

Pose and Motion from Contact

Yan-Bin Jia Michael Erdmann

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213-3890

June 26, 1998

Abstract

In the absence of vision, grasping an object often relies on tactile feedback from the fingertips. As the finger pushes the object, the fingertip can feel the contact point move. If the object is known in advance, from this motion the finger may infer the location of the contact point on the object and thereby the object pose. This paper primarily investigates the problem of determining the pose (orientation and position) and motion (velocity and angular velocity) of a planar object with known geometry from such contact motion generated by pushing.

A dynamic analysis of pushing yields a nonlinear system that relates through contact the object pose and motion to the finger motion. The contact motion on the fingertip thus encodes certain information about the object pose. Nonlinear observability theory is employed to show that such information is sufficient for the finger to “observe” not only the pose but also the motion of the object. Therefore a sensing strategy can be realized as an observer of the nonlinear dynamical system. Two observers are subsequently introduced. The first observer, based on the result of [15], has its “gain” determined by the solution of a Lyapunov-like equation; it can be activated at any time instant during a push. The second observer, based on Newton’s method, solves for the initial (motionless) object pose from three intermediate contact points during a push.

Under the Coulomb friction model, the paper copes with support friction in the plane and/or contact friction between the finger and the object. Extensive simulations have been done to demonstrate the feasibility of the two observers. Preliminary experiments (with an Adept robot) have also been conducted. A contact sensor has been implemented using strain gauges.

1 Introduction

Part sensing and grasping are two fundamental operations in automated assembly. Traditionally, they are performed sequentially in an assembly task. Parts in many assembly

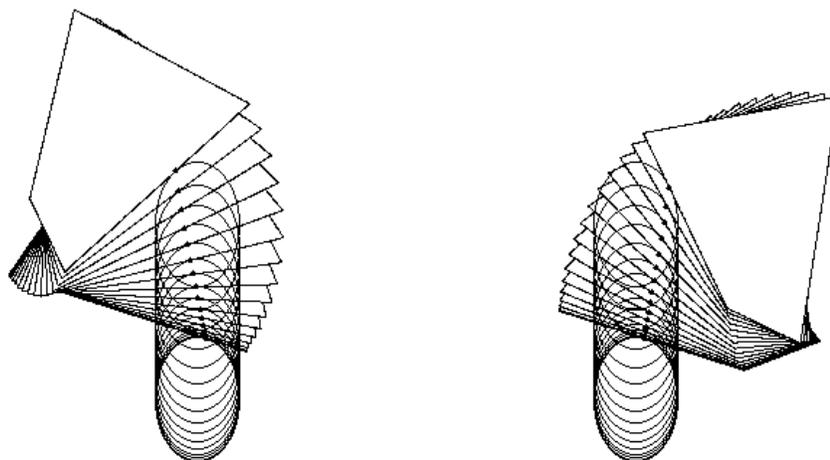


Figure 1: Different motions of contact (drawn as dots) on an ellipse pushing a quadrilateral in two different initial poses.

applications are manufactured to high precisions based on their geometric models. This knowledge of part geometry can sometimes significantly facilitate sensing as well as grasping. It can sometimes also help integrate these two operations, reducing the assembly time and cost.

Consider the task of grasping something, say, a pen, on the table while keeping your eyes closed. Your fingers fumble on the table until one of them touches the pen and (inevitably) starts pushing it for a short distance. While feeling the contact move on the fingertip, you can almost tell which part of the pen is being touched. Assume the pushing finger is moving away from you. If the contact remains almost stable, then the middle of the pen is being touched; if the contact moves counterclockwise on the fingertip, then the right end of the pen is being touched; otherwise the left end is being touched. Immediately, a picture of the pen configuration has been formed in your head so you coordinate other fingers to quickly close in for a grip.

The above example tells us that the pose of a known shape may be inferred from the contact motion on a finger pushing the shape. To better illustrate this idea, Figure 1 shows two motions of a quadrilateral in different initial poses pushed by an ellipse under the same motion. Although the initial contacts on the ellipse were the same, the final contacts are quite far apart. Thinking in reverse leads to the main questions of this paper:

1. *Can we determine the pose of an object with known geometry and mechanical properties from the contact motion on a single pushing finger, or simply, from a few intermediate contact positions during the pushing?*
2. *Can we determine any intermediate pose of the object during the pushing?*
3. *Furthermore, can we estimate the motion of the object during the pushing?*

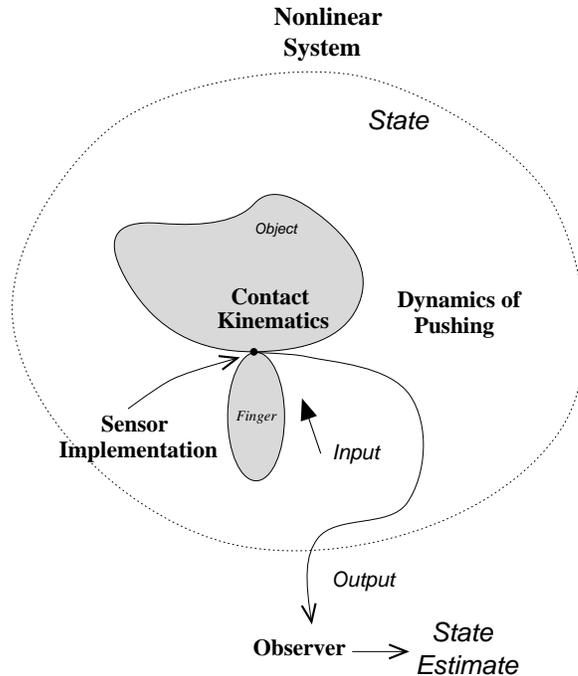


Figure 2: Anatomy of pushing.

In this paper, we will give affirmative answers to the above questions in the general case. To accomplish this, we will characterize pushing as a system of nonlinear differential equations based on its dynamics. As shown in Figure 2, the state of the system will include the configurations (positions, orientations, and velocities) of the finger and object during the push at any time instant. The system input will be the acceleration of the finger. The system output will be the contact location on the finger subject to the kinematics of contact. This output will be fed to nonlinear observers, which serve as the sensing algorithms, to estimate the object’s pose and motion.

Section 2 copes with the dynamics of pushing and the kinematics of contact, deriving a system of differential equations that govern the object and contact motions while resolving related issues such as support friction in the plane and the initial object motion; Section 3 applies nonlinear control theory to verify the soundness of our sensing approach to be proposed, establishing the local observability of this dynamical pushing system from the finger contact; Section 4 describes two nonlinear observers which estimate the object pose (and motion) at any instant and at the start of pushing, respectively, and which require different amounts of sensor data; Section 5 extends the results to incorporate contact friction between the finger and the object; Section 6 presents simulations on both observers and the implementation of a contact sensor, demonstrating that *three* intermediate contact points often suffice to determine the initial pose for the fingers and objects tested; Finally, Section 7 summarizes the paper and outlines future work.

1.1 Related Work

Our work is grounded in robotics where an abundance of previous work exists. It also draws upon the part of nonlinear control theory that concerns nonlinear observability and observers.

1.1.1 Robotics

Dynamics of sliding rigid bodies was treated by MacMillan [36] for non-uniform pressure distributions, and by Goyal *et al.* [19] using geometric methods based on the limit surface description of friction. Howe and Cutkosky [26] experimentally showed that the limit surface only approximates the force-motion relationship for sliding bodies and discussed other simplified practical models for sliding manipulation.

Mason [37] pioneered the study of the mechanics of pushing using quasi-static analysis, predicting the direction in which an object being pushed rotates and plotting out its instantaneous rotation center. For unknown centers of friction,¹ Alexander and Maddocks [2] reduced the problem of determining the motion of a slider under some applied force to the case of a bipod, obtaining analytical solutions for simple sliders. The problem of predicting the accelerations of multiple 3D objects in contact with Coulomb friction has a nonlinear complementarity formulation due to Pang and Trinkle [40]; the existence of solutions to models with sliding and rolling contacts has been established.

Montana [38] derived a set of differential equations describing the motion of a contact point between two rigid bodies in response to a relative motion of these bodies, and employed these equations to sense the local curvature of an unknown object and to follow its surface while steering the contact point to some desired location on the end effector. The kinematics of spatial motion with point contact was also studied by Cai and Roth [6] who assumed a tactile sensor capable of measuring the relative motion at the contact point. The special kinematics of two rigid bodies rolling on each other was considered by Li and Canny [33] in view of path planning in the contact configuration space. In our work, contact kinematics is derived directly from the absolute velocities of the finger and the object rather than from their relative velocity at the contact. Also we are concerned with a finger and object only in the plane not in 3D.

Part of our motivation came from the blind grasping task at the beginning of the paper. The caging work by Rimon and Blake [43] is concerned with constructing the space of all configurations of a two-fingered hand controlled by one parameter that confine a given 2D object; these configurations can lead to immobilizing grasps by following continuous paths in the same space. This work requires an initial image of the object taken by a camera. Work related to caging includes parts feeder design [42] and fixture design [5]. In this paper, we are concerned with how to “feel” a known object using only one finger and how to infer its pose and motion information rather than how to constrain and grasp the object using multiple fingers.

A larger part of the motivation of our work was from parts orienting. Orienting mechanical parts was studied early on by Grossman and Blasgen [22]. They used a vibrating box to constrain a part to a small finite number of possible stable poses and then determined

¹The center of friction is the centroid of the pressure distribution.

the particular pose by a sequence of probes using a tactile sensor. Inspired by their result, Erdmann and Mason [12] constructed a planner that employs sensorless tilting operations to orient planar objects randomly dropped into a tray, based on a simple model of the quasi-static mechanics of sliding. Utilizing the theory of limit surfaces [19], Böhringer *et al.* [4] developed a geometric model for the mechanics of an array of microelectromechanical structures and showed how this structure can be used to uniquely and efficiently align a part up to symmetry. Goldberg [17] showed that every polygonal part with unknown initial orientation can be oriented by a parallel-jaw gripper up to symmetry in the part’s convex hull. He constructed an algorithm with sub-cubic running time as a proof of sensorless parts orienting.

Based on the limit surface model and quasi-static analysis, Lynch *et al.* [35] conducted active sensing of an object’s center of mass during pushing from tactile feedback; and developed a control system that translates and orients objects. Their work assumes known contact edge and does not quantitatively estimate the object’s motion due to the quasi-static nature. Also applying quasi-static analysis, Akella and Mason [1] described a complete open-loop planner that can orient and translate polygonal objects in the plane by pushing with a straight fence.

In our previous work [29], we introduced the methods of cone inscription and point sampling that compute the poses of known shapes from a continuum and a finite number of possibilities, respectively, using simple geometric constraints such as coincidence and containment. Paulos and Canny [41] studied the problem of finding optimal point probes for refining the pose of a polygonal part with known geometry from an approximate pose; they revealed that this problem is dual to the grasping problem of computing optimal finger placements and gave an efficient near-optimal solution.

Model-based recognition and localization traditionally solve a constraint satisfaction problem by searching for a consistent matching between sensory data and model(s). This is often conducted by pruning the search tree subject to pure geometric constraints. Grimson and Lozano-Pérez [21] used tactile measurements of positions and surface normals on a 3D object to identify and locate it from a set of known 3D objects, based on the geometric constraints imposed by these tactile data. Gaston and Lozano-Pérez [14] showed how to identify and locate a polyhedron on a known plane using tactile information that includes contact points and the ranges of surface normals at these points. Grimson [20] advocated that model-based recognition and localization should be regarded as a constraint satisfaction problem that searches for a consistent matching between sensory data (e.g., 2D) and model(s) (e.g., 3D) under some geometric constraints.

A major difference between the sensing approach described in this paper and the listed previous work is that our approach combines geometric constraints with dynamics of manipulation. On the one hand, this would relieve the requirement for sufficient geometric constraints (and hence lessen the hardware load). On the other hand and more importantly, the involvement of dynamics extends sensing to dynamic pose and motion estimation. Subsequently, the tools which we will apply are from nonlinear control theory rather than from AI and computational geometry.

One feasible implementation for contact detection is to employ a force or tactile sensor. The paper by Salisbury [44] proposed the concept of fingertip force sensing with an

approach for determining contact locations and orientations from force and moment measurements. Fearing and Binford [13] designed a cylindrical tactile sensor to determine the principal curvatures of an object through rolling contact. Based on continuum mechanics and photoelastic stress analysis, Cameron *et al.* [7] built a tactile sensor using a layer of photoelastic material along with its mathematical model. Allen and Roberts [3] deployed robot fingers to obtain a number of contact points around an object and then fit (in a least squares manner) the data to a superquadric surface representation to reconstruct the object’s shape. Howe and Cutkosky [25] introduced dynamic tactile sensing in which sensors capture fine surface features during motion, presenting mechanical analysis and experimental performance measurements for one type of dynamic tactile sensor—the stress rate sensor.

1.1.2 Nonlinear Control

The theoretical foundation of our work comes from the part of control theory concerned with the observability and observers of nonlinear systems. For a general introduction to nonlinear control theory, we refer the reader to Isidori [27] and Nijmeijer and van der Schaft [39].

Hermann and Krener [24] first studied observability using the notion of observation space. We will use their observability rank condition to show that the pose and motion of a polygonal object pushed by a disk is locally observable from the disk contact. A result due to Crouch [9] shows that an analytic system is observable if and only if the observation space distinguishes points in the state space.

Luenberger-like asymptotic observers, first constructed by Luenberger [34] for linear systems, remain likely the most commonly used observer forms for nonlinear systems today. Gauthier, Hammouri, and Othman [15] described an observer for affine-control nonlinear systems whose “gain” is determined via the solution of an appropriate Lyapunov-like equation. Their observer has a very simple form: it is a copy of the original system, together with a linear corrective term that depends only on the state space dimension. Our first pose observer will be constructed using the GHO procedure.

Ciccarella *et al.* [8] proposed a similar observer whose gain vector is controlled by the properly chosen eigenvalues of a certain matrix obtained from the original system’s Brunovsky canonical form, thus providing more freedom on optimizing the observer behavior. Extending the results of GHO [15], Gauthier and Kupka [16] characterized non-affine control systems that are observable under any input and constructed a generic exponential observer for these systems.

Zimmer [46] presented a state estimator that conducts on-line minimization over some objective function. His observer, with provable convergence, iteratively uses Newton’s method to modify its state estimate every fixed period of time. Our second pose observer will also make use of Newton’s method but we will estimate the initial motionless pose of the object, relieving the task of evaluating the complex second order partial derivatives of the drift field as Zimmer had encountered.

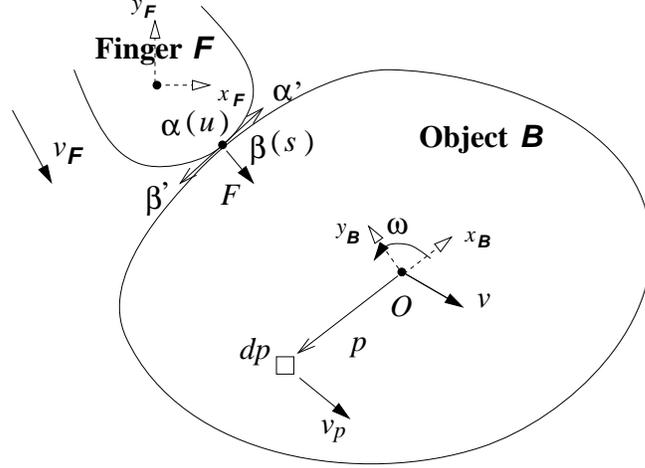


Figure 3: Finger \mathcal{F} translating and pushing object \mathcal{B} .

1.2 Notation

We abide by the following conventions on notation in this paper. Every vector \mathbf{x} is a column vector written as $(x_1, \dots, x_n)^T$ for some variables x_1, \dots, x_n . The derivative of a vector function $\mathbf{x}(t) = (x_1(t), \dots, x_n(t))^T$ with respect to t is denoted by $d\mathbf{x}/dt = (\frac{dx_1}{dt}, \dots, \frac{dx_n}{dt})^T$. The gradient of a scalar function $y(\mathbf{x})$, where $\mathbf{x} = (x_1, \dots, x_n)^T$, is a row vector $\partial y / \partial \mathbf{x} = (\frac{\partial y}{\partial x_1}, \dots, \frac{\partial y}{\partial x_n})$. The partial derivative of a vector field $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))^T$ with respect to vector $\mathbf{x} = (x_1, \dots, x_n)^T$ is a $m \times n$ matrix, given by $(\partial \mathbf{f} / \partial \mathbf{x})_{ij} = \frac{\partial f_i}{\partial x_j}$.

To avoid any ambiguity, the notation “ $\dot{}$ ” means differentiation with respect to time, while the notation “ $\dot{}$ ” means differentiation with respect to some curve parameter. For example, $\dot{\boldsymbol{\alpha}} = \boldsymbol{\alpha}' \dot{u} = \frac{d\boldsymbol{\alpha}}{du} \frac{du}{dt}$ gives the velocity of a point moving on a curve $\boldsymbol{\alpha}(u)$. The cross product of two vectors (e.g., $\boldsymbol{\alpha} \times \mathbf{v}$) is treated as a scalar wherever ambiguity would not arise. A scalar in a cross product (e.g., the angular velocity ω in $\omega \times \boldsymbol{\beta}$) acts as a vector of equal magnitude and orthogonal to the plane.

2 Motion of Contact

Throughout the paper we consider the two-dimensional problem of a *translating* finger \mathcal{F} pushing an object \mathcal{B} . Coulomb’s friction law is assumed and the coefficient of support friction, that is, friction between \mathcal{B} and the plane, is everywhere μ . For simplicity, let us assume uniform mass and pressure distributions of \mathcal{B} . Let us also assume *frictionless* contact between \mathcal{F} and \mathcal{B} at present and deal with contact friction exclusively in Section 5. Let $\mathbf{v}_{\mathcal{F}}$ be the velocity of \mathcal{F} , known to \mathcal{F} ’s controller, \mathbf{v} and ω the velocity and angular velocity of \mathcal{B} , respectively, all in the world coordinate frame (Figure 3).

Let \mathcal{F} ’s boundary be a smooth curve $\boldsymbol{\alpha}$ and \mathcal{B} ’s boundary be a piecewise smooth closed curve $\boldsymbol{\beta}$ such that $\boldsymbol{\alpha}(u)$ and $\boldsymbol{\beta}(s)$ are the two points in contact in the local frames of \mathcal{F} and \mathcal{B} , respectively. Following convention, moving counterclockwise along $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ increases u and s , respectively. Assume that one curve segment of $\boldsymbol{\beta}$ stays in contact with $\boldsymbol{\alpha}$ throughout

the pushing.

2.1 Dynamics of Pushing

That \mathcal{F} and \mathcal{B} maintain contact imposes a velocity constraint:

$$\mathbf{v}_{\mathcal{F}} + \boldsymbol{\alpha}' \dot{u} = \mathbf{v} + \boldsymbol{\omega} \times R\boldsymbol{\beta} + R\boldsymbol{\beta}' \dot{s}, \quad (1)$$

where $R(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$ is the rotation matrix associated with the orientation θ of \mathcal{B} , which is determined by u , s , and the orientation of \mathcal{F} . Newton's and Euler's equations on rigid body dynamics are stated as:

$$F + \int_{\mathcal{B}} -\mu\eta g \hat{\mathbf{v}}_p d\mathbf{p} = m\dot{\mathbf{v}}, \quad (2)$$

$$R\boldsymbol{\beta} \times F + \int_{\mathcal{B}} R\mathbf{p} \times (-\mu\eta g \hat{\mathbf{v}}_p) d\mathbf{p} = I\dot{\boldsymbol{\omega}}, \quad (3)$$

where F is the contact force acting on \mathcal{B} , g the acceleration of gravity, m the mass, η the mass density, and I the angular inertia about the center of mass O (all of \mathcal{B}). Here $\mathbf{v}_p = \mathbf{v} + \boldsymbol{\omega} \times R\mathbf{p}$ is the velocity of $\mathbf{p} \in \mathcal{B}$ and $\hat{\mathbf{v}}_p = \frac{\mathbf{v}_p}{\|\mathbf{v}_p\|}$ its direction.²

With no friction at the contact point, F acts along the inward normal of \mathcal{B} :

$$F \cdot R\boldsymbol{\beta}' = 0; \quad (4)$$

$$R\boldsymbol{\beta}' \times F > 0. \quad (5)$$

Finally, the normals of \mathcal{F} and \mathcal{B} at the contact are opposite to each other; equivalently, we have

$$\boldsymbol{\alpha}' \times R\boldsymbol{\beta}' = 0, \quad (6)$$

$$\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta}' < 0.$$

Given the finger motion $\mathbf{v}_{\mathcal{F}}$, there are seven equations (1), (2), (3), (4), and (6) with seven variables $u, s, \boldsymbol{\omega}, \mathbf{v}$, and F .³ From these equations, we are now ready to derive the differential equations for $u, s, \boldsymbol{\omega}$, and \mathbf{v} .

Let $\mathbf{a}_{\mathcal{F}}$ be the acceleration of \mathcal{F} , $\mathcal{A} = \int_{\mathcal{B}} d\mathbf{p} = \frac{m}{\eta}$ and $\rho = \sqrt{\frac{I}{m}}$ the area and radius of gyration of \mathcal{B} , respectively, and $\Gamma = \int_{\mathcal{B}} R\boldsymbol{\beta}' \times (R\mathbf{p} \times \hat{\mathbf{v}}_p) + (\boldsymbol{\beta}' \cdot \boldsymbol{\beta}) \hat{\mathbf{v}}_p d\mathbf{p}$ an integral associated with support friction. We have

²That \mathcal{F} is translating implies either $\mathbf{v} \neq 0$ or $\boldsymbol{\omega} \neq 0$ after the pushing starts. So \mathbf{v}_p can vanish over at most one point $\mathbf{p} \in \mathcal{B}$, which will vanish in the integrals in equations (2) and (3).

³Note that equations (1) and (2) and variables \mathbf{v} and F are each counted twice.

Theorem 1 Consider the pushing system described by (1)–(6). The points of contact evolve according to

$$\dot{u} = \frac{-(\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta}')^2\omega + (\boldsymbol{\alpha}' \times R\boldsymbol{\beta}'')(\boldsymbol{\alpha}' \cdot (\mathbf{v} + \omega \times R\boldsymbol{\beta} - \mathbf{v}_{\mathcal{F}}))}{(\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta}')(\boldsymbol{\alpha}'' \times R\boldsymbol{\beta}') + \|\boldsymbol{\alpha}'\|^2(\boldsymbol{\alpha}' \times R\boldsymbol{\beta}'')}, \quad (7)$$

$$\dot{s} = -\frac{\|\boldsymbol{\alpha}'\|^2(\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta}')\omega + (\boldsymbol{\alpha}'' \times R\boldsymbol{\beta}')(\boldsymbol{\alpha}' \cdot (\mathbf{v} + \omega \times R\boldsymbol{\beta} - \mathbf{v}_{\mathcal{F}}))}{(\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta}')(\boldsymbol{\alpha}'' \times R\boldsymbol{\beta}') + \|\boldsymbol{\alpha}'\|^2(\boldsymbol{\alpha}' \times R\boldsymbol{\beta}'')}, \quad (8)$$

and the object's angular acceleration and acceleration are

$$\dot{\omega} = \frac{i\boldsymbol{\alpha}'' \times (\mathbf{v}_{\mathcal{F}} - \mathbf{v}) + \boldsymbol{\alpha}' \times \mathbf{a}_{\mathcal{F}} - (i\boldsymbol{\alpha}'' \cdot R\boldsymbol{\beta} + \boldsymbol{\alpha}' \cdot (\omega \times R\boldsymbol{\beta} + R\boldsymbol{\beta}'\dot{s}))\omega + \frac{\mu g}{\mathcal{A}\boldsymbol{\beta}' \cdot \boldsymbol{\beta}}\boldsymbol{\alpha}' \times \Gamma}{\boldsymbol{\alpha}' \cdot R\left(\boldsymbol{\beta} + \frac{\rho^2}{\boldsymbol{\beta}' \cdot \boldsymbol{\beta}}\boldsymbol{\beta}'\right)}, \quad (9)$$

$$\dot{v} = \frac{\mathcal{A}\rho^2\dot{\omega} \times R\boldsymbol{\beta}' - \mu g\Gamma}{\mathcal{A}\boldsymbol{\beta}' \cdot \boldsymbol{\beta}}. \quad (10)$$

Proof Taking the dot products of $\boldsymbol{\alpha}'$ with both sides of (1) and rearranging terms thereafter, we obtain

$$\|\boldsymbol{\alpha}'\|^2\dot{u} - (\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta}')\dot{s} = \boldsymbol{\alpha}' \cdot (\mathbf{v} + \omega \times R\boldsymbol{\beta} - \mathbf{v}_{\mathcal{F}}).$$

Next differentiate both sides of (6):

$$(\boldsymbol{\alpha}'' \times R\boldsymbol{\beta}')\dot{u} + (\boldsymbol{\alpha}' \times R\boldsymbol{\beta}'')\dot{s} + (\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta}')\omega = 0.$$

Immediately, we solve for \dot{u} and \dot{s} from the two equations above and obtain (7) and (8).

Now we move on to derive the differential equations for \mathbf{v} and ω . First take the cross products of $R\boldsymbol{\beta}'$ with both sides of (3), eliminating the resulting term that contains $F \cdot R\boldsymbol{\beta}'$ and substituting (2) in after term expansion:

$$-(\boldsymbol{\beta}' \cdot \boldsymbol{\beta})m\dot{\mathbf{v}} - \mu\eta g\Gamma = R\boldsymbol{\beta}' \times I\dot{\omega}.$$

Here the term

$$\Gamma = \int_{\mathcal{B}} R\boldsymbol{\beta}' \times (R\mathbf{p} \times \hat{\mathbf{v}}_p) + (\boldsymbol{\beta}' \cdot \boldsymbol{\beta})\hat{\mathbf{v}}_p d\mathbf{p}, \quad (11)$$

when multiplied by $\mu\eta g$, combines the dynamic effects of friction. Thus we can write $\dot{\mathbf{v}}$ in the form of (10).

Taking the cross products of $\boldsymbol{\alpha}'$ with both sides of (1) and cancelling the term $\boldsymbol{\alpha}' \times R\boldsymbol{\beta}'$ according to (6), we have after a few more steps of term manipulation

$$\boldsymbol{\alpha}' \times (\mathbf{v}_{\mathcal{F}} - \mathbf{v}) = (\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta})\omega. \quad (12)$$

Differentiating both sides of (12) yields

$$\dot{u}\boldsymbol{\alpha}'' \times (\mathbf{v}_{\mathcal{F}} - \mathbf{v}) + \boldsymbol{\alpha}' \times (\mathbf{a}_{\mathcal{F}} - \dot{\mathbf{v}}) = \left(\dot{u}\boldsymbol{\alpha}'' \cdot R\boldsymbol{\beta} + \boldsymbol{\alpha}' \cdot (\boldsymbol{\omega} \times R\boldsymbol{\beta} + R\boldsymbol{\beta}'\dot{s}) \right) \boldsymbol{\omega} + (\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta})\dot{\boldsymbol{\omega}}. \quad (13)$$

Finally, substituting (10) in (13) gives us (9). \square

Substitute (7) and (8) into (9) and the resulted differential equation into (10). We have thus obtained the differential equations of $\boldsymbol{\omega}$ and \mathbf{v} which, along with (7) and (8), form a system of ordinary differential equations (ODEs). This system is numerically solvable for $u, s, \boldsymbol{\omega}$, and \mathbf{v} . Without any ambiguity, we also let (9) and (10) refer to their corresponding differential equations.

Our derivation of the differential equations (7)–(10) is correct only if the denominators on their right hand sides do not vanish. It is easy to show that these denominators vanish only if $\boldsymbol{\beta}' \cdot \boldsymbol{\beta} = 0$, or equivalently, $R\boldsymbol{\beta} \times F = 0$. Furthermore, the limits of $\dot{\boldsymbol{\omega}}$ and $\dot{\mathbf{v}}$ given by equations (9) and (10), respectively, as $\boldsymbol{\beta}' \cdot \boldsymbol{\beta} \rightarrow 0$, are equal to their degenerate forms derived under the condition $\boldsymbol{\beta}' \cdot \boldsymbol{\beta} = 0$, respectively. (See [28].)

The motion of \mathcal{B} is *independent* of its mass density η , as seen from (7), (8), (9), and (10) or directly from (2) and (3).

If $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are unit-speed curves with curvatures κ_{α} and κ_{β} at the contact point, respectively, such that $\kappa_{\alpha} + \kappa_{\beta} \neq 0$,⁴ then equations (7) and (8) are simplified to

$$\dot{u} = \frac{\boldsymbol{\omega} + \kappa_{\beta}\boldsymbol{\alpha}' \cdot (\mathbf{v} + \boldsymbol{\omega} \times R\boldsymbol{\beta} - \mathbf{v}_{\mathcal{F}})}{\kappa_{\alpha} + \kappa_{\beta}}; \quad (14)$$

$$\dot{s} = \frac{-\boldsymbol{\omega} + \kappa_{\alpha}\boldsymbol{\alpha}' \cdot (\mathbf{v} + \boldsymbol{\omega} \times R\boldsymbol{\beta} - \mathbf{v}_{\mathcal{F}})}{\kappa_{\alpha} + \kappa_{\beta}}. \quad (15)$$

For example, let $\boldsymbol{\alpha}$ be a circle with radius r and $\boldsymbol{\beta}$ a polygon. Hence $\kappa_{\alpha} = \frac{1}{r}$ and $\kappa_{\beta} = 0$.⁵ We have $\dot{u} = \frac{\boldsymbol{\omega}}{\kappa_{\alpha}} = \boldsymbol{\omega}r$. During a push of time Δt , the contact moves an arc of length

$$\int_0^{\Delta t} \dot{u} dt = r \int_0^{\Delta t} \boldsymbol{\omega} dt = r\Delta\theta$$

on $\boldsymbol{\alpha}$, which can be immediately verified from the tangency between $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$.

2.2 Integral of Support Friction

To numerically integrate (7)–(10), it is necessary to evaluate the integral Γ given by (11) which represents the effect of support friction on dynamics. Two-dimensional numerical integration of Γ can be very slow. However, by choosing proper polar coordinates we can reduce the evaluation of Γ to one variable integration, and if \mathcal{B} is polygonal, obtain the closed form of Γ .

When the motion of \mathcal{B} is pure translation ($\boldsymbol{\omega} = 0$), the evaluation is easy:

$$\Gamma = (\boldsymbol{\beta}' \cdot \boldsymbol{\beta})\mathcal{A}\hat{\mathbf{v}}. \quad (16)$$

⁴The term $\kappa_{\alpha} + \kappa_{\beta}$ is the 2-dimensional case of the *relative curvature form* introduced by Montana [38].

⁵Here κ_{β} is the curvature of the polygon edge in contact with $\boldsymbol{\alpha}$. We assume that the finger will not be in contact with any vertex during the pushing.

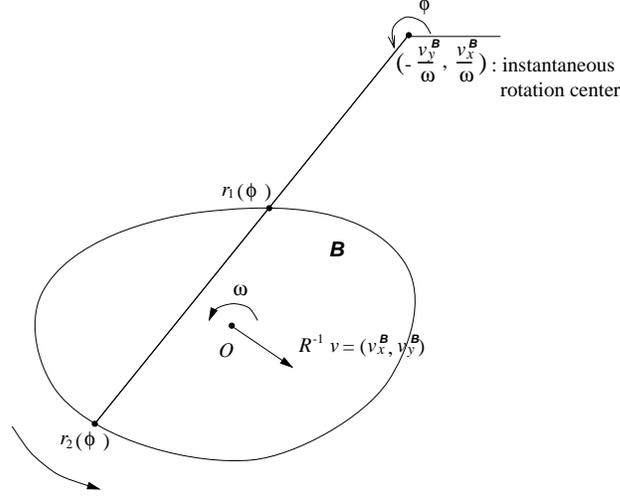


Figure 4: The instantaneous rotation center $(-v_y^{\mathcal{B}}/\omega, v_x^{\mathcal{B}}/\omega)^T$ of object \mathcal{B} with velocity $(v_x^{\mathcal{B}}, v_y^{\mathcal{B}})^T$ and angular velocity $\omega \neq 0$ (about its center of mass O). The integral Γ is evaluated in polar coordinates with respect to the i.r.c..

So we focus our discussion on the case $\omega \neq 0$. The integral Γ can be evaluated in the polar coordinates with respect to the instantaneous rotation center of \mathcal{B} introduced below.

Let us first express Γ in terms of \mathcal{B} 's moving body frame at its center of mass O :

$$\Gamma = R\beta' \times \int_{\mathcal{B}} \mathbf{p} \times \hat{\mathbf{v}}_p^{\mathcal{B}} d\mathbf{p} + (\beta' \cdot \beta) R \int_{\mathcal{B}} \hat{\mathbf{v}}_p^{\mathcal{B}} d\mathbf{p}, \quad (17)$$

where $\mathbf{v}_p^{\mathcal{B}} = R^{-1}\mathbf{v} + \omega \times \mathbf{p}$ is the velocity at $\mathbf{p} \in \mathcal{B}$ in the body frame. At the moment, \mathcal{B} is rotating about the point $(x_0^{\mathcal{B}}, y_0^{\mathcal{B}})^T = \omega \times (v_x^{\mathcal{B}}, v_y^{\mathcal{B}})^T / \omega^2 = (-v_y^{\mathcal{B}}/\omega, v_x^{\mathcal{B}}/\omega)^T$, called the *instantaneous rotation center* (i.r.c.), as shown in Figure 4. For convenience and clarity, we only illustrate the case where \mathcal{B} is convex. The evaluation should be straightforwardly generalized to the case where \mathcal{B} is concave.

Any ray at angle ϕ from the i.r.c. has at most two intersections $(\phi, r_1(\phi))^T$ and $(\phi, r_2(\phi))^T$, $r_1(\phi) < r_2(\phi)$, with the object boundary. Every point \mathbf{p} on the ray is instantaneously moving along the same direction $\hat{\mathbf{v}}_p^{\mathcal{B}} = (-\sin \phi, \cos \phi)^T$ if $\omega > 0$, or along the direction $\hat{\mathbf{v}}_p^{\mathcal{B}} = (\sin \phi, -\cos \phi)^T$ if $\omega < 0$. The two subintegrals in (17) now reduce to one-variable integrals in the polar coordinates:⁶

$$\begin{aligned} I_1 &= \int_{\mathcal{B}} \hat{\mathbf{v}}_p^{\mathcal{B}} d\mathbf{p} \\ &= \pm \int_{\phi_1}^{\phi_2} \int_{r_1(\phi)}^{r_2(\phi)} \begin{pmatrix} -\sin \phi \\ \cos \phi \end{pmatrix} r dr d\phi \\ &= \pm \int_{\phi_1}^{\phi_2} \frac{r_2^2(\phi) - r_1^2(\phi)}{2} \begin{pmatrix} -\sin \phi \\ \cos \phi \end{pmatrix} d\phi; \\ I_2 &= \int_{\mathcal{B}} \mathbf{p} \times \hat{\mathbf{v}}_p^{\mathcal{B}} d\mathbf{p} \end{aligned} \quad (18)$$

⁶If the i.r.c. is in the interior of \mathcal{B} , let $\phi_1 = 0$, $\phi_2 = 2\pi$, and $r_1(\phi) = 0$.

$$\begin{aligned}
&= \pm \int_{\phi_1}^{\phi_2} \int_{r_1(\phi)}^{r_2(\phi)} \left(\begin{pmatrix} x_0^{\mathcal{B}} \\ y_0^{\mathcal{B}} \end{pmatrix} + r \begin{pmatrix} \cos \phi \\ \sin \phi \end{pmatrix} \right) \times \begin{pmatrix} -\sin \phi \\ \cos \phi \end{pmatrix} r dr d\phi \\
&= \pm \int_{\phi_1}^{\phi_2} (x_0^{\mathcal{B}} \cos \phi + y_0^{\mathcal{B}} \sin \phi) \frac{r_2^2(\phi) - r_1^2(\phi)}{2} + \frac{r_2^3(\phi) - r_1^3(\phi)}{3} d\phi.
\end{aligned}$$

For polygonal shapes, the *closed* forms of the above two integrals are given in [28]; for most other shapes, these integrals can only be evaluated numerically.

2.3 Initial Motion

In order to numerically integrate equations (7)–(10), it is necessary to determine the initial accelerations $\dot{\mathbf{v}}$ and $\dot{\omega}$ of \mathcal{B} from the finger acceleration $\mathbf{a}_{\mathcal{F}}$ and the configurations of \mathcal{F} and \mathcal{B} .⁷

At the start of pushing, both the finger \mathcal{F} and the object \mathcal{B} are motionless; that is, we have

$$\mathbf{v}(0) = \mathbf{v}_0 = 0, \quad \omega(0) = \omega_0 = 0, \quad \text{and} \quad \mathbf{v}_{\mathcal{F}}(0) = 0.$$

Plugging the above into (7) and (8) yields the initial contact velocities:

$$\dot{u}(0) = 0 \quad \text{and} \quad \dot{s}(0) = 0.$$

We only consider the non-degenerate case where the contact normal N does not pass through the center of mass O as otherwise the initial accelerations can be easily determined. In other words, we deal with $\omega_0 \neq 0$. The frictional force \mathbf{f}_p at point $\mathbf{p} \in B$ is opposed to the direction of relative motion [18], which, at the start of pushing, is the direction of the acceleration

$$\begin{aligned}
\dot{\mathbf{v}}_p(0) &= \dot{\mathbf{v}}_0 + \dot{\omega}_0 \times \mathbf{p} + \omega_0 \times (\omega_0 \times \mathbf{p}) \\
&= \dot{\mathbf{v}}_0 + \dot{\omega}_0 \times \mathbf{p} \\
&= \dot{\omega}_0 \left(\frac{\dot{\mathbf{v}}_0}{\dot{\omega}_0} + 1 \times \mathbf{p} \right).
\end{aligned}$$

By a simple argument, the sign of $\dot{\omega}_0$ must agree with its sign were there no friction; hence it is easily determined. Consequently, $\hat{\mathbf{v}}_p(0)$, \mathbf{f}_p ,

$$\Gamma_0 = R\boldsymbol{\beta}' \times \int_{\mathcal{B}} R\mathbf{p} \times \hat{\mathbf{v}}_p(0) d\mathbf{p} + (\boldsymbol{\beta}' \cdot \boldsymbol{\beta}) \int_{\mathcal{B}} \hat{\mathbf{v}}_p(0) d\mathbf{p}$$

become functions of $\frac{\dot{\mathbf{v}}_0}{\dot{\omega}_0}$. Thus (10) can be rewritten as

$$\dot{\mathbf{v}}_0 = \frac{\mathcal{A}\rho^2\dot{\omega}_0 \times R\boldsymbol{\beta}' - \mu g\Gamma_0\left(\frac{\dot{\mathbf{v}}_0}{\dot{\omega}_0}\right)}{\mathcal{A}\boldsymbol{\beta}' \cdot \boldsymbol{\beta}} \quad (19)$$

⁷In this section, our focus is on simulating the start of pushing. Hence we temporarily assume the initial configuration of \mathcal{B} is known. Later in Section 4.2, we will see how to use such simulation to solve for the initial pose of \mathcal{B} using Newton's method.

at $t = 0$.⁸ Meanwhile, it follows from (13) that

$$\dot{\omega}_0 = \frac{\boldsymbol{\alpha}' \times \mathbf{a}_{\mathcal{F}}}{\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta} + \boldsymbol{\alpha}' \times \frac{\dot{\mathbf{v}}_0}{\dot{\omega}_0}}. \quad (20)$$

Dividing both sides of (19) by $\dot{\omega}_0$ and substituting (20) in, we get the following equation in $\frac{\dot{\mathbf{v}}_0}{\dot{\omega}_0}$:

$$\frac{\dot{\mathbf{v}}_0}{\dot{\omega}_0} = \frac{\mathcal{A}\rho^2 \times R\boldsymbol{\beta}' - \mu g \Gamma_0 \left(\frac{\dot{\mathbf{v}}_0}{\dot{\omega}_0} \right) \frac{\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta} + \boldsymbol{\alpha}' \times \frac{\dot{\mathbf{v}}_0}{\dot{\omega}_0}}{\boldsymbol{\alpha}' \times \mathbf{a}_{\mathcal{F}}}}{\mathcal{A}\boldsymbol{\beta}' \cdot \boldsymbol{\beta}}. \quad (21)$$

Equation (21) is solvable for $\frac{\dot{\mathbf{v}}_0}{\dot{\omega}_0}$ by Newton’s method using the value of $\frac{\dot{\mathbf{v}}_0}{\dot{\omega}_0}$ for $\mu = 0$, obtainable directly from (9) and (10), as an initial estimate. Intuitively, the method iterates until the computed acceleration and angular acceleration agree with what would be yielded under Newton’s law by the finger acceleration and the frictional force, the latter of which in turn depends on the accelerations themselves under Coulomb’s law of friction.

The partial derivative $\partial \Gamma_0 / \partial \frac{\dot{\mathbf{v}}_0}{\dot{\omega}_0}$ required for the iterations can be evaluated numerically or using its closed form when \mathcal{B} is polygonal [28]. Hence $\dot{\omega}_0$ and $\dot{\mathbf{v}}_0$ are determined from (19) and (20).

2.4 Contact Breaking

The only constraint that was left out in the derivation of the differential equations (7)–(10) is inequality (5). This constraint, however, is used for checking when the contact between the finger and the object breaks. More specifically, the contact breaks when $R\boldsymbol{\beta}' \times F < 0$.

3 Local Observability

In the previous section we saw that the kinematics of contact and the dynamics of pushing are together determined by a system of nonlinear ordinary differential equations (7)–(10). A state of this nonlinear system consists of u and s , which determine the contact locations on the finger and on the object, respectively, the object’s angular velocity ω , velocity \mathbf{v} , and orientation θ ; the input is the finger’s acceleration $\mathbf{a}_{\mathcal{F}}$, generated by the controller of the finger; and the output is u , reported by a tactile sensor mounted on the finger. The sensing task becomes to “observe” s from u , which, as suggested by the system equations, is no easier than to “observe” the whole state of the system.

In this section we shall study local observability of one instantiation of the above system in which the finger is circular and the object is polygonal. This type of pushing is representative in real manipulation tasks. First of all, we introduce the notion of nonlinear observability as

⁸We here make a simplification by identifying the coefficient of static friction with the coefficient of kinetic friction μ .

well as a theorem about local observability; next, we show that the instantiation is locally observable. It will then not be difficult to see that these results can generalize to many other finger and object shapes.

3.1 Observability of a Nonlinear System

Let us consider a smooth affine (or input-linear) control system together with an output map:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \sum_{i=1}^m u_i \mathbf{g}_i(\mathbf{x}), & (u_1, \dots, u_m) \in U \subset \mathbb{R}^m, \\ \mathbf{y} &= \mathbf{h}(\mathbf{x}),\end{aligned}\tag{22}$$

where $\mathbf{x} = (x_1, \dots, x_n)^T$ is the *state* in a smooth n -dimensional manifold $M \subseteq \mathbb{R}^n$ (called the *state space manifold*), $\mathbf{f}, \mathbf{g}_1, \dots, \mathbf{g}_m$ are smooth vector fields on M , and $\mathbf{h} = (h_1, \dots, h_k)^T : M \rightarrow \mathbb{R}^k$ is the smooth output map of the system. We call \mathbf{f} the *drift vector field* and $\mathbf{g}_1, \dots, \mathbf{g}_m$ the *input vector fields*. In the system, u_1, \dots, u_m are the inputs, called the *controls*, over time whose Cartesian product range U defines the system's input space. At state \mathbf{x} , $\mathbf{f}(\mathbf{x})$ is a tangent vector to M representing the rate of change of \mathbf{x} when there is no input, while $\mathbf{g}_j(\mathbf{x})$ for $1 \leq j \leq m$ is a tangent vector showing the change of such rate under unit input of u_j .

Throughout we are only concerned with the class of controls \mathcal{U} that consists of piecewise constant functions that are continuous from the right.⁹ We call these controls *admissible*. The system with constant controls, or no input fields, equivalently, is said to be *autonomous*.

Denote by $\mathbf{y}(t, \mathbf{x}_0, \mathbf{u})$, $t \geq 0$, the output function of the system with initial state \mathbf{x}_0 and under control \mathbf{u} . Two states $\mathbf{x}_1, \mathbf{x}_2 \in M$ are said to be *indistinguishable* (denoted by $\mathbf{x}_1 I \mathbf{x}_2$) if for every admissible control \mathbf{u} the output functions $\mathbf{y}(t, \mathbf{x}_1, \mathbf{u})$ and $\mathbf{y}(t, \mathbf{x}_2, \mathbf{u})$, $t \geq 0$ are identical on their common domain of definition. The system is *observable* if $\mathbf{x}_1 I \mathbf{x}_2$ implies $\mathbf{x}_1 = \mathbf{x}_2$.

To derive a condition on nonlinear observability, the above definition of “observable” is localized in the following way. Let $V \subset M$ be an open set containing states \mathbf{x}_1 and \mathbf{x}_2 . These two states are said to be V -indistinguishable, denoted by $\mathbf{x}_1 I^V \mathbf{x}_2$, if for any $T > 0$ and any constant control $\mathbf{u} : [0, T] \rightarrow U$ such that $\mathbf{x}(t, \mathbf{x}_1, \mathbf{u}), \mathbf{x}(t, \mathbf{x}_2, \mathbf{u}) \in V$ for all $0 \leq t \leq T$, it follows that $\mathbf{y}(t, \mathbf{x}_1, \mathbf{u}) = \mathbf{y}(t, \mathbf{x}_2, \mathbf{u})$ for all $0 \leq t \leq T$ on their common domain of definition. The system is *locally observable* at \mathbf{x}_0 if there exists a neighborhood W of \mathbf{x}_0 such that for every neighborhood $V \subset W$ of \mathbf{x}_0 the relation $\mathbf{x}_0 I^V \mathbf{x}_1$ implies that $\mathbf{x}_0 = \mathbf{x}_1$. The system is called *locally observable* if it is locally observable at every $\mathbf{x}_0 \in M$. Figure 5 illustrates local observability for the case of one output function.

A *one-form* on M is a real-valued and pointwise linear function on the set of all tangent vectors to M . The *cotangent space* of M at state \mathbf{x} includes all the one-forms on M instantiated by \mathbf{x} . In particular, it includes for $1 \leq j \leq k$ the gradient of h_j : $dh_j = (\frac{\partial h_j}{\partial x_1}, \dots, \frac{\partial h_j}{\partial x_n})$. The *Lie derivative* of function $h_j : M \rightarrow \mathbb{R}$ along a vector field X on M , denoted by $L_X h_j$, is the directional derivative $dh_j(X) = dh_j \cdot X$. For simplicity, let notation $L_{X_1} L_{X_2} \cdots L_{X_l} h_j$

⁹So that \mathcal{U} is closed under concatenation.

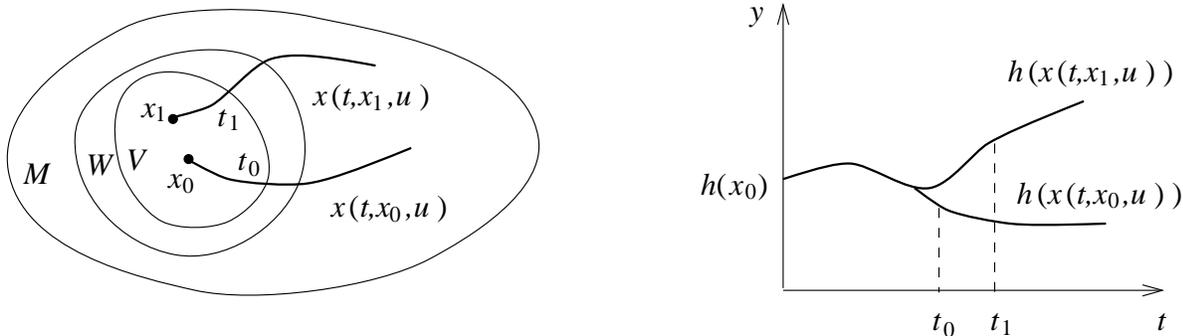


Figure 5: Local observability at state \mathbf{x}_0 . Given the state space M , $W \subset M$ is some neighborhood of \mathbf{x}_0 . For any neighborhood $V \subset W$ of \mathbf{x}_0 , \mathbf{x}_0 is V -distinguishable from all other states in V . More specifically, for any state $\mathbf{x}_1 \neq \mathbf{x}_0$ in V , there exists a constant admissible control \mathbf{u} such that the two state trajectories $\mathbf{x}(t, \mathbf{x}_0, \mathbf{u})$ and $\mathbf{x}(t, \mathbf{x}_1, \mathbf{u})$ will yield different outputs before one of them exits V (at time t_0).

stand for the repeated Lie derivative $L_{X_1}(L_{X_2}(\dots(L_{X_l}h_j)\dots))$ with respect to vector fields X_l, \dots, X_2, X_1 . The *observation space* \mathcal{O} of system (22) is the linear space (over \mathfrak{R}) of functions on M that includes h_1, \dots, h_k , and all repeated Lie derivatives

$$L_{X_1}L_{X_2}\cdots L_{X_l}h_j, \quad j = 1, \dots, k, \quad l = 1, 2, \dots$$

where $X_i \in \{\mathbf{f}, \mathbf{g}_1, \dots, \mathbf{g}_m\}$, $1 \leq i \leq l$. It is not difficult to show that \mathcal{O} is also the linear space of functions on M that includes h_1, \dots, h_k , and all repeated Lie derivatives

$$L_{Z_1}L_{Z_2}\cdots L_{Z_l}h_j, \quad j = 1, \dots, k, \quad l = 1, 2, \dots$$

where

$$Z_i(\mathbf{x}) = \mathbf{f}(\mathbf{x}) + \sum_{j=1}^m u_{ij} \mathbf{g}_j(\mathbf{x}), \quad (23)$$

for some point $\mathbf{u}_i = (u_{i1}, \dots, u_{im}) \in U$.

The observation space shall be better understood with the notion of integral curve. Given a nonlinear system

$$\dot{\mathbf{x}} = Z(\mathbf{x}),$$

defined by some vector field Z on the state space M , the *integral curve* $\sigma_{\mathbf{x}_0}(t)$ is the solution of the system satisfying the initial condition $\sigma_{\mathbf{x}_0}(0) = \mathbf{x}_0$. For every bounded subset $M_1 \subset M$, there exists an interval $(t_1, t_2) \ni 0$ on which the integral curve $\sigma_{\mathbf{x}_0}(t)$ is well-defined for all $t \in (t_1, t_2)$. This allows us to introduce on M_1 a set of maps, called the *flow*,

$$\begin{aligned} Z^t : M_1 &\rightarrow M, & t &\in (t_1, t_2), \\ \mathbf{x}_0 &\mapsto \sigma_{\mathbf{x}_0}(t). \end{aligned}$$

Now choose inputs of system (22) such that it is driven by a sequence of vector fields Z_1, \dots, Z_p of form (23) for small time t_1, \dots, t_p , respectively. The outputs of the system at $t_1 + \dots + t_p$ time are

$$h_i(Z_p^{t_p} \circ Z_{p-1}^{t_{p-1}} \circ \dots \circ Z_1^{t_1}(\mathbf{x}_0)), \quad \text{for } i = 1, \dots, k.$$

Differentiate these outputs sequentially with respect to t_p, t_{p-1}, \dots, t_1 at $t_p = 0, t_{p-1} = 0, \dots, t_1 = 0$ yields $L_{Z_1} L_{Z_2} \dots L_{Z_p} h_i(\mathbf{x}_0)$, for $i = 1, \dots, k$. Hence we see that the observation space in fact consists of the output functions and their derivatives along all possible system trajectories.

The *observability codistribution* at state $\mathbf{x} \in M$, denoted $d\mathcal{O}(\mathbf{x})$, is defined as:

$$d\mathcal{O}(\mathbf{x}) = \text{span}\{dH(\mathbf{x}) \mid H \in \mathcal{O}\}.$$

We are ready for a main theorem concerning local observability:

Theorem 2 (Herman and Krener) *System (22) is locally observable at state $\mathbf{x}_0 \in M$ if $\dim d\mathcal{O}(\mathbf{x}_0) = n$.*

The equation $\dim d\mathcal{O}(\mathbf{x}_0) = n$ is called the *observability rank condition*. Proofs of the above theorem can be found in [24] and [39, pp. 95–96]. Basically, to distinguish between a state and any other state in its neighborhood, it is necessary to consider not only the output functions but also their derivatives along all possible system trajectories. The rank condition ensures the existence of n output functions and/or derivatives that together define a diffeomorphism on some neighborhood of the state, which in turn ensures that the state is locally distinguishable.

3.2 The Disk-Polygon System

Now we study the case in which finger \mathcal{F} is a disk bounded by $\boldsymbol{\alpha} = r(\cos \frac{u}{r}, \sin \frac{u}{r})^T$ and object \mathcal{B} is a simple polygon. The interior of one edge e of \mathcal{B} maintains contact with \mathcal{F} throughout the pushing.¹⁰ We assume that e is known since local observability is concerned, and since a sensing strategy can hypothesize all edges of \mathcal{B} as the contact edge and verify them one by one. Let h be the distance from the centroid O of \mathcal{B} to e . Choose s as the signed distance from the contact to the intersection of e and its perpendicular through O such that s increases monotonically while moving counterclockwise (with respect to \mathcal{B} 's interior) on e . See Figure 6. The orientation of \mathcal{B} is $\theta = u/r - \pi/2$.¹¹ The tangent and normal of \mathcal{F} at the contact are $T = \boldsymbol{\alpha}' = (-\sin \frac{u}{r}, \cos \frac{u}{r})^T$ and $N = r\boldsymbol{\alpha}'' = -(\cos \frac{u}{r}, \sin \frac{u}{r})^T$, respectively. The system is governed by the following nonlinear equations as special cases of (7)–(10), respectively:¹²

$$\dot{u} = \omega r,$$

¹⁰This is easily realizable in a real pushing scenario.

¹¹Given a different contact edge e_1 it follows $\theta = u/r - \pi/2 + \theta_{e_1}$ for some constant θ_{e_1} .

¹²These equations assume that O and the disk center are on different sides of e . Otherwise the term $r + h$ in the equations for $\dot{s}, \dot{\omega}, \dot{\boldsymbol{v}}$ need to be replaced by $r - h$.

the same frame as $(v_{\mathcal{F}_T}, v_{\mathcal{F}_N})^T$ and $(a_{\mathcal{F}_T}, a_{\mathcal{F}_N})^T$, respectively. We find that v_N depends on s , ω , and $v_{\mathcal{F}_N}$ by taking the dot product of N with the velocity constraint (1):

$$v_N = v_{\mathcal{F}_N} + s\omega.$$

From the above equation, $\frac{dT}{dt} = \omega N$, and (24) we have

$$\begin{aligned} \dot{v}_T &= \frac{d(\mathbf{v} \cdot T)}{dt} \\ &= \mathbf{v} \cdot \frac{dT}{dt} + \dot{\mathbf{v}} \cdot T \\ &= \omega v_N - \frac{\mu g}{\mathcal{A}s} \Gamma_T \\ &= \omega v_{\mathcal{F}_N} + s\omega^2 - \frac{\mu g}{\mathcal{A}s} \Gamma_T, \end{aligned}$$

where $\Gamma_T = \Gamma \cdot T$.

System (24) is now rewritten as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + a_{\mathcal{F}_T} \mathbf{g}_T(\mathbf{x}) + a_{\mathcal{F}_N} \mathbf{g}_N(\mathbf{x}). \quad (26)$$

The state \mathbf{x} of the system becomes $(u, s, \omega, v_T, v_{\mathcal{F}_T}, v_{\mathcal{F}_N})^T$ with six variables in total; the inputs are the acceleration components $a_{\mathcal{F}_T}$ and $a_{\mathcal{F}_N}$ along the contact tangent and normal, respectively; and the output is a triple $\mathbf{y} = (u, v_{\mathcal{F}_T}, v_{\mathcal{F}_N})^T$. The drift and input fields are given by

$$\mathbf{f}(\mathbf{x}) = \begin{pmatrix} \omega r \\ v_T - v_{\mathcal{F}_T} - \omega(r + h) \\ \frac{s}{s^2 + \rho^2} \left(\omega^2(r + h) - 2\omega(v_T - v_{\mathcal{F}_T}) \right) - \frac{\mu g}{\mathcal{A}(s^2 + \rho^2)} \Gamma_N \\ \omega v_{\mathcal{F}_N} + s\omega^2 - \frac{\mu g}{\mathcal{A}s} \Gamma_T \\ 0 \\ 0 \end{pmatrix}, \quad (27)$$

$$\mathbf{g}_T(\mathbf{x}) = (0, 0, 0, 0, 1, 0)^T,$$

$$\mathbf{g}_N(\mathbf{x}) = \left(0, 0, -\frac{s}{s^2 + \rho^2}, 0, 0, 1 \right)^T.$$

Theorem 3 *The disk-polygon system (26) is locally observable.*

Proof By Theorem 2 it suffices to show that the observability codistribution $d\mathcal{O}$ has rank 6 at every state. Here the observation space \mathcal{O} consists of the outputs $u, v_{\mathcal{F}_T}, v_{\mathcal{F}_N}$ and their repeated Lie derivatives. We choose from \mathcal{O} the following functions and write out their

differentials:

$$\begin{aligned}
du &= (1, 0, 0, 0, 0); \\
dv_{\mathcal{F}_T} &= (0, 0, 0, 0, 1, 0); \\
dv_{\mathcal{F}_N} &= (0, 0, 0, 0, 0, 1); \\
L_f u &= du \cdot f = \omega r, & dL_f u &= (0, 0, r, 0, 0, 0); \\
L_{g_N} L_f u &= -r \frac{s}{s^2 + \rho^2}, & dL_{g_N} L_f u &= \left(0, r \frac{s^2 - \rho^2}{(s^2 + \rho^2)^2}, 0, 0, 0, 0\right); \\
L_{g_N} L_f L_{g_N} L_f u &= r(r+h) \frac{s(s^2 - \rho^2)}{(s^2 + \rho^2)^3}, & dL_{g_N} L_f L_{g_N} L_f u \Big|_{s=\rho} &= \left(0, \frac{r(r+h)}{4\rho^4}, 0, 0, 0, 0\right).
\end{aligned}$$

Thus $du, dv_{\mathcal{F}_T}, dv_{\mathcal{F}_N}, dL_f u$, and $dL_{g_N} L_f u$ (or $dL_{g_N} L_f L_{g_N} L_f u$ when $s = \rho$) span the cotangent space of the space of all possible 5-tuples $(u, s, \omega, v_{\mathcal{F}_T}, v_{\mathcal{F}_N})^T$. It suffices to find one more function in \mathcal{O} whose partial derivative with respect to v_T will not vanish.

Such a task is quite easy, for we have

$$\begin{aligned}
\frac{\partial(L_f L_{g_N} L_f u)}{\partial v_T} &= r \frac{s^2 - \rho^2}{(s^2 + \rho^2)^2}; \\
\frac{\partial}{\partial v_T}(L_f L_{g_N} L_f L_{g_N} L_f u) \Big|_{s=\rho} &= \frac{r(r+h)}{4\rho^4}.
\end{aligned}$$

In summary, the observability codistribution $d\mathcal{O}$ is spanned by $du, dv_{\mathcal{F}_T}, dv_{\mathcal{F}_N}, dL_f u, dL_{g_N} L_f u$ and $dL_f L_{g_N} L_f u$ (or $dL_{g_N} L_f L_{g_N} L_f u$ and $dL_f L_{g_T} L_f L_{g_N} L_f u$ when $s = \rho$) and thus attains full rank. \square

The above proof in fact constructs several control sequences which, when applied for infinitesimal amounts of time, will distinguish between different states in any neighborhood. Assuming $s \neq \eta$, one of the functions $u, v_{\mathcal{F}_T}, v_{\mathcal{F}_N}, L_f u, L_{g_N} L_f u$, and $L_f L_{g_N} L_f u$ must have different values in any two different states close enough as guaranteed by the observability rank condition. Note that $L_f u$ is in fact the differential output under zero control. Since $L_{g_N} L_f u$ may be written as $\frac{1}{2}L_{f+g_N} L_f u - \frac{1}{2}L_{f-g_N} L_f u$, one of these two functions must distinguish the two states if $L_{g_N} L_f u$ does. Obviously, $L_{f+g_N} L_f u$ (or $L_{f-g_N} L_f u$) is realizable in an arbitrarily small amount of time by the control sequence starting with zero control and ending with $a_{\mathcal{F}_N} = 1$ (or -1). The case with function $L_f L_{g_N} L_f u$ is similar.

Moreover, the proof reveals the relative ‘‘hardness’’ of observing the state variables, especially u, s , and ω . The disk contact u constitutes the system output and thus is the easiest to observe. The angular velocity ω of the polygon needs to be obtained from the first order derivative of u . The polygon contact s , the hardest of the three to observe, requires a Lie derivative of the second order or above, which is obtained using two or more controls.

Support friction does not affect the local observability of the disk-polygon system, as none of the differentials chosen in the proof to span $d\mathcal{O}$ involve the integral Γ or any of its partial derivatives.

The proof makes use of the input vector field \mathbf{g}_N but not \mathbf{g}_T , which suggests that pushing along a tangential direction is unnecessary for the purpose of local observability. Intuition tells us that pushing along the contact normal will more likely help the disk observe the polygon.

We conjecture that the autonomous version of the system (under $\mathbf{a}_{\mathcal{F}} = 0$) is locally observable at all except a finite number of states. Although it seems much more difficult to prove the linear independence of du , $dv_{\mathcal{F}_T}$, $dv_{\mathcal{F}_N}$, $dL_f u$, $dL_f^2 u$, and $dL_f^3 u$ at every state, this conjecture will be supported by our simulation results later in Section 6.

4 Pose Observers

With local observability, we can view sensing strategies as nonlinear observers for the disk-polygon system (26) or for the general pushing system (7)–(10). An *observer* of a nonlinear system is a system whose state converges to the state of the original system. The input of the observer consists of the input as well as the output of the original system.

Luenberger-like asymptotic observers [34] for nonlinear systems are often designed through linearization. The disk-polygon system (26), however, cannot be linearized for we have

$$L_{g_N} L_f L_{g_N} L_f u = r(r+h) \frac{s(s^2 - \rho^2)}{(s^2 + \rho^2)^3},$$

violating one of Nijmeijer’s necessary conditions [39, p. 156] on linearization. Another approach of observer design transforms the original system into a linear system modulo an output injection [32]. The necessary conditions for a nonlinear system to admit linear observer error dynamics are rather restrictive and hardly satisfied by the disk-polygon system, let alone system (7)–(10). Even if these conditions hold, it is still quite burdensome (and sometimes impossible) to find explicit solutions to partial differential equations involving repeated Lie brackets on which the desired coordinate transformation must be based.

Our observer, for the disk-polygon system only, uses a result by Gauthier, Hammouri and Othman to be introduced next.

4.1 A Gauthier-Hammouri-Othman Observer

We shall apply the Gauthier-Hammouri-Othman (GHO) procedure [15] to find an observer for the disk-polygon system (26).

Theorem 4 (Gauthier, Hammouri, and Othman) *Consider the single output nonlinear (and analytic) system*

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}), \\ y &= h(\mathbf{x}), \end{aligned} \tag{28}$$

defined on n -dimensional state space manifold M . Suppose the following two conditions hold:

1. the mapping $Z : \mathbf{x} \mapsto \mathbf{z} = \begin{pmatrix} h(\mathbf{x}) \\ L_f h(\mathbf{x}) \\ \vdots \\ L_f^{n-1} h(\mathbf{x}) \end{pmatrix}$ is a diffeomorphism on M ,

2. $L_f^n h(\mathbf{x})$ can be extended from M to \mathfrak{R}^n by a C^∞ function that is globally Lipschitzian on \mathfrak{R}^n .

Let $C = (1, 0, \dots, 0)$. Let $A = (a_{ij})$ be an $n \times n$ matrix with $a_{ij} = 1$ if $i = j - 1$ and 0 if $i \neq j - 1$, and $S_\infty(\zeta)$ be the $n \times n$ matrix that satisfies the equation

$$-\zeta S_\infty - A^T S_\infty - S_\infty A + C^T C = \mathbf{0}, \quad (29)$$

where ζ is some large enough constant. Then the system

$$\dot{\tilde{\mathbf{x}}} = \mathbf{f}(\tilde{\mathbf{x}}) - (h(\tilde{\mathbf{x}}) - y) \frac{\partial Z^{-1}}{\partial \mathbf{z}}(Z(\tilde{\mathbf{x}})) S_\infty^{-1} C^T \quad (30)$$

is an observer for (28) with error dynamics

$$\|\tilde{\mathbf{x}}(t) - \mathbf{x}(t)\| \leq K(\zeta) e^{-\frac{\zeta t}{3}} \|\tilde{\mathbf{x}}(0) - \mathbf{x}(0)\|,$$

where $K(\zeta)$ is some constant.

The proof of the above theorem given in [15] is based on standard Lyapunov arguments. The parameter ζ control the speed of the observer. The matrix $S_\infty(\zeta) = (s_{ij})$ is the limit of the stationary solution of $\dot{S}_t(\zeta) = -\zeta S_t(\zeta) - A^T S_t(\zeta) - S_t(\zeta) A + C^T C$ as $t \rightarrow \infty$, with the initial value $S_0(\zeta)$ being any symmetric positive definite matrix.¹³ The symmetric matrix S_∞ can be determined by starting from its first row and column simultaneously and progressing to higher ordinal pairs of rows and columns. We observe $s_{11} = \frac{1}{\zeta}$ and let $s_{0j} = s_{j0} = 0$, for $j = 1, 2, \dots, n$. Then the remaining entries of S_∞ satisfy a three-term recurrence relation:

$$s_{ij} = -\frac{1}{\zeta} (s_{i-1,j} + s_{i,j-1}), \quad i > 1 \text{ or } j > 1.$$

The observer (30) is a copy of the original system (28) with a corrective term that does not depend on system (28) but only on the dimension and the desired convergence speed ζ .

The GHO observer for a general nonlinear system (22) with inputs is a copy of the original system plus the error corrective term given in (30). To have such an observer, not only must conditions 1 and 2 in the above theorem hold for the drift system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$, but also the original system must be observable for any input.

Getting back to the disk-polygon system (26), we now need to consider only u, s, ω , and v_T as state variables. The drift and input fields reduce from (27) to

$$\mathbf{f}(\mathbf{x}) = \begin{pmatrix} \omega r \\ v_T - v_{\mathcal{F}_T} - \omega(r + h) \\ \frac{s}{s^2 + \rho^2} (\omega^2(r + h) - 2\omega(v_T - v_{\mathcal{F}_T})) - \frac{\mu g}{\mathcal{A}(s^2 + \rho^2)} \Gamma_N \\ \omega v_{\mathcal{F}_N} + s\omega^2 - \frac{\mu g}{\mathcal{A}s} \Gamma_T \end{pmatrix}; \quad (31)$$

¹³So is $S_t(\zeta)$ for $t > 0$ symmetric positive definite.

$$\begin{aligned}\mathbf{g}_T(\mathbf{x}) &= (0, 0, 0, 0)^T; \\ \mathbf{g}_N(\mathbf{x}) &= \left(0, 0, -\frac{s}{s^2 + \rho^2}, 0\right)^T.\end{aligned}$$

With u being the system's only output, the new coordinates under map χ consist of u and its Lie derivatives, up to the third order:

$$\begin{pmatrix} u \\ s \\ \omega \\ v_T \end{pmatrix} \xrightarrow{\chi} \mathbf{x} = \begin{pmatrix} u \\ \omega r \\ rL_f\omega \\ rL_f^2\omega \end{pmatrix}.$$

Generally, for all except at most a finite number of states, $du, rd\omega, rdL_f\omega$, and $rdL_f^2\omega$ are linearly independent, which implies that the map χ is locally diffeomorphic. The Jacobian of the inverse transformation χ^{-1} is then the inverse of the Jacobian of χ [45, p. 2-17]:

$$\frac{\partial\chi^{-1}}{\partial\mathbf{x}} = \begin{pmatrix} du \\ rd\omega \\ rdL_f\omega \\ rdL_f^2\omega \end{pmatrix}^{-1}.$$

The differential $dL_f\omega$ consists of the following partial derivatives:

$$\begin{aligned}\frac{\partial L_f\omega}{\partial u} &= 2\omega \frac{s}{s^2 + \rho^2} \frac{v_{\mathcal{F}_N}}{r} - \frac{\mu g}{\mathcal{A}(s^2 + \rho^2)} \left(\frac{\partial\Gamma}{\partial u} \cdot N - \frac{\Gamma_T}{r} \right); \\ \frac{\partial L_f\omega}{\partial s} &= \frac{\rho^2 - s^2}{(s^2 + \rho^2)^2} (\omega^2(r + h) - 2\omega(v_T - v_{\mathcal{F}_T})) + \frac{2\mu g s}{\mathcal{A}(s^2 + \rho^2)^2} \Gamma_N - \frac{\mu g}{\mathcal{A}(s^2 + \rho^2)} \frac{\partial\Gamma}{\partial s} \cdot N; \\ \frac{\partial L_f\omega}{\partial \omega} &= 2 \frac{s}{s^2 + \rho^2} (\omega(r + h) - (v_T - v_{\mathcal{F}_T})) - \frac{\mu g}{\mathcal{A}(s^2 + \rho^2)} \frac{\partial\Gamma}{\partial \omega} \cdot N; \\ \frac{\partial L_f\omega}{\partial v_T} &= -2\omega \frac{s}{s^2 + \rho^2} - \frac{\mu g}{\mathcal{A}(s^2 + \rho^2)} \left(\frac{\partial\Gamma}{\partial v} T \right) \cdot N,\end{aligned}$$

where the closed form of $d\Gamma$ (on u, s, ω, v_T) can be derived (see [28]). The differential $dL_f^2\omega$, however, involves second order partial derivatives of Γ whose closed forms is too complicated to obtain. Hence we choose to evaluate $dL_f^2\omega$ numerically.

Solve equation (29) under $n = 4$ and take the inverse of the solution:

$$S_\infty = \begin{pmatrix} \frac{1}{\zeta} & -\frac{1}{\zeta^2} & \frac{1}{\zeta^3} & -\frac{1}{\zeta^4} \\ -\frac{1}{\zeta^2} & \frac{2}{\zeta^3} & -\frac{3}{\zeta^4} & \frac{4}{\zeta^5} \\ \frac{1}{\zeta^3} & -\frac{3}{\zeta^4} & \frac{6}{\zeta^5} & -\frac{10}{\zeta^6} \\ -\frac{1}{\zeta^4} & \frac{4}{\zeta^5} & -\frac{10}{\zeta^6} & \frac{20}{\zeta^7} \end{pmatrix} \quad \text{and} \quad S_\infty^{-1} = \begin{pmatrix} 4\zeta & 6\zeta^2 & 4\zeta^3 & \zeta^4 \\ 6\zeta^2 & 14\zeta^3 & 11\zeta^4 & 3\zeta^5 \\ 4\zeta^3 & 11\zeta^4 & 10\zeta^5 & 3\zeta^6 \\ \zeta^4 & 3\zeta^5 & 3\zeta^6 & \zeta^7 \end{pmatrix}.$$

Finally, from (30) and (31) we obtain an GHO observer for frictionless contact:

$$\begin{pmatrix} \dot{\tilde{u}} \\ \dot{\tilde{s}} \\ \dot{\tilde{\omega}} \\ \dot{\tilde{v}}_T \end{pmatrix} = \mathbf{f}(\tilde{u}, \tilde{s}, \tilde{\omega}, \tilde{v}_T) - (\mathbf{a}_{\mathcal{F}} \cdot N(\tilde{u})) \mathbf{g}_N(\tilde{s}) - \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & r & 0 \\ rdL_f\omega(\tilde{u}, \tilde{s}, \tilde{\omega}, \tilde{v}_T) \\ rdL_f^2\omega(\tilde{u}, \tilde{s}, \tilde{\omega}, \tilde{v}_T) \end{pmatrix}^{-1} \begin{pmatrix} 4\zeta \\ 6\zeta^2 \\ 4\zeta^3 \\ \zeta^4 \end{pmatrix} (\tilde{u} - u). \quad (32)$$

It should be noted that we did not verify condition 2 in Theorem 4. The Lie derivative $L_f^4 u$ is generally not extendible to a globally Lipschitzian function. However, $L_f^4 u$ is locally Lipschitzian. So the observer should work well as long as the state estimate is close to the real state and its trajectory does not exit the local neighborhood in which the Lipschitz condition holds. This will be supported by the simulation results in Section 6.1.

4.2 The Initial Pose Observer

The asymptotic observer presented in Section 4.1 has two drawbacks. First, for finger and object shapes other than disks and polygons, the computation of the Lie derivatives and the Jacobian may become a burden. Second, the observer requires a sequence of contact locations on the finger to be sensed, which may cause difficulties in sensor implementation.

One sensing strategy is to observe the initial object pose. The state of the pushing system at any time will then be determined from equations (7)–(10) given that the finger’s pose and velocity during the pushing are known (to the controller). The initial object pose is determined by the initial contact position s_0 on the object boundary. So is the contact position $u(t)$ on the finger. This fact leads to our second observer which is a variation of the shooting method for integrating ordinary differential equations. This observer is for system (7)–(10) in which the finger and the object have *general* planar shapes.

For each initial object contact s_0 , there is a unique finger contact trajectory $u(t) \equiv u(t; s_0)$, $t \geq 0$, as the solution to the differential equations (7)–(10) with the initial values including s_0 . Note $u(0; s_0) = u_0$ must hold, where u_0 is the initial finger contact. Let the finger sense a second contact position u_1 at time $t_1 > 0$. Then the problem reduces to finding a zero s_0^* of the function $u(t_1; s_0) - u_1$. Figure 7 depicts the contact curves resulting from two different initial object contacts, together with the curve segment $g(s_0) = u(t_1; s_0)$ representing all possible finger contacts at time t_1 resulting from any initial object contact in between.

The root s_0^* of $u(t_1; s_0) - u_1$ can be obtained iteratively with Newton’s method for solving nonlinear equations. Each evaluation of this function now involves solving a separate initial value problem for the system (7)–(10) given the value of s_0 at the present iteration step.

The initial pose (IP) observer is also local and therefore subject to how close the estimate on s_0 at the start of iteration is from the real pose s_0^* . To globalize sensing, we provide Newton’s method with multiple guesses of s_0 along the object boundary. This may yield multiple solutions to $u(t_1; s_0) = u_1$, as we will see in the simulation results in Section 6.2. However, such ambiguities can often be resolved by detecting a third contact u_2 on the finger at time $t_2 > t_1$ and verifying against u_2 the finger contacts at t_1 resulting from all ambiguous s_0 values.

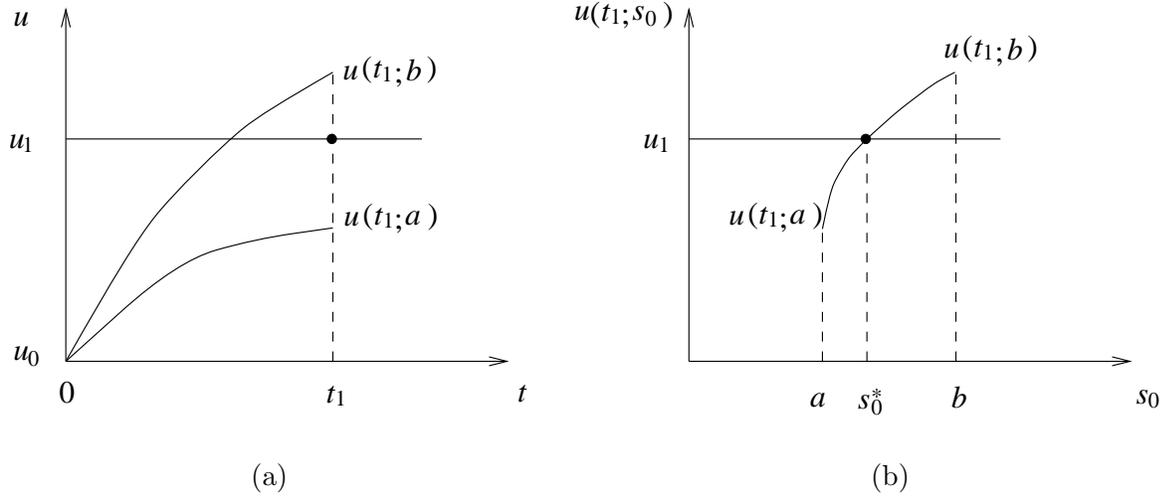


Figure 7: A shooting method for initial pose determination. (a) Two different finger contact motions resulting from the initial object poses $s_0 = a$ and $s_0 = b$, respectively; (b) possible finger contacts at time t_1 resulting from any initial pose $s_0 \in [a, b]$ constitute a curve $g(s_0) = u(t_1; s_0)$. The initial pose observer works by intersecting $g(s_0)$ with the line $u(t_1) = u_1$ to determine the real initial object pose s_0^* .

5 Contact Friction

This section extends the results in the previous sections to include contact friction between the finger and the object. Now we need to consider two modes of contact: *rolling* and *sliding*, according as whether the contact force lies inside the contact friction cone or on one of its two edges (see Figure 8). Each mode is hypothesized and solved; then the obtained contact force is verified with the contact friction cone for consistency. This hypothesis-and-test approach is quite common in solving multi-rigid-body contact problem with Coulomb friction. (See, for instance, Haug *et al.* [23].)

5.1 Rolling

When rolling contact occurs, the contact force F may lie anywhere inside the contact friction cone. Let μ_c be the coefficient of contact friction. Constraint (4) for frictionless contact must now be replaced by

$$R\left(\frac{\pi}{2} + \phi\right)R\beta' \times F < 0 < R\left(\frac{\pi}{2} - \phi\right)R\beta' \times F, \quad (33)$$

where $\phi = \tan^{-1} \mu_c$ is the half angle of the contact friction cone and R, β, F are defined in Section 2. Furthermore, the two points in contact, fixed on α and β , respectively, must have the same instantaneous velocity; that is,

$$\mathbf{v}_{\mathcal{F}} = \mathbf{v} + \boldsymbol{\omega} \times R\beta. \quad (34)$$

Subtracting (34) from the velocity constraint (1) on contact maintenance yields

$$\alpha' \dot{u} = R\beta' \dot{s}. \quad (35)$$

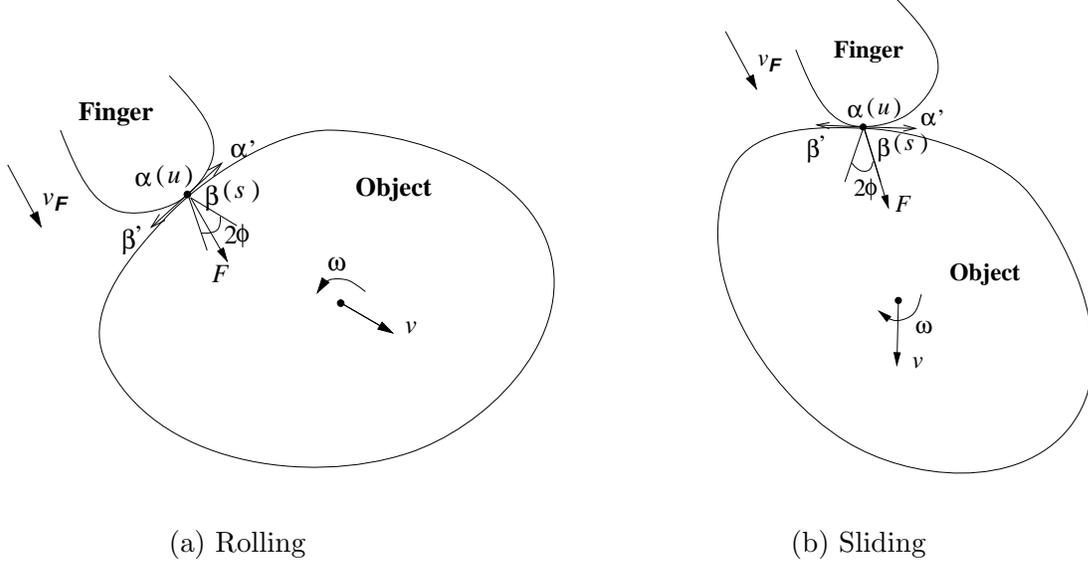


Figure 8: Two modes of motion under contact friction: (a) rolling, in which the contact force F points to the interior of the contact friction cone; (b) sliding, in which F points along one of the edges of the cone.

We are now ready to set up the contact and object motion equations for rolling.

Proposition 5 *In the problem of a translating finger pushing an object considered in Section 2, assume contact friction between the finger and the object as well. In addition to the notation of Section 2, let μ_c and $\phi = \tan^{-1} \mu_c$ be the coefficient and the angle of contact friction, respectively. When the object is rolling along the finger boundary, the pushing system is determined by (2), (3), (6), (33), (34), and (35). The contact and object motions satisfy*

$$\dot{u} = -\omega \frac{(\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta}')^2}{(\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta}')(\boldsymbol{\alpha}'' \times R\boldsymbol{\beta}') + \|\boldsymbol{\alpha}'\|^2(\boldsymbol{\alpha}' \times R\boldsymbol{\beta}'')}, \quad (36)$$

$$\dot{s} = -\omega \frac{\|\boldsymbol{\alpha}'\|^2(\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta}')}{(\boldsymbol{\alpha}' \cdot R\boldsymbol{\beta}')(\boldsymbol{\alpha}'' \times R\boldsymbol{\beta}') + \|\boldsymbol{\alpha}'\|^2(\boldsymbol{\alpha}' \times R\boldsymbol{\beta}'')}, \quad (37)$$

$$\dot{\omega} = \frac{R\boldsymbol{\beta} \times \mathbf{a}_{\mathcal{F}} - (\boldsymbol{\beta} \cdot \boldsymbol{\beta}')\omega\dot{s} + \frac{\mu g}{\mathcal{A}} \int_{\mathcal{B}} R(\boldsymbol{\beta} - \mathbf{p}) \times \hat{\mathbf{v}}_p d\mathbf{p}}{\|\boldsymbol{\beta}\|^2 + \rho^2}, \quad (38)$$

$$\mathbf{v} = \mathbf{v}_{\mathcal{F}} - \omega \times R\boldsymbol{\beta}.$$

Proof Equations (36) and (37) are just the special cases of equations (7) and (8), respectively, under the rolling constraint (34).

Differentiate both sides of (34):

$$\mathbf{a}_{\mathcal{F}} = \dot{\mathbf{v}} + \dot{\omega} \times R\boldsymbol{\beta} - \omega^2 R\boldsymbol{\beta} + \omega \times R\boldsymbol{\beta}'\dot{s}. \quad (39)$$

Meanwhile, substituting Newton's equation (2) into Euler's equation (3) and manipulate the resulting terms to obtain

$$R\boldsymbol{\beta} \times \dot{\mathbf{v}} = \frac{1}{m} \left(I\dot{\omega} + \mu\eta g \int_{\mathcal{B}} R(\mathbf{p} - \boldsymbol{\beta}) \times \hat{\mathbf{v}}_p d\mathbf{p} \right). \quad (40)$$

Taking the cross products of $R\boldsymbol{\beta}$ with both sides of (39) and plugging (40) in, we have after a few steps of term expansion:

$$R\boldsymbol{\beta} \times \mathbf{a}_{\mathcal{F}} = \frac{1}{m} \left(I\dot{\omega} + \mu\eta g \int_{\mathcal{B}} R(\mathbf{p} - \boldsymbol{\beta}) \times \hat{\mathbf{v}}_p d\mathbf{p} \right) + \|\boldsymbol{\beta}\|^2 \dot{\omega} + (\boldsymbol{\beta} \cdot \boldsymbol{\beta}') \omega \dot{s},$$

from which (38) immediately follows. \square

To investigate local observability in the presence of contact friction, we look at the same problem of a disk pushing a polygon considered before. In fact, local observability for the case of rolling can be established more easily. Under rolling contact, \mathbf{v} depends on u, s, ω :

$$\begin{aligned} \mathbf{v} &= \mathbf{v}_{\mathcal{F}} - \omega \times R\boldsymbol{\beta} \\ &= \mathbf{v}_{\mathcal{F}} - \omega \times (h\mathbf{N} - s\mathbf{T}) \\ &= \mathbf{v}_{\mathcal{F}} + \omega(h\mathbf{T} + s\mathbf{N}). \end{aligned} \quad (41)$$

Subsequently, a state can be denoted by $\mathbf{x} = (u, s, \omega)^T$.¹⁴ And the dynamical system (26) has simpler drift and input fields:

$$\begin{aligned} \mathbf{f} &= \begin{pmatrix} \omega r \\ -\omega r \\ \frac{rs\omega^2 + \frac{\mu g}{\mathcal{A}} \int_{\mathcal{B}} R(\boldsymbol{\beta} - \mathbf{p}) \times \hat{\mathbf{v}}_p d\mathbf{p}}{s^2 + h^2 + \rho^2} \end{pmatrix}, \\ \mathbf{g}_T &= \left(0, 0, -\frac{h}{s^2 + h^2 + \rho^2} \right)^T, \\ \mathbf{g}_N &= \left(0, 0, -\frac{s}{s^2 + h^2 + \rho^2} \right)^T. \end{aligned} \quad (42)$$

We leave to the reader the task of verifying that the differentials du , $dL_{\mathcal{F}}u$, and $dL_{\mathbf{g}_T}L_{\mathcal{F}}u$ (or $dL_{\mathbf{g}_N}L_{\mathcal{F}}u$ if $s = 0$) are linearly independent.

Theorem 6 *The disk-polygon system (42) with rolling contact between the disk and the polygon and under support friction in the plane is locally observable.*

¹⁴Unlike in the case of frictionless contact, here $v_{\mathcal{F}_T}$ and $v_{\mathcal{F}_N}$ are not involved in the dynamics of rolling. So they are not considered as state variables.

The GHO observer for the rolling case has the form

$$\begin{pmatrix} \dot{\tilde{u}} \\ \dot{\tilde{s}} \\ \dot{\tilde{\omega}} \end{pmatrix} = \mathbf{f}(\tilde{u}, \tilde{s}, \tilde{\omega}) + (\mathbf{a}_{\mathcal{F}} \cdot T(\tilde{u})) \mathbf{g}_T(\tilde{s}) + (\mathbf{a}_{\mathcal{F}} \cdot N(\tilde{u})) \mathbf{g}_N(\tilde{s}) \\ - \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & r \\ rdL_f\omega(\tilde{u}, \tilde{s}, \tilde{\omega}) \end{pmatrix}^{-1} \begin{pmatrix} 3\zeta \\ 3\zeta^2 \\ \zeta^3 \end{pmatrix} (\tilde{u} - u). \quad (43)$$

The derivation of (43) is similar to that of (32) and thus omitted here.

5.2 Sliding

When sliding contact occurs, F must lie along one edge of the contact friction cone that makes an obtuse angle with the sliding direction. Constraint (4) must be accordingly replaced by

$$F \cdot R(\theta \pm \phi) \boldsymbol{\beta}' = 0, \quad (44)$$

where “ \pm ” is determined by the sliding direction, which is hypothesized. Rewrite constraint (44) as $F \cdot R\tilde{\boldsymbol{\beta}}' = 0$ where $\tilde{\boldsymbol{\beta}}' = R(\pm\phi)\boldsymbol{\beta}'$. Also denote $\tilde{\boldsymbol{\beta}}' = R(\pm\phi)\boldsymbol{\beta}'$ and $\tilde{\Gamma} = \int_{\mathcal{B}} R\tilde{\boldsymbol{\beta}}' \times (R\mathbf{p} \times \hat{\mathbf{v}}_p) + (\tilde{\boldsymbol{\beta}}' \cdot \boldsymbol{\beta}) \hat{\mathbf{v}}_p d\mathbf{p}$. The differential equations governing contact and object motions are similar to those under no contact friction given in Section 2).

Proposition 7 *In the problem of a translating finger pushing an object considered in Section 2, assume contact friction between the finger and the object as well. In addition to the notation of Section 2, let μ_c and $\phi = \tan^{-1} \mu_c$ be the coefficient and the angle of contact friction, respectively. When the object is sliding along the finger boundary, the pushing system is determined by (1), (2), (3), (5), (6), and (44). The contact motions still follow (7) and (8), while the object’s angular acceleration and acceleration satisfy*

$$\dot{\omega} = \frac{\dot{u} \boldsymbol{\alpha}'' \times (\mathbf{v}_{\mathcal{F}} - \mathbf{v}) + \boldsymbol{\alpha}' \times \mathbf{a}_{\mathcal{F}} - \left(\dot{u} \boldsymbol{\alpha}'' \cdot R\boldsymbol{\beta} + \boldsymbol{\alpha}' \cdot (\omega \times R\boldsymbol{\beta} + R\boldsymbol{\beta}' \dot{s}) \right) \omega + \frac{\mu g}{\mathcal{A} \tilde{\boldsymbol{\beta}}' \cdot \boldsymbol{\beta}} \boldsymbol{\alpha}' \times \tilde{\Gamma}}{\boldsymbol{\alpha}' \cdot R \left(\boldsymbol{\beta} + \frac{\rho^2}{\tilde{\boldsymbol{\beta}}' \cdot \boldsymbol{\beta}} \tilde{\boldsymbol{\beta}}' \right)}, \quad (45)$$

$$\dot{\mathbf{v}} = \frac{\mathcal{A} \rho^2 \dot{\omega} \times R\tilde{\boldsymbol{\beta}}' - \mu g \tilde{\Gamma}}{\mathcal{A} \tilde{\boldsymbol{\beta}}' \cdot \boldsymbol{\beta}}. \quad (46)$$

Proof Analogous to the proof of Theorem 1. \square

The resemblance of equations (45) and (46) to equations (9) and (10) suggests the reasoning on local observability for the disk-polygon system in the sliding case to resemble the

proof of Theorem 3. We write the system in this case into the form of (26) and obtain its drift and input fields:

$$\mathbf{f} = \begin{pmatrix} \omega r \\ v_T - v_{\mathcal{F}_T} - \omega(r+h) \\ \text{---} \\ \text{---} \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{g}_N = \begin{pmatrix} 0 \\ 0 \\ -\frac{s \cos \phi \pm h \sin \phi}{s^2 \cos \phi \pm h s \sin \phi + \rho^2 \cos \phi} \\ \pm \sin \phi \frac{\rho^2}{s^2 \cos \phi \pm h s \sin \phi + \rho^2 \cos \phi} \\ 0 \\ 1 \end{pmatrix},$$

and \mathbf{g}_T as given in (27), where “ \pm ” stands for “+” for left sliding of the polygon and “-” for right sliding, and “---” for some complicated terms. Involved calculations will reveal that $du, dv_{\mathcal{F}_T}, dv_{\mathcal{F}_N}, dL_f u, dL_{g_N} L_f u$ (or $dL_{g_N} L_f L_{g_N} L_f u$), and $dL_f L_{g_N} L_f u$ (or $dL_f L_{g_N} L_f L_{g_N} L_f u$) again span the observability codistribution unless $\tan \phi = \frac{r+h}{\rho}$.

Theorem 8 *The disk-polygon system with sliding contact between the disk and the polygon and under support friction in the plane is locally observable if $\tan \phi \neq \frac{r+h}{\rho}$.*

6 Simulations and Experiments

We simulated the GHO observer and the IP observer by the fourth-order Runge-Kutta integration with a stepsize corresponding to 0.01 second (0.01s) real time. The object data in our simulations included polygons and ellipses,¹⁵ all of which were randomly generated.¹⁶

Our proofs of the local observability of the disk-polygon system and its variations in Sections 3.2, 5.1, and 5.2 did not rely on support friction. This suggests that friction would hardly affect the observer’s performance. So we set the coefficient of support friction to be uniformly 0.3. This number was also consistent with the measurements in our experiments, which are to be discussed in the next section. The finger accelerations and velocities used in the simulations are easily achievable on an Adept robot. For convenience, only constant finger accelerations were used.

The simulation code was written in Lisp and run on a Sparcstation 20. The major load of computation turned out to have come from the evaluations of the integrals of friction and their partial derivatives (with respect to the object pose and velocities). To speed up, these integrals and their first order partial derivatives were evaluated via closed forms when the object was a polygon. In such a case, each evaluation took time linear in the number of the polygon vertices (see [28] for the algorithm). For instance, evaluating Γ given by (17) for a 7-gon took 0.183s and evaluating its partial derivatives took 1.118s; while evaluating Γ and its partial derivatives for a triangle took only 0.067s and 0.412s, respectively.

¹⁵The latter shapes are for the initial pose observer only.

¹⁶The polygons were constructed by taking random walks on the arrangement of a large number of random lines precomputed by a topological sweeping algorithm [10].

6.1 On the GHO Observer

We simulated two versions of the GHO observer for the disk-polygon system: (32) in the case of frictionless contact ($\mu_c = 0$) between the disk and the polygon and (43) in the case of rolling contact. The first version has four state variables: the disk contact u , the edge contact s (which determines the polygon’s pose), the polygon’s angular velocity ω , and the tangential component of the polygon’s velocity v_T . The second version has only three: u , s , ω . The case of sliding contact was not simulated mainly because it is very similar to the case of frictionless contact except its nonlinear system is more complicated.

The magnitude of the control parameter ζ of the GHO observer directly affects its performance. When ζ is too small, the observer would either converge its estimate to the real state very slowly or not converge at all. In this case, the error correction would be dominated by the original system’s drift field such that it may not be enough to drive the estimate to some neighborhood of the real state where it can converge. On the other hand, when ζ is too large, the error correction would dominate the original system, causing the state estimate to change dramatically and often to diverge. Based on numerous trials, we chose $\zeta = 10$ in our simulations.

The disk radius was normalized to 1cm in all simulations. All time measurements will refer to how long the pushing would have taken place in the real world rather than how long the computation took.¹⁷

To get an idea of the observer’s behavior, let us look at a simple example of a 7-gon being pushed by the unit disk and rolling on its boundary (see Figure 9). The trajectories of u , s , ω and their estimates \tilde{u} , \tilde{s} , $\tilde{\omega}$ are show in Figure 10. Since the disk contact u is also the output, its estimate \tilde{u} converges faster than the estimates of other state variables. However, this had caused the following problem in many other instances we simulated: The feedback $\tilde{u} - u$ that drives the observer’s error corrective term, would usually diminish fast and become ineffective before other estimates can be corrected. To remedy this problem, our observer turns off error correction in the last 0.04s of every 0.1s interval of pushing so that the error $\tilde{u} - u$ would accumulate a bit for the corrective term to become effective again at the start of the next 0.1s interval. This scheme has turned out to be quite effective at driving other state variable estimates toward convergence.

We first conducted tests assuming known contact edges. In each test, a state and an estimate were randomly generated over the ranges of the state variables.¹⁸ The test would be regarded as a success as soon as the difference between the state and its estimate had become negligible for a period of time;¹⁹ it would be regarded as a failure if one of the state variables had gone out of its range repeatedly or there was no success after a long period of

¹⁷Simulating 0.8s observation of a pushed quadrilateral with rolling contact took 232s, while simulating 0.66s observation of the same quadrilateral with frictionless contact took 1012s.

¹⁸The range of u in terms of the polar angle with respect to the disk center was set to be the interval $[80, 100]$ (degrees); the range of s was determined from the contact edge; the ranges of ω and v_T were set as $[-1, 1]$ (rad/s) and $[-0.4, 0.4]$ (cm/s²), which were based on the velocity range of the Adept robot and on our simulation data of pushing.

¹⁹The length of the period can be arbitrarily set but should be large enough. It was chosen as 0.2s in our simulations.

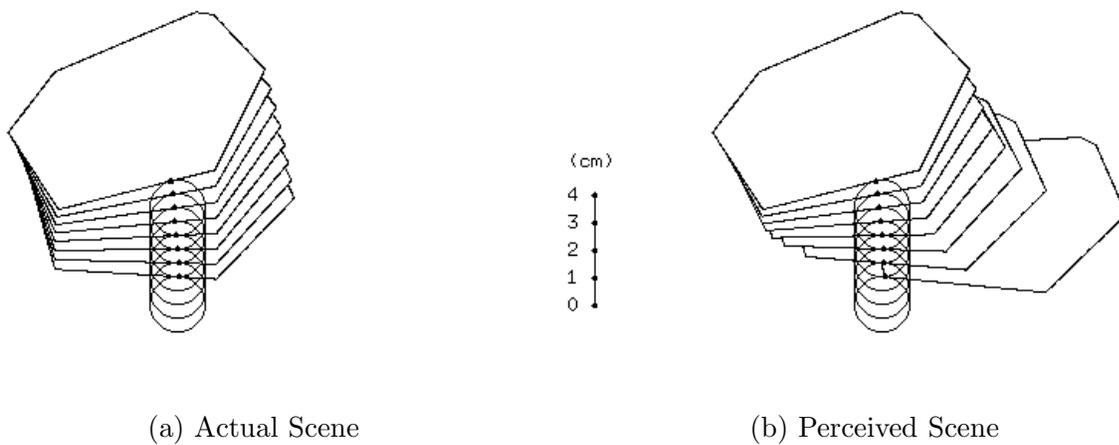
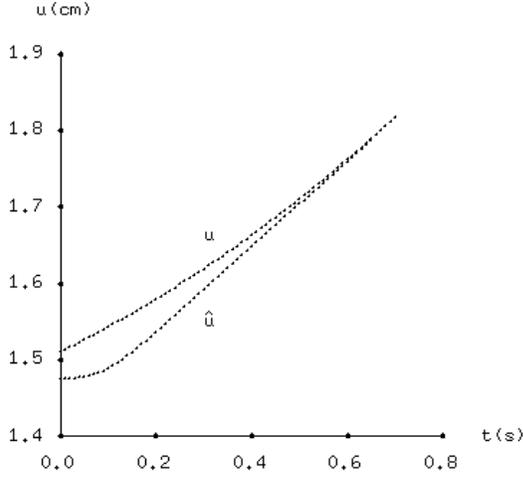
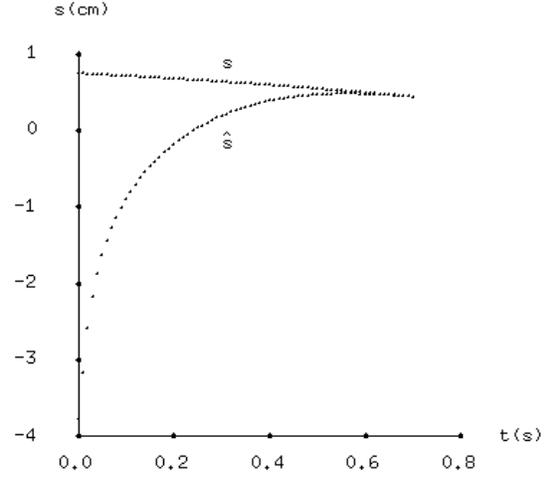


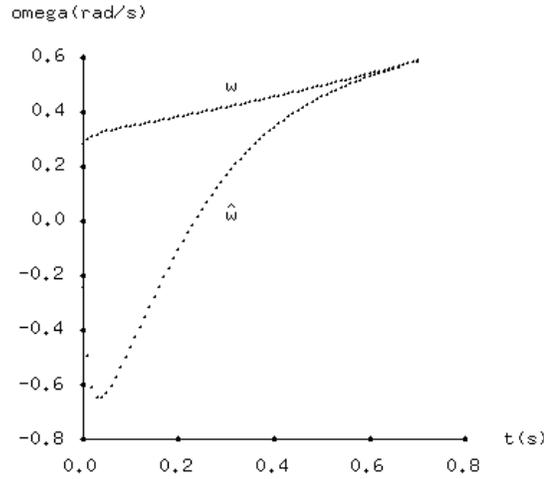
Figure 9: A disk of radius 1cm at constant velocity 5cm/s pushing a 7-gon while observing its pose and motion. The snapshots are taken every 0.1s. Contact friction between the polygon and the disk is assumed to be large enough to allow only the *rolling* on the disk edge. The edge of the polygon in contact is assumed to be known. The coefficient of support friction is 0.3. (a) The scene of pushing for 0.71s. (b) The imaginary scene as “perceived” by the observer (43) during the same time period. The observer constantly adjusts its estimates of the polygon’s pose and motion based on the moving contact on the disk boundary until they converge to the real pose and motion. Although the real contact and its estimate were about 4.5cm apart on the contact edge at the start of estimation, the error becomes negligible in 0.56s.



(c)



(d)



(e)

Figure 10: State variable trajectories vs. state estimate trajectories for the example shown in Figure 9. The sampling rate is 100Hz. Variable u gives the polar angle (scaled by the disk radius 1cm) of the contact from the disk center. Variable s measures the (signed) distance from the contact point to the intersection of the contact edge with its perpendicular through the polygon's center of geometry; it has the range $[-5.82, 2.90]$ (cm). Variable ω is the polygon's angular velocity. These three state variables have estimates \tilde{u} , \tilde{s} , and $\tilde{\omega}$, respectively. (c) The trajectories of u and \tilde{u} . (d) the trajectories of s and \tilde{s} ; and (e) the trajectories of ω and $\tilde{\omega}$. Note that \tilde{u} and $\tilde{\omega}$ converge faster than \tilde{s} .

Type of Pushing	$\mathbf{a}_{\mathcal{F}}$ (cm ² /s)	No. of Tests	Successes		
			No.	Ratio	Avg. Time (s)
Frictionless Contact	0	50	42	84%	0.776
	2.5	50	44	88%	0.738
Rolling	0	500	457	91.4%	0.965
	2.5	500	456	91.2%	0.939

Table 1: Simulations on observing the poses and motions of random polygons being pushed by the unit disk, assuming the contact edges were known. Two versions of the GHO observers were simulated: (32) for the case of no friction between a polygon and the unit disk; and (43) for the case where the polygon is rolling on the disk boundary.

observation.²⁰

Table 1 summarizes the results with known contact edges. There are four groups of data, each representing a different combination of contact mode and disk acceleration. As the table indicates, the finger acceleration $\mathbf{a}_{\mathcal{F}}$ did not affect the observer’s performance. This seems to be in contradiction with our resort to the use of the normal input field \mathbf{g}_N , driven by the normal acceleration $\mathbf{a}_{\mathcal{F}_N}$, in the proof of Theorem 3 on local observability. Nevertheless, the use of \mathbf{g}_N serves to simplify the construction of an algebraic proof of the observability rank condition. We might have used the drift field \mathbf{f} only in the proof, except the rank condition would be very hard or even impossible to establish.

Figure 11(a) shows a simulation example in which a 5-gon making frictionless contact with the unit disk translating at constant velocity. Figure 11(b) plots the “polygon motion” as understood by the observer from the contact motion along the disk boundary. In 0.6s (real time), the observer is able to locate the contact point (thereby determining the pose of the 5-gon) as well as to estimate its velocity and angular velocity. The trajectories of the state variables u, s, ω, v_T paired with the trajectories of their estimates $\tilde{u}, \tilde{s}, \tilde{\omega}, \tilde{v}_T$ are shown in Figure 12 (c), (d), (e), (f), respectively.

Since in every test the estimated contact point, given by s , was randomly chosen on the contact edge, it could be far from the real contact point. Yet, the results in Table 1 seem to suggest that the local GHO observer has “globalness”, at least within one edge.

We also observed that the disk contact estimate \tilde{u} and the angular velocity estimate $\tilde{\omega}$ always converged very fast, and the tangential velocity estimate \tilde{v}_T almost always converged. The pose estimate \tilde{s} , however, was always part of the divergence whenever it occurred. This phenomenon agrees with our previous discussion following the proof of Theorem 3 on the relative “hardness” of observing different state variables of the disk-polygon system.

In the real situation, *only* the finger contact u is known. In other words, the contact edge, the contact location s on the edge, and the velocities ω and v_T are all unknown. Accordingly,

²⁰In the simulations, we set this “long period” as 2s.

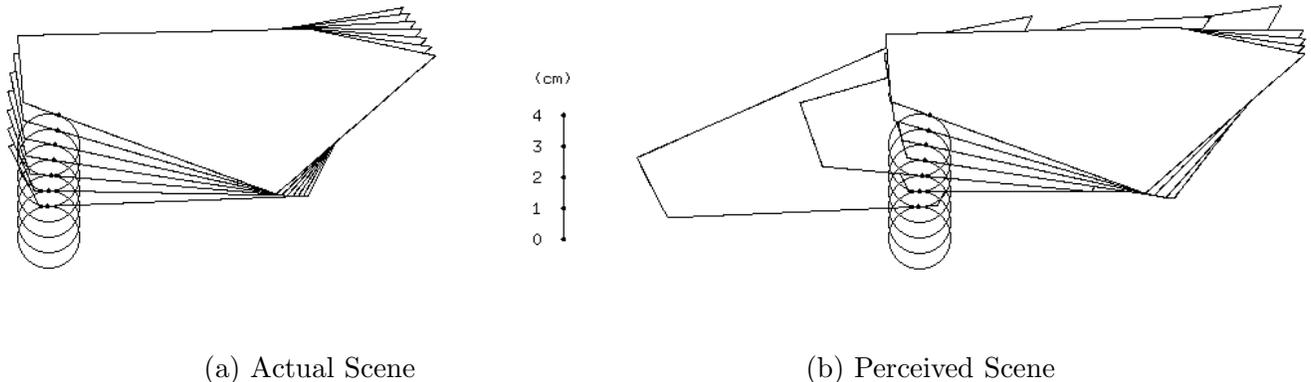
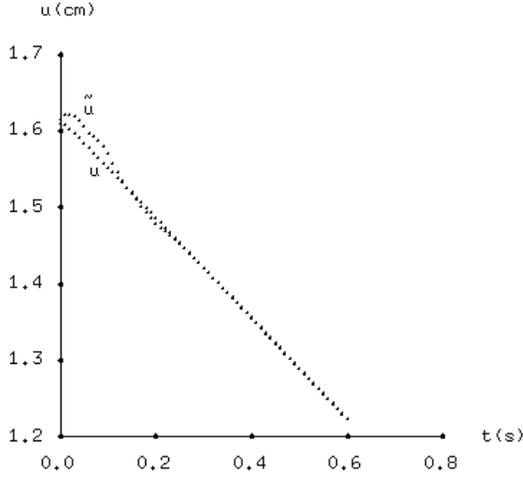


Figure 11: A disk of radius 1cm at constant velocity 5cm/s pushing and observing a 5-gon. The contact between the disk and the polygon is assumed to be *frictionless*. (a) The scene of pushing for 0.6s. (b) The imaginary scene as “perceived” by the observer (32). The real contact and its estimate were about 7.84cm apart on the edge at the start of estimation. The error became negligible in about 0.5s. Figure 12 details the convergence of the estimates of the pose, velocity, and angular velocity of the polygon during the push.

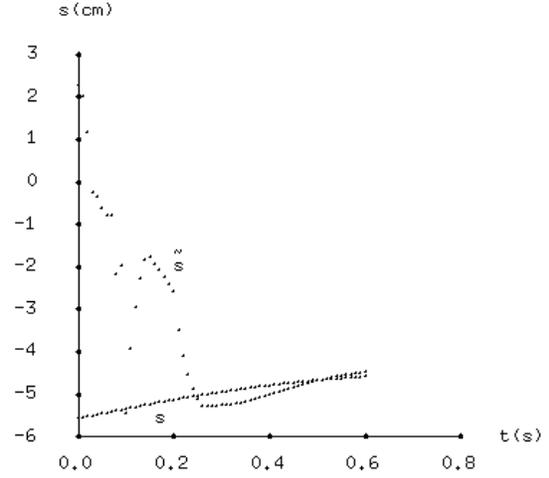
we modify the observer as follows. The observer generates for each edge of the polygon being pushed a state estimate that hypothesizes the edge as in contact. Then it simulates the push starting with these estimates in parallel for a short period of time.²¹ Assuming that the estimate hypothesizing the correct contact edge will likely have converged to the real state by now, the observer then turns off its error correction and continues the simulation of the remaining possible state trajectories. The estimate is chosen from the trajectory that outlasts all the others in having its \tilde{u} stay negligibly close to the the observed disk contact u . The observer fails if all estimates have gone out of their ranges in the first period, or the obtained contact estimate, including the edge and the location on the edge, is incorrect.

Table 2 shows the test results with *unknown* contact edges. A high percentage of the failures reported in the table were due to incorrect contact edges. To explain this, recall in the disk-polygon system (24) that the only parameters reflecting the contact geometry are the distance (or height) h from the contact edge to the polygon’s centroid and the signed distance s from the contact to where the edge intersects its perpendicular from this center. Contact points on different edges of (approximately) the same height and with (approximately) the same s can thus result in (approximately) the same behavior of the disk-polygon system. Finding a wrong contact edge is therefore expected to happen often when the polygon’s centroid is approximately equidistant to the real contact edge and to another edge. In fact, the failures due to incorrect contact edges that we had observed individually were predominantly of this type.

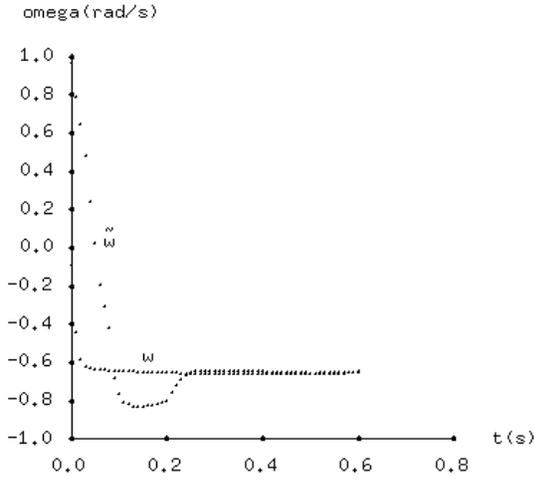
²¹This length of this period was based on the average convergence time in Table 1.



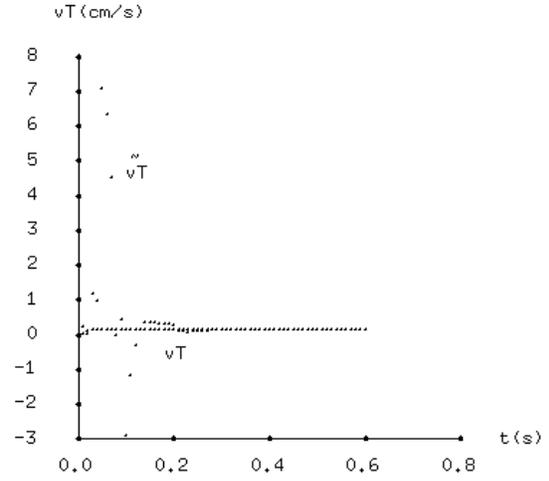
(c)



(d)



(e)



(f)

Figure 12: State variable trajectories vs. state estimate trajectories for the example shown in Figure 11. Variables u , s , and ω are as specified in Figure 10. Variable v_T is the projection of the velocity of \mathcal{P} onto the contact tangent. Variable s has the range $[-5.82, 2.90]$ (cm). The four state variable estimates are \tilde{u} , \tilde{s} , $\tilde{\omega}$, and \tilde{v}_T , respectively. Note that \tilde{u} , $\tilde{\omega}$, \tilde{v}_T converge faster than \tilde{s} .

Type of Pushing	$\mathbf{a}_{\mathcal{F}}$ (cm ² /s)	No. of Tests	Successes			Failures		
			No.	Ratio	Time (s)	Contact	Edge	Divergence
Frictionless	0	50	26	52%	0.733	9	8	7
Contact	2.5	50	26	52%	0.760	9	9	6
Rolling	0	500	294	58.8%	0.893	32	122	52
	2.5	500	287	57.4%	0.891	35	129	49

Table 2: Simulations of the GHO observers (32) and (43) on finding the poses and motions of random polygons being pushed by a unit disk, assuming *unknown* contact edges. There were three types of observation failures, shown from left to right in the three columns under the “Failures” title bar: (1) the observer found the correct contact edge but not the correct contact point; (2) the observer found the incorrect contact edge; (3) the observer diverged on all initial estimates.

6.2 On the IP Observer

Simulations were conducted for three types of pushing: ellipse(finger)-ellipse(object), line-ellipse, and ellipse-polygon. No contact friction was assumed in these simulations.

Closed forms of integral Γ exist for polygons but not for ellipses. On a Sparcstation 20, one evaluation of Γ takes about 2s for an ellipse. The computation of initial accelerations as in Section 2.3 takes about 1.6s for a hexagon and 25s for an ellipse.

During a push, the initial, the final, and one intermediate contact positions on the finger were recorded, along with the times when the contact reached these positions. The initial pose observer in Section 4.2 computed possible resting poses of the object which, under the push, would cause the contact to move to the intermediate position on the fingertip at the recorded time. More specifically, the algorithm guessed a number of initial contacts on the object, and called the Newton-Raphson routine.²² The final contact position was then used to further eliminate infeasible poses.

Table 3 shows the test results under no support friction. These results support our conjecture in Section 4.2 that the object pose can often be determined from *three* instantaneous contacts on the finger during a push.

The slow numerical evaluation of integral Γ prohibits us from conducting large number of tests on elliptic objects under support friction. Simulations under friction were only performed on polygons, for which closed forms of Γ exist. The 105 tests took about 65 hours, yielding 94 successes, 11 failures and ambiguities.

²²In the experiments, 10 guesses were taken for an ellipse and 3 guesses for each edge of a polygon.

Finger	Object	No. of Tests	Successes	
			No.	Ratio
ellipse	ellipse	1000	978	97.8%
line	ellipse	1000	975	97.5%
ellipse	poly	200	189	94.5%

Table 3: Simulations of the IP observer with the frictionless plane.

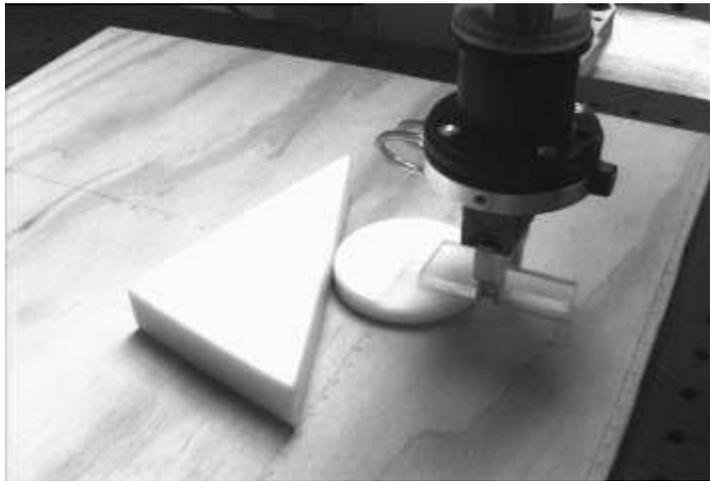


Figure 13: Experimental setup of pose-from-pushing. The coefficient of contact friction between the part and the disk (finger) was small (measured to be 0.213).

6.3 Preliminary Experiments

Later we conducted some experiments with an Adept 550 robot. The “finger” in our experiments was a plastic disc held by the robot gripper. The disk edge was marked with angles from the disk center so a contact position could be read by flesh eyes. Plastic polygonal parts of different material were used as objects. A plywood surface served as the supporting plane for pushing. Figure 13 shows the experimental setup.

Simulation and experimental results on pushing were found to agree closely (Figure 14), with slight discrepancies mainly due to shape uncertainties and non-uniform properties of the disk, the parts, and the plywood, all handmade.

We also did some experiments on sensing. Instead of one push, two consecutive pushes were performed so that the contact position after the first push served as the intermediate contact position.

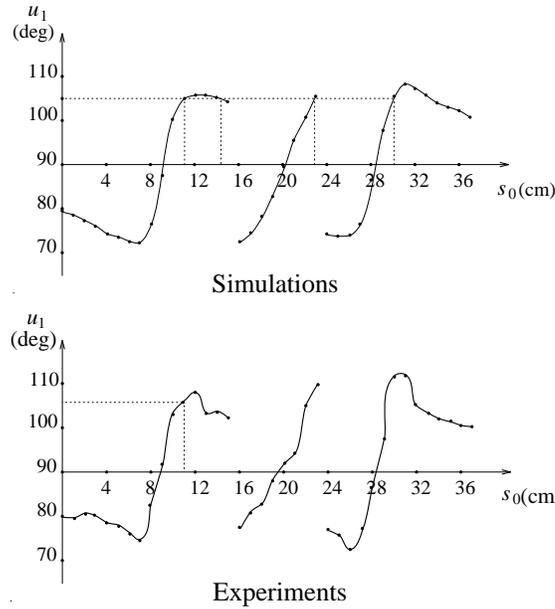


Figure 14: Simulations versus experiments on a triangular part. The graphs show the final sensor contact u_1 as a function of the initial part contact s_0 , which *here* measured the distance from a vertex of the part counterclockwise along the boundary. The part boundary was discretized into a finite set of locations $\{s_0\}$. For each such s_0 we performed a numerical simulation and a physical experiment. The same disk motion lasting 0.75 seconds was used in each of these simulations and experiments. The initial contact u_0 on the pushing disk was always at 90 degrees from its center. The case where s_0 was 11cm in the physical experiment is illustrated by the dotted lines: Four feasible poses were found by the simulator from contact position $u_1 = 105$ degrees after the push. These four poses were later distinguished (and the real pose was thus determined) using a second contact position u_2 determined after a second identical push by the disk.



Figure 15: A force sensor for contact sensing. The sensor is composed of a horizontal disk with diameter 3cm and a cylindrical stainless steel beam erected vertically on the disk and attached to the gripper of an Adept robot at the top. Two pairs of 350Ω strain gauges are mounted on the upper end of the beam where they would be most sensitive to any force exerted on the disk. They measure the contact force along two orthogonal directions, respectively

6.4 Sensor Implementation

We have built a “finger” with tactile capability using four strain gauges as shown in Figure 15. The strain gauges are mounted near the top of a vertical stainless steel beam and connected to an Omega PC plug-in card to form two Wheatstone half bridges. The lower end of the beam is attached a disk which serves as the “finger”. A contact with the disk would result in the bending of the beam, which would be detected by the strain gauges. The components of the contact force exerted on the disk boundary along the x and y axes of the disk, respectively, can then be calculated. When contact friction is small enough, the contact force measured by the gauges would point along the disk normal at the contact, thereby indicating the contact location on the disk boundary.

The sensor is sensitive enough to detect force in microstrains with a frequency over 2000 Hz. It reports the contact in terms of its polar angle with respect to the disk center. After calibration, the sensed static contacts (in 1000 readings) constantly have a mean within one degree away from the real contact and a standard deviation of less than 0.5 degree. For the sensor to be applicable, more work is needed to improve the measurement accuracy and to deal with dynamic friction.

7 Summary

We have introduced a sensing approach based on nonlinear observability theory that makes use of one-finger tactile information. The approach determines the pose of a known planar

object by pushing it with a finger while “feeling” the contact motion on the fingertip. It also estimates the object motion during the pushing. Both the finger and the object are assumed to have piecewise smooth boundaries.

We derived a system of nonlinear differential equations that govern contact and object motions from geometric and velocity constraints, as well as from the dynamics of pushing. The state of this pushing system includes the pose and motion of the object while its output is the moving contact position on the fingertip. We established the local observability of the system for the case of a disk pushing a polygon, a result that can generalize to many other finger and object shapes. This result forms the underlying principle of our sensing algorithms, which essentially are observers of the nonlinear dynamical system for pushing.

Based on the result of [15], we constructed an asymptotic nonlinear observer and demonstrated it by simulations. This observer is a composition of a copy of the original system with an error corrective term constructed over the system output and the solution of a Lyapunov-like equation. It is capable of asymptotically correcting any local error in the estimation of the object pose and motion. The observer accepts a sequence of fingertip contacts beginning any time during a push. Such an on-line property makes the observer quite flexible but it also requires the sensor to continually provide contact data.

We also presented a nonlinear observer based on Newton’s method. It determines the initial resting pose of the object (and thus any pose from then on) from two or three intermediate contact positions on the fingertip. In constructing this observer, we viewed pushing as a mapping from the one-dimensional set of initial motionless object poses to the set of one-dimensional contacts on the fingertip at a time instant; and sensing just as its *inverse* mapping. Simulations and preliminary experiments show that the initial, final, and another intermediate contact positions usually suffice for pose determination. Although this observer requires much less sensor data than the first observer, it is more likely to be affected by sensor noises and uncertainties in object motion due to possible impact between the finger and the object at the beginning of pushing.

Although in certain worst cases global sensing ambiguities can never be eliminated with one push, a sensing failure may be removed by pushing repeatedly at different sections of the object boundary.

Both support friction in the plane and contact friction between the object and the finger have been taken into account.

Another implementation, for the initial pose observer only, may embed three point light detectors in the fingertip. Keep the initial contact always at the first light detector so that it is covered. Different initial object poses would result in different times when the second and third light detectors are covered. To enhance efficiency, we could simply discretize the boundary of a part and precompile a table from which poses can be directly looked up with the recorded time instants.

Contact motion is the subject of our paper for it encodes a range of information that we want to know about a task. It is generated by pushing and decoded with the application of nonlinear observability theory. It is clear that our method will be applicable in other situations where contact motion or similar tactile information exists regardless of the type of manipulation that generates such motion.

From a more general perspective, the information for accomplishing a manipulation task

may exist in various forms, implicitly or explicitly. Accordingly, there may be various ways of obtaining such information and making use of it. This paper uses the example of contact motion and applies nonlinear observability theory to illustrate the above idea. A good strategy for achieving a task usually comes from an understanding of its geometry and mechanics.

We view a sensing algorithm as the part of a sensing strategy that interprets data acquired by physical sensors to derive information essential for a task. It can be a computational algorithm based on task geometry, a control system based on task mechanics, or something else. Just as geometric sensing resorts to computational geometry algorithms, the sensing strategy presented in this paper applies nonlinear control techniques. In devising a sensing strategy we have to decide what portion of the sensing is to be done by physical sensors and what portion to be carried out by the sensing algorithm. A good strategy should exploit to some extent the sensors available as well as the task geometry and mechanics. Here we would like to refer the reader to Erdmann's methodology [11] on task-specific and action-based sensor design.

Acknowledgements

Support for this research was provided in part by Carnegie Mellon University, and in part by the National Science Foundation through the following grants: NSF Presidential Young Investigator award IRI-9157643 and NSF Grant IRI-9213993.

Thanks to Matt Mason for having interesting discussions on the topics of pushing and pressure distribution, for suggesting the use of force sensing in real implementation, and for pointing us to nonlinear observability theory; to Ken Salisbury, Alan Guisewite, and Ralph Hollis for their feedback on the experimental setup, and especially, on the use of strain gauges; to Garth Zeglin for generous help with the experiments; to Kevin Lynch for introducing us to his related work on active sensing by pushing using tactile feedback, and for offering many helpful comments on the paper; to Wes Huang for realizing the connection of our sensing work to observer design; and to Alfred Rizzi for an interesting discussion on nonlinear observers.

Many thanks to the anonymous reviewers whose insightful and constructive comments have helped strengthen the technical soundness as well as improve the intuitive illustration of the paper. Sections 1, 2, 4.2, 6.2, and 6.3 are based on Jia and Erdmann [30] presented at the 1996 IEEE International Conference on Robotics and Automation, while sections 3, 4.1, 6.1, and 6.4 are based on Jia and Erdmann [31] presented at the 1998 IEEE ICRA. Thanks to the anonymous reviewers of these two conference papers for their valuable comments.

References

- [1] Srinivas Akella and Matthew T. Mason. Posing polygonal objects in the plane by pushing. In *Proceedings of the 1992 IEEE International Conference on Robotics and*

- Automation*, pages 2255–2262, 1992.
- [2] J. C. Alexander and J. H. Maddocks. Bounds on the friction-dominated motion of a pushed object. *International Journal of Robotics Research*, 12(3):231–248, 1993.
 - [3] P. K. Allen and K. S. Roberts. Haptic object recognition using a multi-fingered dextrous hand. In *Proceedings of the 1989 IEEE International Conference on Robotics and Automation*, pages 342–347, 1989.
 - [4] K.-F. Böhringer, B. R. Donald, R. Mihailovich, and N. C. MacDonald. Sensorless manipulation using massively parallel microfabricated actuator arrays. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 826–833, 1994.
 - [5] Randy C. Brost and Kenneth Y. Goldberg. A complete algorithm for synthesizing modular fixtures for polygonal parts. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 535–542, 1994.
 - [6] Chunsheng Cai and Bernard Roth. On the spatial motion of a rigid body with point contact. In *Proceedings of the 1987 IEEE International Conference on Robotics and Automation*, pages 686–695, 1987.
 - [7] Alec Cameron, Ron Daniel, and Hugh Durrant-Whyte. Touch and motion. In *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, pages 1062–1067, 1988.
 - [8] G. Ciccarella, M. Dalla Mora, and A. Germani. A Luenberger-like observer for nonlinear systems. *International Journal of Control*, 57(3):537–556, 1993.
 - [9] P. E. Crouch. Dynamical realizations of finite Volterra series. *SIAM Journal on Control and Optimization*, 19(1):177–202, 1981.
 - [10] Herbert Edelsbrunner and Leonidas J. Guibas. Topologically sweeping an arrangement. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing*, pages 389–403. ACM Press, 1986. Berkeley, CA.
 - [11] Michael Erdmann. Understanding action and sensing by designing action-based sensors. *International Journal of Robotics Research*, 14(5):483–509, 1995.
 - [12] Michael Erdmann and Matt Mason. An exploration of sensorless manipulation. *IEEE Journal of Robotics and Automation*, 4(4):369–379, 1988.
 - [13] R. S. Fearing and T. O. Binford. Using a cylindrical tactile sensor for determining curvature. In *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, pages 765–771, 1988.
 - [14] Peter C. Gaston and Tomás Lozano-Pérez. Tactile recognition and localization using object models: The case of polyhedra on a plane. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(3):257–266, 1984.

- [15] J. P. Gauthier, H. Hammouri, and S. Othman. A simple observer for nonlinear systems applications to bioreactors. *IEEE Transactions on Automatic Control*, 37(6):875–880, 1992.
- [16] J. P. Gauthier and I. A. K. Kupka. Observability and observers for nonlinear systems. *SIAM Journal on Control and Optimization*, 32(4):975–994, 1994.
- [17] Kenneth Y. Goldberg. Orienting polygonal parts without sensors. *Algorithmica*, 8(10):201–225, 1993.
- [18] Lawrence E. Goodman and William H. Warner. *Statics and Dynamics*. Wadsworth Publishing Company, Inc., Belmont, CA, 1964.
- [19] Suresh Goyal, Andy Ruina, and Jim Papadopoulos. Planar sliding with dry friction. *Wear*, 143:307–352, 1991.
- [20] W. E. L. Grimson. *Object Recognition by Computer: the Role of Geometric Constraints*. MIT Press, Cambridge, 1990.
- [21] W. E. L. Grimson and T. Lozano-Pérez. Model-based recognition and localization from sparse range or tactile data. *International Journal of Robotics Research*, 3(3):3–35, 1984.
- [22] David D. Grossman and Michael W. Blasgen. Orienting mechanical parts by computer-controlled manipulator. *IEEE Transaction on Systems, Man, and Cybernetics*, 5(5):561–565, 1975.
- [23] E. J. Haug, S. C. Wu, and S. M. Yang. Dynamic mechanical systems with coulomb friction, stiction, impact and constraint addition-deletion—I: Theory. *Mechanisms and Machine Theory*, 21(5):407–416, 1986.
- [24] Robert Hermann and Arthur J. Krener. Nonlinear controllability and observability. *IEEE Transactions on Automatic Control*, AC-22(5):728–740, 1977.
- [25] Robert D. Howe and Mark R. Cutkosky. Dynamic tactile sensing: Perception of fine surface features with stress rate sensing. *IEEE Transactions on Robotics and Automation*, 9(2):140–151, 1993.
- [26] Robert D. Howe and Mark R. Cutkosky. Practical force-motion models for sliding manipulation. *International Journal of Robotics Research*, 15(6):557–572, 1996.
- [27] A. Isidori. *Nonlinear Control Systems: An Introduction*. Springer-Verlag, 1985.
- [28] Yan-Bin Jia. *Geometric and Dynamic Sensing: Observation of Pose and Motion through Contact*. PhD thesis, Carnegie Mellon University, The Robotics Institute, 1997. CMU-RI-TR-97-50.
- [29] Yan-Bin Jia and Michael Erdmann. Geometric sensing of known planar shapes. *International Journal of Robotics Research*, 15(4):365–392, 1996.

- [30] Yan-Bin Jia and Michael Erdmann. Pose from pushing. In *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, pages 165–171, 1996.
- [31] Yan-Bin Jia and Michael Erdmann. Observing pose and motion through contact. In *Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, 1998.
- [32] Arthur J. Krener and Alberto Isidori. Linearization by output injection and nonlinear observers. *Systems Control Letters*, 3:47–52, 1983.
- [33] Zexiang Li and John Canny. Motion of two rigid bodies with rolling constraint. *IEEE Transactions on Robotics and Automation*, 6(1):62–72, 1990.
- [34] David G. Luenberger. An introduction to observers. *IEEE Transactions on Automatic Control*, AC-16(6), 1971.
- [35] Kevin Lynch, Hitoshi Maekawa, and Kazuo Tanie. Manipulation and active sensing by pushing using tactile feedback. In *Proceedings of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 416–421, 1992.
- [36] William Duncan MacMillan. *Dynamics of Rigid Bodies*. McGraw-Hill Inc., 1936. Dover reprint, 1960.
- [37] Matthew T. Mason. Mechanics and planning of manipulator pushing operations. *International Journal of Robotics Research*, 5(3):53–71, 1986.
- [38] David J. Montana. The kinematics of contact and grasp. *International Journal of Robotics Research*, 7(3):17–32, 1988.
- [39] Henk Nijmeijer and Arjan van der Schaft. *Nonlinear Dynamical Control Systems*. Springer-Verlag, New York, 1990.
- [40] Jong-Shi Pang and Jeffrey C. Trinkle. Complementary formulations and existence of solutions of dynamic multi-rigid-body contact problems with Coulomb friction. *Mathematical Programming*, 73:199–226, 1996.
- [41] Eric Paulos and John Canny. Fast construction of near optimal probing strategies. In Jean-Paul Laumond and Mark Overmars, editors, *Algorithms for Robotics Motion and Manipulation*, pages 185–195. A. K. Peters, 1997. Proceedings of the 1996 Workshop on the Algorithmic Foundations of Robotics.
- [42] M. A. Peshkin and K. Goldberg. Curved fences for part alignment. In *Proceedings of the 1993 IEEE International Conference on Robotics and Automation*, pages 467–573, 1993.
- [43] Elon Rimon and Andrew Blake. Caging 2D bodies by 1-parameter two-fingered gripping systems. In *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, pages 1458–1464, 1996.

- [44] Kenneth Salisbury. Interpretation of contact geometries from force measurements. In M. Brady and R. Paul, editors, *Robotics Research*, pages 565–577. The MIT Press, 1984.
- [45] Michael Spivak. *A Comprehensive Introduction to Differential Geometry*, volume 1. Publish or Perish, Inc., Boston, 1970.
- [46] Gerta Zimmer. State observation by on-line minimization. *International Journal of Control*, 60(4):595–606, 1994.