Aerospace and Mechanical Engineering
University of Notre Dame
Notre Dame, Indiana 46556
Phone: +1 219 631 3283
Fax: +1 219 631 8341
Email: jgoodwin@nd.edu

June 23, 1998

Abstract

This paper presents a general, nonholonomic dextrous manipulation and finger gaiting technique for robotic grasping problems. This method is *general* in that it is independent of the geometry of the grasped object and independent of the morphology of the mainipulating hand. This method is based upon a nonholonomic motion planning method for smooth system that has been extended to a certain class of discontinuous systems by recognizing a particular generic geometric structure underlying legged robotic locomotion and grasping problems, called a *stratification* The stratification is a decomposition of the configuration space of the system into subsets which correspond to the various combinations of fingers contacting the object. Additionally, associated with this manipulation methodology is a definition of manipulability that takes into account the fact that fingers may intermittently engage the object.

Keywords: robotic grasping, finger gaiting, robotic manipulation, nonlinear control, motion planning.

1 Introduction

Nonlinear control is often concerned with determining control inputs which generate specified trajectories for general nonlinear systems. In this paper, we present a general, nonholonomic dextrous manipulation and finger gaiting technique for robotic grasping problems that is based upon a motion planning algorithm for nonlinear control systems. This method is *general* in that it is independent of the geometry of the grasped object and independent of the morphology of the manipulating hand. In particular, the method is independent of the type of contact between the finger and object (*e.g.*, "point contact with friction," "soft finger," rolling fingers [13, 14] etc.) and independent of the morphology of the manipulating "fingers" (*i.e.*, independent of the number of joints, number of fingers, etc.).

This method is based upon a nonholonomic motion planning method for smooth system [12] due to Lafferriere and Sussmann that has been extended to a certain class of discontinuous systems by recognizing a particular generic geometric structure underlying legged robotic locomotion and grasping problems, called a *stratification* [7, 8, 9, 10]. The stratification is a decomposition of the configuration space of the system into subsets which correspond to the various combinations of fingers contacting the object. Additionally, associated with this manipulation methodology is a definition of manipulability that takes into account the fact that fingers may intermittently engage the object.

2 Background

Throughout this paper, we distinguish between *smooth systems*, where the vector fields for the control system are smooth, and *stratified systems*, (defined in Section 3) where the vector fields are not necessarily smooth. Stratified systems include legged robotic locomotion and robotic finger gaiting systems.

This section outlines the motion planning method due to Lafferriere and Sussmann [12] for smooth, kinematic nonholonomic systems which are in the form of a non-linear affine driftless system evolving on a configuration manifold, M

$$\dot{x} = g_1(x)u_1 + \cdots + g_m(x)u_m, \qquad x \in M. \tag{1}$$

For an underactuated system (*i.e.*, $m < \dim(M)$) we make use (from [12]) of an "extended system," where "fictitious controls," corresponding to higher order Lie bracket motions, are added. The trajectory generation problem is relatively easy to solve for the extended system. The real controls are then computed from the fictitious controls associated with the extended system.

The flow along g_i is referred to as the *formal exponential* of g_i and is denoted

$$\phi_t^{g_i}(x) := e^{tg_i}(x) = (I + tg_i + \frac{t^2}{2}g_i^2 + \cdots). \tag{2}$$

Note that terms of the form g_i^2 are *not* vector fields and are rigorously considered by associating a Lie algebra of *indeterminates* to the Lie algebra of vector fields associated with the control problem. A complete exposition on the use of indeterminates in this manner is omitted here due to space limitations and we refer the reader to references [12, 17, 18] for a complete explanation.

The Campbell–Baker–Hausdorff formula is given in the following theorem and is of basic importance to the methods outlined in this paper.

Theorem 2.1 *Given two smooth vector fields g_1, g_2 the composition of their exponentials is given by*

$$e^{g_1}e^{g_2} = e^{g_1+g_2+\frac{1}{2}[g_1,g_2]+\frac{1}{12}([g_1,[g_1,g_2]]-[g_2,[g_1,g_2]])\cdots} \tag{3}$$

where the remaining terms may be found by equating terms in the (non-commutative) formal power series on the right– and left–hand sides.

Associate with the system in Equation 1 the *extended system*:

$$\dot{x} = g_1v_1 + \cdots + g_mv_m + g_{m+1}v_{m+1} + \cdots + g_sv_s \tag{4}$$

where the $g_{m+1}, \ldots, g_s$ are higher order Lie brackets of the g_i, chosen so that $\dim(\text{span}\{g_1, \ldots, g_s\}) = \dim(M)$. (Note that this requires that the system is controllable since it is simply a statement of the Lie Algebra Rank Condition.) The v_i's are called *fictitious inputs* since they may not correspond with the actual system inputs. The higher order Lie brackets must belong to the Philip Hall basis (see [15]) for the Lie algebra generated by the control vector fields.

To steer the system from a point p to a point q, define a curve, $\gamma(t)$ connecting p and q (a straight line would work, but is not necessary) and solve

$$\dot{\gamma}(t) = g_1(\gamma(t))v_1 + \cdots + g_s(\gamma(t))v_s \tag{5}$$

for the fictitious controls v_i.

To compute the actual control inputs, determine the Philip Hall basis for the Lie algebra generated by $g_1, \ldots, g_m$, and denote it by $B_1, B_2, \ldots, B_s$. It is possible to represent all flows of Equation 1 in the form

$$S_t(x) = e^{h_s(t)B_s} e^{h_{s-1}(t)B_{s-1}} \cdots e^{h_2(t)B_2} e^{h_1(t)B_1}(x) \tag{6}$$

for some functions $h_1, h_2, \ldots, h_s$, called the (backward) Philip Hall coordinates. Furthermore, $S_t(x)$ satisfies the formal differential equation

$$\dot{S}(t) = S(t)(B_1 v_1 + \cdots + B_s v_s), \quad S(0) = 1, \tag{7}$$

where $S_t(x)$ has been replaced by $S(t)$. If we define the *adjoint mapping*

$$\mathrm{Ad}_{e^{-h_i B_i}} B_j = e^{-h_i B_i} B_j e^{h_i B_i},$$

then it can be shown that

$$\mathrm{Ad}_{e^{-h_i B_i} \cdots e^{-h_{j-1} B_{j-1}}} B_j \dot{h}_j = \left(\sum_{k=1}^{s} p_{j,k}(h) B_k \right) \dot{h}_j,$$

for some polynomials $p_{j,k}(h)$. (For a more detailed explanation, see [14]). Equating coefficients yields the differential equations

$$\dot{h} = A(h)v \qquad h(0) = 0. \tag{8}$$

These equations specify the evolution of the Philip Hall coordinates in response to the fictitious inputs, which were found via Equation 5, and represent how long the system must flow along each basis element in Equation 6.

Determining the real inputs is straight–forward. Note that the fictitious inputs are defined as a function of time which occur *simultaneously*; whereas, the Philip Hall coordinates occur *sequentially*. Since the system flows along each basis element sequentially, the real control inputs can be determined by making use of piece–wise constant approximations to Lie bracket motions, *e.g.*,

$$\phi^t_{[g_1,g_2]}(x_0) \approx \phi^{\sqrt{t}}_{-g_2} \circ \phi^{\sqrt{t}}_{-g_1} \circ \phi^{\sqrt{t}}_{g_2} \circ \phi^{g_1}_{\sqrt{t}}(x_0). \tag{9}$$

If the extended system contains Lie brackets of order higher than two, then we must transform the backward Philip Hall coordinates into forward Philip Hall coordinates (see [12]). The method is particularly elegant in that the higher order error associated with approximating flows along Lie bracket vector fields as in Equation 9 can be directly compensated when approximating Lie brackets of the same order as the error. For simplicity, we will restrict our attention to second order Lie bracket systems in this paper. However, we note that this is not a theoretical limitation of the method. Finally, note that this method works exactly for nilpotent systems, and approximately for non–nilpotent systems. However, arbitrary precision can be obtained by iterating the method for non–nilpotent systems.

3 Stratified Systems

The method outlined in the preceding section does not apply to stratified systems since it requires that the vector fields that define the control system be smooth. Before considering the motion planning problem for stratified systems we will define and consider the fundamental geometry of stratified systems. We will motivate our definition of a stratified configuration space with a simple example.

3.1 Stratified Configuration Spaces

Example 3.1 Consider two cooperating robot arms which intermittently engage a smooth object as illustrated in Figure 1 (for the purposes of this example, ignore the important practical issues of force closure and manipulability). The configuration manifold for the system describes the spatial position and orientation of a reference frame rigidly attached to the object as well as variables such as joint angles which describe the internal geometry of the robots. The set of configurations corresponding to one of the robots engaging the object is a codimension one submanifold 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 codimension 2 submanifold of the configuration space formed by the intersection of the single contact submanifolds. We will refer to each submanifold as a *stratum*. The structure of the configuration manifold for such a system is abstractly illustrated in Figure 2. Note that the equations of motion for the system will be different on each submanifold because the constraints on the system will be different on each submanifold. Our approach is to exploit this type of geometric structure of such configuration spaces.

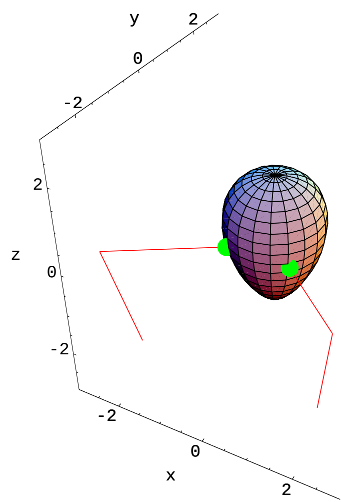

Figure 1. Two cooperating robots manipulating an object.

By considering systems more general than the two cooperating robots from Example 3.1, we can develop a general definition of stratified configuration spaces. Let M denote the system's entire configuration manifold (it will often be convenient to denote this space as S_0). Let $S_i \subset M$ denote the codimension one submanifold of M that corresponds to all configurations where only the ith robot engages the object. Denote, the intersection of S_i and S_j, by $S_{ij} = S_i \cap S_j$. The set S_{ij} physically corresponds to states where both the ith and jth robots engage the object. Further intersections can be similarly defined in a recursive fashion: $S_{ijk} = S_i \cap S_j \cap S_k = S_i \cap S_{jk}$, etc. We will call the stratum with the lowest dimension containing the point x as the *bottom stratum*, denoted S_B, and any other submanifolds containing x as *higher strata*. When making relative comparisons among different strata, we will refer to lower dimensional (*i.e.* higher codimension) strata as *lower strata*, and higher dimensional (*i.e.* lower codimension) strata as *higher strata*.

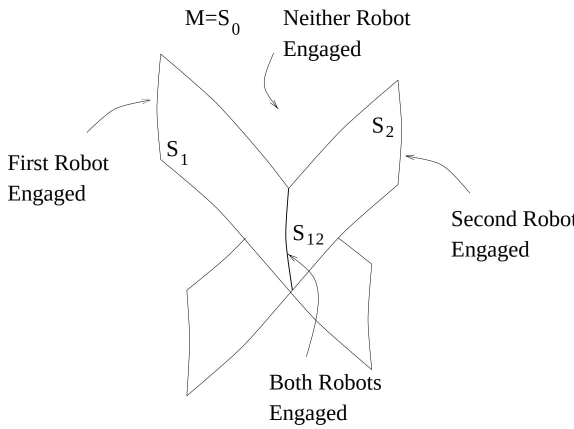

Figure 2. Configuration manifold structure of two cooperating robots.

DEFINITION 3.2: (STRATIFIED CONFIGURATION MANIFOLD)
Let M be a manifold, and n functions $\Phi_i : M \mapsto \mathbb{R}$, $i = 1, \ldots n$ be such that the level sets $S_i = \Phi_i^{-1}(0) \subset M$ are regular submanifolds of M, for each i, and the intersection of any number of the 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 M. Then M and the functions Φ_n define a *stratified configuration space.*

3.2 Stratified Motion Planning

The main difficulty with stratified systems is that the various sets of equations of motion are defined on different spaces. Since we are ultimately required to consider the vector fields associated with each stratum in one common space, vector fields on different strata must have a particular relationship. In this case, that common space will be the bottom stratum and the particular relationship will involve Lie bracket. A simple example will help motivate this.

EXAMPLE 3.3 Consider the simple cooperating robot configuration space as shown in Figure 2. Assume that on stratum S_{12}, the vector field $g_{1,1}$ moves the system off of S_{12} and onto S_1, (arm 2 disengages the object) and correspondingly, $g_{2,1}$ moves the system off of S_{12} onto S_2. Also, consider the vector fields $g_{1,2}$ and $g_{2,2}$, defined on S_1 and S_2 respectively. Consider the following sequence of flows, starting from the point $x_0 \in S_{12}$

$$x_f = \underbrace{\phi^{t_6}_{-g_{2,1}}}_{S_{12}\leftarrow S_2} \circ \underbrace{\phi^{t_5}_{g_{2,2}}}_{\text{on } S_2} \circ \underbrace{\phi^{t_4}_{g_{2,1}}}_{S_2\leftarrow S_{12}} \circ \underbrace{\phi^{t_3}_{-g_{1,1}}}_{S_{12}\leftarrow S_1} \circ \underbrace{\phi^{t_2}_{g_{1,2}}}_{\text{on } S_1} \circ \underbrace{\phi^{t_1}_{g_{1,1}}}_{S_1\leftarrow S_{12}} (x_0). \tag{10}$$

The notation under each flow indicates what the flow is doing, *e.g.*, "$S_{12} \leftarrow S_1$" means that the flow takes the system from S_1 to S_{12} and "on S_1" means that the flow was entirely on S_1. This sequence of flows is illustrated in

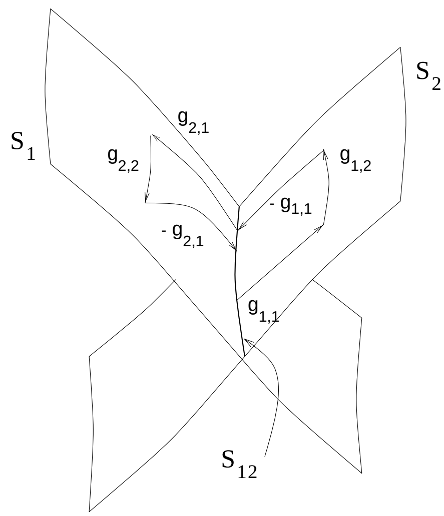

Figure 3. Sequence of flows.

Figure 3. In this sequence of flows, the system first moved off of the bottom stratum into S_1, flowed along the vector field $g_{1,2}$, flowed back onto the bottom stratum, off of the bottom stratum onto S_2, along vector field $g_{2,2}$ and back to the bottom stratum. In finger gaiting, such a sequence of flows corresponds to a finger disengaging the object, undergoing some additional motion and then engaging the object.

In the simplest case, it is clear from the Campbell–Baker–Hausdorff formula (Equation 3) that if the Lie bracket between two vector fields is zero, then their flows commute. Thus, if

$$[g_{1,1}, g_{1,2}] = 0 \quad \text{and} \quad [g_{2,1}, g_{2,2}] = 0, \tag{11}$$

we can reorder the above sequence of flows, by interchanging the flow along $g_{1,1}$ and $g_{1,2}$ and the flows along $g_{2,1}$ and $g_{2,2}$ as follows

$$x_f = \underbrace{\phi^{t_5}_{g_{2,2}} \circ \phi^{t_6}_{-g_{2,1}}}_{\text{interchanged}} \circ \phi^{t_4}_{g_{2,1}} \circ \underbrace{\phi^{t_2}_{g_{1,2}} \circ \phi^{t_3}_{-g_{1,1}}}_{\text{interchanged}} \circ \phi^{t_1}_{g_{1,1}}(x_0). \tag{12}$$

If $t_1 = t_3$ and $t_4 = t_6$, this reduces to

$$x_f = \underbrace{\phi^{t_4}_{g_{2,2}} \circ \phi^{t_2}_{g_{1,2}}}_{\text{on } S_{12}}(x_0). \tag{13}$$

Note that that $g_{1,2}$ and $g_{2,2}$ are vector fields in the equations of motion for the system on S_1 and S_2, respectively, but *not* on S_{12}. However, the sequence of flows in Equation 10, where each flow occurs on a stratum where the associated vector field is in the equations of motion results in the same flow as in Equation 12, where the vector fields are evaluated on the bottom stratum, even though they are not part of the equations of motion there. Furthermore, note that if the vector fields $g_{1,2}$ and $g_{2,2}$ are tangent to S_{12}, then the resulting flow given in Equation 12 will remain in S_{12}.

In the more general case where the Lie bracket "decoupling" expressed by Equation 11 does not hold we can consider the pull back of vector fields defined on higher strata to the bottom stratum along the flow of the vector field which takes the system off of the bottom stratum. Recall that the pull back of a vector field g by a diffeomorphism ϕ is

$$\phi^* g = (T\phi)^{-1} \circ g \circ \phi$$

and the basic fact that the flow of the pull back of a vector field is the pull back of its flow, where the pull back of the flow is defined as

$$\phi^* \phi_g^t = \phi^{-1} \circ \phi_g^t \circ \phi,$$

where ϕ_g is the flow corresponding to the vector field g. In general, the diffeomorphism ϕ and its derivative, $T\phi$ may be hard to compute since it involves integrating a nonlinear differential equation; however, for grasping problems, since this flow corresponds to a finger moving out of contact with the object, this flow will be easy to compute since it is simply the forward kinematics of the finger.

Finally, note that pulling back a vector field as described above defines a vector field at points contained in the bottom stratum, but not necessarily vector fields contained in the tangent space to the bottom stratum. Therefore, we require that the pull back satisfy the following equation

$$\langle \mathbf{d}\Phi(x), \phi^* g \rangle = 0, \qquad x \in S_B \tag{14}$$

where a level set of the function Φ describes the bottom stratum. Alternatively, we can push forward the one form $\mathbf{d}\Phi$ to points in the higher stratum where g is defined by the formula

$$\langle \phi_* \mathbf{d}\Phi(x), g(x) \rangle = \langle \mathbf{d}\Phi \phi^{-1}(m), T_x \phi^{-1} g(m) \rangle, \qquad m = \phi(x).$$

The above discussion shows that it is possible to consider vector fields in higher strata as part of the equations of motion for the system on the bottom stratum. At this point, then, we have essentially increased the class of vector fields that we may use in using the motion planning algorithm presented in Section 2 by recognizing a mechanism whereby vector fields on higher strata can be defined on the bottom stratum. Thus, the method presented in Section 2 could be used with the modification that whenever the system must flow along a vector field in a higher stratum, it switches to that stratum by appropriately disengaging the object, flows along the vector field, and then engages the object as before.

Now for stratified systems, we can proceed just like in Section 2. Pick a nominal trajectory $\gamma(t)$ connecting the starting point with the desired final point. Denoting the stratified extended system by $\dot{x} = b_1(x)v_1 + \cdots + b_k(x)v_k$, (where some of the b_i's are vector fields that come from the equations of motion for the system on higher strata (or their pull back)) set

$$\dot{\gamma}(t) = b_1(\gamma(t))v_1 + \cdots + b_k(\gamma(t))v_k,$$

and solve for the fictitious inputs, $v_i(t)$. Because of the fact that the method only works approximately for non–nilpotent systems, it will be convenient to consider the nominal trajectory as a set of concatenated subtrajectories. Then, the motion planning method will be used for each subtrajectory sequentially. We will address this issue further in Section 4.

Note that we can consider a more general notion of manipulability for a grasp. The "standard" definition of manipulability for a grasp (see, *e.g.*, [14]) does not accommodate fingers making and breaking contact. Here, similar to the definition of controllability for nonlinear systems, the grasp will be manipulable at a point x if $\{b_1, \ldots, b_k\}$ span the tangent space to the configuration manifold. Note that if $\{b_1, \ldots, b_k\}$ do not span the tangent space to the configuration space, it will not be possible to use this method to have the system to follow arbitrary object and finger motion trajectories.

The rest of the method is just as outlined in Section 2: from the fictitious inputs, determine the Philip Hall coordinates which gives the desired sequence of flows along basis elements, and then approximate Lie bracket flows by a sequence of piece–wise constant inputs taking care to switch strata as necessary.

4 Stability and Force Closure

Important issues for finger gaiting manipulation is stability and force closure. Unfortunately there is not an inherent mechanism in the direct application of the method in Section 2 to guarantee the stability of the gait. Recall that the main approach of the method was to pick a trajectory for the extended system, $\gamma(t)$, from which to determine the fictitious inputs. Then, using the fact that any flow can be decomposed to individual flows along the Philip Hall basis vector fields, the real inputs could be determined. The important point to note is that the actual trajectory will, in general, *not* be $\gamma(t)$. Thus, merely picking an initial trajectory $\gamma(t)$ which is always stable or force closure is not sufficient. What also must be guaranteed is that deviations from the initial trajectory be within the stability bounds as well.

Assume that there is a means for determining the stability or force closure properties of the system and assume that this property can be determined by means of a scalar–valued function of the configuration, $\Psi(x)$. For convenience, assume that when $\Psi(x)$ has a negative value, the system is unstable, when $\Psi(x)$ has a positive value, the system is stable, and when $\Psi(x) = 0$, the system is on the boundary between stability and instability. In the trajectory generation method, then, we must pick the initial trajectory, $\gamma(t)$ such that it does not does not intersect any unstable regions and also such that it does not intersect the stability boundary, *i.e.*, $\Psi(\gamma(t)) > 0$, $t \in [0, 1]$. Whether a stable trajectory exists for a given object for any or all initial and final configurations is clearly an important question, but is beyond the scope of this paper.

The overall approach is to take steps that are "small enough" to ensure that the system remains stable. Since we are considering small motions and need a norm to provide a measure of the length of a flow, we will consider the system locally in $\mathbb{R}^n$. Given a desired step along the trajectory, $\gamma(t)$, $t \in [0, 1]$, let $\mathcal{R} = \min\{\|x - c\|, \quad c \in \Psi^{-1}(0)\}$, *i.e.* the distance from the starting point to the closest point on the stability boundary.

The goal is to ensure that the trajectory of the system does not intersect the set $\Psi^{-1}(0)$. If x denotes the starting point, and x_f the final point, let $\gamma(t) = x + t(x_f - x)$ be the desired straight line path between the starting and end points. Also, let $\Delta = \|x_f - x\|$. Recall that the fictitious inputs, v_i were determined by solving the equation $\dot{\gamma}(t) = g_1(\gamma(t))v_1 + \cdots + g_s(\gamma(t))v_s$ for the v_i. Then $\|v_i\| < C\|\dot{\gamma}(t)\| = C\Delta$, for some fixed constant C. By the method of construction of the real inputs from the fictitious inputs, then, $\|u^i\| < C\Delta^{1/k}$, where k is the highest order Lie bracket that is approximated by a sequence of piece–wise constant inputs.

Now, pick a ball, $\mathcal{B}$, of radius $\mathcal{R}$, and let K be the maximum norm of all the (first order) vector fields, g_i for all points in the ball $\mathcal{B}$. Recall that the real inputs, u^i were given by a sequence of inputs which approximate the flow of the extended system. Denote this sequence by u^i_j, where the subscript indexes which input it is, and the superscript indexes its position in the sequence. The maximum distance that the system can possibly flow from the starting point, x, is given by the sum of the distances of the individual flows. Let $x_{max} = max_{t\in[0,1]}\{\|x(t) - x\|\}$ denote the point in the flow that is maximally distant from the starting point. (Note that this is not necessarily the final point, x_f). To guarantee stability, we want to show that $\|x_{max} - x\| < \mathcal{R}$. However, this distance, $\|x_{max} - x\|$ is necessarily bounded by the sum of the norms of each individual flow associated with one real control input, u^i_j, *i.e.*,

$$\|x_{max} - x\| \leq \sum_{i,j} \| \int_0^1 g_i u_j^i dt \|.$$

However, $\|u_j^i\| \leq C\Delta^{1/k}$ and $\|g_i(x)\| \leq K \;\forall x \in \mathcal{B}$. Thus,

$$\|x_{max} - x\| \leq \sum_{i,j} KC\Delta^{1/k}, \tag{15}$$

and since $\Delta = \|x_f - x\|$, by choosing the desired final point close enough to the starting point, the trajectory will not intersect the stability boundary.

Note that because Δ is raised to the power of $1/k$, if k is large, then it may be necessary to make Δ exceedingly small in order to ensure stability. However, the bound expressed in Equation 15 is itself very conservative since it sums the length of a bound on each individual flow in the series. Thus an appropriate step length may be best determined experimentally.

5 Example

This section illustrates the stratified motion planning algorithm with a grasping manipulation problem. Unfortunately, due to space limitations, this example is necessarily relatively simple and does not make use of the full capability of the theory (for example, no Lie bracket approximating motions are necessary for this example). Additionally, much of the machinery associated with robotic grasping manipulation is incorporated by reference, and many equations are left in symbolic form.

Consider the "egg–shaped" object in Figure 4 which is manipulated by four, three degree of freedom manipulators. The object is parameterized by

$$c(u,v) = \begin{pmatrix} \left(1+\frac{u}{\pi}\right)\cos u \, \cos v \\ \left(1+\frac{u}{\pi}\right)\cos u \, \sin v \\ \frac{3}{2}\sin u \end{pmatrix}, \qquad \begin{matrix} u \in \left(\frac{-\pi}{2}, \frac{\pi}{2}\right) \\ v \in (-\pi, \pi) \end{matrix}$$

and the kinematic model for each manipulator is illustrated in Figure 5, where the forward kinematics are

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} l\,\cos\theta_1\,\sin\theta_2 + l\,\cos\theta_1\,\sin(\theta_2-\theta_3) \\ l\,\sin\theta_1\,\sin\theta_2 + l\,\sin\theta_1\,\sin(\theta_2-\theta_3) \\ l\,\cos\theta_2 + l\,\cos(\theta_2-\theta_3) \end{pmatrix},$$

where (x, y, z) is the Cartesian position of the end effector. We assume a "point contact with friction" contact model.

The geometry of the stratified configuration space will consist of a total of 16 different strata, corresponding to all the possible combinations of finger contacts. However, as will be clear shortly, the system is manipulable if we restrict our attention to only 5 strata: when all four fingers are in contact plus each of the four cases where only one of the fingers is out of contact. We will denote these strata as $S_{1234}, S_{123}, S_{124}, S_{134}$, and S_{234} where the subscripts denote which fingers are in contact with the object.

Since the nominal trajectory is away from the kinematic singularities of the fingers, we can treat the velocity of the finger tips as the inputs to the system to simplify the computations (this will also make the equations of motion satisfy Equation 11). We note that, in general, this can not be done (for example, when the the finger tips are in

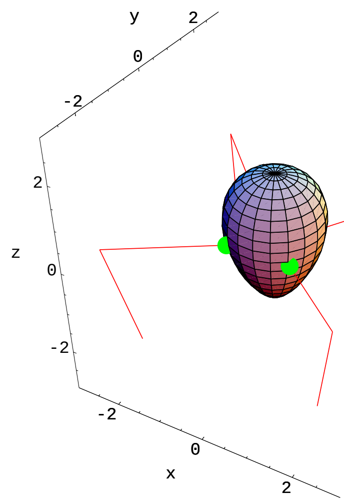

Figure 4. Four fingers manipulating an object.

rolling contact with the object); however, we emphasize that this more general case fits within the framework of the stratified motion planning method outlined in Section 3, and will be the subject of a future (longer) publication.

The following treatment of grasping and much of the associated notation is from [14]. For a grasped system with point contacts, the fundamental grasping constraint is

$$J_h(\theta, x_0)\dot{\theta} = G^T V_{po}^b, \tag{16}$$

where J_h is the hand Jacobian, G is the grasp map and V_{po}^b is the body velocity of the object relative to the "palm" frame. The grasp map is defined as

$$G = \left[\mathrm{Ad}^T_{g_{oc_1}^{-1}} B_{c_1} \cdots \mathrm{Ad}^T_{g_{oc_4}^{-1}} B_{c_4} \right], \tag{17}$$

where the wrench basis, B_{c_i} is defined as

$$B_{c_i} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

for point contacts with friction. The adjoint map in Equation 17 is defined as

$$\mathrm{Ad}^T_{g_{oc_i}^{-1}} = \begin{pmatrix} R_{oc_i} & 0 \\ \hat{p}_{oc_i} R_{oc_i} & R_{oc_i} \end{pmatrix},$$

and the subscript c_i indicates the contact point for the ith finger. Equation 16 requires that the linear velocity of the contact points be compatible with the rigid body velocity of the object and contact model. In this example, due

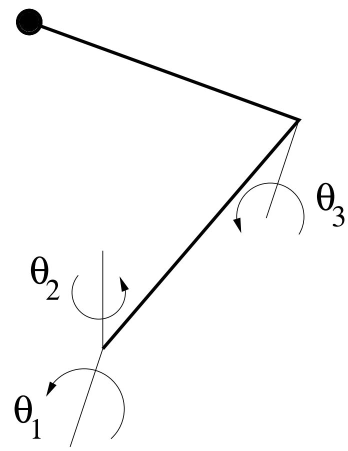

Figure 5. Finger kinematics.

to the form of the wrench basis in Equation 17, each side of Equation 16 is simply the linear velocities of the contact points. Equation 16 is actually 12 equations on the bottom stratum (9 equations on the higher strata). Thus, 6 finger tip velocities will be constrained on the bottom stratum, and 3 finger tip velocities will be constrained on the higher strata. Thus the equations of motion on the bottom stratum are of the form

$$\dot{x} = g_1(x)u_1 + \cdots + g_6(x)u_6,$$

and on the higher strata are of the form

$$\dot{x} = g_1(x)u_1 + \cdots + g_6(x)u_6 + g_7(x)u_7 + \cdots g_9(x)u_9,$$

where the first 6 inputs correspond to the inputs associated with the finger tip velocities for the three fingers contacting the object, and inputs 7 through 9 are the three degrees of freedom for the finger that is not in contact with the object. Note that $g_7(x)$ through $g_9(x)$ will be of the form $(0,0,0,0,0,0,1,0,0)$ since they are just the unconstrained finger tip velocities of the finger which is not contacting the object, and thus they will satisfy Equation 11. Therefore, they may be incorporated into the equations of motion for the bottom stratified extended system.

Incorporating these unconstrained finger tip velocity vector fields for each of the four higher strata gives a stratified extended system of the form

$$\begin{aligned} \dot{x} &= \underbrace{g_1(x) + \cdots + g_6(x)u_6}_{\text{on } S_{1234}} \\ &+ \underbrace{g_7(x)u_7 + g_8(x)u_8 + g_9(x)u_9}_{\text{from } S_{123}} \\ &\vdots \\ &+ \underbrace{g_{16}(x)u_{16} + g_{17}(x)u_{17} + g_{18}(x)u_{18}}_{\text{from } S_{234}}, \end{aligned}$$

where all the vector fields except those on the first line correspond to free finger tip motion. Tedious detailed calculations show that $\{g_1, \ldots, g_{18}\}$ spans the the tangent space to the configuration space, so the system is stratified manipulable. Since no Lie brackets are necessary to make the system stratified manipulable, this system is already in extended form, and the actual control inputs are the same as the "fictitious" inputs presented in Sections 2 and 3.

Assume that the initial and final configurations are identical (as illustrated in Figure 4), and that the desired motion is a pure rotation of 2π about the axis $\omega = (\frac{1/}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}})$. Using exponential coordinates, then, the nominal configuration of the object as a function of time is given by Rodriques' formula:

$$\begin{aligned}\gamma(t) \quad &= \quad e^{\hat{\omega}2\pi t} = I + \hat{\omega}\sin 2\pi t + \hat{\omega}^2(1 - \cos 2\pi t), \\ & \qquad t \in [0, 1].\end{aligned}$$

For the initial and final configuration for the object as shown in Figure 4, each finger is oriented at an angle of $\pi/4$ relative to the x– and y–axes. As the object is rotated, the nominal configuration for each finger is such that it contacts the object along that same axis. This can be determined by equating the forward kinematics for each finger with the the point on the surface of the object that intersects the respective $\pi/4$ radial from the origin, *i.e.*, for the finger in the quadrant where both x and y are positive, solve

$$\gamma(t)c(u, v) = \begin{pmatrix} \frac{s}{\sqrt{2}} \\ \frac{s}{\sqrt{2}} \\ 0 \end{pmatrix}$$

for u, v and s (s parameterizes how far along the radial the surface of the object intersects it), and then, using the kinematics of each finger, determine the desired joint configurations. For this particular example, this trajectory is difficult to compute analytically, but is simple to do numerically for each step in the motion of the system.

We also must check that the grasp maintains force closure during the entire manipulation. Numerically calculating the angles between the normal vectors for each combination of three fingers in contact with the line connecting the contact points along the entire trajectory shows that a coefficient of friction (assuming a static Coulomb friction model) of $\mu > 0.40$ maintains force closure.

For the desired trajectory described above decomposed into 10 subsegments, the evolution of the contact coordinates for the finger in the positive 'x– and y–quadrant is illustrated in Figure 6. Similar plots show the evolution of the contact coordinates for the other fingers. Finally, a sequence of ten "snapshots" from the manipulation are shown in Figure 7.

6 Conclusions

Our method provides a general means to solve the manipulation problem for certain types of robotic grasping problems and the simulations indicate that the approach is simple to apply. The method is general in that it is independent of morphology of the grasping system and shape of the grasped object. A by–product of this manipulation method is a test for manipulability that takes into account the fact that fingers can make and break contact with the object.

The method is limited in several ways. First, as formulated, the object must be a manifold, and so polygonal objects with edges do not fit within the framework as presented here. However, since we typically consider the nominal trajectory as the composition of several subtrajectories, it may be relatively straight–forward to devise a means to have a finger move from one smooth surface of the object to another across an edge which can be

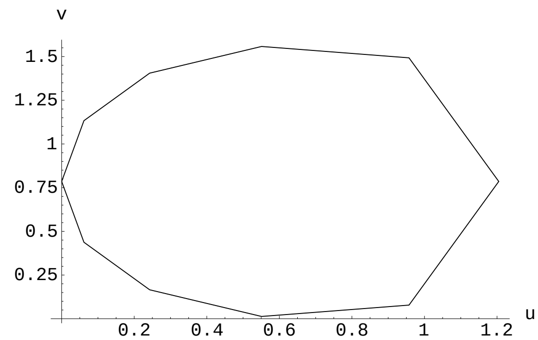

Figure 6. Evolution of contact coordinates.

interspersed between the application of our methodology to the various subtrajectories. Alternatively, manifold "approximations" could be formulated which "smooth" the edges, and the method may be modified to place fingers at points on the surface with some specified maximum curvature which would naturally avoid the edges.

Another area of future work concerns determining a nominal trajectory that has the desired force–closure and stability properties. As presented in this paper, we presume that we are given or can otherwise determine a stable nominal trajectory. The existence of this trajectory is not necessarily guaranteed, however, and, in any case, determining feasible nominal trajectories may prove difficult.

A final area of future work concerns dynamic manipulation. The method upon which ours is based (from [12]) is limited to kinematic systems which can be expressed in the form of Equation 1. Since our method is an adaptation of that in [12], it is similarly limited, and in fact, has this limitation on *each stratum* to which the system may switch. Considering systems with drift on some or all of the strata would greatly broaden the class of systems to which this methodology applies.

References

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

[2] Antonio Bicchi. On the closure properties of robotic grasping. *The International Journal of Robotics Research*, 14(4):319–334, 1995.

[3] N. Brrok, M. Shoham, and J. Dayan. Controllability of grasps and manipulations in multi–fingered hands. *IEEE Transactions on Robotics and Automation*, 14(1):185–192, February 1998.

[4] 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. IEEE, 1993.

[5] 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.

[6] R.S. Fearing. Tactile sensing mechanisms. *The International Journal of Robotics Research*, 9(3):3–23, 1990.

[7] Bill Goodwine and Joel Burdick. Controllability of kinematic systems on stratified configuration spaces. In preparation, available electronically at `http://controls.ame.nd.edu/papers.html`.

[8] Bill Goodwine and Joel Burdick. Trajectory generation for kinematic legged robots. In *IEEE International Conference on Robotics and Automation*, pages 2689–2696. IEEE, 1997.

[9] Bill Goodwine and Joel Burdick. Gait controllability for legged robots. Proceedings of the 1998 International Conference on Robotics and Automation, 1998.

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

[11] 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.

[12] 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.

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

[14] Richard M. Murray, Zexiang Li, and S. Shankar Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Inc., 1994.

[15] Jean-Pierre Serre. *Lie Algebras and Lie Groups*. Springer–Verlag, 1992.

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

[17] Hector J. Sussmann. Lie brackets and local controllability: A sufficient condition for scalar–input systems. *Siam J. Control and Optimization*, 21(5):686–713, 1983.

[18] Hector J. Sussmann. A general theorem on local controllability. *Siam J. Control and Optimization*, 25(1):158–194, 1987.

[19] 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.

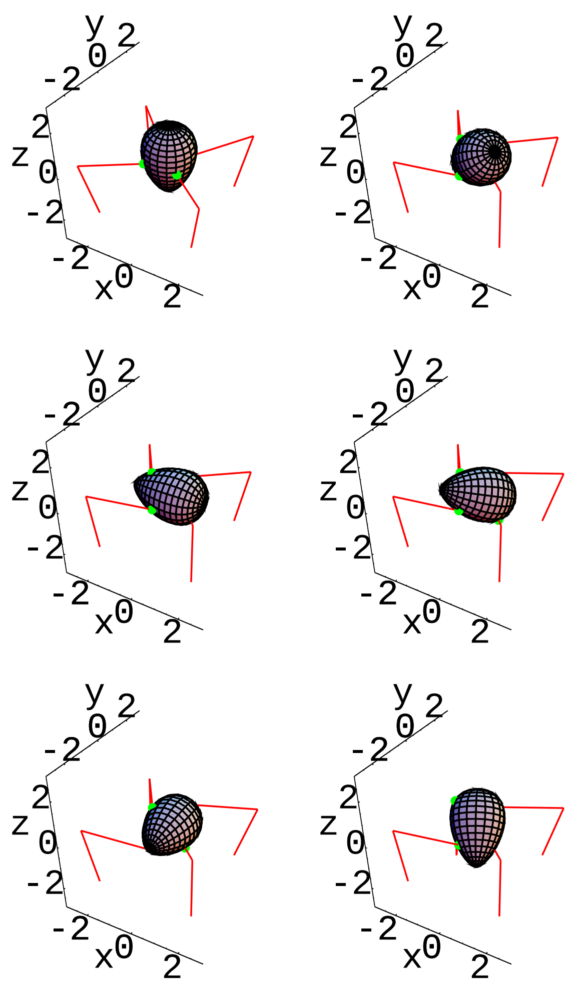

Figure 7. Evolution of manipulation task.