Manipulation and Active Sensing by Pushing Using Tactile Feedback

Kevin M. Lynch*
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213 USA

Hitoshi Maekawa
Cybernetics Division
Mechanical Engineering Laboratory, AIST-MITI
Namiki 1-2, Tsukuba-shi, Ibaraki-ken, 305 Japan

Kazuo Tanie
Biorobotics Division
Mechanical Engineering Laboratory, AIST-MITI
Namiki 1-2, Tsukuba-shi, Ibaraki-ken, 305 Japan

Abstract— **We investigate manipulation and active sensing by a pushing control system using only tactile feedback. The equations of motion of a pushed object are derived using a model of the object's limit surface, and we design a control system to translate and orient objects. The effectiveness of the proposed controller is confirmed through simulation and experiments. Active sensing of the object's center of mass is described.**

I. Introduction

Pushing is a useful robotic capability for positioning and orienting parts. Several researchers have demonstrated the utility of pushing operations by planning open-loop pushing sequences to position and orient polygonal objects despite the presence of uncertainty in the initial state [1, 2, 3, 4, 5, 6, 7]. These operations typically plan for a known object shape and center of mass (CM) and a flat pushing fence or specially designed pusher geometry to exploit the mechanics of pushing.

Others have proposed pushing control systems based on visual feedback [8, 9]. The pusher makes point contact with the object, and the position and orientation of the object is determined by a vision system. The goal is to push the object along a desired trajectory. Uncertainty in the frictional forces governing the object's motion is compensated for by the appropriate design of a feedback controller.

The purpose of this work is to investigate the possibility of useful manipulation by a pushing control system using only tactile feedback. Try closing your eyes and controlling the motion of an object on a table by pushing it with a finger. This is the type of capability we would like to give a robot. In contrast to vision, tactile sensing requires very little data processing. The tactile sensor is mounted directly on the manipulator and the robot requires no peripheral sensors.

The primary difficulty arises from the fact that tactile sensing can only give local contact information. The configuration of the object has three degrees-of-freedom: two position coordinates and an orientation. When the pusher is in contact with the object, these three degrees-of-freedom may be equivalently expressed by the location of the pushing contact, the contact point on the perimeter of the object, and the orientation of the object. The tactile sensor we use is capable of sensing the contact location and the object orientation at the contact, but not the contact point on the object. With vision, all three degrees-of-freedom are directly sensed.

Despite this missing information, we demonstrate that simple manipulation is possible by pushing using only tactile feedback. Specifically, we implement a controller to translate the object and regulate its orientation.

*Kevin Lynch was supported as a participant in the 1991 Summer Institute in Japan program sponsored by the U.S. National Science Foundation, the Science and Technology Agency of Japan, and the Japan Foundation's Center for Global Partnership.

The rest of the paper is organized as follows. The remainder of this section describes the problem formulation and the notation used in the paper. Section II reviews the mechanics of pushing and develops a model of the motion of a pushed object. Section III defines active sensing as it applies to the pushing control system. In Section IV we describe the control algorithm implemented. Section V describes the experimental setup and presents simulation and experimental results. We offer some conclusions in Section VI.

A. Problem Formulation

The pusher is a disk. A tactile sensor detects the contact point on the disk, and the contact normal is in the direction of the vector from the center of the disk to the contact. The contact orientation angle is defined by the tangent to the disk at the contact. The object to be manipulated is a convex polygon. During execution, the pusher remains in contact with one edge of the object. In Section V we give an example of manipulation of non-polygonal objects.

We assume that pusher motions are slow enough that inertial forces are negligible compared to frictional forces. This is the quasi-static assumption. We also assume that frictional forces conform to Coulomb's Law. At any contact, the tangential frictional force f_t felt by an object must satisfy the relationship $f_t \leq \mu f_n$, where μ is the coefficient of friction and f_n is the normal force applied to the object. The static and kinetic coefficients of friction are assumed to be equal. Defining the friction angle $\alpha = \tan^{-1}\mu$, the total contact force must lie on or inside the cone of vectors which intersect the contact and make an angle α with the contact normal. If the contact is sliding, the force felt by the object lies on the boundary of the friction cone maximally opposing the motion of the object relative to the contacting surface.

B. Notation

The following notation will be used throughout the paper (see fig. 1):

$\hat{\mathbf{n}}$	contact normal unit vector (into the object)
θ	angle of the pushed edge (90 degrees clockwise of the angle of $\hat{\mathbf{n}}$)
ϕ	angle of motion of the pusher
d	contact location along the pushed edge of the object ($d = 0$ at the point on the edge closest to the CM, and d is positive to the right of the ray from this point to the CM)
l	distance from the pushed edge to the CM
r	radius of the disk pusher
μ_c	friction coefficient between the pusher and object
μ_s	friction coefficient between the object and support

Bold letters are vectors, and unit vectors are capped with a circumflex, as in $\hat{\mathbf{n}}$. We will denote linear velocities (v_x, v_y) as $\mathbf{v}$, generalized velocities (v_x, v_y, ω) as $\mathbf{q}$, linear forces (f_x, f_y) as $\mathbf{f}$, and generalized forces (f_x, f_y, m) as $\mathbf{p}$. All 2-vectors should be regarded as vectors in 3-space with zero third components. Angular

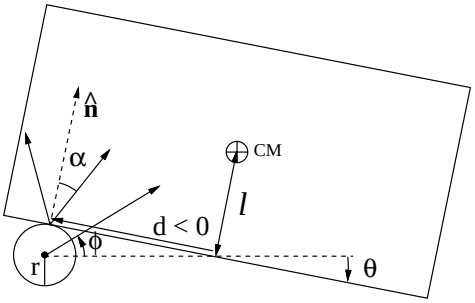

Fig. 1. Pushing system notation.

velocities and moments may be written in boldface as vectors with the first two components zero.

II. Mechanics of Pushing

In this section we review the mechanics of pushing and develop a model of the motion of a pushed object. This model may be used in the design of a pushing controller.

A. The Voting Theorem

Mason [6] analyzed the mechanics of quasi-static pushing operations given a known pusher–object contact point, contact normal, pushing direction, μ_c, and CM of the object. The analysis results in a simple method for determining the sense of rotation of a pushed object, which we will refer to as the *voting theorem*:

> Construct three rays at the contact point: the two friction cone limits and the ray of the pusher motion. Each ray that passes to the left of the CM votes for clockwise rotation, and each ray that passes to the right votes for counterclockwise rotation. The votes are tallied and the majority determines the rotation sense. If two or three rays pass through the CM, or one ray passes through the CM and the other two vote oppositely, the object will translate along the ray from the contact to the CM.[1]

Actually, this result uses the center of friction (CF), not the CM. The CF is located at the centroid of the support friction distribution multiplied by the pressure distribution. For the case of a uniform μ_s, however, the CF is simply the projection of the CM to the support plane. In this paper we assume a uniform μ_s. If this assumption is violated, CF should be substituted where CM appears.

The power of the voting theorem comes from its insensitivity to the form of the pressure distribution. The pressure distribution between an object and support is usually unknown and possibly changing due to microscopic variations in the support surface. Although the exact motion of a pushed object depends on the form of the pressure distribution, Mason showed that the rotation sense depends only on the location of the CM.

We can apply the voting theorem to determine the set of pushing contacts from which an object can be translated. The possible translational pushing contacts are obtained by drawing the friction cone at the CM and extending the cone until it intersects the pushed edge. For a ray of pushing through the CM, translation occurs in the direction of the push for contacts inside the projected friction cone, i.e., $-\mu_c l \leq d \leq \mu_c l$. Contacts to the left of the friction cone result in clockwise rotation regardless of the push direction. Similarly, contacts to the right of the friction cone result in counterclockwise rotation.

B. Limit Surfaces

Whereas the CM is sufficient for determining the rotation sense of a pushed object, the pressure distribution of the object must be considered to find the exact motion. The quasi-static relationship between the instantaneous velocity of a sliding object and the support frictional force is governed by a closed, convex, origin-enclosing limit surface in (f_x, f_y, m) space [10]. For a given reference point, the limit surface is the boundary of the sum (respectively integral) of the frictional forces and moments that each of the object's individual support points (resp. differential elements of support area) can apply to the support plane. The limit surface in (f_x, f_y, m) space encloses the set of generalized forces $\mathbf{p}$ which may be statically applied to the slider.

The limit surface of a sliding object is analogous to the friction cone for a single contact. Any applied force $\mathbf{p}$ inside the limit surface is completely resisted by the support friction, and the object will not move. If $\mathbf{p}$ extends outside the limit surface, static equilibrium is violated and the object will accelerate. During quasi-static motion of the object, $\mathbf{p}$ lies on the limit surface and the direction of the object's velocity is given by the unit vector $\hat{\mathbf{q}}$ normal to the surface at $\mathbf{p}$ in (v_x, v_y, ω) space, which is aligned with the (f_x, f_y, m) space. (See [10] for further details.) The shape of the limit surface is determined by the pressure distribution, and μ_s simply scales the limit surface. Thus, if the pressure distribution is known, the relationship between applied forces and object motions is completely described by the limit surface.

In general, the pressure distribution is unknown and possibly changing as the object moves. For this reason, we will develop a model using an approximation to the actual limit surface. The limit surface will be modeled as an ellipsoid in force-moment space [11, 12, 13]. The procedure to find the ellipsoid is similar to that in [13]:

1. Choose the CM as the reference point about which moments are measured. Find the maximum frictional moment m_{max}, resulting from a pure rotation about the reference point:
$$m_{max} = \mu_s \int_A |\mathbf{x}| p(\mathbf{x}) dA \tag{1}$$
where A is the support region, dA is a differential element of area of A, $\mathbf{x}$ is the position of dA, and $p(\mathbf{x})$ is the pressure at $\mathbf{x}$. This defines two ellipsoid endpoints: $(0, 0, \pm m_{max})$.[2]
2. Find the maximum frictional force (pure translation):
$$f_{max} = \mu_s f_n \tag{2}$$
where f_n is the normal support force.
3. The approximating ellipsoid is given by the equation:
$$\left(\frac{f_x}{f_{max}}\right)^2 + \left(\frac{f_y}{f_{max}}\right)^2 + \left(\frac{m}{m_{max}}\right)^2 = 1 \tag{3}$$

The calculation of m_{max} requires a known pressure distribution $p(\mathbf{x})$. To account for the unknown pressure distribution, the pushing controller should be designed to be robust for limit surfaces with a range of values of m_{max}. If the object's support pressure is concentrated near the boundaries of the object, m_{max} is large and the resulting limit surface is elongated along the m-axis. The motion

[1] If the object's support is confined to a line segment perpendicular to the pushing direction, the object may rotate [6, 10].

[2] We assume the CM is coincident with the center of twist [10]. A pure rotation about the center of twist corresponds to a pure moment.

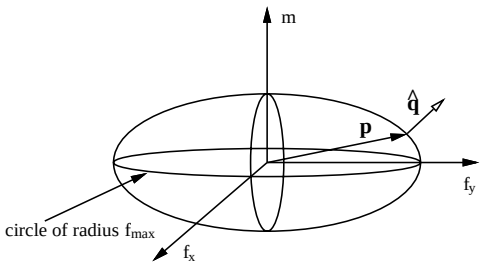

Fig. 2. Ellipsoidal limit surface. $\mathbf{p}$ is the generalized applied force and $\hat{\mathbf{q}}$ is the associated generalized velocity unit vector.

of the pushed object tends to be largely translational. If the object's support pressure is concentrated near the CM, m_{max} is small and the limit surface is relatively flat. The object's motion tends to be mostly rotational.

An example ellipsoidal limit surface is shown in fig. 2. The $m = 0$ cross-section of the ellipsoid is a circle of radius f_{max}. In order to translate the object, a force f_{max} must be applied through the CM in the direction of the desired translation. In addition, positive moment corresponds to counterclockwise rotation, and negative moment corresponds to clockwise rotation. These properties of the ellipsoid are consistent with the voting theorem. For the ellipsoid model, the translational velocity $\mathbf{v}$ of the CM is always parallel to the applied linear force $\mathbf{f}$. If a pure moment is applied, the object will rotate about the CM. The ellipsoid approximation to the limit surface satisfies the voting theorem and gives a nonlinear closed-form relationship between forces and velocities. The accuracy of the approximation depends on the particular pressure distribution; see Section V for an example.

C. Equations of Motion

Defining the parameter $c = m_{max}/f_{max}$ (with units of length), the relative values of the components of $\mathbf{q}$ (defined at the CM) for an applied force $\mathbf{p}$ are derived from the ellipsoid equation:

$$\frac{v_x}{\omega} = c^2\frac{f_x}{m} \quad (4)$$

$$\frac{v_y}{\omega} = c^2\frac{f_y}{m} \quad (5)$$

The magnitude of the applied force $|\mathbf{p}|$ is given by (3) and $|\mathbf{q}|$ is determined by the velocity of the pusher. The object's velocity must be just sufficient to move out of the way of the advancing pusher.

Equations (4) and (5) relate applied forces to object velocities. It remains to find the object's motion in response to a position-controlled push. The method for doing this is taken from [6]:

1. The origin is located at the CM and the pushing contact is located at $\mathbf{x}_c = (x_c, y_c)$. The pusher velocity at the contact is denoted $\mathbf{v}_p = (v_{px}, v_{py})$, and the resulting velocity of the object at the contact is written $\mathbf{v}_o = (v_{ox}, v_{oy})$.
2. Find the unit generalized velocities $\hat{\mathbf{q}}_l = (v_{lx}, v_{ly}, \omega_l)$ and $\hat{\mathbf{q}}_r = (v_{rx}, v_{ry}, \omega_r)$ resulting from forces at the left and right edge of the friction cone, respectively. The corresponding velocities at the contact point $\mathbf{x}_c$ are $\mathbf{v}_l = (v_{lx} - \omega_l y_c, v_{ly} + \omega_l x_c)$ and $\mathbf{v}_r = (v_{rx} - \omega_r y_c, v_{ry} + \omega_r x_c)$. These two velocity vectors form the boundary of the *motion cone* [6]. Any force applied at $\mathbf{x}_c$ inside the friction cone results in a velocity $\mathbf{v}_o$ which is a positive linear combination of $\mathbf{v}_l$ and $\mathbf{v}_r$ (i.e., inside the motion cone).

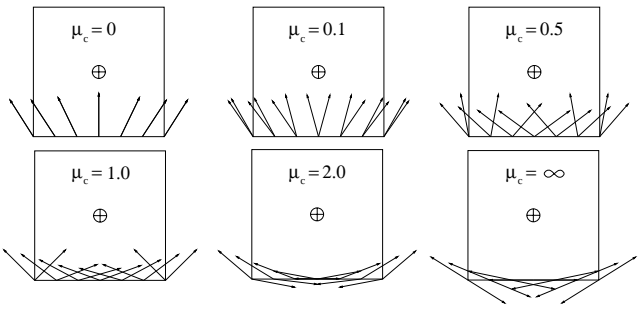

Fig. 3. Motion cones for a square object using the ellipsoidal limit surface approximation and m_{max} calculated using a uniform pressure distribution. Contacts are sampled along the bottom edge with different values of μ_c. For $\mu_c = 0$, the motion cone reduces to a ray, and for $\mu_c = \infty$, the interior angle of the motion cone is 180 degrees. The motion cones in this figure imply two nonintuitive effects: 1) Infinite friction does not necessarily imply sticking contact during pushing. The ray of pushing may lie outside the 180 degree motion cone. 2) Vectors in the motion cone with a negative component in the direction of the contact normal imply that it is possible to "pull" the object by applying a force inside the friction cone.

3. If the pusher velocity $\mathbf{v}_p$ is contained within the motion cone, sticking contact occurs ($\mathbf{v}_o = \mathbf{v}_p$). If $\mathbf{v}_p$ is to the left of the motion cone, the pusher slides to the left with respect to the object, resulting in an applied force at the left edge of the friction cone. The object velocity at the contact therefore lies on the left edge of the motion cone. If $\mathbf{v}_p$ is to the right of the motion cone, the pusher slides to the right with respect to the object, and $\mathbf{v}_o$ lies on the right edge of the motion cone. See fig. 3.

This procedure is applied using the ellipsoidal limit surface model.

1) Object Motion During Sticking Contact: When $\mathbf{v}_p$ is inside the motion cone, two constraints on the object's velocity $\mathbf{q} = (v_x, v_y, \omega)$ are given by $\mathbf{v}_o = \mathbf{v}_p$. Since the line of applied force $\mathbf{f}$ must pass through the contact, this defines a constraint on the applied moment, $\mathbf{m} = \mathbf{x}_c \times \mathbf{f}$. These three constraints may be written

$$v_x = v_{px} + \omega y_c \quad (6)$$

$$v_y = v_{py} - \omega x_c \quad (7)$$

$$m = x_c f_y - y_c f_x \quad (8)$$

Remembering that $\mathbf{f}$ is parallel to $\mathbf{v}$, we solve for $\mathbf{q}$ using (4) and (5):

$$v_x = \frac{(c^2 + x_c^2)v_{px} + x_c y_c v_{py}}{c^2 + x_c^2 + y_c^2} \quad (9)$$

$$v_y = \frac{x_c y_c v_{px} + (c^2 + y_c^2)v_{py}}{c^2 + x_c^2 + y_c^2} \quad (10)$$

$$\omega = \frac{x_c v_y - y_c v_x}{c^2} \quad (11)$$

2) Object Motion During Slipping Contact: When $\mathbf{v}_p$ is outside the motion cone, the contact is slipping and $\mathbf{v}_o$ lies on one of the motion cone boundaries. The slipping velocity $\mathbf{v}_{slip}$ is along the contact tangent, and $\mathbf{v}_o$ and $\mathbf{v}_{slip}$ must satisfy

$$\mathbf{v}_o + \mathbf{v}_{slip} = \mathbf{v}_p \quad (12)$$

as in fig. 4. In order to treat both slipping directions simultaneously, we denote the boundary of the motion cone under consideration $\mathbf{v}_b$ and the corresponding generalized velocity $\hat{\mathbf{q}}_b$. For contact normal $\hat{\mathbf{n}}$ we define the scaling factor $\kappa = (\mathbf{v}_p \cdot \hat{\mathbf{n}})/(\mathbf{v}_b \cdot \hat{\mathbf{n}})$. The motion of the

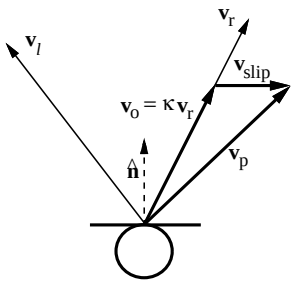

Fig. 4. $\mathbf{v}_o$ and $\mathbf{v}_{slip}$ for $\mathbf{v}_p$ to the right of the motion cone.

object during slipping contact is given by the following equations:

$$\mathbf{v}_o = \kappa \mathbf{v}_b \quad (13)$$

$$\mathbf{v}_{slip} = \mathbf{v}_p - \mathbf{v}_o \quad (14)$$

$$\mathbf{q} = \kappa \hat{\mathbf{q}}_b \quad (15)$$

3) Contact Velocity: The velocity of the contact along the pushed edge has two components: slipping and rolling. The contact velocity $\dot{d}$ is given by

$$\dot{d} = \mathbf{v}_{slip} \cdot \hat{\mathbf{t}} - \omega r \quad (16)$$

where $\hat{\mathbf{t}}$ is the unit tangent vector in the direction of increasing d.

III. Active Sensing

In addition to manipulation, a pushing control system may be used for active sensing. Sensory modalities which project a signal (e.g., light striping, radar, laser range finding) or use strategies to guide the acquisition of data (e.g., tactile probing, active vision) are commonly labeled active sensing. In this paper, however, active sensing refers to the process of exploiting task mechanics by performing an action or set of actions in order to change the system state to a known smaller set of possible resulting states.

The voting theorem constrains the CM of the object to lie on the ray of pushing during a finite length translation.With this knowledge of the task mechanics, we conclude that if our pushing control system can achieve stable translation by pushing, the CM of the object must lie on the ray of pushing.

This constraint on the location of the CM may be used in the following ways. 1) With the aid of a vision system, the location of the CM of an unknown object can be determined by establishing stable translational pushing at two or more contact points. The CM lies at the intersection of the rays of pushing. 2) If the shape and CM of the object are known, and the uncertainty in initial position and orientation is bounded so that the pushed edge is known, the pushing controller can remove uncertainty in the object location. The problem is that d cannot be directly sensed. If the object is translating, however, d is constant and is uniquely determined by the angle of pushing with respect to the edge:

$$d = \begin{cases} l/\tan(\theta - \phi) & \text{if } \phi - \theta < \pi/2 \\ l/\tan(\pi + \theta - \phi) & \text{if } \phi - \theta \geq \pi/2 \end{cases} \quad (17)$$

where l is given by the object model. The sensed value of θ and the estimated value of d completely determine the position and orientation of the object with respect to the pusher.

IV. Control Algorithm

The configuration of the object is controllable (i.e., the object can be pushed to any position and orientation) by pushing an edge if at least two noncollinear $\hat{\mathbf{q}}$ vectors, at least one of which has a nonzero ω component, can be obtained by pushing the edge. Sufficient conditions for the controllability of the object configuration are 1) the edge has nonzero length or a nonzero coefficient of friction (and is not a point at the CM), and 2) the support pressure is bounded everywhere, i.e., the limit surface is not flat at any point.

In determining the form of a pushing control law, we must consider the nonholonomic constraints on the motion of the object arising from the point contact with the pusher. The possible object velocities are constrained by functions of θ and d. One consequence of this fact is that it is impossible to design a controller to push the object to a desired configuration using a smooth feedback function of the system state. (See, for example, [14].) A method for determining bounds on the possible pushed object trajectories, independent of the control law, is presented in [15].

In this work, we focus on designing a control system to achieve stable translation of the pushed object, which allows active sensing of the CM as described in Section III. The discrete version of the control law can be written

$$\phi(k) = \phi(k-1) + k_p[\theta(k) - \theta_d] + k_d[\theta(k) - \theta(k-1)] \quad (18)$$

where $\phi(k)$ is the commanded push angle, $\theta(k)$ is the sensed contact orientation, θ_d is the desired contact orientation, and k_p and k_d are the proportional and derivative gains, respectively. The goal of this simple linear controller is to stabilize the contact orientation to a constant θ_d, which results in object translation (provided translation is possible).

This control scheme does not permit specification of a desired translation direction. In general the translation direction is not independent of the contact orientation. For the case $\mu_c = 0$, for example, the only possible translation direction is in the direction of the contact normal.

V. Simulation and Experiment

A. Experimental Setup

For our experiments we used a miniaturized version of the optical waveguide tactile sensor described in [16] mounted on one finger of the three-fingered hand developed at MEL [17]. The finger is mounted horizontally above the support plane, and the first of the three finger joints is fixed. The final two joints are used for planar finger tip motions. The finger is shown in a typical configuration in fig. 5.

The finger tip is hemispherical and the tactile sensor returns two angles, the azimuth and the elevation of the contact point. The projection of the hemisphere to the support plane is semicircular, allowing us to treat the finger as a disk, as only the hemispherical portion of the finger tip will contact the object. The elevation of the contact point and the finger joint angles are used to determine the contact normal in the plane of motion.

The radius of the tactile sensor is 16 mm, including a 2.5 mm spacing between the rubber covering and the optical waveguide. During contact, the rubber cover directly contacts the optical waveguide, giving the finger tip an effective radius of 13.5 mm.

The finger is mounted above a conveyor belt which moves the support at a constant velocity in the $-y$ direction. The quasi-static mechanics of pushing are the same as for the finger moving with a constant velocity v_y in the $+y$ direction. The finger tip generates the specified push angle ($0 < \phi < \pi$; in practice, we use $33^o < \phi < 147^o$)

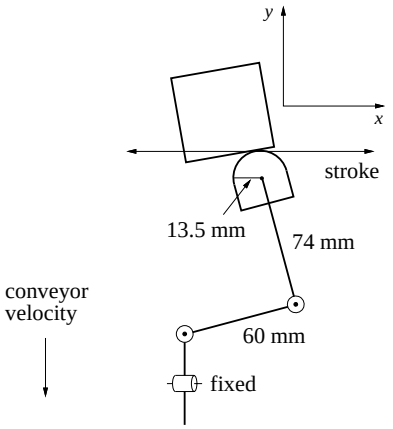

Fig. 5. Experimental setup.

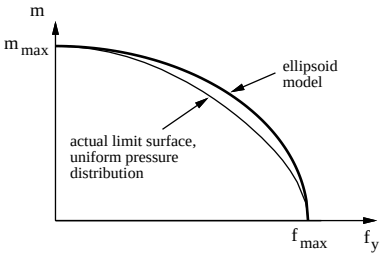

Fig. 6. Uniform pressure limit surface and the ellipsoid model in the $f_x = 0$ plane. The figure is symmetric about the f_y and m axes, and $v_x = 0$ for forces in the $f_x = 0$ plane.

by moving with velocity $v_x = v_y / \tan\phi$.[3] As a result of using the conveyor belt, the length of constant time pushes varies with ϕ and is equal to $v_y t / \sin\phi$, where t is the duration of the push.

Due to the quasi-static nature of the system, it is the length of each push, not the sampling rate, which determines whether or not the system can be controlled. In our experiments, the constant conveyor speed is 5 mm/s and the sampling rate is 10 Hz. This results in a push length of between 0.5 mm and 0.9 mm, depending on ϕ.

The object to be manipulated is a 5 cm x 4.75 cm rectangular steel block with a CM at the center. The 5 cm edge is pushed and $l = 2.375$ cm. The coefficient of friction μ_c between the finger and the block is very high; for simulation purposes, we assume $\mu_c = \infty$.

B. Simulation

In order to better understand the pushing model developed in Section II, we wrote a simulation of the derived equations of motion. The simulation may be used to determine appropriate control gains.

The rectangular object is modeled using three different values for c: $c_{max} = 3.45$ cm, $c_{uni} = 1.87$ cm, and $c_{min} = 0.67$ cm. c_{max} is found by assuming that the support is concentrated at the corners and c_{uni} is found by assuming a uniform support distribution. (Fig. 6 shows the $f_x = 0$ plane of the actual and modeled limit surfaces for a uniform pressure distribution.) In principle, c_{min} could be zero if there is only a single point of support at the CM. In this case, the limit surface is confined to the $m = 0$ plane and there is no quasi-static solution to the motion of the object. Instead, we chose c_{min} for the case of uniformly distributed support in a circle of radius 1 cm about the CM.

[3]We simply control the x velocity of the pole of the finger tip. This results in added linear velocity components at other points on the finger tip due to rotation of the finger tip. These velocities are relatively small, and contact is usually near the pole of the finger tip, so we ignore them.

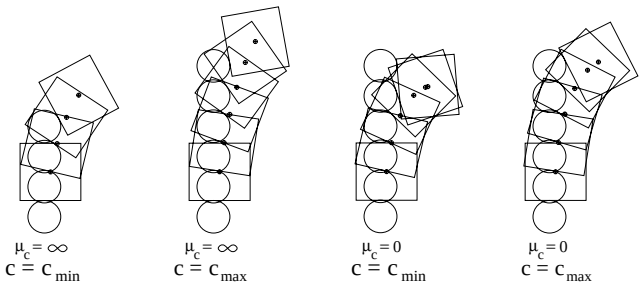

Fig. 7. Simulated object trajectories for open-loop pushing.

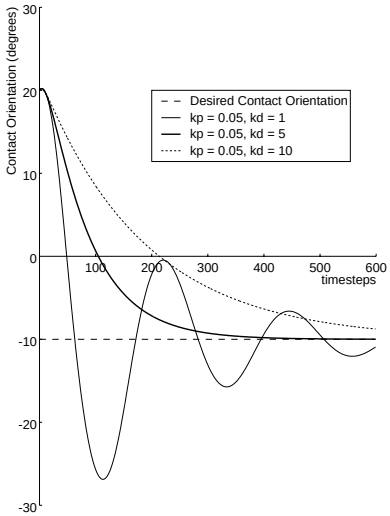

Fig. 8. Simulated step response for $\theta(0) = 20^o$, $d(0) = -0.6$ cm.

Fig. 7 shows the simulated trajectories of the rectangular object for $\phi = 90^o$, initial orientation $\theta(0) = 0^o$, initial contact $d(0) = -0.5$ cm and four combinations of c and μ_c. Fifty pushes are simulated between each snapshot.

Simulations indicate that the control law (18), with a proper choice of control gains (in particular, k_p, $k_d \geq 0$), is effective for a wide range of objects, μ_c, and initial conditions. The contact slips and rolls to a point from which the object can be translated. Fig. 8 shows the system response for three different combinations of k_p and k_d (for θ and ϕ measured in radians) for the rectangular object with $c = c_{uni}$, $\theta(0) = 20^o$, $d(0) = -0.6$ cm, $\phi(0) = 90^o$, and $\theta_d = -10^o$.

C. Experimental Results

We experimented with several different combinations of k_p and k_d (for θ and ϕ measured in radians) and empirically determined that $k_p = 0.05$, $k_d = 5.0$ gave the best performance. Fig. 9 shows an example time history of the system for these gain settings and $\theta_d = -10^o$. The thin line represents θ and the thick line represents $\phi - 90^o$. After 173 timesteps (17.3 s or 8.65 cm pushing in $+y$), θ stays within three degrees of θ_d. After 400 timesteps, $-10.4^o \leq \theta \leq -8.9^o$ and $80.2^o \leq \phi \leq 85.8^o$. The object is essentially translating. During this period the minimum value of $\phi - \theta$ is 90.6^o and the maximum value is 95.0^o. Plugging these values and l into (17), we determine bounds

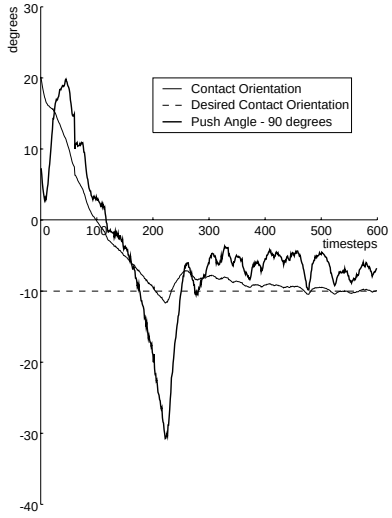

Fig. 9. Experimental results for the rectangular object.

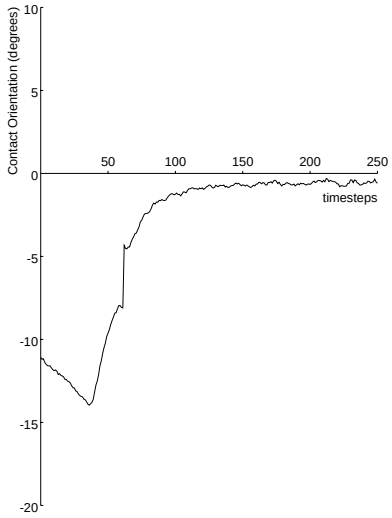

Fig. 10. Experimental results for the disk, $\theta_d = 0^o$.

on d during this time: 0.03 cm $\leq d \leq 0.21$ cm. By active sensing the possible values of d have been reduced from -2.5 cm $\leq d \leq 2.5$ cm.

We tried the same control law for a non-polygonal object: a steel disk of radius 2.85 cm with a CM at the center. The results for $\theta_d = 0^o$ are shown in fig. 10. After 110 timesteps, the contact orientation is maintained within one degree of θ_d.

D. Discussion

The controller works encouragingly well in these examples. Our implementation of the pushing controller is limited, however, by a relatively small finger workspace, occasional sensor error, and limits on detectable contact orientations due to the sensor design. These factors restrict the range of stabilizable initial system states.

The simulation suggests that larger values of k_p and k_d give better system response, but large experimental gains resulted in instability. The primary reason for this is that the simulation assumes perfect pusher position control. A more precise system model would include a model of the response of the robot finger.

VI. Conclusions

This work demonstrates through simulation and experimentation the feasibility of manipulation and active sensing by pushing using only tactile feedback. To increase the utility of a pushing controller, other control algorithms, including nonsmooth feedback laws, should be studied.

Acknowledgments

We thank the U.S. NSF and the STA of Japan for making this collaboration possible, and the members of the Cybernetics and Biorobotics Divisions at MEL and the Manipulation Lab at CMU for support and helpful comments.

References

[1] S. Akella and M. T. Mason, "Posing polygonal objects in the plane by pushing," in *IEEE International Conference on Robotics and Automation*, pp. 2255–2262, 1992.

[2] Z. Balorda, "Reducing uncertainty of objects by robot pushing," in *IEEE International Conference on Robotics and Automation*, pp. 1051–1056, 1990.

[3] R. C. Brost, "Automatic grasp planning in the presence of uncertainty," *International Journal of Robotics Research*, vol. 7, pp. 3–17, Feb. 1988.

[4] R. C. Brost, *Analysis and Planning of Planar Manipulation Tasks*. PhD thesis, Carnegie Mellon University, School of Computer Science, Jan. 1991.

[5] K. Y. Goldberg, *Stochastic Plans for Robotic Manipulation*. PhD thesis, Carnegie Mellon University, School of Computer Science, Aug. 1990.

[6] M. T. Mason, "Mechanics and planning of manipulator pushing operations," *International Journal of Robotics Research*, vol. 5, pp. 53–71, Fall 1986.

[7] M. A. Peshkin and A. C. Sanderson, "Planning robotic manipulation strategies for workpieces that slide," *IEEE Journal of Robotics and Automation*, vol. 4, pp. 524–531, Oct. 1988.

[8] F. Gandolfo, M. Tistarelli, and G. Sandini, "Visual monitoring of robot actions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 269–275, 1991.

[9] M. Inaba and H. Inoue, "Vision-based robot programming," in *International Symposium on Robotics Research*, 1989.

[10] S. Goyal, A. Ruina, and J. Papadopoulos, "Planar sliding with dry friction. Part 1. Limit surface and moment function," *Wear*, vol. 143, pp. 307–330, 1991.

[11] R. D. Howe, I. Kao, and M. R. Cutkosky, "The sliding of robot fingers under combined torsion and shear loading," in *IEEE International Conference on Robotics and Automation*, pp. 103–105, 1988.

[12] I. Kao and M. R. Cutkosky, "Quasistatic manipulation with compliance and sliding," Center for Design Research Technical Report 199110212, Stanford University, 1991. *International Journal of Robotics Research*, in press.

[13] S. H. Lee and M. R. Cutkosky, "Fixture planning with friction." *ASME Journal of Engineering for Industry*, in press.

[14] A. M. Bloch and N. H. McClamroch, "Control of mechanical systems with classical nonholonomic constraints," in *IEEE International Conference on Decision and Control*, pp. 201–205, 1989.

[15] K. M. Lynch, "The mechanics of fine manipulation by pushing," in *IEEE International Conference on Robotics and Automation*, pp. 2269–2276, 1992.

[16] N. Nakao, M. Kaneko, N. Suzuki, and K. Tanie, "A finger shaped tactile sensor using an optical waveguide," in *Conference of the IEEE Industrial Electronics Society*, pp. 300–305, 1990.

[17] H. Maekawa, K. Yokoi, K. Tanie, M. Kaneko, N. Kimura, and N. Imamura, “Position/stiffness based manipulation by three-fingered robot hand,” in *International Symposium on Advanced Robot Technology*, pp. 597–603, 1991.