

Exact Bounds on the Contact Driven Motion of a Sliding Object, With Applications to Robotic Pulling

Eric Huang[†], Ankit Bhatia[†], Byron Boots* and Matthew T. Mason[†]

[†]Robotics Institute

Carnegie Mellon University, Pittsburgh, Pennsylvania 15213

Email: {erich1,ankitb}@andrew.cmu.edu, matt.mason@cs.cmu.edu

*Interactive Computing

Georgia Institute of Technology, Atlanta, Georgia 30332

Email: bboots@cc.gatech.edu

Abstract—This paper explores the quasi-static motion of a planar slider being pushed or pulled through a single contact point assumed not to slip. The main contribution is to derive a method for computing exact bounds on the object’s motion for classes of pressure distributions where the center of pressure is known but the distribution of support forces is unknown. The second contribution is to show that the exact motion bounds can be used to plan robotic pulling trajectories that guarantee convergence to the final pose. The planner was tested on the task of pulling an acrylic rectangle to random locations within the robot workspace. The generated plans were accurate to $4.00\text{mm} \pm 3.02\text{mm}$ of the target position and $4.35\text{ degrees} \pm 3.14\text{ degrees}$ of the target orientation.

I. INTRODUCTION

Pushing (or pulling) planar objects with fixed contact is difficult to model in both theory and practice. First, pressure distributions of objects are statically indeterminate (barring the case of three-point support with known center of mass). Second, surface imperfections lead to spatial variability in both the pressure distribution and coefficient of friction [17]. Though several force-motion models for pushing exist [19, 7, 4], the above sources of indeterminacy ultimately lead to errors in the predicted velocity of the pushed object.

If the motion cannot be predicted, then another option is to find bounds on the velocity of the pushed object. This problem was first raised in Mason’s thesis on robotic pushing [11]. In the case of fixed contact pushing, this is equivalent to finding bounds on angular velocity of the object as it is pushed through the contact point. To this end, we develop the first algorithm that finds *exact* angular velocity bounds on the object’s motion over all pressure distributions with shared center of pressure. Moreover, the bounds are exact for many additional classes of pressure distributions that have not been considered before.

Dealing with uncertainty is a fundamental challenge in robotics [16]. We demonstrate how our bounds can be applied to planning for robotic pulling under action uncertainty. Robotic pulling is a general-purpose manipulation skill for positioning and orienting objects. The proposed planner uses the angular velocity bounds to find actions that reduce the uncertainty in the system, i.e. close the distance between the integrated orientation bounds. Moreover, given a suitable

initialization, the planner finds trajectories that guarantee the uncertainty at the final pose converges to a very small value.

The rest of the paper is organized as follows. Section II discusses related work. Section III summarizes the relevant background on planar pushing needed to understand our work. Section IV develops several theoretical results needed to prove the correctness of our algorithmic contributions. Section V introduces the exact angular velocity bound algorithm and the algorithm for planning pulling trajectories under action uncertainty. Section VI presents our experimental results. Section VII gives concluding remarks on the paper.

II. RELATED WORK

We can categorize prior work bounding the motion of a pushed object according to the tightness of the bounds and the basic assumptions from which those bounds are deduced, such as knowledge of the center of pressure location. The bisector bound restricts the feasible rotation centers to a half-space delimited by the perpendicular bisector between the contact point and center of pressure [12]. Alexander and Maddocks bounded the set of feasible rotation centers to lie within the minimum object-enclosing vertical strip perpendicular to the wrench applied by the pusher [1]. Peshkin and Sanderson bounded the motion of a pushed object by computing the set of feasible rotation centers of the minimum object-enclosing disk centered at the object’s center of pressure [13]. In all prior work, the bounds are conservative but not exact. That is, the bounds include every feasible motion, but also include infeasible motions.

To the extent of our knowledge, prior work has used robotic pulling significantly less compared to its counterpart, robotic pushing. The stable equilibrium that occurs when pulling along a straight line was recognized, without proof, by Mason [10], Lynch and Mason [8], and Berretty et al. [2]. For completeness, we provide a proof in Subsection IV-A. To the best of our knowledge, Berretty et al. is the only work, apart from ours, to explicitly take advantage of this stability [2]. However, the method of Berretty et al. is limited to orienting asymmetrical convex polygons. In contrast, our method can both position and orient any pullable object, regardless of its geometry. Our planner is also related to the work of Lynch

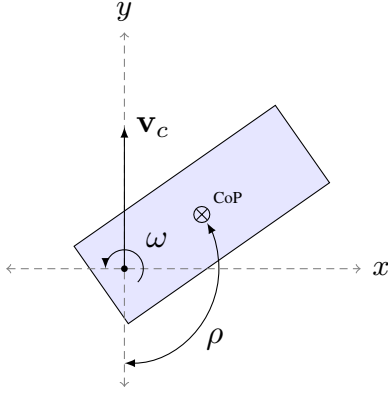


Fig. 1: Coordinate frame associated with a press-pulled slider, where \mathbf{v}_c is the contact point velocity, ω is the angular velocity of the slider, ρ is the angular deviation from the stable pulling configuration.

and Mason [9]. They invented an algorithm that plans stable pushes using two contact points. However, their use of weaker bounds [13, 1, 12] leads to more conservative strategies.

III. BACKGROUND

This section provides the relevant background needed to understand our contributions. Subsection III-A serves two purposes. First, it defines the notation and terminology used throughout our paper. Second, it frames our work in the context of the quasi-static theory of planar pushing. Subsection III-B summarizes the moment envelope construct used to jointly reason about pressure and frictional torques. This construct is particularly relevant for Subsection V-B, which introduces an algorithm that computes the feasibility of a given pushed object velocity.

A. Planar Pushing Subject to Friction

In this paper, we treat planar pushing as the manipulation skill where the robot contacts a rigid body at a point and pushes (or pulls) that fixed contact point along a trajectory. Our following presentation of the quasi-static analysis of planar pushing follows that of [1, 12]. A quasi-static analysis seeks to balance contact forces, gravity, and other applied forces while neglecting inertial forces [12].

Let the *generalized velocity*, or *twist*, of a planar rigid body be $\mathbf{v}^+ = [v_x, v_y, \omega]^T$, where v_x and v_y are the linear velocities of a reference point and ω is the angular velocity about that point. Taking the origin as the reference point, the velocity of a point $\mathbf{x} = [x_1, x_2]^T$ on the body is then given by $\mathbf{v}(\mathbf{x}) = [v_x, v_y]^T + \omega \hat{\mathbf{k}} \times \mathbf{x}$ with $\hat{\mathbf{k}} = [0, 0, 1]^T$, and can be written in matrix notation as

$$\mathbf{v}(\mathbf{x}) = A(\mathbf{x})\mathbf{v}^+, \quad (1)$$

where

$$A(\mathbf{x}) = \begin{bmatrix} 1 & 0 & -x_2 \\ 0 & 1 & x_1 \end{bmatrix}. \quad (2)$$

Where convenient, we will convert freely between twists and their equivalent formulation, rotation centers. The mappings

$$[v_x, v_y, \omega] \rightarrow \left[-\frac{v_y}{\omega}, \frac{v_x}{\omega}, \omega\right] \quad (3)$$

$$[r_x, r_y, \omega] \rightarrow [r_y\omega, -r_x\omega, \omega] \quad (4)$$

map a twist at the origin to a rotation center and angular velocity about that rotation center and vice versa.

We define the *contact frame* to be the coordinate frame where the origin is the contact point and the y -axis aligns with the contact point velocity (see Figure 1 for an example). Let ρ be the *angular deviation* from the stable pulling configuration to the rigid body's center of pressure. The stable pulling configuration occurs when the center of pressure is collinear with and behind the direction of the pulling motion (proof in Subsection IV-A). The contact point is assumed to be pushed with unit speed, yielding a body twist $\mathbf{v}^+ = [0, 1, \omega]^T$. Assuming Coulomb's law of sliding friction, the total frictional force and moment of the body at the origin are

$$\mathbf{f}_f = -\mu \int_R \frac{A(\mathbf{r})\mathbf{v}^+}{\|A(\mathbf{r})\mathbf{v}^+\|} p(\mathbf{r}) dA \quad (5)$$

$$\mathbf{m}_f = -\mu \int_R \mathbf{r} \times \frac{A(\mathbf{r})\mathbf{v}^+}{\|A(\mathbf{r})\mathbf{v}^+\|} p(\mathbf{r}) dA, \quad (6)$$

where μ is the coefficient of friction (static and dynamic), R is the region of the rigid body in contact with the plane, \mathbf{r} is a point in R , $\mathbf{v}(\mathbf{r})$ is the body point velocity given by (1) and $p(\mathbf{r})$ is a pressure distribution over R .

When the contact point is pushed at constant velocity, the quasi-static assumption states that the total moment at the contact point is zero. Because a point contact cannot generate any torque, this implies the total frictional moment (6) must also be zero. This leads to the following constraint on the possible motions of an object.

Definition 1. An angular velocity ω and its corresponding twist $\mathbf{v}^+ = [0, 1, \omega]^T$ are said to be **feasible** if there exists a pressure distribution $p(\mathbf{r})$ such that the resulting total frictional moment is zero.

Note that both the angular velocity and its corresponding twist are taken with respect to the contact point frame. However, using equations (5) and (6) directly to check feasibility can lead to difficulties when the integrand's denominator $\|A(\mathbf{r})\mathbf{v}^+\|$ is zero. The next formulation obviates that difficulty.

The principle of minimal dissipation states that the motion of the pushed body minimizes the instantaneous work dissipated by friction [1]. That is, the motion minimizes the following:

$$\begin{aligned} &\underset{\mathbf{v}^+}{\text{minimize}} && \mu \int_R \|A(\mathbf{r})\mathbf{v}^+\| p(\mathbf{r}) dA \\ &\text{subject to} && \mathbf{v}^+ \in \mathcal{C}. \end{aligned} \quad (7)$$

We take $\mathcal{C} = \{[0, 1, \omega]^T, \omega \in \mathbb{R}\}$ so that the contact point motion is aligned with the coordinate frame. We call the objective in (7) the *frictional dissipation function*, $P(\mathbf{v}^+)$. The

principle of minimal dissipation is equivalent to the quasi-static model of planar sliding with friction [1]. In fact, the quasi-static motion constraints are identical to the first order optimality conditions of (7). To see this, take the Lagrangian of (7)

$$L(\mathbf{v}^+, \lambda) = P(\mathbf{v}^+) + \lambda_x v_x^+ + \lambda_y (v_y^+ - 1), \quad (8)$$

where λ_x and λ_y are Lagrangian multipliers, and set gradient of L with respect to \mathbf{v}^+ ,

$$\nabla L(\mathbf{v}^+, \lambda) = \mu \int_R \frac{A(\mathbf{r})^T A(\mathbf{r}) \mathbf{v}^+}{\|A(\mathbf{r}) \mathbf{v}^+\|} p(\mathbf{r}) dA + [\lambda_1, \lambda_2, 0]^T, \quad (9)$$

to zero. When equation 9 is written out element-wise, we end up with the following first order conditions on the force and the moment

$$\mathbf{f}_f = [\lambda_x, \lambda_y]^T \quad (10)$$

$$\mathbf{m}_f = 0, \quad (11)$$

that is, the quasi-static motion model.

In addition to avoiding the zero denominator issue of equations (5) and (6), we prefer using the principle of minimal dissipation in the proofs of our main results (Section IV) because the frictional dissipation equation (7) is continuous and convex in \mathbf{v}^+ .

B. Frictional Moment Envelopes

The frictional moment envelope provides a nice geometric model of the constraints between the center of pressure and moment at the contact point [12]. In this subsection, we show that the moment envelope is important because it reduces the feasibility test of a particular angular velocity to a point-in-convex-hull test. An example frictional moment envelope is illustrated step-by-step in Figure 2.

A frictional moment envelope has as its parameters the support region R and twist \mathbf{v}^+ . Let f_0 be the total normal force. Then the function

$$g(\mathbf{x}) = -\mu f_0 \mathbf{x} \times \frac{A(\mathbf{x}) \mathbf{v}^+}{\|A(\mathbf{x}) \mathbf{v}^+\|} \quad (12)$$

evaluates the frictional torque that would result from a unit normalized pressure at \mathbf{x} . Let G map R into a surface in \mathbb{R}^3 by associating each point $\mathbf{x} \in R$ with its maximum potential moment $g(\mathbf{x})$, i.e.

$$G(\mathbf{x}) = \begin{bmatrix} x \\ y \\ g(\mathbf{x}) \end{bmatrix}, \quad (13)$$

and let $\hat{p} = p/f_0$ be the normalized pressure. Then the set $\{\int_R G(\mathbf{r}) \hat{p}(\mathbf{r}) dA \mid \int_R \hat{p}(\mathbf{r}) dA = 1\}$ is the convex hull of the surface $G(R)$. Moreover, any point in the convex hull of $G(R)$

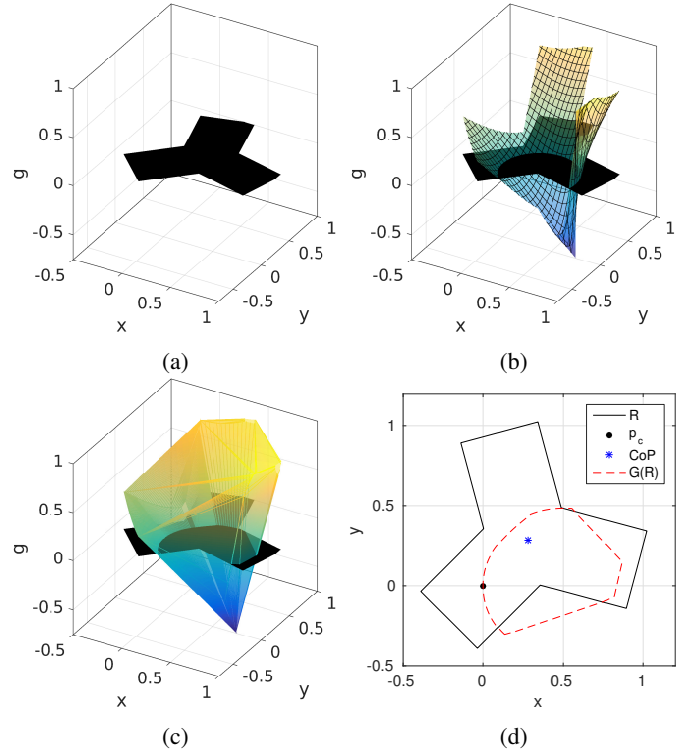


Fig. 2: Example moment envelope for a trigonal 2D object with rotation center $x_{IC} = 0.75$. (a) Support region R . (b) Normalized-moment surface $G(R)$. (c) Convex moment envelope of $G(R)$. (d) Intersection of the moment envelope and the xy -plane. The intersection bounds the set of feasible centers of pressure with zero moment.

satisfies

$$\begin{aligned} \int_R G(\mathbf{r}) \hat{p}(\mathbf{r}) dA &= \int_R \begin{bmatrix} x \\ y \\ g(\mathbf{x}) \end{bmatrix} \hat{p}(\mathbf{r}) dA \\ &= \begin{bmatrix} x_0 \\ y_0 \\ \int_R g(\mathbf{x}) \hat{p}(\mathbf{r}) dA \end{bmatrix}. \end{aligned} \quad (14)$$

Thus, the convex hull of $G(R)$ is the set of all feasible centers of pressure and frictional moments for a given support region R and twist \mathbf{v}^+ . We refer to the convex hull of $G(R)$ as the *moment envelope generated by \mathbf{v}^+* . Therefore, given a center of pressure $[x_0, y_0]^T$, an angular velocity ω is feasible if and only if the point $[x_0, y_0, 0]^T$ is contained in the moment envelope generated by ω .

IV. THEORY

This section covers our theoretical contributions. Subsection IV-A proves the existence and uniqueness of the stable equilibrium during pulling. This result motivates our main contributions. Subsection IV-B lays the theoretical groundwork necessary for proving the correctness of the exact angular velocity bound algorithm introduced in V-A. Subsection IV-C extends the angular velocity bounds to orientation bounds.

This subsection completes the theoretical tools needed in Subsection V-C to generate robotic pulling trajectories that guarantee convergence to the final pose.

A. Stable Equilibrium When Pulling

For completeness, we prove the existence and uniqueness of the stable equilibrium that occurs when pulling a rigid body. The existence of the stable equilibrium was observed, without proof, in Mason [10], Lynch and Mason [8], and Berretty et al. [2]. The robotic pulling trajectories generated in Subsection V-C automatically use the stable equilibrium to reduce pose uncertainty.

Theorem 1. *For pulling of a rigid body in the plane, the rigid body converges to the state where its center of pressure is collinear with the pulling direction.*

Proof: We know from Theorem 7.4 in [12] that the rigid body translates when the center of pressure is already collinear with the pulling direction. Now, suppose the center of pressure is strictly to the right of the pulling direction (as in Figure 1). Then by Theorem 7.4 in [12], the rigid body rotates clockwise about the contact point, i.e. has angular velocity $\omega < 0$. Let $\rho \in (\pi, 0)$ be the angular deviation of the rigid body. Since ρ is monotonically decreasing and $\omega = 0$ if $\rho = 0$ or π , we see that ρ converges to 0 in the limit as $t \rightarrow \infty$. The case when the center of pressure is strictly to the left of the line of motion follows from symmetry. ■

B. Properties of Angular Velocities Bounds

We prove that the set of feasible angular velocities for an object with known center of pressure is connected and bounded. These two properties justify the use of a bisection search to locate the minimum and maximum angular velocities in Subsection V-A.

We begin by citing a proposition from variational analysis used in our proofs of the main results.

Proposition 1. *Suppose $P(u) := \arg \min_x f(x, u)$ with $f : \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{R}$ continuous and level-bounded in x locally uniformly in u . Then the set-valued mapping $P(u)$ is outer-semicontinuous and locally bounded.*

Proof: Proposition adapted from Corollary 7.42 and Theorem 1.17 in [14]. ■

Theorem 2. *For pulling of a planar rigid body with known center of pressure, the set of all feasible angular velocities is connected.*

Proof: Let Ω be the set of feasible angular velocities, w.r.t. (7), for a given support region R with known center of pressure. Suppose Ω is non-empty. Let $\omega_1, \omega_2 \in \Omega$ and let p_1, p_2 be their corresponding pressure distributions. Then the new distribution $p_t = tp_1 + (1-t)p_2$, with $t \in [0, 1]$, shares the same center of pressure as p_1, p_2 . Define the function

$$f(\omega, t) = \mu \int_R \|A(\mathbf{r})\mathbf{v}^+(\omega)\| p_t(\mathbf{r}) dA, \quad (15)$$

where $\mathbf{v}^+ : \omega \rightarrow [0, 1, \omega]^T$ and $\text{dom } f := \mathbb{R} \times [0, 1]$. By inspection, f is continuous. For all $t \in [0, 1]$ and $\alpha \in \mathbb{R}$, the set $\{(\omega, t) \mid f(\omega, t) \leq \alpha\}$ is bounded because $f \rightarrow \infty$ as $|x| \rightarrow \infty$. Hence, f is level-bounded in ω locally uniformly in t , and f satisfies the conditions of Proposition 1.

The image of a connected set by an outer-semicontinuous set-valued mapping whose values are nonempty and connected is connected [6]. Let $P(t) := \arg \min_\omega f(\omega, t)$. For a given t , the set $P(t)$ is convex-valued and therefore connected. Since f is continuous and level-bounded, the set is also nonempty (Theorem 1.9 [14]). Therefore, the image $P([0, 1])$ contains the interval connecting ω_1 and ω_2 . Because the choice of ω_1, ω_2 was arbitrary, Ω is connected. ■

In general, Theorem 2 holds for any convex set of pressure distributions with known center of pressure.

Corollary 1. *For pulling of a planar rigid body with known center of pressure, the set of all feasible angular velocities is bounded.*

Proof: We prove Corollary 1 separately for the cases when the rigid body's center of pressure lies strictly in the right-half plane, left-half plane, or on the y -axis.

Suppose the center of pressure lies in the right-half plane of the contact frame. Then by Theorem 7.4 of [12] the angular velocity of the pulled body is strictly negative and hence, 0 bounds Ω from above. To establish a lower bound, we appeal to the following two properties about the set of feasible rotation centers when pushing or pulling with sticking contact. First, the rotation center must lie on the x -axis. Second, the rotation center must lie behind the line bisecting the contact point and the center of pressure [12]. We observe that any feasible rotation center must have the form $[x, 0]$ with $x > x^* > 0$, where x^* is the intersection of the bisecting line and the x -axis. Recall that, by convention, we fix the contact point velocity to $\mathbf{v}_c = [0, 1]^T$. We compute the angular velocity ω^* at $[0, x^*]$ using \mathbf{v}_c and see that Ω is bounded from below by

$$\omega^* = -\frac{\|\mathbf{v}_c\|}{x^*}. \quad (16)$$

Thus, Ω is bounded. The case when the center of pressure lies in the left-half plane follows from a symmetrical argument.

When the center of pressure is collinear with the pulling direction, the rigid body translates and $\Omega = \{0\}$ [12]. ■

C. Integrated Orientation Bounds

We prove that the angular velocity bounds derived in Subsection IV-B integrate into bounds on the orientation of the pulled body.

Let θ be the orientation of the rigid body in the world frame and let u and l be upper and lower bounds on the orientation in the world frame. Let the function $\omega(\rho)$ map from the angular deviation to the angular velocity of the rigid body. Likewise, let the functions $\alpha(\rho)$ and $\beta(\rho)$ map from the angular deviation to the upper and lower angular velocity bounds. An example phase-plot of $\alpha(\rho)$ and $\beta(\rho)$ is illustrated in Figure 3. Note the stable equilibrium at $\theta = 0$.

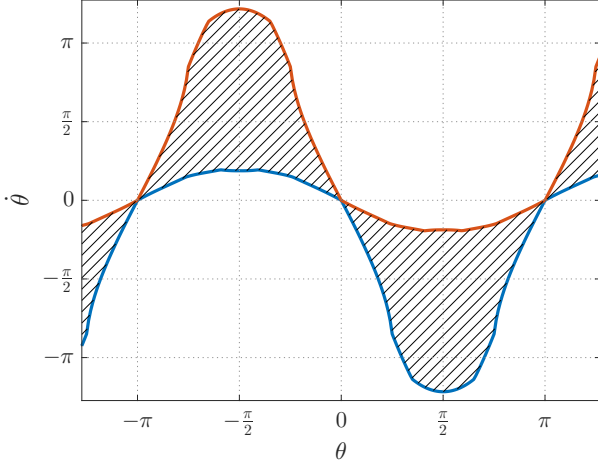


Fig. 3: Example angular velocity bounds for the trigonal 2D object from Figure 2. The object is oriented such that the stable pulling configuration corresponds to $\theta = 0$. The orange curve is the upper bound α . The blue curve is the lower bound β .

We assume that the pulling trajectory $\gamma : \mathbb{R} \rightarrow \mathbb{R}^2$ can be approximated by a finite number of straight line segments of equal length. Given such a γ , the pulling angle $\phi(t) = \tan^{-1}(\dot{\gamma}_y(t), \dot{\gamma}_x(t))$ is a piece-wise constant (step) function. Let $v(t) = \|\dot{\gamma}(t)\|$. As the planar rigid body is pulled along γ with unit velocity, the state and bounds change according to the dynamical system

$$\dot{x} = \cos(\phi(t)) \quad (17)$$

$$\dot{y} = \sin(\phi(t)) \quad (18)$$

$$\dot{\theta} = \omega(\theta - \phi(t) + \pi) \quad (19)$$

$$\dot{u} = \alpha(u - \phi(t) + \pi) \quad (20)$$

$$\dot{l} = \beta(l - \phi(t) + \pi), \quad (21)$$

where the expression $z - \phi(t) + \pi$ maps an orientation z in the global frame to the angular deviation.

Proposition 2. *For pulling of a rigid body with known initial pose, the orientation of the body is bounded above and below by u and l .*

Proof: Suppose the proposition is false and θ crosses the bound u at time t_0 , i.e. $u(t_0) = \theta(t_0)$ and $\theta(t) > u(t)$ immediately afterwards. Let the line segment of γ at t_0 be indexed by i and have length ε . Then $\phi(t)$ is constant for t in the range $[t_0, (i+1)\varepsilon]$. Pick t_1 from said range such that $\theta(t_1) > u(t_1)$. Because ϕ is constant in $[t_0, t_1]$, we can apply separation of variables to solve differential equation (20) and get

$$\int_{u(t_0)}^{u(t_1)} \frac{1}{\alpha(x - \phi(t_0))} dx = \int_{t_0}^{t_1} dt. \quad (22)$$

This result shows that we can integrate the inverse of an angular velocity function to compute the amount of time required to reach a particular orientation. However, we can

Algorithm 1 Exact Angular Velocity Bounds

```

1: function FIND EXTREMA( $R, x_0, y_0$ )
2:   if  $x_0$  is 0 then return  $[0, 0]$ 
3:    $P \leftarrow \min_{\mathbf{x}} \mathbf{0}$  s.t.  $R\mathbf{x} = [x_0, y_0]^T, L \leq \mathbf{x} \leq U$ 
4:    $x_r \leftarrow$  COMPUTE ROTATION CENTER( $R, P$ )
5:    $\omega \leftarrow -\|\mathbf{v}_c\|/x_r$ 
6:    $l \leftarrow 0$ 
7:    $\omega_1 \leftarrow$  BISECTION SEARCH( $R, x_0, y_0, l, \omega$ )
8:    $u \leftarrow \omega$ 
9:   do
10:     $u \leftarrow 2u$ 
11:     $\mathbf{v}^+ \leftarrow [\mathbf{v}_c^T, u]^T$ 
12:     $G \leftarrow \{-\mathbf{x} \times A(\mathbf{x})\mathbf{v}^+ / \|A(\mathbf{x})\mathbf{v}^+\| \mid \mathbf{x} \in R\}$ 
13:    while  $[x_0, y_0, 0]^T \in \text{CONVHULL}(G)$ 
14:     $\omega_2 \leftarrow$  BISECTION SEARCH( $R, x_0, y_0, u, \omega$ )
15:     $l \leftarrow \text{MIN}(\omega_1, \omega_2)$ 
16:     $u \leftarrow \text{MAX}(\omega_1, \omega_2)$ 
17:  return  $[l, u]$ 
18: function BISECTION SEARCH( $R, x_0, y_0, \alpha, \beta$ )
19:  while  $\varepsilon < |\alpha - \beta|$  do
20:     $\omega \leftarrow (\alpha + \beta)/2$ 
21:     $\mathbf{v}^+ \leftarrow [\mathbf{v}_c^T, \omega]^T$ 
22:     $G \leftarrow \{-\mathbf{x} \times A(\mathbf{x})\mathbf{v}^+ / \|A(\mathbf{x})\mathbf{v}^+\| \mid \mathbf{x} \in R\}$ 
23:    if  $[x_0, y_0, 0]^T \in \text{CONVHULL}(G)$  then
24:       $\beta \leftarrow \omega$ 
25:    else
26:       $\alpha \leftarrow \omega$ 
27:  return  $(\alpha + \beta)/2$ 

```

also apply separation of variables to the function ω . Observe that because α is an upper bound on ω

$$\alpha(x - \phi(t_0)) \geq \omega(x - \phi(t_0)), \quad (23)$$

the resulting integral of ω satisfies

$$\int_{u(t_0)}^{u(t_1)} \frac{1}{\alpha(x - \phi(t_0))} dx \leq \int_{u(t_0)}^{u(t_1)} \frac{1}{\omega(x - \phi(t_0))} dx. \quad (24)$$

This indicates that it takes less time for u to reach $u(t_1)$ compared to θ , and therefore, $\theta(t_1) \leq u(t_1)$, a contradiction. Therefore, the upper bound holds and the lower bound follows from a similar argument. ■

V. METHODS

In this section, we synthesize the materials in Sections III and IV into an algorithm for computing exact angular velocity bounds and a method for planning convergent trajectories using the computed bounds. The former is detailed in Subsection V-A and extended in Subsection V-B and the latter is detailed in Subsection V-C.

A. Exact Angular Velocity Bound Algorithm

Algorithm 1 finds exact angular velocity bounds for a given support region R and center of pressure $[x_0, y_0]^T$. It uses bisection search to estimate the end-points of Ω . This choice

is justified because Ω is connected and bounded by Theorem 2 and Corollary 1.

To initialize the bisection search, we need an angular velocity in Ω and two angular velocities above and below the bounds of Ω . First, we compute an angular velocity ω in Ω . We find a feasible assignment of pressures P such that the center of pressure is $[x_0, y_0]$ (line 3). From P , the resultant rotation center $[x_r, 0]$ can be computed using the root-finding method in [11] (line 4). Lastly, we convert the rotation center into the angular velocity $\omega \in \Omega$ (line 5). Of the two out-of-bound angular velocities l and u , we can set l to 0 (line 6). The other can be found by repeatedly doubling ω until the resulting angular velocity is no longer feasible (lines 8-13). As a reminder, we test the feasibility of a given angular velocity ω' by checking whether the point $[x_0, y_0, 0]^T$ is contained in the associated frictional moment envelope (see Section III-B). Now that ω , u , and l have been computed, we pass them into the bisection search to compute the boundary points of Ω (lines 7 and 14).

The run-time of the algorithm is $\mathcal{O}(dn \log n)$, where d is the number of significant digits returned and n is the number of points in the discretization of R . This computation is relatively expensive to perform online. In Section V-C, we avoid recomputing angular velocity bounds by fitting Fourier series to a set of pre-computed orientation-bound pairs.

B. Improving on Exact Angular Velocity Bounds

The exact angular velocity bounds computed in Section V-A result in slow convergence towards the stable pulling equilibrium point (for experimental measurements, see Subsection VI-A). Consequently, wide bounds cause our planner to generate long trajectories that exceed the robot's workspace in order to satisfy tolerances on the final pose uncertainty.

In this subsection, we show how to modify the constraints on the pressure distributions from which the bounds were computed. This allows us to restrict pressure distributions to smaller subclasses and thus achieve tighter angular velocity bounds. Let \mathcal{C} be the class of normalized pressure distributions over a region R with center of pressure $[x_0, y_0]$. Now, suppose we had a convex subclass \mathcal{K} of pressure distributions such that $\mathcal{K} \subset \mathcal{C}$. Regrettably, the point-in-convex-hull feasibility test only works for \mathcal{C} . However, we can setup an alternative feasibility test with respect to \mathcal{K} by solving the linear program

$$\begin{aligned} & \underset{p}{\text{minimize}} && \left\| \sum_R g(\mathbf{r})p(\mathbf{r}) \right\| \\ & \text{subject to} && p \in \mathcal{K}, \end{aligned} \quad (25)$$

where p is a discretized pressure distribution, $g(\mathbf{r})$ is the unit-torque function from equation (12), and the summation is over points $\mathbf{r} \in R$. A given angular velocity ω is feasible if and only if the linear program (25) finds a pressure distribution p such that the objective $\|\sum_R g(\mathbf{r})p(\mathbf{r})\|$ is 0 and $p \in \mathcal{K}$.

Several options exist for the choice of \mathcal{K} . In our experiments, we use

$$\mathcal{K} = \{p \mid 0 \leq p_i \leq U, p \in \mathcal{C}\}, \quad (26)$$

where $U \leq 1$ is an upper bound on the discretized pressures. The upper bound U controls the percentage of R guaranteed to be in contact with the surface, i.e. has non-zero pressure. For example, if we set $U = 2/N$, where N is the number of points in the discretization of R , then, by the pigeon-hole principle, at least 50% of R is always in contact with the surface. Our implementation solves linear program (25) using Gurobi [5].

C. Planning Convergent Trajectories for Robotic Pulling

We use control-limited Differential Dynamic Programming (DDP) [15] to plan convergent trajectories for robotic pulling. At a high level, DDP solves a trajectory optimization problem where the objective is to reach a target pose $[x_T, y_T, \omega_T]^T$ with small uncertainty, i.e. the integrated orientation bounds are within ϵ of ω_T .

Let the discretized state be $\mathbf{x}_i = [x_i, y_i, u_i, l_i]^T$, $i \in [1, N]$. Let the discretized controls be $\mathbf{u}_i = [d_i, \phi_i]^T$, $i \in [1, N]$, where d_i is the distance to travel and ϕ_i is the heading in the global frame. We use the following first order approximation of the discretized dynamics in our trajectory optimizer

$$x_{i+1} = x_i + d_i \cdot \cos(\phi_i) \quad (27)$$

$$y_{i+1} = y_i + d_i \cdot \sin(\phi_i) \quad (28)$$

$$u_{i+1} = u_i + d_i \cdot \hat{\alpha}(u_i - \phi_i + \pi) \quad (29)$$

$$l_{i+1} = l_i + d_i \cdot \hat{\beta}(l_i - \phi_i + \pi) \quad (30)$$

$$h_{i+1} = h_i + d_i, \quad (31)$$

where the functions $\hat{\alpha}$ and $\hat{\beta}$ are Fourier series approximations of the upper and lower angular velocity bounds (to avoid costly in-loop computations) and the additional state h_i measures the cumulative distance pulled.

We use the following cost functions to bias the trajectory optimizer towards finding convergent trajectories. We set the running cost $\mathcal{L}(\mathbf{x}_i, \mathbf{u}_i)$, $i \in [1, N-1]$ to zero. We set the final cost to be

$$\mathcal{L}_F(\mathbf{x}_N, h_N) = \mathbf{k}^T \mathcal{L}_\delta(\mathbf{x}_N - \mathbf{x}_F) + \lambda h_N^2, \quad (32)$$

where \mathcal{L}_δ is the vectorized version of the Pseudo-Huber loss function¹

$$\mathcal{L}_\delta(a) = \sqrt{a^2 + \delta^2} - \delta, \quad (33)$$

$\mathbf{x}_F = [x_T, y_T, \omega_T, \omega_T]^T$ is the target state, \mathbf{k} and δ are the slope and width, respectively, of the vectorized Pseudo-Huber loss function, and λ is the distance penalty coefficient. Note that the target upper and lower orientation bounds are equal to ω_T in the target state \mathbf{x}_F . This ensures the generated trajectory minimizes uncertainty in the final orientation (due to Proposition 2). Finally, we initialize our trajectory optimizer using paths generated from Dubins' curves. Pushing with sticking contact shares similar dynamics with the simple car [3, 18].

¹This function approximates an ℓ_1 norm for $a > \delta$.

	Exact	Exact-50%	Peshkin
MIT	0.670±0.141	0.362±0.073	1.354±0.501
	0.172±0.047	0.243±0.053	0.162±0.041
Bipod	0.762±0.210	0.692±0.213	0.899±0.239
	0.516±0.218	0.574±0.216	0.273±0.095
Tripod	0.765±0.133	0.688±0.143	1.180±0.244
	0.522±0.162	0.576±0.157	0.340±0.108
Quadrapod	0.880±0.120	0.749±0.114	1.207±0.235
	0.417±0.099	0.489±0.098	0.329±0.096

TABLE I: Comparison of distance-to-convergence (in meters) for different objects and angular velocity bounds. The top and bottom values in each cell correspond to distances from the upper and lower angular velocity bounds, respectively. See Section VI-A for the experimental setup.

VI. EXPERIMENTS

A. Comparison of Angular Velocity Bounds

In this experiment, we compare distance-to-convergence for our exact angular velocity bounds and the previous best bound, i.e. the Peshkin bound [13]. We test the bounds over the objects in the MIT Pushing Dataset [17] and randomly generated bipods, tripods, and quadrapods. The generated n -pods were chosen to have circumcircle diameters similar to the MIT objects, roughly 0.16m.

For each MIT object, we pick 10 even spaced contact points on the boundary of the object. We generate 30 random n -pods for each category and took the contact point to be the center of a random pod (similar to pulling the leg of a chair). We compute distance-to-convergence in the following manner. Let γ be an angular velocity bound (can be upper or lower). We orient the object such that the center of pressure is 90 degrees away from the stable configuration. Next, we simulate a pulling trajectory while integrating γ and stop when the integral converges to within 1 degree of the stable configuration. The distance travelled is the distance-to-convergence².

The experimental results are collected in Table I. The Peshkin bound computes the feasible angular velocities for the circumcircle enclosing the object. As a result, it underestimates the slowest angular velocity bound and its distance-to-convergence can be twice as far as compared to the exact bound. When feasible pressure distribution are restricted such that at least 50% of the object is in contact with the surface, the distance-to-convergence of the exact bound is reduced by another factor of two. Because the exact bound converges within 3/4 a meter, it is serviceable for manipulating the MIT objects on a large table. Naturally, smaller objects or tighter bounds are required for smaller tables.

B. Robotic Pulling on a Tabletop

Figure 4 shows the experimental setup that we used to test the robotic pulling trajectories generated by the planning algorithm in Subsection V-C.

²Note that this distance is independent of the pulling velocity, see Equation (??).

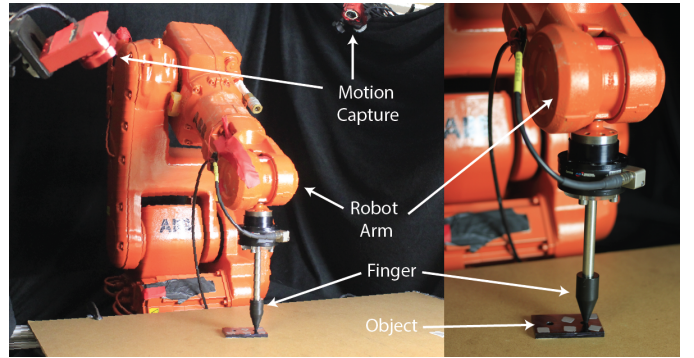


Fig. 4: Hardware setup for robotic pulling experiments.

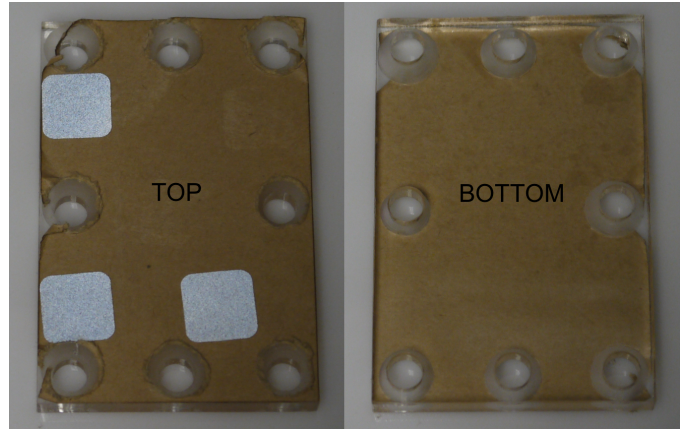


Fig. 5: The top and bottom view of the acrylic rectangle with motion capture markers and 8 holes for pulling.

Experimental data was collected using an ABB 140 manipulator equipped with a conical finger. The test object was a laser-cut acrylic rectangle (75mmx50mmx6.35mm) with 8 holes at the edges and corners. The conical finger moved the acrylic rectangle by pulling inside the holes. A 5 camera OptiTrack motion capture system was set up to record ground truth position of the object in 2D with a accuracy of 2mm. To compensate sensing error, the holes on the rectangle were oversized to have a 3mm radius. We used MDF board as our surface material.

We computed angular velocity bounds for the acrylic rectangle over pressure distributions restricted to have at least 25% of the object is in contact with the surface. The slope k of the Pseudo-Huber Loss function for the DDP planner was set to [5000, 5000, 1000, 1000] and the width δ was set to [0.01, 0.01, 0.02, 0.02]. The distance penalty λ was set to 40. For each pulling trial, we generated random start and end poses within the vision system’s field of view. The planner was evaluated for all eight contact points and the lowest cost trajectory that remained within the robot workspace was executed on the robot at 25mm/s linear speed. The final pose was then recorded by the motion capture system.

We collected eighty trials of robotic pulling. Of those eighty,

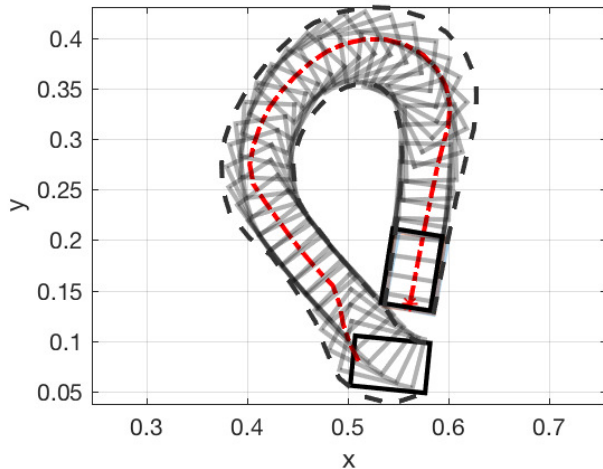


Fig. 6: (Dashed red line) The planned robotic pulling trajectory. (Dashed black line) The area swept by the possible poses computed by our bounds during the planned trajectory. (Grey rectangles) The measured poses of the object when the trajectory was executed on a real robot.

we discarded the four trials where our planner failed to find trajectories that satisfied workspace constraints. The average absolute displacement from the target pose was $4.00\text{mm} \pm 3.02\text{mm}$. The average absolute angular displacement from the target pose was $4.35 \text{ degrees} \pm 3.14 \text{ degrees}$. Note the hole radius introduces a systematic error of 3mm to the final pose because the puller contacts the edge of the hole, not the center. Overall, our experimental results support the claim that the planner finds convergent pulling trajectories. An example trial is visualized in Figure 6.

VII. CONCLUSION

In this paper, we derive a method for computing exact bounds on the object's motion for classes of pressure distributions where the center of pressure is known but the distribution of support forces is unknown. We also show these exact motion bounds can be used to plan robotic pulling trajectories that guarantee the pulled object converges to the final pose. We validate our planner on a real robotic system and show that the generated trajectories obtain low errors on the final pose of the object.

ACKNOWLEDGMENTS

Eric Huang was supported by the Department of Defense (DoD) through the National Defense Science & Engineering Graduate Fellowship (NDSEG) Program. This material is based upon work supported by the National Science Foundation under Grants No. IIS1409003 and IIS1637908. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

- [1] JC Alexander and JH Maddocks. Bounds on the friction-dominated motion of a pushed object. *The International Journal of Robotics Research*, 12(3):231–248, 1993.
- [2] R-P Berretty, Ken Goldberg, Mark H Overmars, and A Frank Van Der Stappen. Orienting parts by inside-out pulling. In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 1053–1058. IEEE, 2001.
- [3] Lester E Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of mathematics*, 79(3):497–516, 1957.
- [4] Suresh Goyal, Andy Ruina, and Jim Papadopoulos. Planar sliding with dry friction part 1. limit surface and moment function. *Wear*, 143(2):307–330, 1991.
- [5] Inc. Gurobi Optimization. Gurobi optimizer reference manual, 2016. URL <http://www.gurobi.com>.
- [6] J-B Hiriart-Urruty. Images of connected sets by semicontinuous multifunctions. *Journal of Mathematical Analysis and Applications*, 111(2):407–422, 1985.
- [7] Robert D Howe and Mark R Cutkosky. Practical force-motion models for sliding manipulation. *The International Journal of Robotics Research*, 15(6):557–572, 1996.
- [8] Kevin M Lynch and Matthew T Mason. Pulling by pushing, slip with infinite friction, and perfectly rough surfaces. *The International journal of robotics research*, 14(2):174–183, 1995.
- [9] Kevin M. Lynch and Matthew T. Mason. Stable pushing: Mechanics, controllability, and planning. *The International Journal of Robotics Research*, 15(6):533–556, 1996. doi: 10.1177/027836499601500602.
- [10] M. T. Mason. On the scope of quasi-static pushing. *ISRR*, 1986.
- [11] Matthew T Mason. *Manipulator Grasping and Pushing Operations*. PhD thesis, Cambridge, MA, USA, 1982.
- [12] Matthew T. Mason. *Mechanics of Robotic Manipulation*. MIT Press, Cambridge, MA, USA, 2001. ISBN 0-262-13396-2.
- [13] Michael A Peshkin and Arthur C Sanderson. The motion of a pushed, sliding workpiece. *IEEE Journal on Robotics and Automation*, 4(6):569–598, 1988.
- [14] R Tyrrell Rockafellar and Roger J-B Wets. *Variational Analysis*, volume 317. Springer Science & Business Media, 2009.
- [15] Yuval Tassa, Nicolas Mansard, and Emo Todorov. Control-limited differential dynamic programming. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 1168–1175. IEEE, 2014.
- [16] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005. ISBN 0262201623.
- [17] Kuan-Ting Yu, Maria Bauzá, Nima Fazeli, and Alberto

Rodriguez. More than a million ways to be pushed: A high-fidelity experimental data set of planar pushing. *CoRR*, abs/1604.04038, 2016. URL <http://arxiv.org/abs/1604.04038>.

- [18] Jiaji Zhou and Matthew T Mason. Pushing revisited: Differential flatness, trajectory planning and stabilization. *Manuscript in submission*, 2017.
- [19] Jiaji Zhou, Robert Paolini, J Andrew Bagnell, and Matthew T Mason. A convex polynomial force-motion model for planar sliding: Identification and application. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 372–377. IEEE, 2016.