Stochastic Complementarity for Local Control of Discontinuous Dynamics

Yuval Tassa
Interdisciplinary Center for Neural Computation
Hebrew University
tassa@alice.nc.huji.ac.il

Emo Todorov
Computer Science and Engineering
University of Washington
todorov@cs.washington.edu

Abstract— **We present a method for smoothing discontinuous dynamics involving contact and friction, thereby facilitating the use of local optimization techniques for control. The method replaces the standard Linear Complementarity Problem with a Stochastic Linear Complementarity Problem. The resulting dynamics are continuously differentiable, and the resulting controllers are robust to disturbances. We demonstrate our method on a simulated 6-dimensional manipulation task, which involves a finger learning to spin an anchored object by repeated flicking.**

I. Introduction

Classic control methods focus on forcing a dynamical system to a reference trajectory. This approach is powerful but limited. For complex behaviors in underactuated domains, planning the desired trajectory cannot be easily separated from the control strategy. Optimal Control offers a comprehensive framework that solves both the planning and control problems by finding a policy which minimizes future costs. However, global methods for finding optimal policies scale exponentially with the state dimension, making them prohibitively expensive.

Local Optimal Control methods, or *trajectory optimizers*, play a key role in the search for nonlinear feedback controllers that are on par with biological motor systems. These methods find a solution in a small part of the state-space, but because their complexity scales polynomially, they may constitute the only feasible approach to tackling high-dimensional problems.

Locomotion and *hand-manipulation*, behaviors which encompass some of the most interesting control problems, are crucially dependent on contact and friction. These phenomena pose a problem for efficient variants of local methods, which require differentiability to some order. Hard contacts and joint limits are examples of dynamic discontinuities where velocities change instantaneously upon impact. Though these two phenomena can conceivably be modeled with stiff nonlinear spring-dampers, friction is inherently discontinuous and does not readily admit a smooth approximation.

If our goal is to produce an accurate simulation, it might be sensible to use a deterministic model of the dynamics, since macro-physical systems are often well-modeled as such. If however we wish to control the system, *noise* might qualitatively alter the optimal behavior. We argue that when the dynamics are discontinuous, an optimal controller must account for stochasticity in order to generate an acceptable policy. Because stochastic dynamics are inherently smooth (in the mean), differentiability issues automatically disappear.

A popular and established method for modeling hard unilateral constraints and friction, involves defining *complementarity conditions* on forces and velocities. Simulators based on this principle are called *time-stepping integrators* and solve a Linear Complementarity Problem at each time step. We propose to instead solve a Stochastic Linear Complementarity Problem, using the approach proposed in [1]. This method transforms the complementarity problem into a smooth nonlinear optimization which is readily solved.

The solution effectively describes new deterministic dynamics, that implicitly take into account noise around the contact. These modified dynamics can qualitatively be described as featuring a fuzzy “force-field” that extends out from surfaces, allowing both contact and friction to act at a distance. The size and shape of this layer are naturally determined by the noise distribution, without any free parameters.

We test our method on a simplified manipulation task. A two link planar finger must learn to spin an ellipse that is anchored to the wall by flicking at it. We use an off-line Model Predictive Control strategy to patch together a global, time-independent policy, which robustly performs the task for both the smoothed system and the original discontinuous one. We constructed an interactive simulation which allows the user to actively perturb the controlled system. The controller proved to be robust, withstanding all these disturbances.

II. A Qualitative Argument

Consider the simple and familiar task of holding an object. The force which prevents the object from falling is friction, related by Coulomb’s law to the normal force exerted by the fingers. If we now slowly loosen our grip, there is no discernable change in the positions or velocities of the system until suddenly, when the the weight of the object penetrates the friction cone, sticking changes into slipping and the object drops from our hand.

In the Optimal Control context, the control signal realizes a tradeoff between a state-cost and a control-cost. Because the state-cost cannot change until the object begins to slip, a controller that is optimal with respect to deterministic dynamics will attempt to hold the object with the *minimum possible force*, i.e. on the very edge of the friction cone.

This delicate grip would be disastrously fragile – the smallest perturbation would cause the object to fall. If, however, there is uncertainty in our model, we can only maintain our grip *in probability*. By grasping with more force than is strictly necessary, we push probability-mass from the slip-regime into the stick-regime.

The moral is that optimal control of discontinuous dynamics cannot give acceptable results with a deterministic model. Our intuition of what constitutes a reasonable solution implicitly contains the notion of robustness, which requires the explicit modeling of noise near the discontinuities.

III. Background

A. Dynamics with unilateral contacts

The modeling and simulation of multibody systems with contacts and friction is a broad and active field of research [2]. One reasonable approach, not investigated here, is to model discontinuous phenomena with a continuous-time *hybrid* dynamical system which undergoes switching [3]. These methods require accurate resolution of collision and separation times, so fixed time-step integration with a fixed computational cost is impossible. Moreover, collision events can in principle occur infinitely often in a finite time, as when a rigid elastic ball bounces to rest. The appropriate control strategy would be to chop the trajectory into several first-exit problems, where contact surfaces serve as an exit manifold for one segment, while the post-collision state serves as an initial condition for the next one. It is not clear how a trajectory-optimizer for such a system could deal with unforeseen changes to the switch sequence, as when a foot makes grazing contact. Furthermore, such a scheme would mandate that every contact/stick-slip *configuration* be considered a hybrid system component, so their number would grow combinatorially.

Time-stepping avoids all of these problems. We give a brief exposition here and refer the interested reader to [4]. The equations of motion of a controlled mechanical system are

$$\begin{aligned}\mathbf{M}\dot{\mathbf{v}} &= \mathbf{r} + \mathbf{u}\\ \dot{\mathbf{q}} &= \mathbf{v}\end{aligned}$$

where $\mathbf{q}$ and $\mathbf{v}$ are respectively the generalized coordinates and velocities, $\mathbf{M} = \mathbf{M}(\mathbf{q})$ is the mass matrix, $\mathbf{r} = \mathbf{r}(\mathbf{q}, \mathbf{v})$ the vector of total external forces (gravity, drag, centripetal, coriolis etc.) and $\mathbf{u}$ is the applied control signal (e.g. motor torques).

It is non-trivial to model discontinuous phenomena like rigid contact and Coulomb friction in continuous time. Doing so requires instantaneous momentum transfers via unbounded forces and the formalism of Measure Differential Inclusions to fully characterize solutions [5]. Time-stepping integrators rewrite the equations of motion in a discrete-time momentum-impulse formulation. Because the integral of force (the impulse) is always finite, unbounded quantities are avoided, and both *bounded* and *impulsive* forces can be properly addressed in one step.

For a timestep h, an Euler integration step for the momenta[1] $\mathbf{M}\mathbf{v}' = \mathbf{M}\mathbf{v}(t+h)$ and coordinates $\mathbf{q}' = \mathbf{q}(t+h)$ is:

$$\begin{aligned}\mathbf{M}\mathbf{v}' &= h(\mathbf{r} + \mathbf{u}) + \mathbf{M}\mathbf{v}\\ \mathbf{q}' &= \mathbf{q} + h\mathbf{v}'.\end{aligned}$$

We want a unilateral constraint vector function $\mathbf{d}(\mathbf{q})$ to remain non-negative, for example a *signed distance* between objects, which is positive for separation, zero for contact and negative for penetration. We therefore demand that impulses λ be applied so that

$$\mathbf{d}(\mathbf{q}') \approx \mathbf{d}(\mathbf{q}) + h\mathbf{J}\mathbf{v}' \geq 0,$$

where $\mathbf{J} = \nabla\mathbf{d}(\mathbf{q})$. This leads to the following complementarity problem for $\mathbf{v}'$ and λ:

$$\mathbf{M}\mathbf{v}' = h(\mathbf{r} + \mathbf{u}) + \mathbf{M}\mathbf{v} + \mathbf{J}^{\mathsf{T}}\lambda \tag{1a}$$

$$\lambda \geq 0, \tag{1b}$$

$$\mathbf{d}(\mathbf{q}) + h\mathbf{J}\mathbf{v}' \geq 0, \tag{1c}$$

$$\lambda^{\mathsf{T}}(\mathbf{d}(\mathbf{q}) + h\mathbf{J}\mathbf{v}') = 0. \tag{1d}$$

Conditions (1b) and (1c) are read element-wise, and respectively constrain the contact impulse to be non-adhesive, and the distance to be non-penetrative. Condition (1d) asserts that $\mathbf{d}(\mathbf{q}') > 0$ (broken contact) and $\lambda > 0$ (collision impact), are mutually exclusive.

Since the mass matrix is always invertible, we can solve (1a) for $\mathbf{v}'$, and plug into (1c). Defining

$$\mathbf{A} = \mathbf{J}\mathbf{M}^{-1}\mathbf{J}^{\mathsf{T}} \tag{2a}$$

$$\mathbf{b} = \mathbf{d}(\mathbf{q})/h + \mathbf{J}\mathbf{v} + h\mathbf{J}\mathbf{M}^{-1}(\mathbf{r} + \mathbf{u}), \tag{2b}$$

we can now write (1) in standard LCP form:

$$\text{Find } \lambda \quad s.t. \quad 0 \leq \lambda \;\perp\; \mathbf{A}\lambda + \mathbf{b} \geq 0. \tag{3}$$

In order to solve for frictional impulses λ_f, the Coulomb friction law $\|\lambda_f\| \leq \mu\lambda$ must be incorporated. To maintain linearity, a polyhedral approximation to the friction cone is often used, though some new methods can handle a smooth one [7]. In the model of [8], the matrix $\mathbf{A}$ is no longer positive-definite, though a solution is guaranteed to exist. In the relaxed model of [9], $\mathbf{A}$ remains positive-definite. With either model the problem retains the LCP structure (3).

B. Smoothing methods for LCPs

The Linear Complementarity Problem (3) arises in many contexts and has stimulated considerable research [10][11]. One method of solving LCPs involves the use of a so-called NCP function $\phi(\cdot, \cdot)$ whose root satisfies the complementarity condition

$$\phi(a, b) = 0 \quad \Longleftrightarrow \quad a \geq 0,\ b \geq 0,\ ab = 0.$$

Due to the element-wise nature of complementarity, is suffices to consider NCP functions with scalar arguments. Two popular examples are

$$\phi(a, b) = \min(a, b) \tag{4}$$

[1] In the derivation we assume $\mathbf{M}$ and $\mathbf{r}$ to be constant throughout the time-step, but for our simulations used a more accurate approximation [6].

and

$$\phi(a,b) = a + b - \sqrt{a^2 + b^2}.$$

These functions reformulate the complementarity problem as a system of (possibly nonsmooth) nonlinear equations, whose residual can then be minimized. A particular method, closely related to the one used here, was presented by Chen and Mangasarian in [12] and proceeds as follows. Rewrite (4) as $\min(a,b) = a - \max(a-b,0)$ and replace $\max(\cdot,0)$ with a smooth approximation

$$s(x,\epsilon) \ \rightarrow \ \max(x,0) \quad \text{as} \quad \epsilon \downarrow 0.$$

One of several such functions proposed there is

$$s(x,\epsilon) = \epsilon \log(1 + e^{x/\epsilon}). \tag{5}$$

The authors then present an iterative algorithm whereby for a positive ϵ, the squared residual

$$r(a,b,\epsilon) = \big(a - s(a-b,\epsilon)\big)^2 \tag{6}$$

is minimized with a standard nonlinear minimization technique, ϵ is subsequently decreased, and the procedure repeated until a satisfactory solution is obtained.

C. Local optimization

Local Optimal Control methods, with roots in the venerable Maximum Principle [13], solve the control problem along a single *trajectory*, giving either an open-loop or a closed-loop policy that is valid in some limited volume around it. They are efficient enough to be used for real-time control of fast dynamics [14], and scale well enough to handle high-dimensional nonlinear mechanisms [15]. These methods can get stuck in local minima, though if neural controllers set the golden standard, it might be noted that biological suboptimal minima exist, somewhat anecdotally evidenced by the high-jump before Dick Fosbury. Their main detraction is that they solve the problem in only a small volume of space, namely around the trajectory. To remedy this, new methods [16] use Sum of Squares verification tools to measure the size of local basins-of-attraction and constructively patch together a global feedback controller. Finally, using the control-estimation duality, the powerful framework of estimation on graphical models is being brought to bear [17] on trajectory optimizers.

Though policy gradient methods like *shooting* can also be considered, we concentrate on Local Dynamic Programming, i.e. the construction of a local approximation to the Value function, and restrict ourselves to the finite-horizon case.

The control $\mathbf{u} \in \mathbb{R}^m$ affects the propagation of the state $\mathbf{x} \in \mathbb{R}^n$ through the general Markovian dynamics

$$\mathbf{x}' = \mathbf{f}(\mathbf{x},\mathbf{u}). \tag{7}$$

The cost-to-go starting from state $\mathbf{x}$ at time i with a control sequence $\mathbf{u}_{i:N-1} \equiv \{\mathbf{u}_i, \mathbf{u}_{i+1} \ldots, \mathbf{u}_{N-1}\}$, is the sum of running costs[2] $\ell(\mathbf{x},\mathbf{u})$ and final cost $\ell_f(\mathbf{x})$:

$$J_i(\mathbf{x}_i, \mathbf{u}_{i:N-1}) = \sum_{k=i}^{N-1} \ell(\mathbf{x}_k, \mathbf{u}_k) + \ell_f(\mathbf{x}_N),$$

We define the optimal Value function at time i as the the cost-to-go given the minimizing control sequence

$$V_i^*(\mathbf{x}) \equiv \min_{\mathbf{u}_{i:N-1}} J_i(\mathbf{x}, \mathbf{u}_{i:N-1}).$$

Setting $V_N^*(\mathbf{x}) \equiv \ell_f(\mathbf{x}_N)$, the Dynamic Programming principle reduces the minimization over the entire sequence of controls to a sequence of minimizations over a single control, proceeding backwards in time:

$$V_i^*(\mathbf{x}) = \min_{\mathbf{u}} [\ell(\mathbf{x},\mathbf{u}) + V_{i+1}^*(\mathbf{f}(\mathbf{x},\mathbf{u}))] \tag{8}$$

1) First-Order Dynamic Programming: To derive a discrete-time equivalent of the Maximum Principle, we observe the following: Given a first-order approximation of the Value at $i+1$, if $\mathbf{f}$ is affine in $\mathbf{u}$ (which holds for mechanical systems) and ℓ is convex and smooth in $\mathbf{u}$ (so that $\nabla_{\mathbf{u}}\ell$ is invertible), then the minimizing $\mathbf{u}$ is given by:

$$\mathbf{u}_i^* = -\nabla_{\mathbf{u}} \ell^{-1} \left(\nabla_{\mathbf{u}} \mathbf{f}^\mathsf{T} \nabla_{\mathbf{x}} V_{i+1} \right) \tag{9}$$

with dependencies on $\mathbf{x}$ and $\mathbf{u}$ suppressed for readability. Once $\mathbf{u}_i^*$ is known, the approximation at time i is given by

$$\nabla_{\mathbf{x}} V_i(\mathbf{x}) = \nabla_{\mathbf{x}} \big(\ell(\mathbf{x}, \mathbf{u}_i^*) + V_{i+1}(\mathbf{f}(\mathbf{x}, \mathbf{u}_i^*)) \big). \tag{10}$$

The first-order local dynamic programming algorithm proceeds by alternatingly propagating the dynamics forward with (7), and propagating $\mathbf{u}_i$ and $\nabla_{\mathbf{x}} V_i(\mathbf{x})$ backward with (9) and (10).

2) Second-Order Dynamic Programming: By propagating a quadratic model of $V_i(\mathbf{x})$, second-order methods can compute locally-linear policies. These provide both quadratic convergence rate and a more accurate, closed-loop controller. We define the *unminimized* Value function

$$Q_i(\delta\mathbf{x}, \delta\mathbf{u}) = \ell(\mathbf{x} + \delta\mathbf{x}, \mathbf{u} + \delta\mathbf{u}) + V_{i+1}(\mathbf{f}(\mathbf{x} + \delta\mathbf{x}, \mathbf{u} + \delta\mathbf{u}))$$

and expand to second order

$$\approx \frac{1}{2} \begin{bmatrix} 1 \\ \delta\mathbf{x} \\ \delta\mathbf{u} \end{bmatrix}^\mathsf{T} \begin{bmatrix} Q_0 & Q_x{}^\mathsf{T} & Q_u{}^\mathsf{T} \\ Q_x & Q_{xx} & Q_{xu} \\ Q_u & Q_{ux} & Q_{uu} \end{bmatrix} \begin{bmatrix} 1 \\ \delta\mathbf{x} \\ \delta\mathbf{u} \end{bmatrix}. \tag{11}$$

Solving for the minimizing $\delta\mathbf{u}$ we have

$$\delta\mathbf{u}^* = \operatorname*{argmin}_{\delta\mathbf{u}} [Q_i(\delta\mathbf{x}, \delta\mathbf{u})] = -Q_{uu}^{-1}(Q_u + Q_{ux}\delta\mathbf{x}), \tag{12}$$

giving us both open-loop and linear-feedback control terms. This control law can then be plugged back into (11) to obtain a quadratic approximation of V_k. As in the first-order case, these methods proceed by alternating a forward pass which propagates the dynamics using the current policy, and a backward pass which re-estimates the Value and produces a new policy.

[2] The possible dependence of ℓ on i is suppressed for compactness.

D. Optimal Control of discontinuous systems

To date, there has not been a profusion of research on Optimal Control of discontinuous dynamics. Stewart and Anitescu's recent contribution [18] includes a literature review. In that paper, the authors describe a method whereby a smooth approximation replaces the discontinuities. They show that in some limit, the solution of the smooth system converges to the solution of the nonsmooth one, and proceed to apply their method to the "Michael Schumacher" racing car problem. We would argue that though a solution is obtained, it might be a solution to the wrong problem. Qualitatively, it reaches the very limits of tire traction, and passes within a whisker of the walls. The smallest disturbance would send the car crashing. Their example is appropriate since this extreme driving well-describes the car racing profession, but the optimum is clearly non-robust. A driver who thinks that the road is slippery or the steering wheel is inaccurate, would most likely be considered a "better driver" by other standards.

The method presented in the next section describes a specific way of accounting for stochasticity when controlling dynamics with complementarity conditions, but in general, noise has a profound effect on optimal solutions. For deterministic continuous-time systems, even infinitely differentiable ones, the Value function, which satisfies the Hamilton Jacobi Bellman PDE, is often discontinuous. A solution always exists in the *Viscosity Solution* sense [19], but the fact that a special formalism is required is conspicuous. If, however, the dynamics are a stochastic diffusion with positive-definite noise covariance, the HJB equation gains a second-order Itô term, and the solution is always unique and smooth, regardless of discontinuities in the underlying dynamics.

IV. The Proposed Method

Instead of solving for contact impulses with an LCP, we propose instead to solve a Stochastic LCP. The incorporation of stochasticity makes the contact impulses a differentiable function of the state, facilitating the use of local control methods. Because a controller must always overcome noise (process, observation, modeling), the noise-induced smooth dynamics constitute a better model for control purposes, even if they are a worse approximation of the real physical mechanism.

A. New and old formulas for SLCPs

The study of Stochastic Linear Complementarity Problems is a fairly recent endeavor, see [20] for a survey. The general problem could be written

$$0 \le \mathbf{x} \;\perp\; \mathbf{A}(\omega)\mathbf{x} + \mathbf{b}(\omega) \ge 0. \tag{13}$$

Where ω is random variable. In this form the problem is obviously not well-posed, since it is not clear in what sense $\mathbf{x}$ satisfies the constraints. The Expected Residual Minimization approach of [1], proposes that we minimize

$$r_{\text{ERM}}(\mathbf{x}) = \mathbf{E}\left[\|\Phi(\mathbf{x},\omega)\|^2\right] \tag{14}$$

where

$$\Phi(\mathbf{x},\omega)_i = \phi\Big(\mathbf{x}_i,\ \big(\mathbf{A}(\omega)\mathbf{x}+\mathbf{b}(\omega)\big)_i\Big)$$

is a vector of NCP residuals.

As detailed in section IV-B, in our case $\mathbf{A}$ can be considered constant to first order. This variant is discussed in section 3 of [1], where after a simple proof that $r_{\text{ERM}}(\mathbf{x})$ is continuously differentiable, it is explicitly computed for a uniformly distributed $\mathbf{b}$. Because we are ultimately trying to model a diffusion, the Normal distribution would be more appropriate. Considering scalar arguments a and $b_{\mathcal{N}}$, with

$$b_{\mathcal{N}} \sim p(b_{\mathcal{N}}) = \mathcal{N}(b_{\mathcal{N}}|b,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\tfrac{1}{2}\left(\tfrac{b_{\mathcal{N}}-b}{\sigma}\right)^2\right)$$

and cumulative distribution

$$P(b_{\mathcal{N}}) = \int_{-\infty}^{b_{\mathcal{N}}} p(t)dt = \frac{1}{2}\left(1+\text{erf}\left(\tfrac{b_{\mathcal{N}}-b}{\sigma\sqrt{2}}\right)\right),$$

the expectation produces

$$\mathbf{E}\left[\min(a,b_{\mathcal{N}})^2\right] = \\ a^2 - \sigma^2(a+b)p(a) + \left(\sigma^2+b^2-a^2\right)P(a). \tag{15}$$

A possible variant of equation (14), would be to exchange squaring and expectation, giving what might be termed Residual Expectation Minimization:

$$r_{\text{REM}}(\mathbf{x}) = \|\,\mathbf{E}\left[\Phi(\mathbf{x},\omega)\right]\|^2 \tag{16}$$

Due to Jensen's inequality, $r_{\text{REM}}(\mathbf{x}) \le r_{\text{ERM}}(\mathbf{x})$, and is thus a weaker bound, though their minima coincide as $\sigma \downarrow 0$. A benefit of this residual is that the expectation gives simpler formulae. In particular, if $b_{\mathcal{L}}$ has a Logistic distribution

$$b_{\mathcal{L}} \sim \mathcal{L}(b_{\mathcal{L}}|b,\sigma) = \frac{1}{\sigma}\Big(\exp\left(\tfrac{b_{\mathcal{L}}-b}{2\sigma}\right) + \exp\left(\tfrac{b-b_{\mathcal{L}}}{2\sigma}\right)\Big)^{-2}$$

then the expectation produces

$$\mathbf{E}[\min(a,b_{\mathcal{L}})] = a - \sigma\log(1+e^{\frac{a-b}{\sigma}}). \tag{17}$$

It is clear that this residual is identical to (6) with the smoothing function (5), where σ takes the place of ϵ. This immediately provides us with a new interpretation to the method of [12], and gives us access to the literature which investigates it (e.g. [21]).

With these smooth approximations[3] to ϕ, the minimizing $\mathbf{x}$ is now a differentiable function of $\mathbf{A}$ and $\mathbf{b}$.

[3] Although we only used formulae (15) and (17) in our experiments, for completeness we also provide the expectation of the $\min(\cdot,\cdot)$ function with a Normally distributed argument:

$$\mathbf{E}[\min(a,b_{\mathcal{N}})] = a - \sigma^2 p(a) - (a-b)P(a),$$

and of $\min(\cdot,\cdot)^2$ with a Logistically distributed argument:

$$\mathbf{E}[\min(a,b_{\mathcal{L}})^2] = a^2 - 2a\sigma\log(1+e^{\frac{a-b}{\sigma}}) + 2\sigma^2\,\text{Li}_2(-e^{\frac{a-b}{\sigma}}),$$

where $\text{Li}_2(x)$ is the Dilogarithm function $\text{Li}_2(x) = -\int_x^0 \frac{\log(1-t)}{t}dt$.

B. Determining the noise covariance

How should the noise covariance σ be determined? Assume that a gaussian state distribution is estimated from observations by some filter, and we are given

$$\Sigma_{\mathbf{q}} = \mathbf{E}[\mathbf{q}\mathbf{q}^{\mathsf{T}}] \quad \text{and} \quad \Sigma_{\mathbf{v}} = \mathbf{E}[\mathbf{v}\mathbf{v}^{\mathsf{T}}].$$

Examining (2a) we see that $\mathbf{A}$ is a function of the mass matrix $\mathbf{M}$ and the constraint Jacobian $\mathbf{J}$. Both of these depend on $\mathbf{q}$ but often smoothly and slowly. In contrast, since h must be small due to the Euler integration, its presence in the denominator of the first term of the RHS of (2b), suggests that the appropriate first-order approximation is

$$\Sigma_{\mathbf{b}} = \mathbf{E}[\mathbf{b}\mathbf{b}^{\mathsf{T}}] \approx \mathbf{J}\Sigma_{\mathbf{q}}\mathbf{J}^{\mathsf{T}}h^{-2}$$

To account for the second term as well, one would use

$$\Sigma_{\mathbf{b}} \approx \mathbf{J}(\Sigma_{\mathbf{q}}h^{-2} + \Sigma_{\mathbf{v}})\mathbf{J}^{\mathsf{T}}.$$

The element-wise nature of the complementarity conditions allows us to ignore the off-diagonal terms and use

$$\mathbf{b}(\omega)_i \sim \mathcal{N}(\omega|\mathbf{b}_i, \sqrt{(\Sigma_{\mathbf{b}})_{ii}}) \quad \text{or} \quad \mathcal{L}(\omega|\mathbf{b}_i, \sqrt{(\Sigma_{\mathbf{b}})_{ii}}), \tag{18}$$

to define the vector residual as

$$\Phi(\mathbf{x}, \omega)_i = \phi\Big(\mathbf{x}_i,\ \mathbf{A}_i\mathbf{x} + \mathbf{b}(\omega)_i\Big).$$

C. Making global controllers from trajectories

The output of the second-order trajectory optimizer of section III-C.2 is a time-dependent sequence of linear feedback policies $\mathbf{u}()_{1:N-1}$

$$\mathbf{u}(\mathbf{x})_i = \mathbf{u}_i - Q_{i\,uu}^{-1}Q_{i\,ux}(\mathbf{x} - \mathbf{x}_i).$$

Ideally, we would like to use an online Model Predictive Control strategy, whereby we iteratively use $\mathbf{u}()_1$ for one time-step and re-solve the problem. Our simulations were not fast enough for that (see below), so we resort to an off-line MPC strategy, as in [15]. Once a solution trajectory is obtained for some $\mathbf{x}_1$, we save $\mathbf{u}()_1$, and proceed to solve for a new trajectory starting at $\mathbf{x}_{1+d}$ for a small offset d. We can use the previous control sequence shifted by $-d$ and appended with d copies of the last control

$$\mathbf{u}^{\text{NEW}}()_{1:N-1} = [\mathbf{u}()_{1+d:N-1}\ \ \mathbf{u}()_{N-1}\ \ \mathbf{u}()_{N-1}\cdots], \tag{19}$$

as our initial guess for the policy of the shifted trajectory. Because we are usually not far from the optimum, we enjoy the quadratic convergence properties of second-order methods. The trajectory thus propagates forward, leaving behind it a trail of time-independent controllers $\mathbf{u}()_k$, all with horizon $T = hN$. We now use this collection of local linear controllers to construct a global controller that can be used online, by following a simple nearest neighbor rule:

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}(\mathbf{x})_j \quad \text{with} \quad j = \underset{k}{\operatorname{argmin}} \|\mathbf{x} - \mathbf{x}_k\|^2. \tag{20}$$

This is similar to the Trajectory Library concept [22]. If, as in the case below, the solution lies on a limit cycle, the time-independent trajectory will converge to it. Now we can perturb the initial state $\mathbf{x}_1$ every several time steps, and explore the state space around the limit cycle.

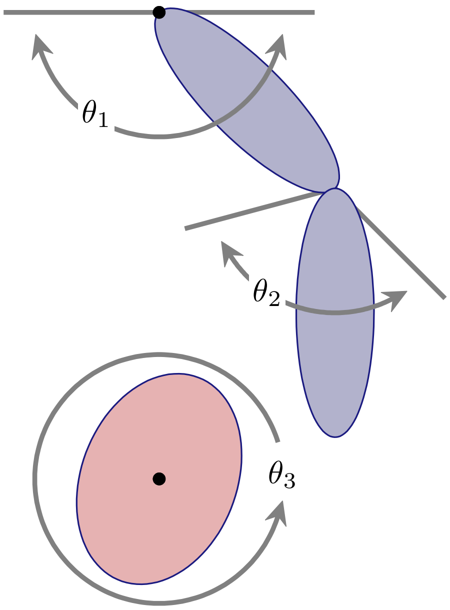

Fig. 1. The "flicking finger" dynamical system. The controller actuates θ_1 and θ_2 in order to spin the free ellipse around θ_3.

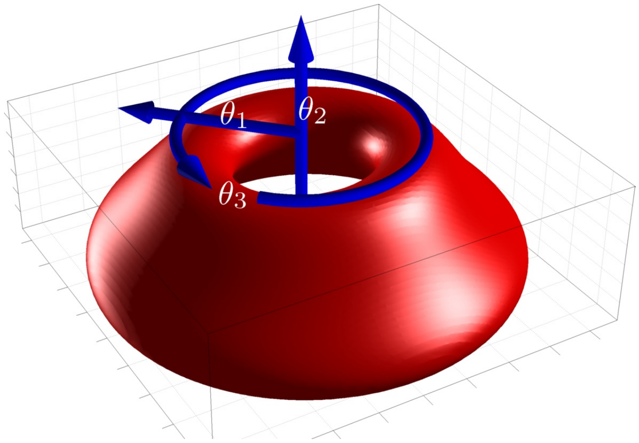

Fig. 2. The contact surface in the configuration space $[\theta_1\ \theta_2\ \theta_3]^{\mathsf{T}}$. The torus-like shape is the zero-valued isosurface of the distance function $\mathbf{d}(\mathbf{q})$. It corresponds to the set of points where the finger-tip makes contact with the free ellipse. Points inside and outside the surface correspond to penetration and broken contact respectively. A cylindrical coordinate system was chosen because θ_3 is a periodic variable (with period π rather than 2π due to symmetry).

V. Experiments

A. Setup

We performed our experiments with the 6-dimensional system of Figure 1. This planar system is composed of two articulated ellipses, with angles θ_1 and θ_2, and a free-spinning ellipse, whose angle is given by θ_3. The full state of the system is thus

$$\mathbf{x} = [\theta_1\ \theta_2\ \theta_3\ \dot{\theta}_1\ \dot{\theta}_2\ \dot{\theta}_3]^{\mathsf{T}}$$

The controller can apply two torques $\mathbf{u} = [u_1\ \ u_2]^{\mathsf{T}}$ to θ_1 and θ_2. We make it the controller's goal to spin the free ellipse in the positive direction, by defining a negative state-cost proportional to θ_3 with a quadratic control-cost:

$$\ell(\mathbf{x}, \mathbf{u}) = -c_x\theta_3 + c_u\|\mathbf{u}\|^2$$

Note that it would be very difficult to solve this task with control techniques that force the system to a pre-planned trajectory, since it is not clear how such a trajectory would be found.

In Figure 2, we depict the $\mathbf{d}(\mathbf{q}) = 0$ isosurface in the configuration space $[\theta_1\ \theta_2\ \theta_3]^\mathsf{T}$.

B. Parameters and methods

The following values can all be assumed to have the appropriate units of a self consistent unit system (e.g. MKS). The major and minor radii of the finger ellipses are .8 and .25 respectively. Those of the free ellipse are .7 and .5. The masses and moments of inertia correspond to a mass density of 1. The vertical distance between the two anchors (black dots in Figure 1) is 3. Angle limits are $-\pi \le \theta_1 \le 0$ and $-2\pi/3 \le \theta_2 \le 0$. The drag coefficients of the finger joints and of the free ellipse axis are .2, .2 and .7, respectively. Gravity in the vertical direction is -4. The control-cost coefficients were $c_u = 0.05$ and $c_x = 1$. The time step $h = 0.05$ and the number of time steps per trajectory $N = 75$, for a time horizon of $T = 3.5$.

Both angle constraints and the contact constraint were satisfied with impulses, as described in section III-A. We used a friction coefficient of $\mu = 0.5$, with the friction model of [9]. Since we did not estimate the dynamics, we used a constant $\sqrt{(\Sigma_\mathbf{b})_{ii}} = 0.1$, which was chosen as a trade-off between good convergence and small softening of the contact (described below). For the SLCP residual, both (15) and (17) gave qualitatively similar results, and we used (17) in the results below, because it is computationally cheaper.

We avoided the matrix inversions of (2) by directly solving the (stochastic) mixed complementarity problem of (1). This is easy to do with NCP functions, by setting $\phi(a, b) = b$ for those indexes where equality is desired.

We propagated the receding-horizon trajectory 400 times, with an offset $d = 4$, giving us a library of 400 locally linear feedback controllers. Every 15 time-steps, the velocities of initial state were perturbed by normal noise of variance 4. The reason only the velocities were perturbed is that random perturbations of the angles might lead to illegal (penetrative) configurations. When selecting the nearest neighbor of equation (20), we used a Euclidian norm scaled by the covariance of all the $\mathbf{x}_i$. In Figure 3, we plot the locations of the origins $\mathbf{x}_i$, projected onto the first 3 dimensions of the state-space.

Our simulator[4] was written in MATLAB for versatility and code readability, but is therefore not very efficient. One forward-backward pass of the local method took $\sim 5_s$ on a quad-core Core i7 machine. Starting from the zero policy, convergence was attained in $\sim$60 passes. In receding-horizon mode, starting from the shifted previous policy (19), $\sim$8 passes sufficed, due to quadratic convergence.

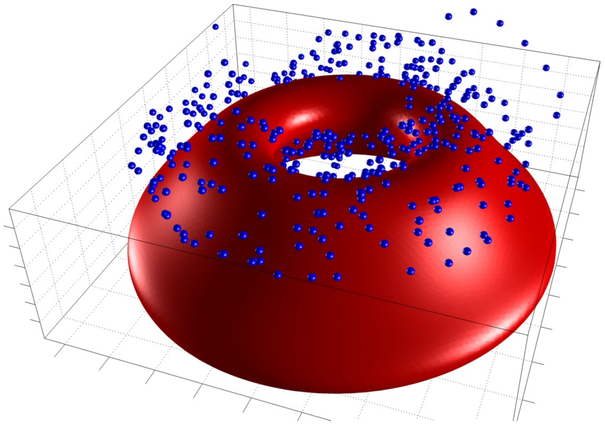

Fig. 3. The trajectory library. The dots correspond to the origins of 400 linear feedback controllers which we select with the nearest-neighbor rule (20). The global policy is effectively a Voronoi tessellation of these policies.

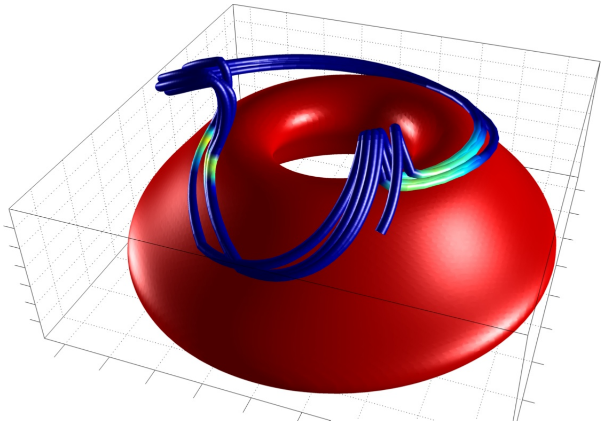

Fig. 4. The limit-cycle of a controlled trajectory. The color of the trajectory is proportional to the contact impulses, showing the points when flicking occurs.

C. Results

Our use of the SLCP when solving for velocities and impulses has the qualitative effect of creating a “force field” which extends from surfaces and lets contact and friction impulses act at a distance. If we could simulate the true stochastic system, some probability mass would experience contact before the mean would. The “fuzzy” force layer in our new deterministic system is an approximation to the mean force that would act on the mean of the distributed state.

The controller that was synthesized is very robust. We tested it on both the smoothed system and the original non-smooth one. For both systems, the controller dependably spun the ellipse, and recovered from disturbances. As an indicator of robustness, the controller worked well for time-steps different than the one which it was trained. We built a real-time simulator with a graphical front-end, which allows the user to perturb the system as it is being controlled, by interactively pulling on the ellipses with the mouse cursor. We were unable to perturb the system out of the controller’s basin-of-attraction.

In order to assess the contribution of the smoothing to robustness, we solved again, with a smaller noise covariance

[4] Available online at http://alice.nc.huji.ac.il/~tassa/

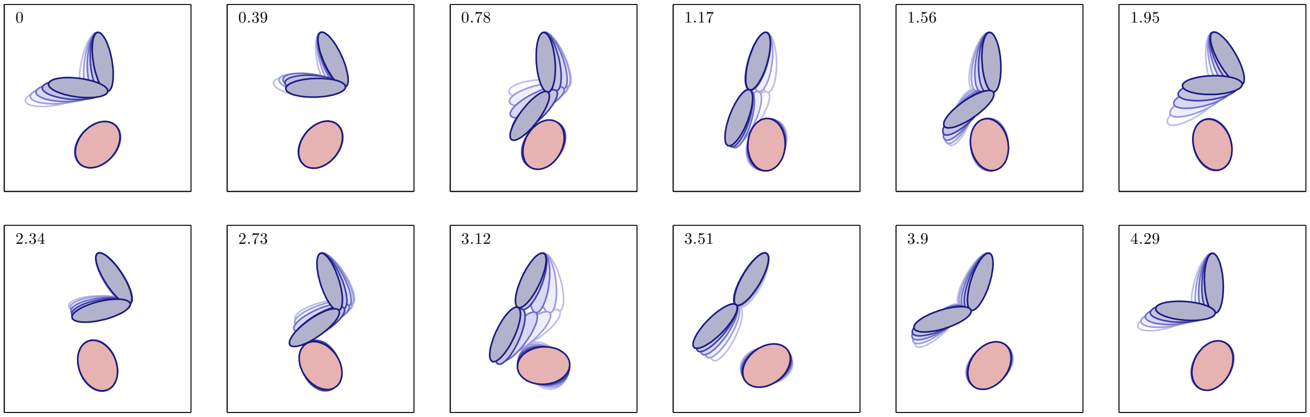

Fig. 5. Animation frames from the limit cycle. The third frame (time=0.78) and eighth frame (time=2.73) correspond to the contacts on the left and right side of Figure 4, respectively.

$\sqrt{(\Sigma_{\mathbf{b}})_{ii}} = 0.05$. The resulting controller was less robust in all of the senses described above. In particular, this controller sometimes entered into a non-productive limit-cycle where no contact was achieved. For smaller noise covariances the trajectory optimizer did not converge, so a direct comparison was not possible.

In Figure 4, we show the limit-cycle that a controlled trajectory makes in configuration space, with $h = 0.03$. The dynamics used were those of the original, nonsmooth system. The color of the trajectory is proportional to the contact impulses, showing the points when flicking occurs. The contact on the left of the figure corresponds to a weak tap that repositions the ellipse at a favorable angle, while the second one on the right side, exploits this position to deliver a stronger flick, that spins the ellipse. In Figure 5 we show animation frames from this sequence. The first and second contacts correspond to frame 3 (t=0.78) and frame 8 (t=2.73).

VI. Discussion

Our purpose here was to investigate the applicability of local methods to nonlinear control problems involving contact and friction. Our proposed method involves modifying the contact solver of the modeled dynamics, in a way which takes into account the controller's uncertainty regarding the state of the system. Though we present promising preliminary results that include a working robust controller, open issues remain.

Folding the noise into the dynamics, as we do here, is an approximation of what we would really like to do, namely simulate the full stochastic system, and measure costs WRT distributions rather than points. This option however has its own problems. The actual distributions which are propagated by the true dynamics become multimodal upon contact and require either a nontrivial parameterization, or a large number of "particles" for a non-parametric representation. In the case where distributions are represented as a mixture of samples, whether particles or σ-points, the dynamics would still be non-differentiable.

Local minima are still a significant problem for local methods like the ones used here. If we had not included gravity in the simulation, the initial trajectory with $\mathbf{u}_i = 0$ would have never achieved contact with the free ellipse, and the controller wouldn't have "known" about the possibility of contact. One option is to inject noise into the system during the initial stages of learning, for exploration purposes. Another option is to use the action-at-a-distance effect of the SLCP solution. By first solving for a system with very large postulated noise covariance, and then gradually reducing it, a scaffolding effect might be achieved.

The SLCP solution effectively replaces hard contact with a nonlinear spring-damper. However, unlike some arbitrary spring, it does not require setting the unknown parameters (spring and damper coefficients, form of the nonlinearity), and is derived in a principled way from the noise covariance. Additionally, unlike conventional springs which deform in proportion to the applied force, the forces which we compute scale with the effective inertia, so that a heavy and a light object experience the same smoothing.

A non-differentiable distance function $\mathbf{d}(\mathbf{q})$ would result in non-differentiable dynamics. For the ellipses used here, the signed distance is indeed differentiable, but this is not true for other shapes.

The complementarity conditions in (1) apply for quantities of different units: λ is an impulse while $\mathbf{d}(\mathbf{q}')$ is a distance. Because the smoothed NCP function $\phi(\cdot,\cdot)$ effectively mixes these quantities, a different choice of units would ostensibly lead to different results, which is clearly undesirable. This is a fundamental issue that requires further investigation. As famously observed by Stewart [4], "in many ways it is easier to write down a numerical method for rigid-body dynamics than it is to say exactly what the method is trying to compute".

Acknowledgment

This work was supported by the US National Science Foundation.

References

[1] X. Chen and M. Fukushima, "Expected residual minimization method for stochastic linear complementarity problems," *Math. Oper. Res.*, vol. 30, no. 4, pp. 1022–1038, 2005.

[2] F. Pfeiffer and C. Glocker, *Multibody dynamics with unilateral contacts*. Wiley-VCH, 1996.

[3] E. Westervelt, *Feedback control of dynamic bipedal robot locomotion*. Boca Raton: CRC Press, 2007.

[4] D. E. Stewart, "Rigid-Body dynamics with friction and impact," *SIAM Review*, vol. 42, no. 1, pp. 3–39, 2000.

[5] J. J. Moreau, "Unilateral contact and dry friction in finite freedom dynamics," *Nonsmooth mechanics and Applications*, p. 1–82, 1988.

[6] F. A. Potra, M. Anitescu, B. Gavrea, and J. Trinkle, "A linearly implicit trapezoidal method for integrating stiff multibody dynamics with contact, joints, and friction," *International Journal for Numerical Methods in Engineering*, vol. 66, no. 7, pp. 1079–1124, 2006.

[7] E. Todorov, "Implicit nonlinear complementarity: a new approach to contact dynamics," in *International Conference on Robotics and Automation*, 2010.

[8] M. Anitescu and F. A. Potra, "Formulating dynamic Multi-Rigid-Body contact problems with friction as solvable linear complementarity problems," *Nonlinear Dynamics*, vol. 14, no. 3, pp. 231–247, Nov. 1997.

[9] M. Anitescu, "Optimization-based simulation of nonsmooth rigid multibody dynamics," *Mathematical Programming*, vol. 105, no. 1, pp. 113–143, 2006.

[10] R. W. Cottle, J. Pang, and R. E. Stone, *The Linear Complementarity Problem*. SIAM, Oct. 2009.

[11] S. C. Billups and K. G. Murty, "Complementarity problems," *Journal of Computational and Applied Mathematics*, vol. 124, no. 1-2, pp. 303–318, Dec. 2000.

[12] C. Chen and O. L. Mangasarian, "A class of smoothing functions for nonlinear and mixed complementarity problems," *Computational Optimization and Applications*, vol. 5, no. 2, pp. 97–138, Mar. 1996.

[13] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko, *The mathematical theory of optimal processes*. Interscience New York, 1962.

[14] P. Abbeel, A. Coates, M. Quigley, and A. Y. Ng, "An application of reinforcement learning to aerobatic helicopter flight," in *Advances in Neural Information Processing Systems 19: Proceedings of the 2006 Conference*, 2007, p. 1.

[15] Y. Tassa, T. Erez, and W. Smart, "Receding horizon differential dynamic programming," in *Advances in Neural Information Processing Systems 20*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. Cambridge, MA: MIT Press, 2008, p. 1465–1472.

[16] R. Tedrake, "LQR-Trees: feedback motion planning on sparse randomized trees," in *Proceedings of Robotics: Science and Systems (RSS)*, 2009.

[17] M. Toussaint, "Robot trajectory optimization using approximate inference," in *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009.

[18] D. E. Stewart and M. Anitescu, "Optimal control of systems with discontinuous differential equations," *Numerische Mathematik*, 2009.

[19] W. H. Fleming and H. M. Soner, *Controlled Markov processes and viscosity solutions*. Springer Verlag, 2006.

[20] M. Fukushima and G. lin, "Stochastic equilibrium problems and stochastic mathematical programs with equilibrium constraints: A survey," *Pacific Journal of Optimization*, vol. to appear, 2010.

[21] X. Chen and P. Tseng, "Non-Interior continuation methods for solving semidefinite complementarity problems," *Mathematical Programming*, vol. 95, no. 3, pp. 431–474, Mar. 2003.

[22] M. Stolle and C. G. Atkeson, "Policies based on trajectory libraries," in *Proceedings of the International Conference on Robotics and Automation (ICRA 2006)*, 2006.