Motion Planning for Kinematically Overconstrained Vehicles Using Feedback Primitives

Todd D. Murphey
Electrical and Computer Engineering
University of Colorado at Boulder
Boulder, Colorado 80309
Email: murphey@colorado.edu

***Abstract*— In this paper we consider motion planning for kinematically overconstrained vehicles. Such vehicles are reasonably common in applications that require many axles for static stability. When a system is kinematically overconstrained, typically some contacts with the environment must slip, violating the constraint. This introduces nonsmooth behavior into the equations of motion, making classical motion planning strategies inapplicable. As an example, we consider a vehicle that has a simplified version of the kinematic structure of the rover from the first Mars mission. We introduce a provably complete motion planner for purposes of illustration. However, the primary purpose of this paper is to clearly identify some of the open problems in motion planning for these mechanisms and to propose a kinematic modeling framework that reveals the underlying complications due to slipping while maintaining the relative simplicity associated with kinematic systems over dynamic ones. The planner we describe has properties that we anticipate will be relevant to a general methodology for motion planning for both kinematically overconstrained systems as well as more general systems that have uncertain dynamics.**

I. Introduction

Mechanisms that experience mechanical contact with their environment are common in robotics. For many systems, design constraints and environmental factors often guarantee that some of these contacts must transition between "stick" and "slip" states. Such transitions introduce uncertainty into the system dynamics, partially due to the transitions themselves and partially due to uncertainties in modeling the contact interface itself. Often these uncertainties take on an explicitly hybrid form (discussed shortly). The question we face is how to create provably complete motion plans in the face of this model uncertainty.

This paper approaches the problem of motion planning with model uncertainty by considering motion planning from the perspective of motion primitives. Although a primitives-based approach has the disadvantage of being inherently heuristic, it often gives insight into what type of structure is needed for a more general motion planning framework.

The example we use is based on a famous example of a kinematically overconstrained system–the Rocky 7 rover, the first rover to travel to Mars. As we will describe shortly, this rover is intrinsically overconstrained, leading to equations of motion that are nondeterministic. However, it is a highly structured uncertain nonlinear system, and we use an analysis approach that allows this structure to be more clear. We believe that the relative simplicity of our approach will have advantages over the other (also correct) more complex analyses of such complicated mechanisms, such as [1].

Vehicles are not the only common overconstrained systems. Similar scenarios of kinematic overconstraint arise in grasping and in multi-point manipulation [2], [3]. In each of these examples actuators can also be out of contact entirely, leading to a discrete change in support. Hence, the ability to incorporate and account for the effects of stick/slip and changing support is inherent to the study of a broad class of mechanisms. Moreover, in each one of these cases the driving force behind the contact changes is environmentally determined, indicating that in an uncertain environment one cannot expect to be able to explicitly predict the changes in contact state based on modeling principles. Therefore, we will consider the contact state as a model uncertainty, and will discuss motion planning with this in mind.

Mechanisms involving mechanical contact, such as the Rocky 7, multiple point manipulation, and slip-steered vehicles, have been studied for many decades. Examples of such devices include autonomous vehicles for both civilian and military purposes [4], "smart" vehicles capable of integrating sensor data for purposes of collision avoidance [5]–[7]. Prior work addressed motion planning [5]–[10], and control [11]–[14] for such devices. All these techniques have implicitly required single-valued, differentiable equations of motion, properties that the mechanisms discussed in this paper do not have.

Overconstrained systems have been studied for a number of years in the context of slip-steered vehicles [15], [16]. These early works developed slip-steering heuristics based on assumptions that all wheels exhibit sliding friction. Geometric properties were typically ignored, thereby ignoring the rich literature on nonholonomic mechanics [17]–[22] that could be applied to these vehicles. Recently, however, with the development of high-visibility vehicles like the Mars rovers, where autonomous, fine-scale motion planning is a requirement, more emphasis has been placed on motion control of overconstrained vehicles [15], [16], [23].

We should note that although motion planning with sensor uncertainty has been a very active area of research for many years [24], [25], relatively little has been done with model uncertainty. This paper is intended to start filling that gap.

This paper is organized as follows. Section II discusses a simplified model of the Rocky 7 rover as an example of a kinematically overconstrained system. This example illustrates many of the difficulties associated with motion planning for such systems as well as the advantages of the proposed methodology. Section III discusses modeling, particularly the modeling of friction and how and why one should avoid it. One of the main points of this paper is that explicit modeling of friction should be avoided because real environmental uncertainties rarely allow one to pick a particular friction model. Instead, one should bound equations of motion and develop the motion planner with these bounds in mind. Given such a model, Section IV discusses how to create a motion planner for such a system. We initially discuss a primitives-based approach, and provide a planner that is sufficiently rich in its structure that one can prove that it is complete. We then discuss how insights from this problem should be applied to more general motion planning strategies, such as the use of Probabilistic Roadmaps (PRMs) and Rapidly Exploring Random Trees (RRTs). We end in Section V with conclusions about this preliminary work and where we think it will lead us over the next few years.

II. Example

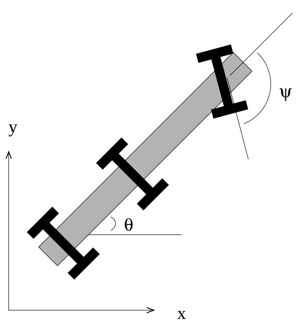

Fig. 1. 1 driven and steered wheel, 2 passive wheels

As an example of an overconstrained mechanism, consider the vehicle shown in Fig. 1 with one front wheel that can turn and two back axles that are fixed. Such a vehicle mimics the kinematic structure of trucks with multiple axles as well as the original Mars rover (the Rocky 7). If we assume that the front wheel always maintains contact with the ground, this vehicle is, at least intuitively, always capable of motion. Let's denote the configuration of this vehicle by $q = [x, y, \theta, \psi, \phi]$ where (x, y, θ) represent the $SE(2)$ configuration of the body, ψ represents the front wheel angle, and ϕ represents the front wheel rolling angle. The constraints are of the form $A(q)\dot{q} = 0$ where each row of A represents a nonholonomic constraint (the form of A can be found in any standard robotics textbook that includes nonholonomic systems–see [12] for example). In this case the only $\dot{q}$ that satisfies this equation are $\dot{q} = [0, 0, 0, 1, 0]^T$ (if $\psi \neq 0$) and $\dot{q} = [Cos[\theta], Sin[\theta], 0, 0, 1]^T$ and $[0, 0, 0, 1, 0]^T$ (if $\psi = 0$). Hence, for $\psi \neq 0$ and $u_1 \neq 0$, this system is not kinematically well-posed. When $\psi = 0$, the constraints are linearly dependent, thus admitting the additional kinematically admissible vector field (that corresponds to moving straight forward). The point is that the degeneracy of the constraints at $\psi = 0$ provides a non-overconstrained (and therefore non-error-accumulating) motion for the vehicle.

We will want to be able to concisely express the dynamics of this system in a way that is amenable to motion planning analysis. The next section discusses how we do this. In particular, we justify the use of the vehicle's *kinematic* description instead of a dynamic description. We do this despite the fact that friction generally introduces dynamic effects into a system's equations of motion. However, some systems, including our example, satisfy the necessary and sufficient conditions for a system to be kinematic even with frictional (and other dissipative) forces acting on it.

III. Modeling

We assume that the systems we are interested in are finite-dimensional simple mechanical systems (as described for smooth systems in [26]). That is, their equations of motion may be found using a Lagrangian of the form kinetic energy minus potential energy ($L = K.E. - V$) along with a set of constraints on the system of the form $\omega(q)\dot{q} = 0$ (or equivalently $A(q)\dot{q} = 0$), where q is the configuration. Moreover, there may be external forces acting on the system. If we ignore potential energy (as is appropriate for many planar systems including the one in Section II), such a system's dynamics may be written down as: $\nabla_{\dot{q}}\dot{q} = u^\alpha Y_\alpha$, where the notation $u^\alpha Y_\alpha$ implies summation over the α. In this expression, ∇ is the constrained affine connection encoding the free kinetic energy and the constraints, in our case the nonslip constraints. Moreover, u represents external forces (not necessarily inputs) and Y represents the associated vector fields on the configuration manifold Q (i.e., $Y \in T_qQ$). If we wish to include potential energy, it will show up as a vector field on the right-hand side of the equation.

The systems of interest have two types of external forces–those that correspond to inputs and those that correspond to external disturbances. In the case of multiple point contact, the external disturbance forces generally correspond to reaction forces due to friction when a contact slips. Therefore, it will be useful to write the dynamic equations as: $\nabla_{\dot{q}}\dot{q} = u^\alpha Y_\alpha + d^\beta V_\beta$ so that we can distinguish between the different types of external forces. (Note that if a constraint is satisfied so that the contact is not slipping, there is still a reaction force. In that case the reaction force is incorporated into the definition of of the constrained affine connection ∇).

Lastly, because the contact state changes over time (as the contacts transition between stick and slip), the constraints change over time. This implies that ∇ is not a single constrained affine connection, but rather comes from a set of constrained affine connections ∇^σ, each of which represents a different set of stick/slip states of the mechanism. The same holds true for Y^σ and V^σ. Hence, if we index the set of possible stick/slip states by σ, we get equations of motion of the following form:

$$\nabla^\sigma_{\dot{q}}\dot{q} = u^\alpha Y^\sigma_\alpha + d^\beta V^\sigma_\beta \qquad (1)$$

where u are input forces and d are external forces. Equation (1) represents the equations of motion for any multiple contact system or overconstrained system mentioned in Section I. (Note that for this equation to make sense, one must assume that the switching signal σ is at least measurable, and often it is assumed that it is piecewise continuous.) Lastly, it is important to point out that the representation $\nabla_{\dot{q}}\dot{q} = u^{\alpha}Y_{\alpha}$ is neither more nor less than the Euler-Lagrange equations [26].

Consider what this modeling methodology implies for a vehicle such as that in Fig. 1. If we apply a torque to the front wheel, the output of the system will depend on which wheels are in contact with the ground and which are sticking or slipping. This, in turn, depends on the mass distribution and normal forces, which even statically may be difficult to model accurately. Therefore, efforts in motion planning and control must take these transitions into account.

A. Kinematic Descriptions of Systems that Slip

We use the affine connection formalism to describe mechanical systems because it is in the context of this formalism that a useful technical connection between 2nd-order mechanical systems and 1st-order kinematic systems has been made (found for smooth systems in [27] and for nonsmooth systems in [28]). Because of the importance that kinematic systems have played in motion planning and control for robotic systems that are not overconstrained, we believe that a well-posed kinematic description of overconstrained systems will be useful in their analysis. That is, it would be useful to be able to write Eq. (1) in the form:

$$\dot{q} = \overline{u}^{a} X^{\sigma}_{a}, \tag{2}$$

where $\overline{u}$ are *velocity* inputs instead of force inputs. Roughly speaking, a system is kinematic if it can be written as a first order differential equation in q without losing any information about what trajectories the system is capable of producing. More precisely, this kinematic description is only useful if it satisfies two requirements. First, for every solution of the dynamic system in Eq. (1) there must exist a kinematic solution of the form in Eq. (2). In the case of a vehicle, this corresponds to requiring that for every *trajectory* of the vehicle there exists a corresponding *path* that can be obtained from kinematic considerations alone. Secondly, for every kinematic solution there must exist a dynamic solution that is equal to the kinematic solution coupled with its time derivative (so that it lies in TQ). This means that there must exist a dynamic solution for every feasible kinematic path. This way of viewing smooth kinematic systems has been studied extensively, including [27], [29]. Motion planning has been studied using these concepts in [30]–[32], but these works were all intended for smooth systems. However, it was shown in [28] that the kinematic reduction of a nonsmooth system of the form in Eq. (1) to one of the form in Eq. (2) is equivalent to the reduction of each smooth model of the multiple model system. The associated algebraic test of kinematic reducibility is that the *symmetric product* between two vector fields Y^{σ}_{i} and Y^{σ}_{j} (defined by $\langle Y^{\sigma}_{i} : Y^{\sigma}_{j} \rangle = \nabla^{\sigma}_{Y^{\sigma}_{i}} Y^{\sigma}_{j} + \nabla^{\sigma}_{Y^{\sigma}_{j}} Y^{\sigma}_{i}$ for given i, j, σ) lie within the distribution of the vector fields and that any reaction forces lie within the span of the input vector fields. That is,

$$\begin{aligned} \langle Y^{\sigma}_{i} : Y^{\sigma}_{j} \rangle &\in span\{Y_i | i = 1, \ldots, m\} \ \forall\ i, j, \sigma \quad (3) \\ V^{\sigma}_{\beta} &\in span\{Y_i | i = 1, \ldots, m\} \ \forall\ \beta, \sigma \quad (4) \end{aligned}$$

Notice that this need only hold for each σ, so the calculation is a smooth calculation, despite the fact that our system is nonsmooth. That is, even with the nonunique solutions these systems can have, one may test for each model independently (i.e., holding σ constant) whether a system is kinematic.

It is important to note that the only assumption made regarding friction is that it creates stick/slip effects. In the context of this paper, no other assumptions are necessary. This is an important consequence of the formal modeling approach just outlined. At the expense of quite a bit of formalism, one can almost completely remove the dependence on the contact interface modeling by merely stating that the contact interface mechanics induce stick/slip transitions. This statement (which originated in [28]) allows one to be reasonably certain that the uncertainties in real-world motion planning problems where the environment is not well-characterized ahead of time will fit within the motion planning schema developed here. Moreover, we will see that a simplified version of the Mars rover is kinematic, even though its wheels must slip against the ground in order for it to turn.

B. Uncertainty Representations

It is additionally worth noting that the contact state enters solely in the σ dynamics in Eq. (1) and (2). Hence, the uncertainty has a hybrid, discrete-valued structure rather than a continuous one like that typically addressed in the robust control community. Moreover, the consequences of this uncertainty can be more drastic; it is often true that choices of σ (in particular dependencies on the configuration q) can lead to equations of motion that have no solution. The primary (and rather conservative) manner in which we can guarantee solutions is by replacing Eq. (1) and (2) with differential inclusions that allow σ to switch arbitrarily quickly. Although this provides a viable notion of solution, it implies that motion planning and stabilization must typically take into account these non-unique solutions.

C. Example Model

We now return to the example in Section II. If the vehicle's location in space can be identified with $SE(2)$, (i.e., $[x, y, \theta]^T$) and the controls u_1 and u_2 are associated with the drive and steering velocities respectively, we may consider the configuration of the total vehicle to be $q = [x, y, \theta, \psi, \phi]^T$, where ψ is the turning angle and ϕ is the drive angle. We would like to be able to show that the set of paths this vehicle can follow are not affected by its dynamic characteristics or by a particular choice of friction model (under reasonable assumptions). That is, we must show that the system is kinematic, even when there are frictional reaction forces due to slipping and discontinuities due to changes in contact.

Let us call the no-sideways-slip constraint associated with the front wheel ω_1, the no-forward-slip constraint associated with the front wheel ω_2, and the no-sideways-slip constraint associated with the back two wheels ω_3 and ω_4 respectively. We assume that the normal load on the front wheel is sufficiently high that the wheel does not slip (i.e., ω_1 and ω_2 are always satisfied)–otherwise the system is trivially uncontrollable because it cannot move if the other contacts are in a "stick" state. Given this assumption, and the fact that the two back wheels are passive (so that they may accommodate any rolling velocity required to not slip in their forward rolling direction), the only switching that can occur is between the middle wheel sliding sideways (i.e., the constraint ω_3 is broken) and the back wheel sliding sideways (i.e., the constraint ω_4 is broken). Note that this only happens when $\psi \neq 0$ and $u_1 \neq 0$. Because these are the only switching signals allowed, the system always satisfies the property found in Eq. (3) for a system to be kinematic, even with the reaction forces due to frictional slipping. That is, this system satisfies $\left\langle Y_i^\sigma : Y_j^\sigma \right\rangle \in span\{Y_i | i = 1, \ldots, m\} \; \forall \; i, j, \sigma$ and $V_\beta^\sigma \in span\{Y_i | i = 1, \ldots, m\} \; \forall \; \beta, \sigma$. See [28] for their description of these calculations.

Hence, the kinematic equations of motion are precisely those kinematics that, for each possible set of constraints that are not overconstrained, annihilate the constraints. In this case there are two such sets, $\Omega_1 = [\omega_1, \omega_2, \omega 3]$, and $\Omega_2 = [\omega_1, \omega_2, \omega_4]$. This gives us kinematic equations of the form in Eq. (2):

$$\dot{q} = g_\sigma(q) u_1 + g_3(q) u_2 \qquad \sigma : (q, t) \rightarrow \{a, b\} \tag{5}$$

$$\begin{aligned} g_a &= \begin{bmatrix} \cos(\psi)\cos(\theta) & \cos(\psi)\sin(\theta) & \frac{1}{l}\sin(\psi) & 0 \end{bmatrix}^T \\ g_b &= \begin{bmatrix} \cos(\psi)\, \cos(\theta) - \frac{r\, \sin(\theta)\, \sin(\psi)}{l+r} \\ \cos(\psi)\, \sin(\theta) + \frac{r\, \cos(\theta)\, \sin(\psi)}{l+r} \\ \frac{1}{l+r}\sin(\psi) \\ 0 \end{bmatrix} \\ g_3 &= \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T \end{aligned}$$

where r is the distance from the back wheel to the middle wheel and l is the distance from the middle wheel to the front wheel. In this description, g_a and g_3 annihilate Ω_1 and g_b and g_3 annihilate Ω_2. As this system evolves, if $\psi = 0$ then $g_a = g_b$, so the system is not overconstrained, and there is therefore no uncertainty due to switching. If $\psi \neq 0$, then at any given time the system either evolves with $\sigma = a$ or $\sigma = b$, thus changing the vector field g_σ. This uncertainty in σ is the uncertainty we will address in a moment in our motion planning algorithm.

IV. Motion Planning

Motion planning for nonlinear systems with model uncertainty like the one being discussed here is nontrivial, but provably complete planners to exist. We now move forward with our description of such a planner based on motion primitives.

A. Controllability Properties of Nonsmooth Systems

Many results in motion planning and stabilization require a system to be *controllable* as a prerequisite. For smooth systems, controllability (and its many flavors) are comparatively well understood. For nonlinear driftless systems, Chow's theorem [33] gives a sufficient condition for small-time local controllability (STLC)–the property that, in a given neighborhood $B_\epsilon(q_0)$ of an initial configuration q_0, the system can get to any final q_f in finite time T without leaving $B_\epsilon(q_0)$. STLC thus guarantees the existence of a motion plan for smooth systems. One may reasonably ask whether an overconstrained mechanism (such as the Mars rover or the slip-steered vehicle) is STLC. Nonsmooth systems, such as those systems discussed here, have only recently had their associated controllability properties studied [23], [34]. With assumptions on the admissible dependence between the switching signal σ and configuration q, an analog to Chow's theorem may be obtained as shown in [23]. In particular, that result shows that the example system in Section II is STLC under some assumption on σ. That is, σ cannot degenerate the controllability properties for the system. This result gives us reason to anticipate that our system is STLC, and we will build our motion planning tools with this assumption in mind.

B. Primitives-based Motion Planning

The use of motion primitives for motion planning is intuitive because it allows us to use physical insight to produce motions. The downside is that primitives almost always must be designed on a case-by-case basis. What is desired is a set of guidelines for how to look for useful primitives and how to prove the completeness of a motion plan.

Consider again the example in Fig.1. The only primitive that is kinematically well-posed for this system is the straight forward motion (because no slipping is required). Any turning motion necessarily accumulates error depending on how the contact state changes over time. These effects can be incorporated into a formally complete motion planner, however. Our goal is to drive the vehicle in Fig. 1 from (x_0, y_0, θ_0) to (x_f, y_f, θ_f) within ϵ relative to a metric defined by $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (\theta_1 - \theta_2)^2}$, the Euclidean metric on $SE(2)$. We will consider three primitives. The first primitive worth noting is that the vehicle can move forward and backward without accumulating any error due to contact changes. The second is the turning primitive–that the vehicle can turn left or right even if we cannot dictate how quickly it turns. This is effectively because switching in the model causes the wheel base to change length unpredictably, thereby changing the radius of curvature that the vehicle is moving around. Lastly, we can stabilize to any line submanifold of $\mathbb{R}^2$ if we are sufficiently well aligned with it initially. That is, if the submanifold is described by $N = (x_0 + s\cos\theta_0, y_0 + s\sin\theta_0)$ for $s \in \mathbb{R}$ and $|\theta - \theta_0|$ is sufficiently small, we can get arbitrarily close to N using a linear control law (described below) that is valid for both models in the multiple model system for the vehicle. These three primitives are enough to produce a motion planner, as follows.

Motion Planning Algorithm for Vehicle in Fig. 1

P1 Turn vehicle until at an angle of $\frac{\pi}{6}$ to the line leaving the point (x_0, y_0) (the vehicle's initial position) at an angle of θ_f (the desired final orientation).

P2 Use the control law

$$\begin{aligned} u_1 &= k \\ u_2 &= -k_1(g^{-1}(q_0 - q_f)) - k_2(\theta - \theta_f) - k_3\phi \end{aligned}$$

until vehicle is within $\epsilon/2$ of the line and within $\frac{\epsilon}{2\sqrt{(x - x_f)^2 + (y - y_f)^2}}$ of the orientation θ_f. Here g is the mapping from the world coordinates to the body coordinates.

P3 Travel on the forward-backward primitive to within $\epsilon/2$ of (x_f, y_f).

We call these three primitives (P1, P2, and P3) "Feedback Primitives" because each one involves some level of feedback in its statement. The first and third primitives (P1 and P3) use feedback to determine when they should terminate, and the second primitive (P2) uses continuous feedback for purposes of stabilization to the manifold N.

Lemma 4.1: The motion plan in the above algorithm has a finite-time admissible solution for every $((x_0, y_0, \theta_0), (x_f, y_f, \theta_f))$ pair.

Proof: We work backwards from (x_f, y_f, θ_f). Define the vehicle coordinates as (x, y, θ) and define the manifold N by $N = (x_f + s\cos\theta_f, y_f + s\sin\theta_f)$ (for $s \in \mathbb{R}$). The distance from a point $(x, y) \in \mathbb{R}^2$ to N is defined by $dist_N((x, y)) = \min_{s\in\mathbb{R}} \|(x, y) - (x_f + s\cos\theta_f, y_f + s\sin\theta_f)\|$. If the vehicle satisfies $dist_N((x, y)) < \epsilon/2$, and $|\theta - \theta_f| < \frac{\epsilon}{2\|(x,y)-(x_f,y_f)\|}$, then the vehicle will be within $\frac{\epsilon}{2}$ after backing up a distance $\|(x, y) - (x_f, y_f)\|$.

Now we need to show that we can make the vehicle satisfy $dist_N((x, y)) < \epsilon/2$ and $|\theta-\theta_f| < \frac{\epsilon}{2\|(x,y)-(x_f,y_f)\|}$. To do so, we use the fact that for $(\theta - \theta_f)$ sufficiently small the linear control law

$$\begin{aligned} u_1 &= k \\ u_2 &= -k_1(g^{-1}(q_0 - q_f)) - k_2(\theta - \theta_f) - k_3\phi \end{aligned}$$

is stable to N for both parts of the multiple model system. (This can be verified using the common Lyapunov function candidate $V = dist_N(\cdot)$). Therefore, because the controlled system is asymptotically stable, there exists a time such that the vehicle will be within $\epsilon/2$ of N and the orientation will be within $\frac{\epsilon}{2\sqrt{(x-x_f)^2+(y-y_f)^2}}$ of the desired orientation θ_f.

Lastly, we need to be able to orient the vehicle so that $|\theta - \theta_f|$ is small (in this case $|\theta - \theta_f| < \frac{\pi}{6}$ works well). However, we can see from the equations of motion that for both models, turning the wheel to some choice of ϕ guarantees that the vehicle will turn. That is, for $\psi \neq 0$ we know that $0 \notin \dot{\theta}$. We know this because zero is not in the convex hull of the models from the multiple model system. That is, either $\dot{\theta} < 0$ or $\dot{\theta} > 0$ for $\psi \neq 0$–hence the vehicle must turn in one direction for both models.

We have shown that for any choice of initial condition (x_0, y_0, θ_0) and final condition $(x_f, y_f, \theta_f))$ this motion plan will bring the vehicle from (x_0, y_0, θ_0) to within ϵ of $(x_f, y_f, \theta_f))$. ∎

Hence, the use of these three primitives provides a planner that is guaranteed to have a solution for any initial condition and any final condition in $SE(2)$, at the expense of requiring feedback as an *encoded part of the primitive definition*. This motion plan does not account for obstacles nor does it account for lack of dead reckoning. It does produce a functional motion planner in the absence of these complexities that accommodates the errors generated by the contact state uncertainty. Moreover, in practice, accounting for obstacles can be achieved reasonably by choosing intermediate waypoints that avoid the obstacles. However, we do not address obstacles here.

Although this approach is effective for this example, currently there are no guidelines for how to generalize this approach. We must be able to show the existence of feedback primitives and, moreover, show how to construct a motion planner based on them. We intend to do this by extending the work on "kinematically decoupled vector fields" [32] by finding conditions under which the decoupled vector fields are invariant with respect to σ dynamics.

C. Simulation

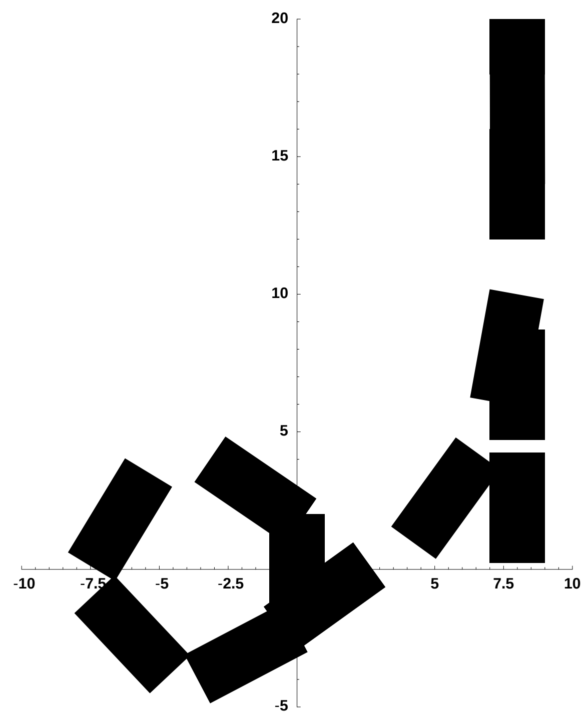

Fig. 2. Simulation (in *Mathematica*) of overconstrained rover-type vehicle driving from $(x_0, y_0, \theta_0) = (0, 0, \frac{\pi}{2})$ to $(x_f, y_f, \theta_f) = (8, 2, \frac{\pi}{2})$

We simulated this system using the equations of motion in Eq. (5) and by having σ_1 be a random variable. We used *Mathematica* for the numerical simulation. Primitive 1 requires that the wheel be turned at some some angle ψ which we chose to be $\frac{\pi}{6}$. We ran the algorithm over many initial conditions and never came across a failure mode. Figure 2 shows snapshots of one simulation that illustrates nicely the functionality (and some shortcomings) of the algorithm. The initial condition was $(x_0, y_0, \theta_0) = (0, 0, \frac{\pi}{2})$ and the final condition was $(x_f, y_f, \theta_f) = (8, 2, \frac{\pi}{2})$. Therefore, $N = (8, s)$ for $s \in \mathbb{R}$. Although the vehicle does eventually arrive at the final destination, it clearly turned the wrong direction at the beginning of the algorithm. However, if the final condition was $(x_f, y_f, \theta_f) = (-8, 2, \frac{\pi}{2})$, the algorithm works much more quickly because the vehicle does not need to turn nearly $2\pi\ rad$ before stabilizing to N. Details such as these can be fixed in more sophisticated motion planners–our purpose here is to show that a motion planner can indeed be designed that incorporates the effects of uncertainty in contact state (and possibly other uncertainties) while guaranteeing the termination of the planner.

V. CONCLUSIONS

In this paper we introduce both a class of interesting motion planning problems and give a preliminary solution we believe will be relevant to a reasonably wide range of mechanisms. Despite the heuristic nature of motion planning using primitives, these primitives give us insight into the general structure one should look for. Loosely speaking, we believe that this structure is essentially that each primitive should at least act predictably on a submanifold of the configuration manifold. In the case of our example, Primitive 1 (P1) acts predictably on the manifold described by the orientation θ. Primitive 2 (P2) acts predictably relative to the manifold N by stabilizing to it. Primitive 3 (P3) acts predictably *on* N by just backing down it. In some sense, this is similar to the work on manipulation in [35] in that it just requires that the uncertainty be reduced with each step in the plan.

It is also worth noting that although we do use feedback, it is not our goal to create a "stabilized" system. Instead, we see feedback as functioning purely to make the statement of the motion primitives well-posed. In this way we avoid many of the somewhat superficial concerns with global convergence of nonlinear systems while simultaneously taking advantage of the uncertainty mitigation feedback provides.

Future work will include the encoding of these primitives into a Probabilistic Roadmap setting and the development of numerical tools for automatically calculating "feedback primitives." Although this work is certainly preliminary, it seems likely that this approach will eventually lead to substantial improvements in motion planning for more general systems with uncertain dynamics.

REFERENCES

[1] M. Tarokh and G. McDermott, "Kinematics modeling and analyses of articulated rovers," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 539–553, 2005.

[2] K. Böhringer and H. Choset, Eds., *Distributed Manipulation*. Kluwer, 2000.

[3] J. Luntz, W. Messner, and H. Choset, "Closed-loop distributed manipulation using discrete actuator arrays," in *Workshop on the Algorithmic Foundations of Robotics (WAFR)*, 2000.

[4] R. Vidal, O. Shakernia, H. Kim, D. Shim, and S. Sastry, "Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation," *IEEE Tran. on Robotics and Automation*, vol. 18, no. 5, pp. 662–669, Oct. 2002.

[5] L. Yang and S. LaValle, "The sampling-based neighborhood graph: An approach to computing and executing feedback motion strategies," *IEEE Tran. on Robotics and Automation*, vol. 20, no. 3, pp. 419– 432, June 2004.

[6] H. Choset, K. Nagatani, and N. Lazar, "Topological simultaneous localization and mapping (slam): toward exact localization without explicit localization," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 2, pp. 125–137, Apr. 2001.

[7] A. Ladd and L. Kavraki, "Measure theoretic analysis of probabilistic path planning," *IEEE Tran. on Robotics and Automation*, vol. 20, no. 2, pp. 229–242, April 2004.

[8] F. Bullo and K. Lynch, "Kinematic controllability and decoupled trajectory planning for underactuated mechanical systems," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2001, pp. 3300–3307.

[9] S. Roumeliotis and G. Bekey, "Distributed multirobot localization," *IEEE Tran. on Robotics and Automation*, vol. 18, no. 5, pp. 781–795, Oct. 2002.

[10] F. Bullo, N. E. Leonard, and A. D. Lewis, "Controllability and motion algorithms for underactuated lagrangian systems on lie groups," *IEEE Transactions on Automatic Control*, vol. 45, no. 8, pp. 1437–1454, Aug. 2000.

[11] R. Featherstone, "Modeling and control of contact between constrained rigid bodies," *IEEE Tran. on Robotics and Automation*, vol. 20, no. 1, pp. 82–92, Feb. 2004.

[12] R. Murray, Z. Li, and S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[13] R. Murray and S. Sastry, "Nonholonomic motion planning: Steering using sinusoids," *IEEE Transactions on Automatic Control*, vol. 38, pp. 700–716, 1993.

[14] ——, "Steering nonholonomic systems in chained form," in *Proc. Int. Conf. on Decision and Control (CDC)*, 1991, pp. 1121–1126.

[15] J. C. Alexander and J. H. Maddocks, "On the kinematics of wheeled mobile robots," *The International Journal of Robotics Research*, vol. 8, no. 5, pp. 15–27, October 1989.

[16] K. Waldron and C. Hubert, "Control of contact forces in wheeled and legged off-road vehicles," in *Preprints, 6th Symp. on Experimental Robotics*, Syndney, Australia, March 1999, pp. 265–274.

[17] W. Koon and J. Marsden, "The geometric structure of nonholonomic mechanics," in *Proc. Int. Conf. on Decision and Control*, 1997, pp. 2356–2361.

[18] J. Marsden and T. Ratiu, *An Introduction to Mechanics and Symmetry*. Springer–Verlag, 1992.

[19] C. Lanczos, *The Variational Principles of Mechanics*. Dover, 1986.

[20] A. Bloch, P. S. Krishnaprasad, J. Marsden, and R. Murray, "Nonholonomic mechanical systems with symmetry," *Arch. Rational Mech. Anal.*, vol. 136, no. 1, pp. 21–99, 1996.

[21] I. Kolmanovsky and N. H. McClamroch, "Developments in nonholonomic control problems," *IEEE Control Systems Magazine*, pp. 20–36, December 1995.

[22] A. Teel, R. Murray, and G. Walsh, "Nonholonomic control systems: From steering to stabilization with sinusoids," *Int. J. of Control*, vol. 62, no. 4, pp. 849–870, 1995.

[23] T. D. Murphey and J. W. Burdick, "A controllability test for multiple model systems," in *Proc. IEEE American Controls Conference (ACC)*, Anchorage, Alaska, 2002.

[24] D. R. B.R. Donald, J. Jennings, *Algorithmic Foundations of Robotics (WAFR)*. A.K. Peters, Ltd, Wellesley, MA, 1995, ch. Information invariants for distributed manipulation, pp. 431–459.

[25] J. Latombe, *Robot Motion Planning*. Kluwer Academic Press, 1991.

[26] A. Lewis, "Simple mechanical control systems with constraints," *IEEE Transactions on Automatic Control*, vol. 45, no. 8, pp. 1420–1436, 2000.

[27] ——, "When is a mechanical control system kinematic?" in *Proc.* 38^{th} IEEE *Conf. on Decision and Control*, Dec. 1999, pp. 1162–1167.

[28] T. D. Murphey and J. W. Burdick, "Kinematic reducibility for multiple model systems," in *Proc. IEEE Conf. on Decision and Control (CDC)*, Maui, Hawaii, 2003, pp. 5307–5313.

[29] F. Bullo and A. Lewis, "Low-order controllability and kinematic reductions for affine connection control systems," *To appear in SIAM Journal on Control and Optimization*, 2004.

[30] K. Lynch, N. Shiroma, H. Arai, and K. Tanie, "Collision-free trajectory planning for a 3-dof robot with a passive joint," *International Journal of Robotics Research*, vol. 19, no. 12, pp. 1171–1184, December 2000.

[31] F. Bullo, A. Lewis, and K. Lynch, "Controllable kinematic reductions for mechanical systems: Concepts, computational tools, and examples," *Int Symp on Math. Theory of Networks and Systems (MTNS)*, August 2002.

[32] F. Bullo and K. M. Lynch, "Kinematic controllability for decoupled trajectory planning in underactuated mechanical systems," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 4, pp. 402–412, August 2001.

[33] W. Chow, "Über systeme von linearen partiellen differentialgeichungen erster ordinung," *Math Ann.*, vol. 117, pp. 98–105, 1939.

[34] F. Rampazzo and H. J. Sussmann, "Set-valued differentials and a nonsmooth version of Chow's theorem," in *Proc. of 40th Conf. Decision Control*, 2001.

[35] M. Erdmann and M. Mason, "An exploration of sensorless manipulation," *IEEE Journal of Robotics and Automation*, vol. 4, no. 4, 1988.