Harnessing Task Mechanics for Robotic Pushing and Grasping

Jiaji Zhou

CMU-RI-TR-18-16
April 2018

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Matthew T. Mason, Co-chair, CMU RI
J. Andrew Bagnell, Co-chair, CMU RI
Christopher G. Atkeson, CMU RI
Russ Tedrake, MIT CSAIL

*Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy.*

Abstract

A high-fidelity and tractable mechanics model of physical interaction is essential for autonomous robotic manipulation in complex and uncertain environments. This thesis studies several aspects of harnessing task mechanics for robotic pushing and grasping operations: mechanics model learning, pose and model uncertainty reduction, and planning and control synthesis in the minimal coordinate space.

We begin with a study of the mechanics of planar sliding as captured by a proposed efficiently identifiable convex polynomial force-motion model. Given a position-controlled manipulator action, we derive a kinematic contact model that resolves the contact modes and instantaneous object motion. This enables a generic quasi-static planar contact simulation validated with extensive robotic grasping and pushing experiments.

Based on the contact model, we prove that quasi-static pushing with a sticking contact is differentially flat using both graphical and algebraic derivations. The pusher-slider system is reducible to the Dubins car problem where the sticking contact constraints translate to bounded curvatures. For trajectory stabilization in the presence of model uncertainty, we design closed-loop control using dynamic feedback linearization or open-loop control using two contact points as a form of mechanical feedback. To reduce initial pose perception uncertainty, we propose a probabilistic algorithm that generates sequential uncertainty reduction actions based on finger encoder feedback until the object pose is uniquely known (subject to symmetry).

Going beyond position-controlled planar manipulation, we demonstrate grasping a large-size book where the robot needs to perform a sequence of non-prehensile contact-rich actions to finish the task while obeying kinematic and hardware limitations. Such actions include reorienting the book by force-controlled twisting and pulling, and push-grasping in the gravity plane treating the table as a third supporting finger.

Acknowledgments

I have been fortunate to spend my graduate school years in the Robotics Institute, where I engage in robotics research and improve problem solving skills. Like all the robograds, I feel privileged to be surrounded by many gifted talents in such a collaborative environment.

I am very thankful to have Drew Bagnell and Matt Mason as my advisors and friends. Drew taught me how to break down hard research problems into tractable pieces and provided many insightful thoughts and discussions during the early stage of my graduate school. Matt introduced me to the fascinating field of Robotic Manipulation, taught me mechanics and graphical methods, and provided endless supply of Franziskaner Hefe-Weisse beer and Hershey dark chocolate. Both gave me the full freedom to explore and grow as an independent researcher.

I would also like to thank my external thesis committee members: Chris Atkeson and Russ Tedrake. Their extensive experiences and unique perspectives help shape my research. It's a pleasant experience to collaborate with and learn from lab mates in MLab and Lairlab. Thank you to everyone in Mlab: Robbie, Ankit, Zhenzhong, Yifan, Eric, Alex, Gilwoo, Reuben, Siyi, Pragna, Erol, Zhiwei, and Jean for making sure the lab operates smoothly. Thank you to everyone in Lairlab: Paul, Stephane, Daniel, Kris, Katherina, Alex, Kevin, Dey, Shervin, Arun, Allie, Wen, Echo, Nick and Shaurya. To my other collaborators including Siyuan Feng, Michael Posa, Aaron Johnson, Yisong Yue, Alberto Rodriguez and Martial Hebert: thank you for providing helpful discussions and insights at various stages of my graduate school.

Finally I thank my parents, Yuxing Zhou and Yuling Chen, who have always been there to give me unconditional support and love, and to whom I am forever indebted.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Manipulation Mason (2018) is the process of changing the physical state of the environment such as picking up a pencil from the table and twirling it in hand. The human hand and arm system is a truly marvelous piece of art. Animals are also impressive at manipulation: apes utilize tools from nature and birds with minimal degree of freedom beaks build sophisticated nests. Industrial robots performing tasks such as pick-and-place and welding are making a major impact in manufacturing where the environment and assembly process are carefully controlled with minimized uncertainty. Yet factory robots are merely programmable machines that execute fixed sequences of actions. When leaving the factory floor, robots are far from being capable of interacting with the unstructured world confidently and gracefully. A common manipulation pipeline often consists of two stages. The first stage is to plan a collision free path to a pre-action pose followed by an open-loop action. The second stage is to adjust the in-hand pose with pick-and-place actions or fingertip motions. Contacts with the environment are minimized during the process. In sharp contrast, humans actively embrace environment contacts using compliant motions under rich feedback to guide manipulation, reduce uncertainty, and achieve dexterity. Mastering physical interaction with frictional contact is essential to autonomous robotic manipulation in a complex, cluttered and dynamically changing environment.

Planning and control without explicit reasoning about uncertainty and the task mechanics can lead to undesirable results. For example, grasp planning (Miller et al. (2003); Ferrari and Canny (1992)) is often prone to failure: the object moves while the fingers close and ends up in a final relative pose that differs from planned. Consider the process of closing a parallel jaw gripper shown in Fig. 1.1, the object will slide when the first finger engages contact and pushes the object before the other one touches the object. If the object does not end up slipping out, it can be jammed at an undesired position or grasped at an unexpected position. A high fidelity and easily identifiable model with minimum adjustable parameters capturing all these possible outcomes would enable synthesis of robust manipulation strategy.

This thesis explores physics-based mechanics models for robotic grasping and pushing. To summarize, we first address the following aspects in the context of planar manipulation:

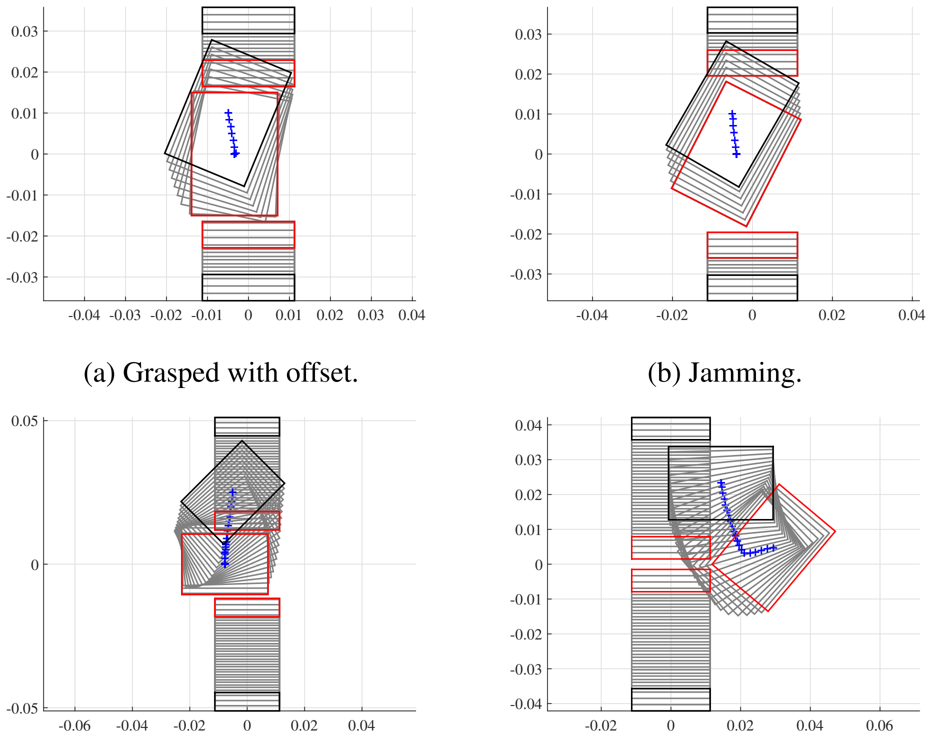
1. Identification of computationally tractable models that obey physics principles.
2. High-fidelity simulation of sliding manipulation including pushing and grasping.
3. Use of reduction techniques to plan and control in minimal coordinate spaces.
4. Use of simple feedback for uncertainty reduction with sequential strategies.

We then study a case of book grasping that requires a sequence of contact-rich actions where we employ compliance, force-control and external contacts to demonstrate the necesssity and benefit of harnessing task mechanics. We conclude the thesis with some outlooks on manipulation research with an emphasis on addressing the perception problem.

1.1 Exploitation of Task Mechanics

The general importance of task mechanics for robotic manipulation is well described in Lynch (1996):

A good model of the mechanics of a task is a resource for the robot system, just as actuators, sensors, and computers are resources. This model can be embedded in the structure of the mechanism, encoded in a feedback scheme, or used explicitly in a manipulation planner. Dafle



(a) Grasped with offset.

(b) Jamming.

(c) Grasped with offset.

(d) Slipped to free space.

Figure 1.1: Simulation results using the proposed contact model illustrating the process of a parallel jaw gripper squeezing along the y axis when the object is placed at different initial poses. The initial, final and intermediate gripper configurations and object poses are in black, red and grey respectively. Blue plus signs trace out the center of mass trajectory of the object.

et al. (2014) explores extrinsic dexterity for regrasp. Gravity, inertia, arm motions and external contacts can be exploited to manipulate an object in the hand. The idea is validated with hand scripted motion execution using a simple hand Mason et al. (2012) with only one degree of actuation.

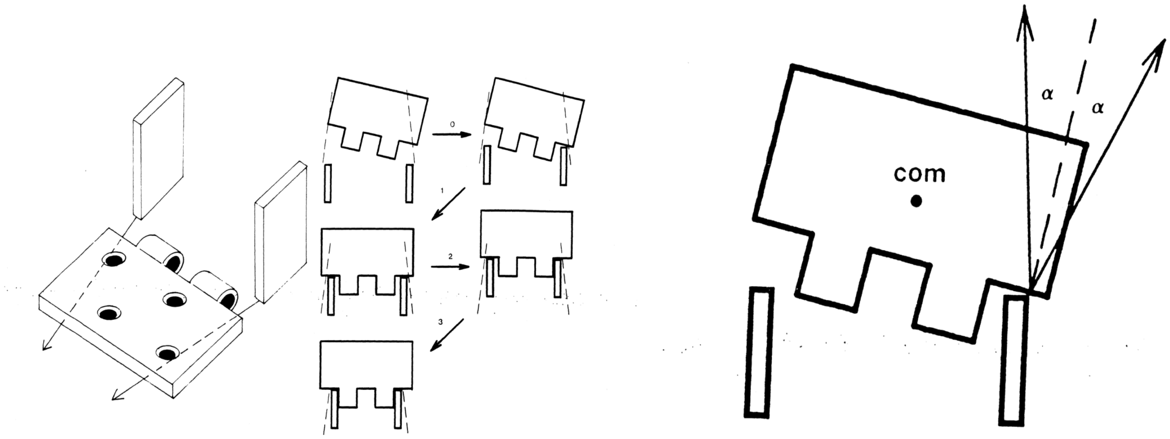
We consider four major perspectives of harnessing task mechanics. They are not mutually exclusive and many of the listed representative works fall under multiple categories.

1.1.1 Compliance

Compliant motion Mason et al. (1982) is key to contact-rich manipulation including turning a crank, inserting a peg, reaching into clutter to grasp objects, etc. In the context of contact-rich assembly, Whitney (1983) analyzed the mechanics of wedging and jamming during peg-in-hole insertion and designed the Remote Center Compliance device to increases the success of the operation under motion uncertainty. With a well defined generalized damper model, Lozano-Perez et al. (1984) and Erdmann (1986) developed strategies to chain a sequence of operations, each with a certain funnel, to guarantee operation success despite uncertainty. These successes stem from robustness analysis using simple physics models. In the context of grasping, some recent notable work on underactuated soft hand Dollar and Howe (2010); Odhner et al. (2014) adapt to object shapes through passive compliance built into the mechanism.

1.1.2 Pushing

As early as in the 70s, Shakey the robot Nilsson (1984) and the programmable assembly robot Bolles and Paul (1973) developed in Stanford AI Lab have employed pushing to help achieve their respective tasks: mobile manipulation and uncertainty reduction in sensor-based assembly. The mechanics of pushing involving finite object motion with frictional support was first studied in Mason (1986b). A notable result is the voting theorem which dictates the sense of rotation given a push action and the center of pressure regardless of the uncertain pressure distribution. Figure 1.2 shows a hinge push-grasp example where the straightline two-finger push help reduce orientation uncertainty. Brost (1988) took such idea one step further and construct the operational space for planning squeezing and push-grasping actions under uncertainty assuming infinitely long fingers approaching the object from infinitely far away. Peshkin and Sanderson (1988b) provided an analysis on the bound of rotation speed given a single point push and used the result to design fences for parts feeding Peshkin and Sanderson (1988a). Lynch and Mason (1996) derived conditions for stable edge pushing such that the object will remain attached to the pusher without slipping or breaking contact.



(a) The process of push-grasping a hinge.

(b) The voting theorem Mason (1986b) dictates that the sense of rotation is counterclockwise.

Figure 1.2: Figure reprinted from Mason (1986b) with permission. Application of voting theorem for push-grasping a hinge. The fingers perform a straightline motion while closing. The pose uncertainty with respect to the fingers are elinimated without any sensory feedback.

1.1.3 External Contact

Often, external contacts including tables, walls and fixtures can act as extra "fingers" to help task execution. Kazemi et al. (2013) increased robustness for grasping small and thin objects through compliant sliding against the supporting surface while closing the fingers. Deimel et al. (2016) analyzed human grasp behavior and demonstrated robust robotic grasping based on constraint-exploiting grasp strategies. Chavan-Dafle and Rodriguez (2015) studied prehensile pushing when the robot pushes a grasped object against environment contacts. The object is held by a simple parallel jaw gripper where the arm motion together with fixed external contact causes different in-hand motion including translational sliding, pivoting and rolling. Chapter 3 provides a case study.

1.1.4 Dynamics

Inertia, gravitational, centrifugal and Coriolis forces enable minimalistic robots performing complicated taskes. Lynch and Mason (1999) demonstrated that a one joint robot capable of controlling a planar object to full dimensional subset of six-dimensional state space by chaining action phases including dynamic grasp, slipping, rolling and free flight. Juggling, or dynamic pick-and-place by throwing and catching, has been studied by Buehler et al. (1994); Schaal and Atkeson (1993). Furukawa et al. (2006) developed a fast multifingered hand system with high-speed vision feedback to catch a falling object.

Dynamics is well exploited in locomoting machines. Raibert (1986) developed dynamically stable running and hopping machines. McGeer et al. (1990) built a passive walker with no motors and sensors by just cleverly harness the stable intrinsic dynamic behavior.

1.2 Related Work

Section 1.1 covers some related work that fall into the aforementioned categories. In this section, we describe additional background and related work for the several aspects of harnessing task mechanics for robotic manipulation: frictional contact models, funnels for uncertainty reduction and control, minimal coordinate space planning and control. We also provide references for cluttered manipulation and learning-based manipulation.

1.2.1 Frictional Contact Modeling

Isotropic Coulomb's Law states that tangential point friction force is opposite to the sliding velocity, with magnitude proportional to the normal force. The scaling factor is the coefficient of friction. When the point is static, the tangential point friction force is bounded but the direction is indeterminate without information about external contact forces. A geometrical characterization of the Isotropic Coulomb's law is the friction cone (both in two-dimensional and three-dimensional). Erdmann (1994) proposed a configuration space embedding of friction cone by including the torque component. For multiple contact points, the composite friction cone is the convex hull of each cone. Moreau (1988) formulated a generalized friction law that velocity and tangential friction force may not be parallel, but only need to obey maximum work inequality. Hence the set of friction forces is a general convex set not restricted to a circle and the velocities are normals to the set. In most practical applications, the contact geometry is surface-to-surface rather than point-to-point.

For surface-to-surface contacts, the normal force distributions are often indeterminate, rendering friction chacterization hard. Given pairs of pointer pusher velocities and the resultant object motion, Yoshikawa and Kurisu (1991) solved an unconstrained least-squares problem to estimate the center of friction and the pressure distribution over discrete grids on the contact surface. With similar set up, Lynch (1993) proposed a constrained linear programming procedure to avoid negative pressure assignment. However, methods based on discretization of the support surface introduce two sources of error in both localization of support points and pressure assignment among those points. Coarse discretization loses accuracy while fine discretization suffers from the curse of dimensionality. Additionally, the instantaneous center of rotation of the object can coincide with one of the support points, rendering the kinematic solution computationally hard due to combinatorial sliding/sticking mode assignment for each support point. In our work, we do not explicitly estimate the support point locations but directly estimate the aggregate effects. The representation is smooth and computationally efficient.

Goyal et al. (1991) noted that all the possible static and sliding frictional wrenches, regardless of the pressure distribution, form a convex set whose boundary is called a limit surface. Analytic construction of limit surface from Minkowsky sum of frictional limit curves at individual support points, however, is intractable. Howe and Cutkosky (1996) presented an ellipsoid approximation of the limit surface assuming known pressure distribution. The ellipsoid is constructed by computing or measuring the major axis lengths (maximum force during pure translation and maximum torque during pure rotation). Facets can be added by intersecting the ellipsoid with planes determined by each support point. The pressure distribution (except for computable 3 points support with known center of pressure), nevertheless, is non-trivial to measure. We also show that the ellipsoid approximation, as the convex quadratic special case of our convex polynomial representation, is less accurate due to lack of expressiveness particularly when the support regions are scattered.

The linear complementarity formulation of point frictional contact Stewart and Trinkle (1996); Anitescu and Potra (1997) provides a convenient representation and computation framework that captures unilateral normal force and sliding/sticking Coulomb's friction law. Posa et al. (2014) applies direct methods for trajectory optimization through frictional contacts. The problem is posed as mathematical programming with complementarity constraints solved using sequential quadratic programming with iteratively relaxed complementarity constraints.Todorov (2011) relaxed the complementarity constraint to allow contact forces at nonzero distance such that the inverse dynamics problem is convex. Mordatch et al. (2012); Tassa et al. (2012) used this smooth model for trajectory optimization to synthesis simulated complex humanoid motion behaviors.

1.2.2 Funnels

Our work is related to the general framework of planning manipulation actions with funnels, i.e., the set of poses associated with a robot action that are guaranteed to reach a particular goal set. Sensorless uncertainty reduction techniques have proven to be successful in many applications, often using a possibilistic approach and assuming worst case motion error. Erdmann and Mason (1988) demonstrated a parts feeding strategy using tray-tilting where mechanical motion alone can eliminate uncertainty. Lozano-Perez et al. (1984) (Preimage backchaining) and Erdmann (1986) (Backprojection) developed strategies to chain a sequence of operations to guarantee operation success despite uncertainty. With sensors, sequential composition of active feedback controllers with overlapping funnels can achieve the desired result, as studied in Burridge et al. (1999); Tedrake et al. (2010).

1.2.3 Planning and Control in Reduced Space

Much of the work in Section 2.3 is inspired by differential flatness techniques from nonlinear control theory Murray et al. (1995). Such techniques are powerful for trajectory generation and control for underactuated mechanical systems. Oriolo et al. (2002) designed dynamic feedback linearization for wheeled mobile robots. A major success of applying the differential flatness techniques is for quadrotor trajectory generation and control Mellinger and Kumar (2011).

1.2.4 Manipulation in Clutter

Suppose the robot needs to reach into a cluttered and constrained environment to grasp an object of interest, a traditional planner would treat all objects as obstacles and try to find a collision free path and grasping points. Suppose such grasps exist, the robot will then close the hand assuming the fingers will arrive at the planned grasping points at roughly the same time. Such approach is prone to failures due to inevitable uncertainty and unmodeled interactions. Dogar and Srinivasa (2011, 2010) studied planning push-grasps actions for rearrangment and grasping for a cluttered table top setting. Each action addresses a single object-to-robot interaction while avoiding contact with others. In Dogar et al. (2012), the planner is extended to a clutter-centric perspective where the robot can make simultaneous contacts with many objects while the object-to-object interaction is ignored.

1.2.5 Learning-based Manipulation

Many tasks can be difficult or computationally intractable for strategies that rely purely on physics-based modeling and control. Learning-based methods, by imitation learning of given demonstrations or reinforcement learning, often prove to be powerful in mastering task-specific skills. Kober et al. (2013) provides a recent survey on this topic.

Arimoto et al. (1984) explored the simple and effective iterative learning control ideas. Chris Atkeson and his colleges (Atkeson et al. (1997); Schaal and Atkeson (2010)) have pioneered locally weighted learning and trajectory-centric methods for learning to control robots.

Learning to push objects are explored in Hermans et al. (2013); Walker and Salisbury (2008); Kopicki et al. (2009); Clavera et al. (2017). Boularias et al. (2014, 2011) address learning to grasp based on geometrical feature similarities. Boularias et al. (2015) learns pushing and grasping action reward and optimize policies for cluttered unknown objects.

Recent advances in deep learning demonstrate impressive results in computer vision Krizhevsky et al. (2012) where data is abundant. Pinto and Gupta (2016) explores using robots to collect grasping data and trains a deep learning network to predict table-top grasping quality. Levine et al. (2016) learns a visual servoing strategy from raw pixel input for bin picking applications. In Pinto and Gupta (2016) and Levine et al. (2016), the robot supervises itself from experimental data. Gualtieri et al. (2016); Mahler et al. (2017) apply deep learning to sythesis anti-podal grasps from simulated depth point cloud. Byravan and Fox (2017); Agrawal et al. (2016) train deep nets to predict object behavior under pushing actions.

Mitchell et al. (1989) explores the ability to explain failures and to deduce error recovery, which is under-addressed in recent robot learning literature.

1.3 Thesis Outline

We start with analyzing the task mechanics for manipulating a single object on the horizontal plane.

1. Section 2.1 proposes a convex polynomial force-motion model for the mechanics of planar sliding Zhou et al. (2016). [1]

2. Section 2.2 derives the kinematic contact model to resolve the contact modes and instantaneous object motion given a position controlled manipulator action Zhou et al. (2017a). [2]

   Section 2.3 shows the results on the differential flatness property of the pusher-slider sys-

[1] Robot experiments video: https://www.youtube.com/watch?v=BW6QaPOlpCk

[2] Robot experiments video: https://www.youtube.com/watch?v=LDSP1WOAri0

tem and its application in motion planning and tracking control Zhou and Mason (2017). [3]

3. Section 2.4 proposes a probabilistic algorithm that generates both sensored and sensorless tree structured plans such that the post-action object pose is uniquely known (subject to symmetry) Zhou et al. (2017b). [4]

4. Chapter 3 describes a case study of a book grasping application where the gripper cannot directly do a top down grasp due to limited finger width and needs to perform a series of non-prehensile contact-rich strategy: reorienting the book by pressing and twisting, force controlled pulling, straightline pushing for uncertainty reduction and push-grasping in the gravity plane treating the table as a third supporting finger while obeying the workspace constraints. [5]

5. Chapter 4 concludes the thesis with the author's thoughts on improving robotic manipulation via building better physics models with data, deep learning powered 2d to 3d perception trained from synthetic images and compliant manipulator hardware with visual guidance.

[3] Robot experiments video: https://www.youtube.com/watch?v=fH8GEQGBqQY
[4] Robot experiments video: https://www.youtube.com/watch?v=G8q8HmD36xw
[5] Robot experiments video: https://www.youtube.com/watch?v=k_mpCreIhDE

Chapter 2

Single Object Quasi-static Planar Manipulation

This chapter studies sliding manipulation of a single object on a plane. We assume quasi-static rigid body planar mechanics (Mason (1986a)) where inertia forces and out-of-plane moments are negligible. The challenge lies in the indeterminate and stochastic nature of friction distribution between the object and the supporting plane. Source code for the algorithms and simulation packages developed in the chapter can be found at `https://github.com/robinzhoucmu/PlanarManipulationToolBox`.

2.1 Convex Polynomial Force-Motion Model

This section proposes a precise and statistically-efficient force-motion model with a computationally efficient identification procedure. We propose a framework of representing planar sliding force-motion models using homogeneous even-degree sos-convex polynomials, which can be identified by solving a semi-definite programming. The set of applied friction wrenches is the 1-sublevel set of a convex polynomial whose gradient directions correspond to incurred sliding body twist.

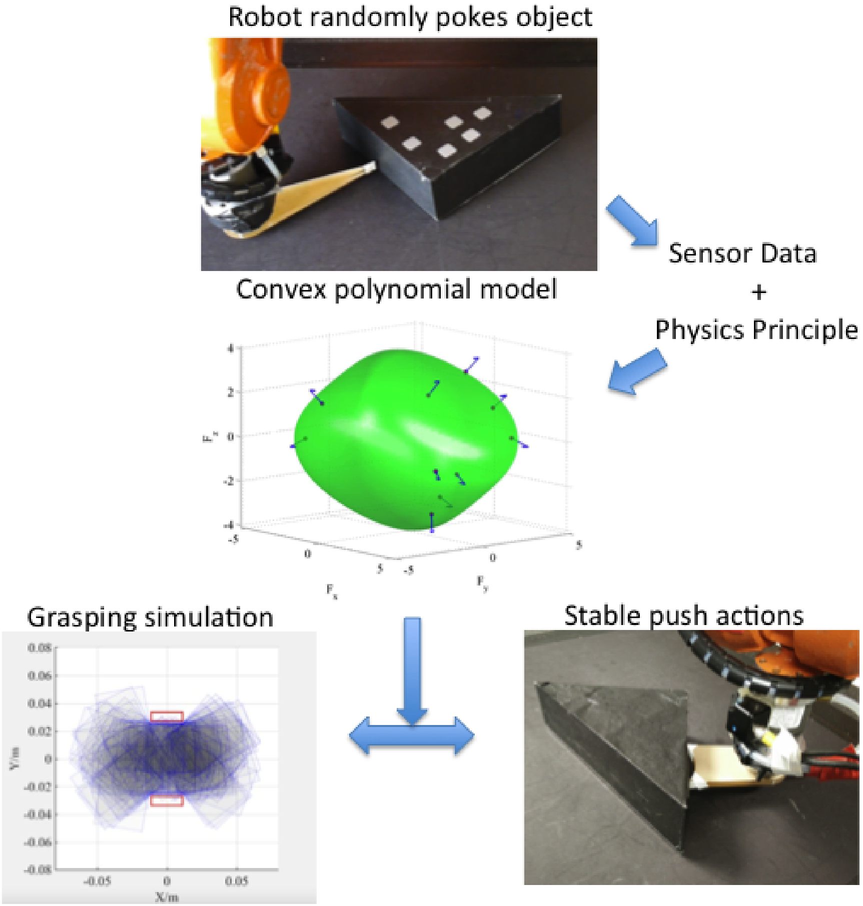


Figure 2.1: The robot randomly pokes the object of known shape with a point finger to collect force-motion data. We then optimize a convex polynomial friction representation with physics-based constraints. Stable pushing and grasping simulations under pose uncertainty are two example applications of the model.

Fig. 2.1 illustrates the outline of this section.

We first introduce the following notations:

- O: the object center of mass used as the origin of the body frame[1]. We assume vector quantities are with respect to body frame unless specially noted.
- R: the region between the object and the supporting surface.
- $\mathbf{f_r}$: the surface friction force applied by the object at a point $\mathbf{r}$ in R.
- $\mathbf{V} = [V_x; V_y; \omega]$: the body twist (generalized velocity).

[1] Throughout the paper, we use a local coordinate frame with the origin set as the projection of the COM onto the supporting surface. However, the choice of the origin can be any other point of convenience.

- $\mathbf{F} = [F_x; F_y; \tau]$: the generalized friction load. $\mathbf{F}$ equals the applied body wrench by the manipulator when the object is in quasi-static balance.
- $\mathbf{p}_i$: each contact point between the manipulator end effector and object in the body frame.
- $\mathbf{v}_{p_i}$: applied velocities by the manipulator end effector at each contact point in the body frame.
- $\mathbf{n}_{p_i}$: the inward normal at contact point $\mathbf{p}_i$ on the object.
- μ_c: coefficient of friction between the object and the manipulator end effector.

2.1.1 Background on Planar Friction

The classical Coulomb friction law states that for a point contact with instantaneous planar velocity $\mathbf{v} = [v_x, v_y]^T$, the incurred friction force $\mathbf{f} = [f_x, f_y]^T$ the point applies on the surface is parallel to $\mathbf{v}$, i.e., $\mathbf{f}/|\mathbf{f}| = \mathbf{v}/|\mathbf{v}|$. We refer the readers to Mason (1986b) for details of friction analysis for planar sliding under isotropic Coulomb friction law. In this paper, we build our analysis on a generalized friction law formulated first in Moreau (1988), in which $\mathbf{v}$ and $\mathbf{f}$ may not be parallel, but only need to obey the maximum work inequality:

$$(\mathbf{f} - \mathbf{f}') \cdot \mathbf{v} \geq 0, \tag{2.1}$$

where $\mathbf{f}'$ is an arbitrary element from the set of all possible static and sliding friction forces.

We can compute the generalized friction load $\mathbf{F}$ by integration over R:

$$F_x = \int_R f_{rx}\, dr, \quad F_y = \int_R f_{ry}\, dr, \quad \tau = \int_R (r_x f_{ry} - r_y f_{rx})\, dr. \tag{2.2}$$

The maximum work inequality in equation (2.1) can be extended for the generalized friction load

F and twist **V**:

$$\begin{aligned}\mathbf{F}\cdot\mathbf{V} &= \int_R f_{rx}(V_x - \omega r_y)\,dr + \int_R f_{ry}(V_y + \omega r_x)\,dr \\ &= \int_R f_{rx}v_{rx} + f_{ry}v_{ry}\,dr = \int_R \mathbf{f_r}\cdot\mathbf{v_r}\,dr \geq \mathbf{F}'\cdot\mathbf{V},\end{aligned} \tag{2.3}$$

among any other possible generalized friction load $\mathbf{F}'$. Due to the converse supporting hyperplane theorem (Boyd and Vandenberghe (2004)), the set of all generalized friction loads form a convex set $\mathscr{F}$. An important work that inspires us is Goyal et al. (1991) who found that all possible generalized friction loads during sliding form a limit surface (LS) constructed from the Minkowsky sum of limit curves at individual support points. Points inside the surface correspond to static friction loads. Points on the surface correspond to friction loads with normals parallel to sliding velocity directions, forming a mapping between generalized friction load and sliding velocity. An ideal LS is always convex due to the maximum work inequality but may not be strictly convex when a single point supports finite pressure. As shown in Fig. 2.3b, facets can occur since the object can rotate about one of the three support points whose velocity is zero with indeterminate underlying friction.

Erdmann (1994) proposed a configuration space embedding of friction. In his work, the third component of **F** is $F_z = \tau/\rho$ and the third component of **V** is $V_z = \omega\rho$, where ρ is the radius of gyration. In doing so, all three components in **F** and **V** have the same unit. Observe that such normalized representation also obeys maximum work inequality with ρ being any characteristic length. In our experiments, we have found that the normalized representation yields better numerical condition and different values of ρ including radius of gyration, average edge length and minimum enclosing circle radius lead to similar performance.

2.1.2 Representation and Identification

In this section, we propose the sublevel set representation of friction with desired properties and show that convex even-degree homogeneous polynomials are valid solutions. Then we formulate an efficient convex optimization procedure to identify such polynomials.

2.1.2.1 Polynomial sublevel set representation

Let $H(\mathbf{F})$ be a differentiable convex function that models the generalized friction load and velocity as follows:

- The 1-sublevel set $L_1^-(H) = \{\mathbf{F} : H(\mathbf{F}) \le 1\}$ corresponds to the convex set $\mathscr{F}$ of all generalized friction loads.
- The 1-level set $L_1(H) = \{\mathbf{F} : H(\mathbf{F}) = 1\}$ corresponds to generalized friction loads (during slip) on the boundary surface of $\mathscr{F}$.
- The surface normals given by gradients $\{\nabla H(\mathbf{F}) : \mathbf{F} \in L_1(H)\}$ represent instantaneous generalized velocity directions during slip, i.e., $V = s\nabla H(\mathbf{F})$ where $s > 0$.

Theorem 1. *The set of friction loads represented by the* 1*-sublevel set of a differentiable convex function follows the maximum work inequality.*

Proof. When the object remains static, $\mathbf{F}$ belongs to the interior of $L_1^-(H)$ and $\mathbf{V}$ equals zero, the inequality holds as equality. When the object slips, $\mathbf{F} \in L_1(H)$ and $\mathbf{V}$ is nonzero, we have for any other generalized friction load $\mathbf{F}' \in L_1^-(H)$:

$$\mathbf{V} \cdot (\mathbf{F}' - \mathbf{F}) = s(\nabla H(\mathbf{F}) \cdot (\mathbf{F}' - \mathbf{F})) \le s(H(\mathbf{F}') - H(\mathbf{F})) \le 0,$$

where the first inequality is due to the convexity of $H(\mathbf{F})$. □

In addition to enforcing convexity (discussed in 2.1.2.2), we choose $H(\mathbf{F})$ to obey the following properties:

1. Symmetry: $H(\mathbf{F}) = H(-\mathbf{F})$ and $\nabla H(\mathbf{F}) = -\nabla H(-\mathbf{F})$.
2. Scale invariance: $\nabla H(s\mathbf{F}) = g(s)\nabla H(\mathbf{F})$, where $g(s)$ is a positive scalar function over scalar s.
3. Efficient invertibility: efficient numerical procedures exist for finding a $\mathbf{F} \in L_1(H)$ such that $\nabla H(\mathbf{F})/||\nabla H(\mathbf{F})|| = \mathbf{V}$ for a given query unit velocity $\mathbf{V}$. We denote such operation as $\mathbf{F} = H_{inv}(\mathbf{V})$.

Symmetry is based on the assumption that negating the velocity direction would only result in a sign change in the friction load. Scale invariance is desired for two reasons: 1) scaling in mass and surface coefficient of friction could only result in a change of scale but not other geometrical properties of the level-set representation; and 2) predicting directions of generalized velocities (by computing gradients and normalizing to a unit vector) only depends on the direction of generalized force. Such a property is useful in the context of pushing with robot fingers where applied loads are represented by friction cones. The inverse problem of finding the friction load for a given velocity naturally appears in seeking quasi-static balance for stable pushing or computing deceleration during free sliding. In general, efficient numerical solution to the inverse problem, which our representation enables, is key to planning and simulation. One solution family for $H(\mathbf{F})$ that obeys these properties is the set of **strictly convex even-degree homogeneous polynomials**.

Theorem 2. *A strictly convex even degree-d homogeneous polynomial* $H(\mathbf{F};a) = \sum_{i=1}^{m} a_i F_x^{i_1} F_y^{i_2} F_z^{d-i_1-i_2}$ *with m (bounded by* $\binom{d+2}{2}$*) monomial terms parametrized by a satisfies the properties of symmetry, scale invariance, and efficient invertibility.*

Proof. Proving symmetry and scale invariance are trivial due to the homogeneous and even-degree form of $H(\mathbf{F})$. Here, we sketch the proof that efficient invertibility can be achieved by first solving a simple non-linear least square problem followed by a rescaling.

Construct an objective function $G(\mathbf{F}) = \frac{1}{2}||\nabla H(\mathbf{F}) - \mathbf{V}||^2$ whose gradient $\frac{\partial G}{\partial \mathbf{F}} = \nabla^2 H(\mathbf{F})(\nabla H(\mathbf{F}) - \mathbf{V})$. Note that its stationary point F^*, which iterative methods such as Gauss-Newton or trustregion algorithms will converge to, satisfies $\nabla H(\mathbf{F}^*) - \mathbf{V} = 0$. Hence F^* is globally optimal with value zero. Let $\Delta\mathbf{F}_t = \nabla^2 H(\mathbf{F}_t)^{-1}(\mathbf{V}_t - \mathbf{V})$,[2] then the update rule for Gauss-Newton algorithm is $\mathbf{F}_{t+1} = \mathbf{F}_t - \Delta\mathbf{F}_t$. Although the final iteration point $\mathbf{F}_T$ may not lie on the 1-level set of $H(\mathbf{F})$, we can scale $\mathbf{F}_T$ by $\hat{\mathbf{F}}_T = H(\mathbf{F}_T)^{-1/d}\mathbf{F}_T$ such that $H(\hat{\mathbf{F}}_T) = 1$ and $\nabla H(\hat{\mathbf{F}}_t)/\|\nabla H(\hat{\mathbf{F}}_t)\| = V$ due to the homogeneous form of $H(\mathbf{F})$. Therefore $H_{inv}(\mathbf{V}) = \hat{\mathbf{F}}_T$. □

2.1.2.2 Sum-of-squares Convex Relaxation

Enforcing strong convexity for a degree-2 homogeneous polynomial $H(\mathbf{F};A) = \mathbf{F}^T A\mathbf{F}$ has a straightforward set up as solving a semi-definite programming problem with constraint of $A \succeq \varepsilon I$. Meanwhile, for a polynomial of degree greater than 2 whose hessian matrix $\nabla^2 H(\mathbf{F};a)$ is a function of both $\mathbf{F}$ and a, certification of positive semi-definiteness is NP-hard. However, recent progress (Parrilo (2000); Magnani et al. (2005)) in sum-of-squares programming has given powerful semi-definite relaxations of global positiveness certification of polynomials. Specifically, let $\mathbf{z}$ be an arbitrary non-zero vector in $\mathbb{R}^3$ and $y(\mathbf{F},\mathbf{z}) = [z_1F_x, z_1F_y, z_1F_z, z_2F_x, z_2F_y, z_2F_z, z_3F_x, z_3F_y, z_3F_z]^T$. If there exists a positive-definite matrix Q such that

$$\mathbf{z}^T \nabla^2 H(\mathbf{F};a)\mathbf{z} = y(\mathbf{F},\mathbf{z})^T Q y(\mathbf{F},\mathbf{z}) > 0, \tag{2.4}$$

then $\nabla^2 H(\mathbf{F};a)$ is positive definite for all non-zero $\mathbf{F}$ under parameter a and $H(\mathbf{F};a)$ is called as sos-convex. Further, equation (2.4) can be written as a set of K sparse linear constraints on Q and a.

$$\begin{aligned} \mathrm{Tr}(A_k Q) &= b_k^T a, \quad k \in \{1\ldots K\} \\ Q &\succeq \varepsilon I, \end{aligned} \tag{2.5}$$

[2] In practice, $\mathbf{F}_t$ will not be near the point of origin whose $H(\mathbf{F}_t)$ is all zero. A small diagonal regularization can be added to $H(\mathbf{F}_t)$ before inversion to improve numerical stability.

where A_k and b_k are constant sparse element indicator matrix and vector that only depend on the polynomial degree d. The number of constraints K equals 27 for $d = 4$.

2.1.2.3 Identification

This section sets up an efficient convex optimization for identifying the coefficient a of the polynomial $H(\mathbf{F};a)$ given a set of measured noisy generalized force-motion $\{\mathbf{F}_{i\in\{1\ldots N\}}, \mathbf{V}_{i\in\{1\ldots N\}}\}$ pairs. In our experiments, we use homogeneous 4th order polynomial. The optimization should find the coefficient a such that the measured forces $\mathbf{F}_i$ are close to the 1-level set surface and the corresponding gradients are aligned well (up to scale) w.r.t measured velocities $\mathbf{V}_i$. Let $\alpha_i = ||\nabla H(\mathbf{F}_i;a) - (\nabla H(\mathbf{F}_i;a)\cdot V_i)V_i||_2^2$ be the L2-projection residual of $\nabla H(\mathbf{F}_i;a)$ onto the measured unit velocity vector V_i, and let $\beta_i = (H(\mathbf{F}_i;a)-1)^2$ be a distance measurement of $\mathbf{F}_i$ from the 1-level set of $H(\mathbf{F}_i;a)$. We set up the optimization as follows:

$$\underset{a,Q}{\text{minimize}} \quad \|a\|_2^2 + \sum_{i=1}^{N}(\eta_1\alpha_i + \eta_2\beta_i) \tag{2.6}$$

$$\text{subject to} \quad \mathrm{Tr}(A_kQ) = b_k^T a,\ k = 1,\ldots,K, \tag{2.7}$$

$$Q \succeq \varepsilon I. \tag{2.8}$$

The first term is for parameter regularization. η_1 and η_2 are trade-off parameters determined by cross-validation. Equations (2.7) and (2.8) enforce convexity. Note that the objective is quadratic in a with sparse linear constraints and a semi-definite constraint on Q. We would like to point out that the formulation can be adapted online using projected gradient descent so that the importance of historical data is diminishing as the object moves, enabling the estimation to adapt to changing surface conditions.

2.1.3 Identification Experiments

We conduct simulation and robotic experiments to demonstrate the accuracy and statistical-efficiency of our proposed representation. The model converges to a good solution with few available data which saves experimental time and design efforts. We compare the following four different force-motion model representations $\mathscr{H}$: 1) degree-4 convex homogeneous polynomial (poly4-cvx); 2) degree-4 homogeneous polynomial (poly4) without convexity constraints 3) convex quadratic (quad) as degree-2 polynomial, i.e., $H(\mathbf{F}) = \mathbf{F}^T A \mathbf{F}$ with ellipsoid sublevel set; and 4) gaussian process (GP) with squared exponential kernel[3].

Denote by $\mathbf{V}_i$ the ground truth instantaneous generalized velocity direction and $\mathbf{V}_p(\mathbf{F}_i; \mathscr{H})$ as the predicted generalized velocity direction based on $\mathscr{H}$ for the input generalized load $\mathbf{F}_i$, we use the average angle $\delta(\mathscr{H}) = \frac{1}{N}\sum_{i=1}^{N} \arccos(\mathbf{V}_p(F_i; \mathscr{H}) \cdot \mathbf{V}_i)$ between $\mathbf{V}_p(F_i; \mathscr{H})$ and $\mathbf{V}_i$ as an evaluation criterion.

2.1.3.1 Simulation Study

Two kinds of pressure distribution are studied.

- “Legged” support: Randomly sampled three support points on a unit circle with randomly assigned pressure.
- “Uniform” support: Uniformly distributed 360 support points on a unit circle and 400 support points within a unit square. Each point has the same support pressure.

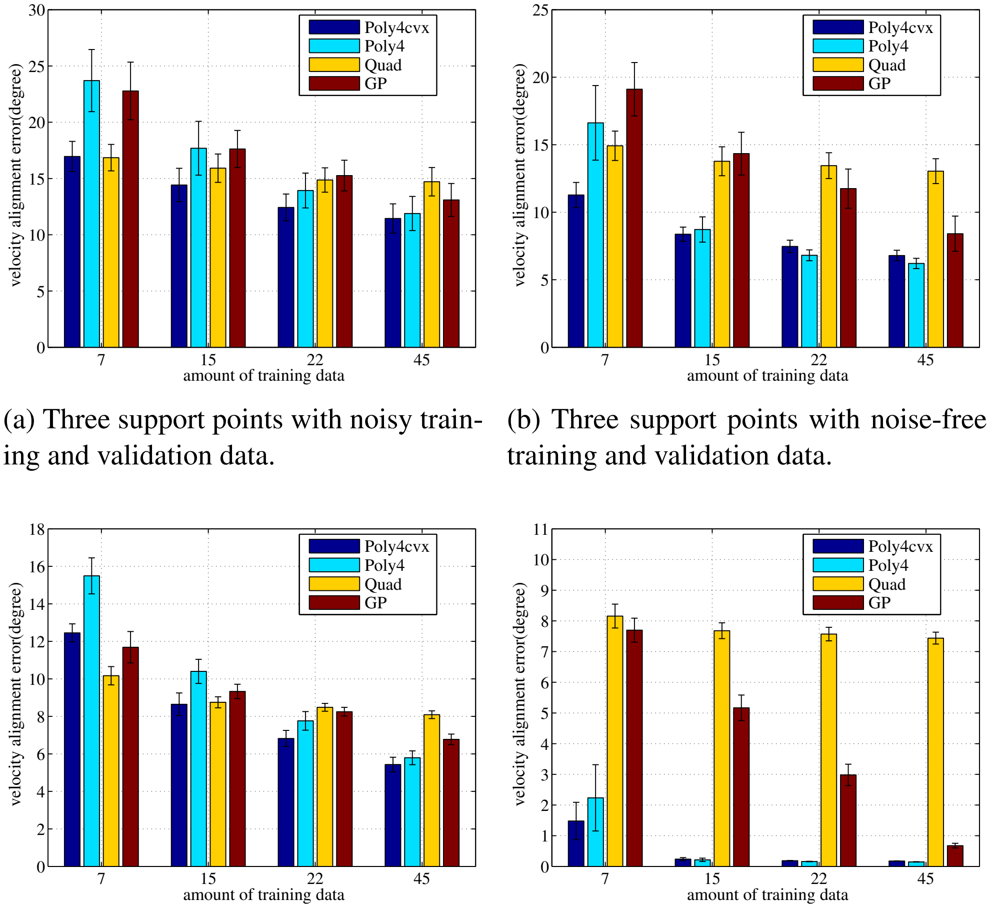
For each pressure configuration, we conduct 50 experimental trials. To generate the simulated force-motion data, we assume a Coulomb friction model at each support point with a uniform coefficient of friction. Without loss of generality, sum of pressure over all contact points is normalized to one and the origin is set as the center of pressure. For each trial of “uniform” support, we sampled 150 instantaneous generalized velocities directions $\mathbf{V}_i$ uniformly on the unit

[3] The squared exponential kernel gives better performance over linear and polynomial. Normalizing the input load to a unit vector improves performance by requiring the GP to ignore scale. Every $(\mathbf{F}, \mathbf{V})$ input pair is augmented with $(-\mathbf{F}, -\mathbf{V})$ for training.

sphere and compute the corresponding generalized friction loads $\mathbf{F}_i$. For each trial of "legged" support, 75 $(\mathbf{F}_i, \mathbf{V}_i)$ pairs are uniformly sampled on the facets (same $\mathbf{V}_i$ but different $\mathbf{F}_i$ for each facet) and another 75 pairs are uniformly sampled in the same fashion as "uniform" support. In doing so, the dataset has a diverse coverage. Among the 150 pairs, 50% is used for hold-out testing, 20% is used for cross validation and four different amounts (7, 15, 22, 45) from the rest of 30% are used as training. In order to evaluate the algorithms' robustness under noise, we additionally corrupt the training and validation set using Gaussian noise of standard deviation $\sigma = 0.1$ to each dimension of both $\mathbf{F}_i$ and $\mathbf{V}_i$ (renormalized to unit vector). From Fig. 2.2 we can reach the following conclusions. 1) Poly4-cvx has the smallest $\delta(\mathscr{H})$ for different amounts of training data and pressure configurations. 2) Both poly4-cvx and convex quadratic show superior performance when data is scarce and noisy, demonstrating convexity is key to data-efficiency and robustness. Poly4-cvx model additionally shows larger improvement as more data is available due to stronger model expressiveness. 3) Poly4 (without convexity constraint) performs the worst when only few data is available, but gradually improves as more data is available for shaping the surface. For noise-free experiments shown in Fig. 2.2b and 2.2d, when enough training data (more than 22) is presented, poly4 performs slightly better than poly4-convex. We conjecture such difference is due to the gap between sos-convex polynomials and convex polynomials (Ahmadi and Parrilo (2012)). GP has similar performance trends as poly4 but worse on average. 4) Polynomial models enjoy significant performance advantages when limit surface is smoother as in uniform point support (approximation of uniform patch contact). Such advantage is smaller for three-points support whose limit surface has large flat facets.

2.1.3.2 Robotic Experiment

We mount three screws at four different sets of locations underneath an alluminium right-angle triangular work object that weighs 1.508kg with edge lengths of 150mm, 150mm and 212.1mm. The four different set of support point locations (in mm) with respect to the right angle corner



(a) Three support points with noisy training and validation data.

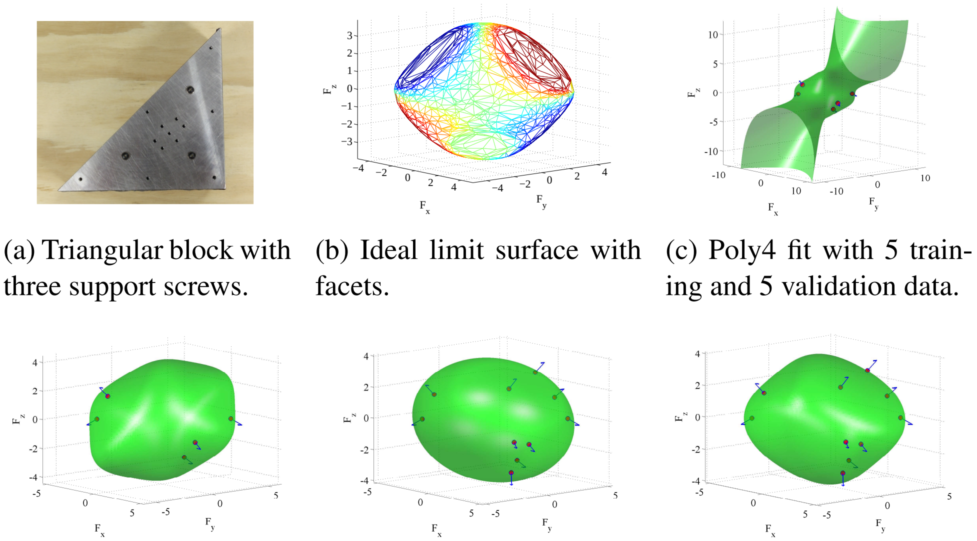
(b) Three support points with noise-free training and validation data.

(c) Uniform circular support points with noisy training and validation data.

(d) Uniform circular support points with noise-free training and validation data.

Figure 2.2: Test error comparison for simulation experiments with 95% confidence bar (50 random evaluations) among different methods as amount of training data increases for three random support points and 360 support points on a ring respectively. Results for uniform pressure distribution within a square are similar to uniform circular support and omitted for space.

vertex are: [(10,10), (10,130), (130,10)], [(30,30), (30,90), (90,30)], [(10,10), (10,130), (90,30)], [(30,30), (63.33,43.33), (43.33,63.33)]. Given known mass and COM projection, ideal ground truth pressure for each support point can be computed by solving three linear equations assuming each screw head approximates a point contact. Fig. 2.3a shows a flipped view of one arrangement whose ideal LS is illustrated in Fig. 2.3b, constructed by Minkowski addition of generalized friction at each single point support assuming Coulomb friction model with uniform coefficient of



(a) Triangular block with three support screws.

(b) Ideal limit surface with facets.

(c) Poly4 fit with 5 training and 5 validation data.

(d) Poly4-cvx fit with 5 training and 5 validation data.

(e) Convex quadratic fit with 10 training and 10 validation data.

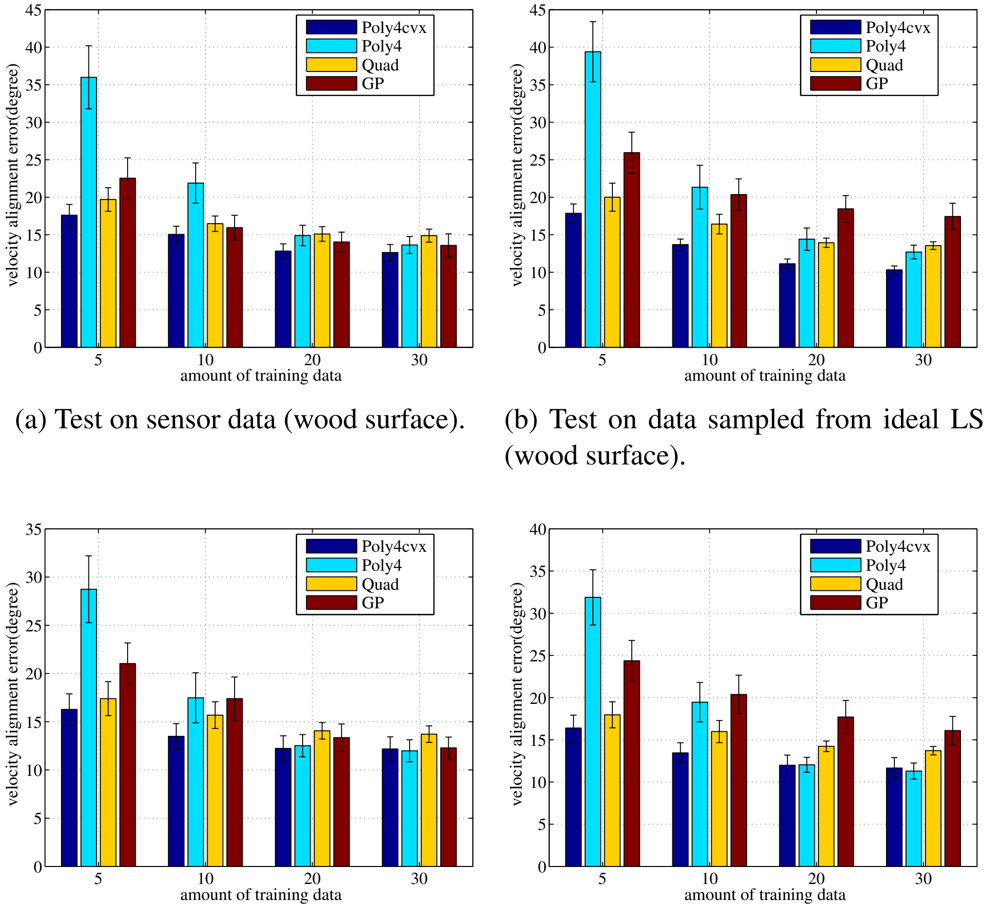
(f) Poly4-cvx fit with 10 training and 10 validation data.

Figure 2.3: Level set friction representations for the pressure arrangement in Fig. 2.3a. Red dots and blue arrows are collected generalized forces and velocities from force-torque and motion capture sensor respectively. Fig. 2.3c and 2.3d, Fig. 2.3e and 2.3f share the same data.

friction. Three pairs of symmetric facets[4] characterize indeterminate friction force when rotating about one of the three support points. Comparison among identified fourth-order homogeneous polynomials with and without convexity constraint is shown in Fig. 2.3c and 2.3d. We can see that convex-shape constraint is essential to avoid poor generalization error when little data is available. Fig. 2.3e and 2.3f compare the level sets of a convex quadratic (ellipsoid) and a sos-convex degree-4 homogeneous polynomial, demonstrating that the higher degree polynomial captures the facets effect better than quadratic models.

We conduct robotic poking (single point pushing) experiments on wood and paper board surfaces. In each experiment, we generate 50 pokes (30 for training set, 10 for validation set and 10 for test set) with randomly chosen contact points and pushing velocity directions. During each pushing action, the robot moves at a slow speed of 2.5mm/s with a total small push-in distance

[4] The third one is in the back not visible from presented view.



(a) Test on sensor data (wood surface).

(b) Test on data sampled from ideal LS (wood surface).

(c) Test on sensor data (paper board surface).

(d) Test on data sampled from ideal LS (paper board surface).

Figure 2.4: Test error comparison for robotic experiments with 95% confidence bar (50 random evaluations) among different methods as amount of training data increases for three support points (averaged over four different arrangements) on wood and hard paper board surfaces.

of 15mm. Each generalized velocity direction is approximated as the direction of pose displacement and generalized force is averaged over the action duration. Fig. 2.4 shows model accuracy (averaged over four different pressure arrangments) with respect to increase in amount of training data for different methods evaluated on both the hold-out test sensor data and samples from ideal LS. We can see similar performance trends as in simulation experiments. Note that both evaluations only serve as certain reference criteria. Sensor data is noisy and all possible force measurements from a single point pusher only cover a limited space of the set of friction loads.

We also do not intend to treat the idealized limit surface as absolute ground truth as there is no guarantee on uniform coefficient of friction between the support points and the underlying surface. Additionally, point contact and isotropic Coulomb friction model are only approximations of reality. Nevertheless, both evaluations demonstrate performance advantage of our proposed poly4-cvx model.

2.1.4 Stable Push Action Generation

Prediction of the resultant object twist under a single point push action cannot be exactly accurate. A two-points push action against an edge of the object, however, can be stable such that the object will remain attached to the pusher without slipping or breaking contact (Lynch and Mason (1996)). That is, the slider and pusher will move about the same center of rotation (COR) point p_c. Given the level set representation $H(\mathbf{F})$, the condition of determining whether a two-points push with instantaneous generalized velocity $\mathbf{V}_{p_c}$ is stable or not is equivalent to check if the corresponding generalized friction load $\mathbf{F}_{p_c} = H_{inv}(\mathbf{V}_{p_c})$ lies in the applied composite wrench cone $\mathbf{F}_c$. To validate predictions based on the model, we sampled 60 random CORs and execute with the robot for three different pressure arrangements on a novel support surface material (hard poster paper). We use the same triangular block in Fig. 2.3a with two three-points contacts [(10,10), (10,130), (130,10)], [(30,30), (30,90), (90,30)] as well as full patch contact. The 60 CORs are tight rotation centers within a 400mm×400mm square centered at the COM. 15 out of the 60 CORs are labelled as stable. The training force-motion data are collected from pushing the object on a wood surface. Table 2.1 and 2.2 summarize the classification accuracy and positive (stable) class recall measurements of three invertible methods with respect to increase in the amount of training data. Fig. 2.5 shows an example (full patch contact) that the stable regions generated from the identified poly4-cvx model is much larger than the conservative analysis as in Lynch and Mason (1996) which misses the tight/closer rotation centers.

Table 2.1: Comparison of average accuracy with 95% confidence interval as amount of training data increases.

	10	20	30
poly4-cvx	**88.13**±1.80	**91.33**±1.61	**93.07**±1.45
poly4	85.27±2.12	89.40±1.98	93.00±1.62
quadratic	87.93±1.72	87.20±1.65	88.00±1.39

Table 2.2: Comparison of average positive recall with 95% confidence interval as amount of training data increases.

	10	20	30
poly4-cvx	**90.13**±3.54	**96.69**±1.93	**98.18**±1.32
poly4	79.96±5.25	92.76±2.90	97.18±1.84
quadratic	73.18±4.61	73.38±4.69	73.87±4.63

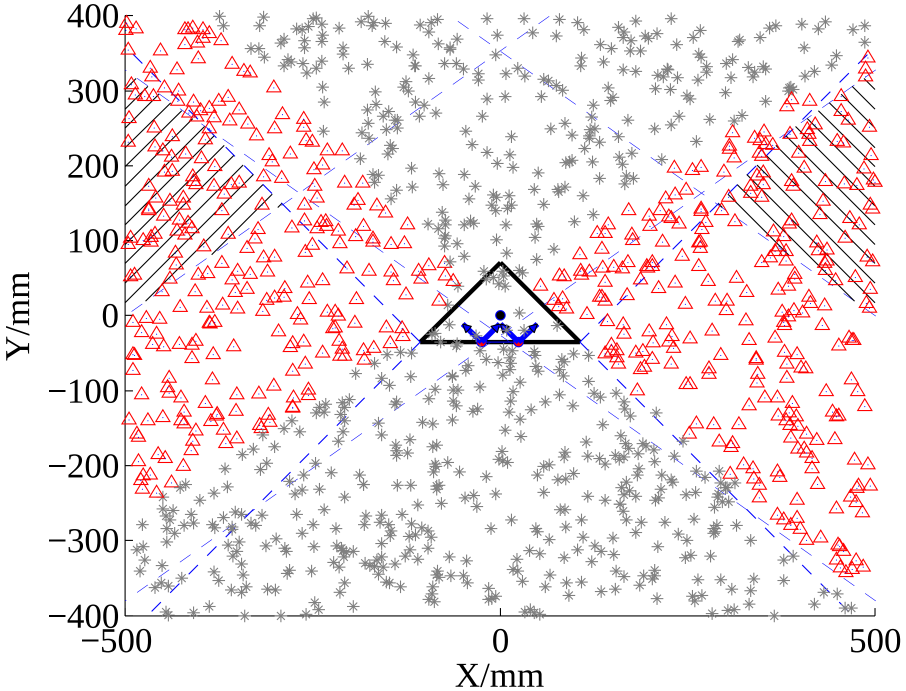


Figure 2.5: Hatched areas correspond to stable CORs region based on the conservative analysis (Lynch and Mason (1996)). Red triangles are stable CORs and gray stars are non-stable CORs based on the poly4-cvx model. The two push points are 50mm in width. The pusher and the object are covered with electrical tape and gaffer tape respectively with measured coefficient of friction equaling one.

2.2 Kinematic Contact Model

This section derives a quasi-static kinematic contact solution for manipulation problems with finite planar sliding motion. The algorithm maps a commanded rigid position-controlled end effector motion to the instantaneous resultant object motion, with detection of equilibrium state (jamming or grasping). The applied wrench is solved as an intermediate output. We show that single contact with a convex quadratic force-motion model has a unique analytic linear solution which extends Lynch et al. (1992). The case for a high order convex polynomial force-motion model is reduced to solving a sequence of such subproblems. For multiple contacts (e.g., pushing with multiple points or grasping) we need to add linear complementarity constraints (Stewart and Trinkle (1996)) at the pusher points, and the entire problem is a standard linear complementarity problem (LCP). The inherent stochasticity in frictional sliding is modelled by sampling the physics parameters from proper distributions. We validate the model by comparing simulation with large scale experimental data on robotic pushing and grasping tasks. The model serves as a good basis for both open loop planning and feedback control.

With a position-controlled manipulator, we are given a single point finger contact at $\mathbf{p}$ with inward normal $\mathbf{n}_p$, pushing velocity $\mathbf{v_p}$ and coefficient of friction μ_c between the pusher and the object. The task is to resolve the incurred body twist $\mathbf{V}$ and contact mode (sticking, slipping, breaking contact): find a $\mathbf{V}$ consistent with the contact mode at $\mathbf{p}$ while the applied wrench, solved as an intermediate output (not a supplied control), equals the corresponding generalized friction load.

2.2.1 Single point pusher

We introduce the concept of motion cone first proposed in Mason (1986b). Let the Jacobian matrix $J_p = \begin{bmatrix} 1 & 0 & -p_y \\ 0 & 1 & p_x \end{bmatrix}$, and denote by $\mathbf{F}_l = J_p^T \mathbf{f}_l$ and $\mathbf{F}_r = J_p^T \mathbf{f}_r$ the left and right edges of the applied wrench cone with corresponding resultant twist directions $\mathbf{V}_l = \nabla H(\mathbf{F}_l)$ and $\mathbf{V}_r =$

$\nabla H(\mathbf{F}_r)$ respectively. The left edge of the motion cone is $\mathbf{v}_l = J_p \mathbf{V}_l$ and the right edge of the motion cone is $\mathbf{v}_r = J_p \mathbf{V}_r$. Mason (1986b) showed that if the contact point pushing velocity $\mathbf{v_p}$ is inside the motion cone, i.e., $\mathbf{v_p} \in \mathbf{K}(\mathbf{v}_l, \mathbf{v}_r)$, the contact sticks. When $\mathbf{v_p}$ is outside the motion cone, sliding occurs. If $\mathbf{v_p}$ is to the left of $\mathbf{v}_l$, the pusher will slide left with respect to the object. Otherwise if $\mathbf{v_p}$ is to the right of $\mathbf{v}_r$, the pusher will slide right as shown in Fig. 2.12.

The following constraints hold assuming sticking contact:

$$v_{px} = V_x - \omega p_y \tag{2.9}$$

$$v_{py} = V_y + \omega p_x \tag{2.10}$$

$$\mathbf{V} = k \cdot \nabla H(\mathbf{F}), \qquad k > 0 \tag{2.11}$$

$$\tau = -p_y F_x + p_x F_y \tag{2.12}$$

In the case of ellipsoid (convex quadratic) representation, i.e., $H(\mathbf{F}) = \mathbf{F}^T A \mathbf{F}$ where A is a positive definite matrix, the problem is a full rank linear system with a unique solution. Lynch et al. (1992) give an analytical solution when A is diagonal. We show that a unique analytic solution exists for any positive definite symmetric matrix A. Let $\mathbf{t} = [-p_y, p_x, -1]^T$, $D = [J_p^T, A^{-1}\mathbf{t}]^T$ and $\mathbf{V}_p = [\mathbf{v_p}^T, 0]^T$, equations 2.9-2.12 can then be combined into one linear equation:

$$\mathbf{V} = D^{-1} \mathbf{V}_p \tag{2.13}$$

Theorem 3. *Pushing with single sticking contact and the convex quadratic representation of limit surface (abbreviated as P.*1*) has a unique solution from a linear system.*

Proof. From equation 2.13, we only need to prove D is invertible.

1. The row vectors of J_p are linearly independent and span a plane.
2. $J_p \mathbf{t} = 0$ implies $\mathbf{t}$ is orthogonal to the spanned plane.
3. If D is not full rank, then $A^{-1}\mathbf{t}$ must lie in the spanned plane and is therefore orthogonal

to $\mathbf{t}$. This contradicts with the fact that $\langle \mathbf{t}, A^{-1}\mathbf{t} \rangle > 0$ for positive definite matrix A^{-1} and nonzero vector $\mathbf{t}$.

□

Corollary 1. *Pushing with single sticking contact and the general homogeneous convex polynomial representation of limit surface is reducible to solving a sequence of sub-problems* $P.1$.

For general convex polynomial representation $H(\mathbf{F})$, the following optimization is equivalent to equations 2.9-2.12:

$$\underset{\mathbf{F}}{\text{minimize}} \quad \|J_p \nabla H(\mathbf{F}) - \mathbf{v_p}\| \tag{2.14}$$

$$\text{subject to} \quad \mathbf{t}^T F = 0 \tag{2.15}$$

When $H(\mathbf{F})$ is of the convex quadratic (ellipsoidal) form, the analytical minimizer is $\mathbf{F} = A^{-1}D^{-1}\mathbf{V}_p$. In the case of high order convex homogeneous polynomials, we can resort to an iterative solution where we use the Hessian matrix as a local ellipsoidal approximation, i.e., set $A_t = \nabla^2 H(\mathbf{F_t})$ and compute $\mathbf{F_{t+1}} = A_t^{-1}D^{-1}\mathbf{V}_p$ until convergence.

When $\mathbf{v_p}$ is outside of the motion cone, assuming right sliding occurs without loss of generality, the wrench applied by the finger equals $\mathbf{F}_r$. The resultant object twist $\mathbf{V}$ follows the same direction as $\mathbf{V}_r$ with proper magnitude such that the contact is maintained:

$$\mathbf{V} = s\mathbf{V}_r \tag{2.16}$$

$$s = \frac{\mathbf{n_p}^T \mathbf{v}_p}{\mathbf{n_p}^T \mathbf{v}_l} \tag{2.17}$$

2.2.2 Multi-contacts

Mode enumeration is tedious for multiple contacts. The linear complementarity formulation for frictional contacts (Stewart and Trinkle (1996)) provides a convenient representation. Denote by

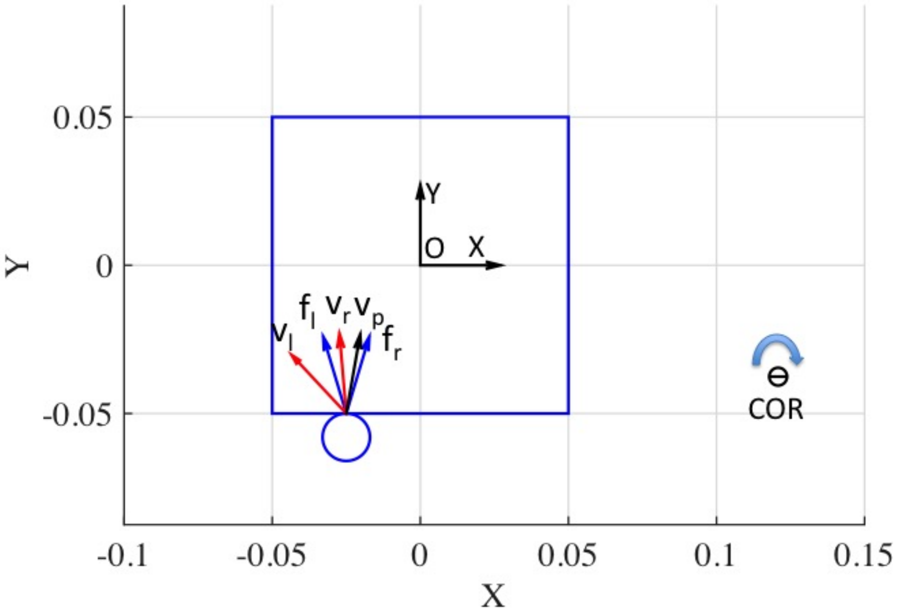


Figure 2.6: Mechanics of single point pushing. The square has a uniform pressure distribution over 100 support grid points sharing the same coefficient of friction. We use a fourth-order convex polynomial to represent the limit surface. The finger's pushing velocity is to the right of the motion cone and hence the finger will slide to the right. The instantaneous center of rotation, computed using the model described in section 2.2.1, is marked as a circle with a negative sign indicating clockwise rotation.

m the total number of contacts, the quasi-static force-motion equation is given by:

$$\mathbf{V} = k\nabla H(\mathbf{F}), \tag{2.18}$$

where the total applied wrench is the sum of normal and frictional wrenches over all applied contacts:

$$\mathbf{F} = \sum_{i=1}^{m} J_{p_i}^T (f_{n_i}\mathbf{n}_{\mathbf{p_i}} + D_{\mathbf{p_i}}\mathbf{f}_{t_i}). \tag{2.19}$$

f_{n_i} is the normal force magnitude along the normal $\mathbf{n_i}$, and $\mathbf{f}_{t_i}$ is the vector of tangential friction force magnitudes along the column vector basis of $D_{\mathbf{p_i}} = [\mathbf{t}_{p_i}, -\mathbf{t}_{p_i}]$. The velocity at contact point $\mathbf{p_i}$ on the object is given by $J_{p_i}\mathbf{V}$. The first order complementarity constraints on the normal force magnitude and the relative velocity are given by:

$$0 \leq f_{n_i} \perp (\mathbf{n}_{p_i}^T(J_{p_i}\mathbf{V} - \mathbf{v}_p)) \geq 0. \tag{2.20}$$

The complementarity constraints for Coulomb friction are given by:

$$0 \leq \mathbf{f}_{t_i} \perp (D_{\mathbf{p_i}}^T(J_{p_i}\mathbf{V} - \mathbf{v}_p) + \mathbf{e}\lambda_i) \geq 0, \tag{2.21}$$

$$0 \leq \lambda_i \perp (\mu_i f_{n_i} - \mathbf{e}^T\mathbf{f}_{t_i}) \geq 0, \tag{2.22}$$

where μ_i is the coefficient of friction at $\mathbf{p_i}$ and $\mathbf{e} = [1;1]$. In the case of ellipsoid (convex quadratic) representation, i.e., $H(\mathbf{F}) = \mathbf{F}^TA\mathbf{F}$ where A is a positive definite matrix, equations 2.18 to 2.22 can be written in matrix form:

$$\begin{bmatrix} 0 \\ \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} A^{-1}/k & -N^T & -L^T & 0 \\ N & 0 & 0 & 0 \\ L & 0 & 0 & E \\ 0 & \mu & -E^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V} \\ \mathbf{f}_n \\ \mathbf{f}_t \\ \lambda \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbf{a} \\ \mathbf{b} \\ 0 \end{bmatrix}, \tag{2.23}$$

$$0 \leq \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \perp \begin{bmatrix} \mathbf{f}_n \\ \mathbf{f}_t \\ \lambda \end{bmatrix} \geq 0,$$

where the binary matrix $E \in R^{2m\times m}$ equals $\begin{bmatrix} \mathbf{e} & & \\ & \ddots & \\ & & \mathbf{e} \end{bmatrix}$, $\mu = [\mu_1, \ldots, \mu_m]^T$, the stacking matrix $N \in R^{m\times 3}$ equals $[\mathbf{n}_{p_1}^T J_{p_1}; \ldots; \mathbf{n}_{p_m}^T J_{p_m}]$, the stacking matrix $L \in R^{2m\times 3}$ equals $[D_{p_1}^T J_{p_1}; \ldots; D_{p_m}^T J_{p_m}]$, the stacking vector $\mathbf{a} \in R^m$ equals $[-\mathbf{n}_{p_1}^T \mathbf{v}_{p_1}, \ldots, -\mathbf{n}_{p_m}^T \mathbf{v}_{p_m}]^T$ and vector $\mathbf{b} \in R^{2m}$ equals $[-D_{p_1}^T \mathbf{v}_{p_1}, \ldots, -D_{p_m}^T \mathbf{v}_{p_m}]^T$.

Note that the positive scalar k will not affect the solution value of $\mathbf{V}$ since $\mathbf{f}_n$ and $\mathbf{f}_t$ will scale accordingly. Hence, we can drop the scalar k and further substitute $\mathbf{V} = A(N^T \mathbf{f}_n + L^T \mathbf{f}_t)$ into equation 2.23 and reach the standard linear complementarity form as follows:

$$\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} NAN^T & NAL^T & 0 \\ LAN^T & LAL^T & E \\ \mu & -E^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{f}_n \\ \mathbf{f}_t \\ \lambda \end{bmatrix} + \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \\ 0 \end{bmatrix}, \tag{2.24}$$
$$0 \leq \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \perp \begin{bmatrix} \mathbf{f}_n \\ \mathbf{f}_t \\ \lambda \end{bmatrix} \geq 0.$$

Similarly, for the case of high order convex homogeneous polynomials, we can iterate between taking the linear Hessian approximation and solving the LCP problem in equation 2.24 until convergence.

Lemma 1. *The object is quasi-statically jammed or grasped if equation 2.24 yields no solution.*

Fig. 2.7 provides a graphical proof. When equation 2.24 yields no solution, either there is no feasible kinematic motion of the object without penetration or all the friction loads associated with the feasible instantaneous twists cannot balance against any element from the set of possible applied wrenches. In this case, the object is quasi-statically jammed or grasped between the fingers. Neither the object nor the end effector can move.

2.2.3 Stochastic Modelling

Frictional interaction is inherently stochastic. Two major sources contribute to the uncertainty in planar motion: 1) indeterminacy of the supporting friction distribution $\mathbf{f_r}$ due to changing pressure distribution and coefficients of friction between the object and support surface; 2) the

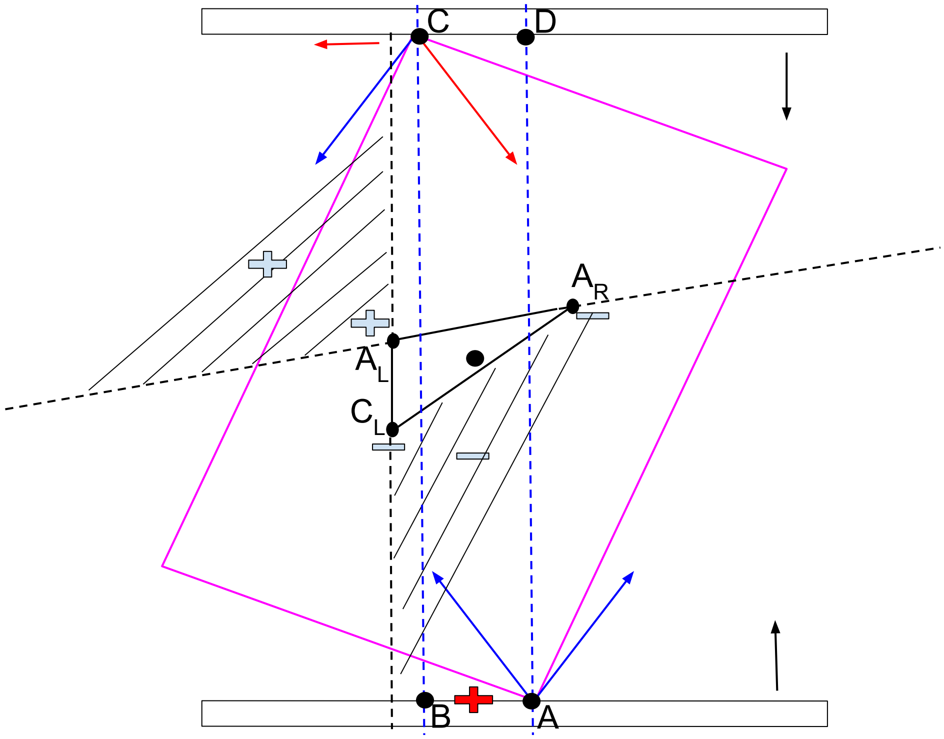


Figure 2.7: Using moment labeling (Mason (2001)), the center of rotation (COR) has positive sign (counter-clockwise) and can only lie in the band between the two blue contact normal lines. Further, the COR must lie on segment AB (contact point A sticks) or segment CD (contact point C sticks) since otherwise both contacts will slip, but the total wrench from the two left edges of the friction cones has negative moment which cannot cause counter-clockwise rotation. Without loss of generality, we can assume COR (red plus) lies on segment AB, leading to sticking contact at A and left sliding at C. Following a similar analysis using the force dual graphical approach (Brost and Mason (1991)), each single friction force can be mapped to its instantaneous resultant signed COR whose convex combination forms the set of all possible CORs under the composite friction forces. The COR can either be of positive sign in the upper left hatched region or negative sign in the lower right hatched region which contradicts with the proposed AB segment. Hence jamming occurs and neither the gripper nor the object can move. This corresponds exactly to the no solution case of equation 2.24.

coefficient of friction μ_c between the object and the robot end effector. We sample μ_c uniformly from a given range and model the effect of changing support friction distribution by sampling the parameters a in $H(\mathbf{F};a)$ from a distribution that satisfies:

1. Samples from the distribution should result in a even degree homogeneous convex polynomial to represent the limit surface.

2. The mean can be set as a prior estimate and the amount of variance controlled by one parameter.

The n_{df} degree of freedom Wishart distribution (Wishart (1928)) $S \sim W(\hat{S}, n_{df})$ with mean $n_{df} S_{est}$ and variance $Var(S_{ij}) = n_{df}(\hat{S}_{ij}^2 + \hat{S}_{ii}\hat{S}_{jj})$ is defined over symmetric positive semidefinite random matrices as a generalization of the chi-squared distribution to multi-dimensions. For ellipsoidal (convex quadratic) $H(\mathbf{F}; A)$, we can directly sample from $\frac{1}{n_{df}} W(A_{est}, n)$ where A_{est} is some estimated value from data or fitted for a particular pressure distribution. Sampling from general convex polynomials is hard. Fortunately, we find that sampling from the sos-convex polynomials subset is not. The key is the coefficient vector a of a sos-convex polynomial $H(\mathbf{F}; a)$ has a unique one-to-one mapping to a positive definite matrix Q so that we can first sample $\tilde{Q}$ from $\frac{1}{n_{df}} W(Q, n_{df})$ [5] and then map back to $\tilde{a}$ through equation 2.5. The degree of freedom parameter n_{df} determines the sampling variance. The smaller n_{df} is, the noisier the system will be.

2.2.4 Deterministic Pushing Model Evaluation

We evaluate the single contact deterministic model on the large scale MIT pushing dataset (Yu et al. (2016)) and the data from the identification experiments in section 2.1.3.2. For the MIT pushing dataset, we use 10mm/s velocity data logs for 10 objects[6] on 3 hard surfaces including delrin, abs and plywood. The force torque signal is first filtered with a low pass filter and 5 wrench-twist pairs evenly spaced in time are extracted from each push action log file. 10 random train-test splits (20 percent of the logs for training, 10 percent for validation and the rest for testing) are conducted for each object-surface scenario.

Given two poses $q_1 = [x_1, y_1, \theta_1]$ and $q_2 = [x_2, y_2, \theta_2]$, we define the deviation metric $d(q_1, q_2)$ which combines both the displacement and angular offset as $d(q_1, q_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 +$

[5] We have noted that adding a small constant on the diagonal elements of $\tilde{Q}$ improves numerical stability.

[6] Despite having the same experimental set up and similar geometry and friction property to the other two triangular shapes, the results for object Tr2 is about 1.5 -2 times worse. Due to time constraint, we have not ruled out the possibility that the data for object Tr2 is corrupted.

	rect1	rect2	rect3	tri1	tri3	ellip1	ellip2	ellip3	hex	butter
poly4-delrin	8.28±0.29	5.37±0.23	6.10±0.21	9.71±0.33	7.54±0.23	7.68±0.51	8.90±1.40	7.35±0.38	6.38±0.28	4.83±0.27
quad-delrin	8.60±0.35	5.92±0.14	8.20±0.16	9.90±0.41	8.18±0.15	6.85±0.25	6.29±0.24	8.08±0.51	6.42±0.12	5.97±0.23
delrin	35.48	40.53	35.98	36.91	34.66	32.18	38.05	33.37	33.55	34.09
poly4-abs	5.86±0.11	7.48±0.80	3.59±0.12	7.13±0.26	5.17±0.38	8.45±1.13	9.18±1.26	5.93±0.19	7.56±0.39	3.94±0.11
quad-abs	6.07±0.16	6.74±0.27	6.19±0.18	8.00±0.37	7.17±0.37	6.66±0.28	7.69±0.27	5.78±0.21	8.19±0.21	5.39±0.15
abs	34.14	39.74	33.98	35.43	32.37	32.68	33.53	32.45	33.23	33.53
poly4-plywood	6.86±0.71	6.86±0.13	5.93±0.33	4.61±0.13	7.21±0.47	4.39±0.16	4.99±0.31	5.72±0.31	8.41±0.24	4.72±0.17
quad-plywood	6.20±0.20	7.22±0.18	6.88±0.18	5.96±0.19	9.43±0.56	4.42±0.12	5.84±0.20	6.46±0.26	8.85 ±0.17	6.05±0.22
plywood	31.86	33.22	32.94	32.81	33.78	27.24	28.23	33.29	32.77	34.10

Table 2.3: Average deviation metric (in mm) between the simulated final pose and actual final pose with 95 percent confidence interval. The 3rd and 6th rows are the deviation from the ground truth initial pose and final pose to indicate how much the object is moved due to the push. In most cases, the fourth order convex (poly4) polynomial has better accuracy. The average normalized percentage error for poly4 is 20.05% and for quadratic is 21.39%. However, the accuracy of a fixed deterministic model is bounded by the inherent variance of the system.

$\rho \cdot \min(|\theta_1 - \theta_2|, 2\pi - |\theta_1 - \theta_2|)$, where ρ is the characteristic length of the object (e.g., radius of gyration or radius of minimum circumscribed circle). A one dimensional coarse grid search over the coefficient of friction μ_c between the pusher and object is chosen to minimize average deviation of the predicted final pose and ground truth final pose on training data. Table 2.3 shows the average metric with a 95% percent confidence interval. Interestingly, we find that using more training data does not improve the performance much. This is likely due to the inherent stochasticity and changing surface conditions as reported in Yu et al. (2016).

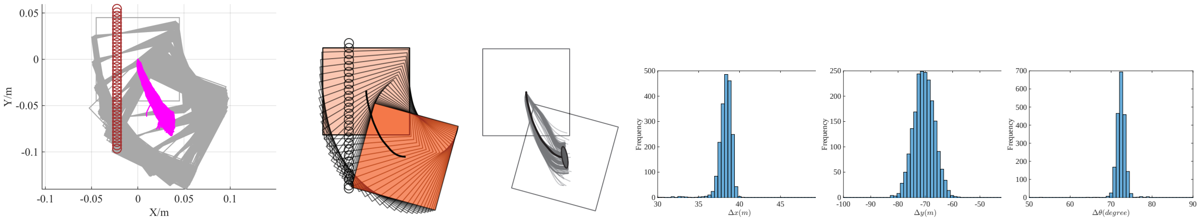
The well-machined objects in the MIT pushing dataset are close to uniform patch pressure. We also test on discrete pressures. The triangular object used in the identification experiments in section 2.1.3.2 are given different configurations of three points discrete support. We use wrench twists pairs sampled from the ideal limit surface for training. The coefficient of friction between the object and pusher is determined by a grid search over 40 percent of the logs. We use the remaining 60 percent to evaluate simulation accuracy. Results are reported in Table 2.4.

2.2.5 Stochastic Pushing Model Simulation

Yu et al. (2016) reported the same 2000 straight-line pushes in a highly controlled setting result in a distribution of final poses, demonstrating the inherent stochastic nature of pushing. We perform

	3pts1	3pts2	3pts3	3pts4
poly4-hardboard	3.52±0.21	2.75±0.25	2.92±0.27	2.80±0.23
quad-hardboard	3.82±0.24	3.63±0.27	3.35±0.23	3.96±0.28
hardboard	16.63	13.86	14.83	15.15
poly4-plywood	3.78±0.11	2.80±0.15	2.84±0.16	3.26±0.11
quad-plywood	4.24±0.15	3.56±0.17	3.28±0.08	4.12±0.13
plywood	16.56	13.81	15.27	14.20

Table 2.4: Average deviation (in mm) between the simulated final pose and actual final pose with 95 percent confidence interval for 3-point support. The wrench-twist pairs used for training the model are generated from the ideal limit surface. The 3rd, 6th and 9th rows are the deviation from the ground truth initial pose and final pose to indicate how much the object is moved due to the push. The fourth order convex (poly4) polynomial has better accuracy for each pressure-surface combination. The average normalized error for poly4 is 20.48% and for quadratic is 24.97%.



(a) Stochastic simulation results.

(b) Figure 9 of Yu et al. (2016), reprinted with permission.

(c) Δx histogram

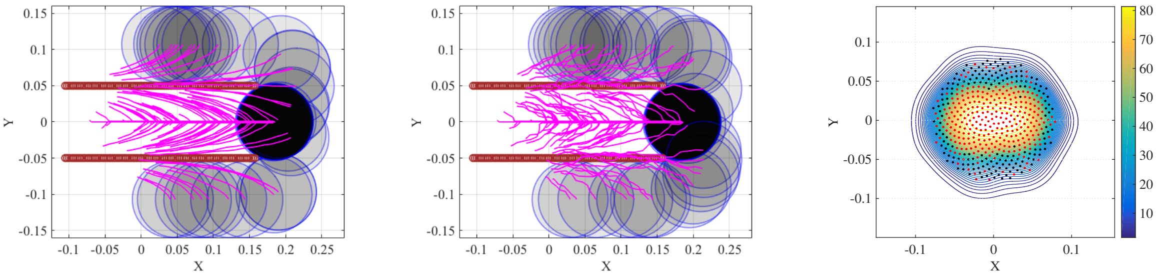
(d) Δy histogram

(e) $\Delta\theta$ histogram

Figure 2.8: Stochastic modelling of single point pushing with fourth order sos-convex polynomial representation of the limit surface using wrench twist pairs generated from 64 grids with uniform pressure. The degree of freedom in the sampling distribution equals 20. The contact coefficient of friction between the pusher and the object is uniformly sampled from 0.15 to 0.35. The trajectories are qualitatively similar to the experimental results in Figure 9 of Yu et al. (2016).

simulations using the same object and pusher geometry and push distance. The 2000 resultant trajectories and histogram plot of pose changes are shown in Fig. 2.8. We note that although the mean and variance pose changes are similar to the experiments with abs material in Yu et al. (2016), the distribution resemble a single Gaussian distribution which differs from the multiple modes distribution in Figure 10 of Yu et al. (2016). We conjecture this is due to a time varying stochastic process where coefficients of friction between surfaces drift due to wear.

We also simulate the effects of initial pose uncertainty reduction with 2 point fingers under the stochastic contact model. The circular object has a radius of 5.25cm. The two fingers separated by 10cm perform a straight line push of 26.25cm. The desired goal is to have the object centered with respect to the two fingers. Fig. 2.9a and 2.9b compare the resultant trajectories under different amount of system noise. We find that despite larger noise in the resultant trajectories, the convergent region of the stable goal pose differs by less than 5% and the difference is mostly around the uncertainty boundary. A kernel density plot of the convergence region is shown in in Fig. 2.9c for $n_{df} = 10$. This demonstrates multiple constraints can induce a large region of attraction despite uncertainty.



(a) 100 pushed trajectories of different initial poses using ellipsoid representation of $H(\mathbf{F})$ with $n_{df} = 200$.

(b) 100 pushed trajectories of different initial poses using ellipsoid representation of $H(\mathbf{F})$ with $n_{df} = 10$ (larger noise).

(c) Kernel density plot of the convergence region for $n_{df} = 10$. Convergent initial poses are in red, and the rest are in black.

Figure 2.9: Simulation results using the proposed contact model illustrating the process of two point fingers pushing a circle to reduce initial pose uncertainty. A total of 500 initial object center positions are uniformly sampled from a circle of radius 7.88cm.

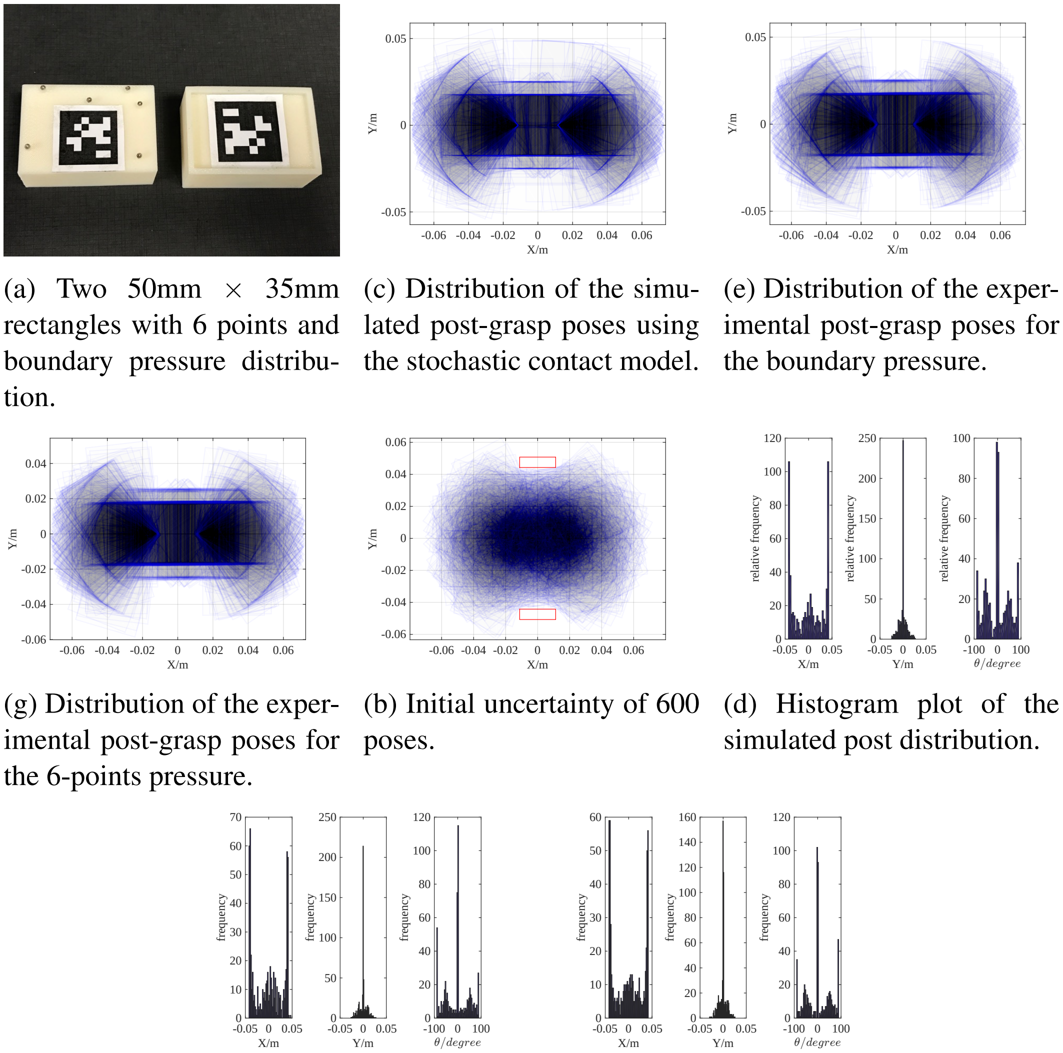
2.2.6 Grasping Simulation

We conduct robotic experiments to evaluate our contact model for grasping. Fig. 2.10a shows two rectangular objects with the same geometry but different pressure distributions. Another experiment is conducted for a butterfly shaped object shown in Fig. 2.11a. We use the Robotiq C-85 2-finger gripper (ROBOTIQ (2017)) and represent it as a planar parallel-jaw gripper with rectangular fingers in simulation as shown in Fig. 2.10b and Fig. 2.11b. Convex quadratic representations of $H(\mathbf{F})$ are trained from wrench-twist pairs assuming a uniform friction distribution along the object boundary. The sampling degree of freedom n_{df} equals 250 with contact friction coefficient μ_c sampled uniformly from [0.015, 0.02]. For the rectangles of both pressure distribution, the simulated results with the stochastic contact model match well with experimental data. However, the model fails to capture the stability of grasps and the deformation of objects. In the case of a butterfly-shaped object, many unstable grasps and jamming equilibria exist, but as the fingers increase the gripping force the object will "fly" away as the stored elastic energy turns into large accelerations which violates the quasistatic assumptions of our model, as revealed in the scattered post-grasp distribution in Fig. 2.11e. We also compare the cases where dynamics do not play a major role: Fig. 2.11g shows the zoomed in plots to compare with simulation results in Fig. 2.11c. Despite qualitative similarity, the simulation results deviate more compared with the case for the rectangular geometry. As shown in the histogram plots in Fig. 2.11d and Fig. 2.11h, the simulation returns more jamming and grasping final states as illustrated by the spikes in θ.

2.3 Differential Flatness, Trajectory Planning and Control

In this section, we prove that quasi-static pushing with a sticking contact and ellipsoid approximation of the limit surface is differential flat. Both graphical and algebraic derivations are given. A major conclusion is the pusher-slider system is reducible to the Dubins car problem where the sticking contact constraints translate to bounded curvature. Planning is as easy as computing Dubins curve with the additional benefit of time-optimality. For trajectory stabilization, we design closed-loop control using dynamic feedback linearization or open-loop control using two contact points as a form of mechanical feedback. We conduct robotic experiments using objects with different pressure distributions, shape and contact materials placed at different initial poses that require difficult maneuvers to the goal pose. The average error is within 1.67mm in translation and 0.5 degrees in orientation over 60 experimental trials. We also show an example of pushing among obstacles using a RRT planner with exact steering.

The quasi-static pusher-slider system is a canonical hybrid system with model uncertainty due



(a) Two 50mm × 35mm rectangles with 6 points and boundary pressure distribution.

(c) Distribution of the simulated post-grasp poses using the stochastic contact model.

(e) Distribution of the experimental post-grasp poses for the boundary pressure.

(g) Distribution of the experimental post-grasp poses for the 6-points pressure.

(b) Initial uncertainty of 600 poses.

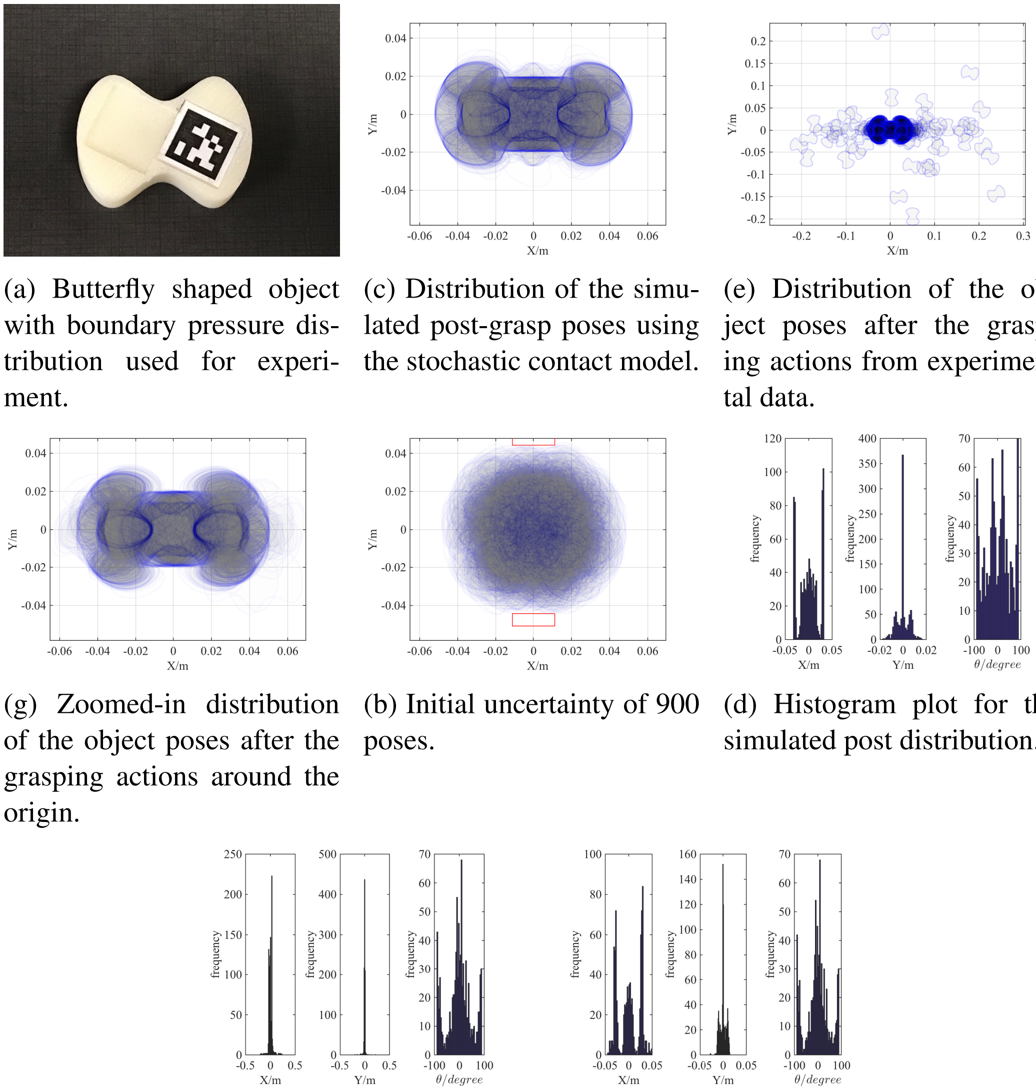
(d) Histogram plot of the simulated post distribution.

(f) Histogram of the experimental post distribution for the boundary pressure.

(h) Histogram of the experimental post distribution for the 6-points pressure.

Figure 2.10: Experiments on the rectangular objects with different pressure distributions. 600 initial poses are sampled whose centers are uniformly distributed in a circle of radius of 20mm and angles are uniformly distributed from -90 to 90 degrees.

to indeterminate and stochastic friction distribution. We assume pushing with rolling/sticking contact where the object is constrained to a two dimensional space embedded in SE(2). The nonholonomic constraint appears to be challenging but is fortunately not. Intuitively, the applied force through the contact point is bounded inside the friction cone and hence the turning rate of the object must be bounded, indicating similarity to the steering car system with bounded front



(a) Butterfly shaped object with boundary pressure distribution used for experiment.

(c) Distribution of the simulated post-grasp poses using the stochastic contact model.

(e) Distribution of the object poses after the grasping actions from experimental data.

(g) Zoomed-in distribution of the object poses after the grasping actions around the origin.

(b) Initial uncertainty of 900 poses.

(d) Histogram plot for the simulated post distribution.

(f) Histogram plot of the experimental post distribution.

(h) Histogram plot of the experimental post distribution around the origin.

Figure 2.11: Experiments on the butterfly object. The longer diameter between the convex curves is 39mm and the shorter diameter between the concave curves is 28.6mm. 900 initial poses are sampled where the centers lie uniformly in a circle of radius 30mm and the frame angles are uniformly distributed in -90 to 90 degrees.

wheel turning angle.

We use differential flatness techniques from nonlinear control theory Murray et al. (1995), which offers some advantage for trajectory generation and control of underactuated mechanical systems. A particular interesting system is the kinematic steering car whose flat output is the center of the rear axle. The problem is well-studied: the time-optimal motion planning solution is given by Dubins curve Dubins (1957) and globally stable controller sythesized with dynamic feedback linearization Oriolo et al. (2002). In this paper, we show that the pusher-slider system with single sticking contact is differential flat, which opens new avenue for trajectory planning and stabilization. We first give an intuitive graphical analysis and continue with algebraic derivation.

2.3.1 Motion Equations

We describe the single point pusher quasi-static motion model assuming rigid body mechanics with Coulomb friction. The following notations are used:

- $\mathbf{q}$: the object pose (x, y, θ) in the world frame W.
- $\mathbf{p}$: the contact point (p_x, p_y) in the local frame O.
- $\mathbf{n_p}$: the inward normal in the local frame at $\mathbf{p}$.
- $\mathbf{f}$: the applied force by the pusher.
- $\mathbf{f_l}, \mathbf{f_r}$: the left and right edges of the friction cone.
- $\mathbf{u}$: the pushing velocity of the pusher in the object local frame O.
- $\mathbf{u_l}, \mathbf{u_r}$: the left and right edges of the motion cone.
- $\mathbf{F}, \mathbf{V}$: the applied wrench and resultant twist in the object local frame.

The force-motion model for quasi-static pushing given certain pressure distribution can be efficiently established through limit surface representation Goyal (1989); Zhou et al. (2016); Howe and Cutkosky (1996). Points inside the surface correspond to static friction wrenches. Points on the surface correspond to friction wrenches with normals parallel to sliding twist directions, forming a mapping between friction wrench and sliding twist.

Using a homogeneous even-degree convex polynomial $H(\mathbf{F})$ representation for the limit surface Zhou et al. (2016), the resultant object twist V follows the same direction as the gradient evaluated at the applied wrench **F**:

$$\mathbf{V} = k\nabla H(\mathbf{F}), \qquad k > 0, \tag{2.25}$$

A global diagonal ellipsoid approximation Lynch et al. (1992); Hogan and Rodriguez (2016) is often adopted as a convenient representation by existing literature on model-based pushing, i.e., assuming $H(\mathbf{F}) = \mathbf{F}^T A\mathbf{F}$, where $A = \begin{bmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & b \end{bmatrix}$. In this paper, we find the closest vector of form $[a,a,b]$ to the diagonal vector of $\nabla^2 H(\mathbf{F_{np}})$ as a local approximation, where $\mathbf{F_{np}}$ is the normal contact wrench at the contact point **p**, i.e., the wrench applied as if the contact is frictionless.

$$a = (\nabla^2 H(\mathbf{F}_{n_p})_{11} + \nabla^2 H(\mathbf{F}_{n_p})_{22})/2 \tag{2.26}$$

$$b = \nabla^2 H(\mathbf{F}_{n_p})_{33} \tag{2.27}$$

Choose the positive y axis of the local body frame to align with the vector pointing from the contact point to the center of mass O. In doing so, $p_x = 0$. After absorbing the scalar k into **f**, we have that in local frame

$$\mathbf{V_x} = af_x \tag{2.28}$$

$$\mathbf{V_y} = af_y \tag{2.29}$$

$$\mathbf{V}_\theta = b\tau = -bp_y f_x. \tag{2.30}$$

The quasi-static motion equations in global frame are written as:

$$\dot{x} = a(f_x \cos\theta - f_y \sin\theta) \tag{2.31}$$

$$\dot{y} = a(f_x \sin\theta + f_y \cos\theta) \tag{2.32}$$

$$\dot{\theta} = -bp_y f_x, \tag{2.33}$$

and the friction cone constraint is given by

$$\|f_x\| \leq \mu f_y \tag{2.34}$$

$$f_y \geq 0 \tag{2.35}$$

Denote by $J_p = \begin{bmatrix} 1 & 0 & -p_y \\ 0 & 1 & p_x \end{bmatrix}$ the Jacobian matrix at $\mathbf{p}$ and $R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$, equations 2.31 to 2.33 is equivalent to:

$$\dot{\mathbf{q}} = R(\theta) A J_p^T \mathbf{f} \tag{2.36}$$

Note that the magnitude of **f** does not have physical meaning and is proportional to the magnitude of the input velocity. Since the pusher is position controlled, we will need to relate the applied force to pushing velocity. It can be shown **?** that for sticking contact the applied force **f** and pushing velocity **u** are linearly related, and have a one-to-one mapping:

$$\mathbf{u} = J_p A J_p^T \mathbf{f}. \tag{2.37}$$

Further, the friction cone constraint is translated into motion cone $\mathscr{K}(\mathbf{u_l}, \mathbf{u_r})$ where the left edge

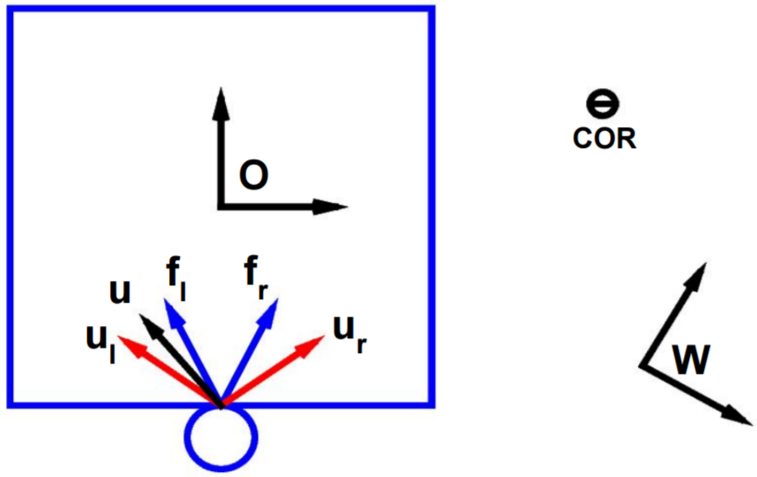


Figure 2.12: A rectangle pushed by a round finger. Blue arrows correspond to the friction cone $\mathscr{K}(f_l, f_r)$ edges and red arrows correspond to the motion cone $\mathscr{K}(u_l, u_r)$ edges. The instantaneous clockwise center of rotation (COR) is marked as a circle with negative sign. The contact sticks since the pushing direction u is inside the motion cone.

$\mathbf{u_l}$ and right edge $\mathbf{u_r}$ of the cone are given by:

$$\mathbf{u}_l = J_p A J_p^T \mathbf{f_l}, \qquad \mathbf{u}_r = J_p A J_p^T \mathbf{f_r}. \tag{2.38}$$

If $\mathbf{u}$ is within the motion cone, then the contact sticks as shown in Fig. 2.12. Left sliding occurs if $\mathbf{u}$ is to the left of $\mathbf{u_l}$ and right sliding occurs if $\mathbf{u}$ is to the right of $\mathbf{u_r}$. The planner and controller described in the rest of the paper assumes the control input is applied force, which will be converted to pusher velocity using equation 2.37.

2.3.2 Differential Flatness

The rough idea of differential flatness is to find flat output states (of the same dimension as control input) as a function of the original states and control inputs. Such mapping also admits an inverse function such that the original states and control inputs can be recovered from the flat output states and their high order derivatives without any integration step.

2.3.2.1 Graphical Derivation

From equations 2.28 to 2.30, an applied body wrench $\mathbf{F} = [\mathbf{f_x}, \mathbf{f_y}, \tau]$ is mapped to a body twist V. A twist in a plane can be further represented as a center of rotation $(\mathbf{V_x}/\mathbf{V}_\theta, \mathbf{V_y}/\mathbf{V}_\theta)$ with the same sign as $\mathbf{V}_\theta)$. Let $r = |p_y|$ be the distance from the line of force to the local frame origin. The distance $\tilde{r}$ from the center of rotation to the origin is inverse proportion to r:

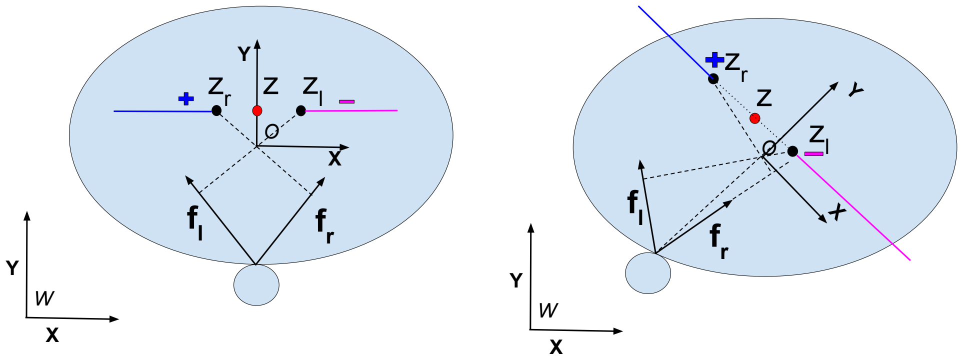
$$\tilde{r} = \sqrt{\frac{\mathbf{V_x^2} + \mathbf{V_y^2}}{\mathbf{V}_\theta^2}} = \frac{a}{br}. \tag{2.39}$$

If we define the unit length as a/b, then the center of rotation lies on the opposite side along the perpendicular line to the line of force through the contact point, with distance to the origin equals $1/r$. As in projective geometry, the dual of a line is a point. Here, the line of applied force is mapped to the resultant center of rotation point.

Proposition 1. *The instantaneous rotation centers corresponding to applied frictional forces through the contact point form a line perpendicular to the vector from the origin to the contact point. The distance from the line to the origin equals $\frac{a}{br}$, where r is the distance from the contact point to the origin.*

This is similar to the force-dual method Brost and Mason (1991) that maps a line of force to the acceleration center. The matrix A can be treated as a damping matrix that connects force to velocity, analogous to the inertia matrix in Brost and Mason (1991) that maps force to acceleration.

In Figure 2.13a, the friction cone is symmetric with respect to the origin as the contact point's normal passes through the origin. The friction cone constraints are represented using the force dual graphical method. Denote by z_l and z_r the instantaneous rotation centers given applied forces on the left edge f_l and the right edge f_r of the friction cone respectively. The allowable motion of z can be characterized by a ray (blue) of counter-clockwise rotation center starting from z_r or a ray (magenta) of clockwise rotation center starting from z_l. Figure 2.13b illustrates the general asymmetric case. Choose the positive y axis to be aligned with the vector pointing from the contact point to the center of mass, the trajectory of the pushed object can be exactly recovered from the trajectory of such point (flat output).



(a) Symmetric case.

(b) Asymmetric case.

Figure 2.13: Graphical analysis.

Theorem 4. *Any point on the line of center of rotations is a differential flat output.*

Proof. Section 2.3.2.2 provides an algebraic proof. We also give a geometrical proof sketch here.

1. Since the line of center of rotation is perpendicular to the positive y axis, the tangents along the trajectory point in the directions of body positive y axis (heading) of the object.
2. After knowing the orientation of the body frame, we can compute the position of the body frame since the point is fixed in the body frame.
3. The instantaneous center of rotation can be further determined from the curvature along the trajectory. Therefore the velocity control input is also known.

□

A key observation is that if we choose the mid point (in red color) between z_l and z_r, then the instantaneous motion constraints from the sticking contact are simply minimum turning radius constraints. We now have a reduction to a Dubins car model LaValle (2006); Dubins (1957) where the heading aligns with the positive body y axis and the mid point (red) of z_l and z_r is the center of rear axle.

2.3.2.2 Algebraic Derivation

This section derives the function mappings between cartesian pose and control to flat outputs.

Symmetric Case The symmetric case as shown in Figure 2.13a is when the pushing point's normal aligns with the vector pointing from the point p to the center of mass O. Hence we have $p_x = 0$ and $r = -p_y$. $\cos\theta \times$ (2.31) $+ \sin\theta \times$ (2.32) and $-\sin\theta \times$ (2.31) $+ \cos\theta \times$ (2.32) yield

$$f_x = \frac{1}{a}(\dot{x}\cos\theta + \dot{y}\sin\theta) \tag{2.40}$$

$$f_y = \frac{1}{a}(-\dot{x}\sin\theta + \dot{y}\cos\theta). \tag{2.41}$$

Together with equation (2.33) yields

$$-\frac{a}{br}\dot{\theta} + \dot{x}\cos\theta + \dot{y}\sin\theta = 0. \tag{2.42}$$

A choice of flat outputs are given by

$$z_1 = x - \frac{a}{br}\sin\theta \tag{2.43}$$

$$z_2 = y + \frac{a}{br}\cos\theta, \tag{2.44}$$

whose derivative are

$$\dot{z}_1 = \dot{x} - \frac{a}{br}\cos\theta\dot{\theta} \tag{2.45}$$

$$\dot{z}_2 = \dot{y} - \frac{a}{br}\sin\theta\dot{\theta} \tag{2.46}$$

Rewrite equation (2.42) using $z, \dot{z}$ we get

$$\dot{z}_1\cos\theta + \dot{z}_2\sin\theta = 0. \tag{2.47}$$

Therefore,

$$\theta = \arctan(\frac{-\dot{z}_1}{\dot{z}_2}) \tag{2.48}$$

$$x = z_1 - \frac{a\dot{z}_1}{br\sqrt{\dot{z}_1^2+\dot{z}_2^2}} \tag{2.49}$$

$$y = z_2 - \frac{a\dot{z}_2}{br\sqrt{\dot{z}_1^2+\dot{z}_2^2}} \tag{2.50}$$

$$f_x = \frac{\dot{z}_1\ddot{z}_2 - \dot{z}_2\ddot{z}_1}{br(\dot{z}_1^2+\dot{z}_2^2)} \tag{2.51}$$

$$f_y = \frac{\sqrt{\dot{z}_1^2+\dot{z}_2^2}}{a} \tag{2.52}$$

The friction cone constraints represented in flat output space can be written as

$$|\frac{\dot{z}_1\ddot{z}_2 - \dot{z}_2\ddot{z}_1}{(\dot{z}_1^2+\dot{z}_2^2)^{\frac{3}{2}}}| \leq \frac{br\mu}{a} \tag{2.53}$$

Note that constraints (2.53) is exactly the curvature of the trajectory of $z(t)$. We can now conclude that pushing with sticking constraint is equivalent to finding curves with bounded curvature that connects two 2D oriented points. In particular, Dubins curve Dubins (1957) is the time-optimal solution.

Figure 2.14 demonstrate two examples of trajectory planning with a single point sticking contact. The friction cone constraint is converted to minimum turning radius constraint. Dubins curve is generated in flat output space. Then the SE(2) poses of the object and control actions are mapped back to the cartesian space as given in equations 2.48 and 2.49 to 2.52.

General Case We also derive the general case when the contact point's normal is not aligned with the the vector as shown in Figure 2.13b. Let the local frame origin be the COM and the positive y-axis aligned with the vector pointing from the contact point p to center of mass O. We show that with this choice of reference frame, any point on the dual line of the friction cone is

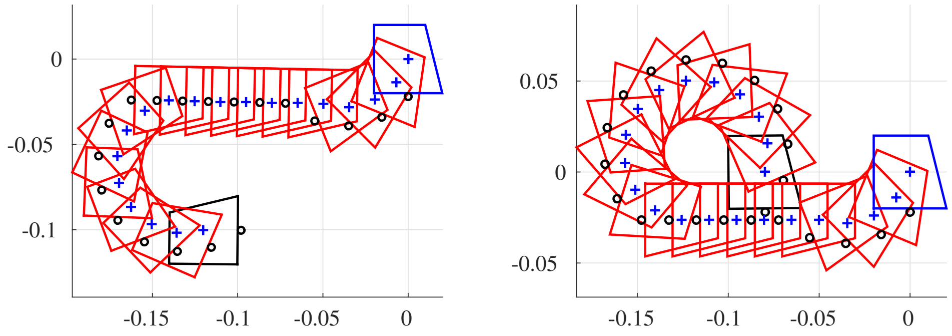


Figure 2.14: Example planned trajectories with the initial pose in black and the final pose in blue.

differential flat, and conveniently we can simply choose the mid point between the two extreme center of rotations that correspond to left and right edges of the friction cone. Denote by c_x the x component along the line of center of rotations, and $\rho = \frac{a}{b}$. The flat outputs are given by

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} c \\ \rho/r \end{bmatrix}, \tag{2.54}$$

where the vector $\begin{bmatrix} c \\ \rho/r \end{bmatrix}$ represents a point on the line of center of rotations (dual line of friction cone) in body frame.

Following similar steps in section 2.3.2.2, we can map the flat outputs to cartesian pose and applied force:

$$\theta = \arctan(\frac{-\dot{z}_1}{\dot{z}_2}) \tag{2.55}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} - \frac{1}{\sqrt{\dot{z}_1^2 + \dot{z}_2^2}} \begin{bmatrix} \dot{z}_2 & \dot{z}_1 \\ -\dot{z}_1 & \dot{z}_2 \end{bmatrix} \begin{bmatrix} c \\ \rho/r \end{bmatrix} \tag{2.56}$$

$$\begin{bmatrix} f_x \\ f_y \end{bmatrix} = \frac{1}{\dot{z}_1^2 + \dot{z}_2^2} \begin{bmatrix} \frac{\dot{z}_1\ddot{z}_2 - \ddot{z}_1\dot{z}_2}{br} \\ \frac{(\dot{z}_1^2+\dot{z}_2^2)^{\frac{3}{2}} - c(\dot{z}_1^2+\dot{z}_2^2)}{a} \end{bmatrix}. \tag{2.57}$$

Further, let c_l and c_r be the x components of the center of rotations corresponding to $\mathbf{f_l}$ and $\mathbf{f_r}$. If we set $c = (c_l + c_r)/2$, then the friction cone constraint is turned into a curvature upper bound of $2/|c_l - c_r|$ (or minimum turning radius of $|c_l - c_r|/2$.)

2.3.3 Stabilization

Uncertainty for robotic pushing mainly come from two sources: 1) perception uncertainty for the initial object pose and 2) model uncertainty due to changing friction distribution. Single contact pushing cannot be open-loop stable and needs active feedback control strategy. Section 2.3.3.1 derives a linear tracking controller in flat output space through dynamic feedback linearization. Section 2.3.3.2 addresses improving robustness against model uncertainty through open-loop stable two-points push that naturally induces mechanical feedback.

2.3.3.1 Dynamic Feedback Linearization Control

Equation 2.36 is in the form of driftless underactuated system with three degrees of freedom state and two degrees of freedom control input:

$$\dot{\mathbf{q}} = G(\mathbf{q})\mathbf{f}. \tag{2.58}$$

For such systems, dynamic feedback linearization finds a feedback compensator of the form:

$$\dot{\zeta} = \alpha(\mathbf{q}, \zeta) + \beta(\mathbf{q}, \zeta)\mathbf{w} \tag{2.59}$$

$$\mathbf{f} = \gamma(\mathbf{q}, \zeta) + \delta(\mathbf{q}, \zeta)\mathbf{w}, \tag{2.60}$$

where the kth derivative of flat output z can be directly controllable via w

$$z^k = w. \tag{2.61}$$

Differentiating the flat output z with respect to time yields

$$\dot{z} = \begin{bmatrix} 0 & -a\sin\theta \\ 0 & a\cos\theta \end{bmatrix} \begin{bmatrix} f_x \\ f_y \end{bmatrix}. \tag{2.62}$$

We need to keep taking derivative since only f_y affects $\dot{z}$. Denote by $\zeta = f_y$ the dynamic feedback compensator and $s = \dot{\zeta}$.

$$\ddot{z} = a \begin{bmatrix} -\sin\theta & -\zeta br\cos\theta \\ \cos\theta & -\zeta br\sin\theta \end{bmatrix} \begin{bmatrix} s \\ f_x \end{bmatrix} \tag{2.63}$$

Let

$$\begin{bmatrix} s \\ f_x \end{bmatrix} = (a \begin{bmatrix} -\sin\theta & -\cos\theta \\ \cos\theta & -\sin\theta \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \zeta br \end{bmatrix})^{-1} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \tag{2.64}$$

$$= \begin{bmatrix} -\sin\theta/a & \cos\theta/a \\ -\cos\theta/(a\zeta br) & -\sin\theta/(a\zeta br) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}, \tag{2.65}$$

which leads to

$$\ddot{z} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = w \tag{2.66}$$

The dynamic feedback compensator is of the following form:

$$\dot{\zeta} = -w_1\sin\theta/a + w_2\cos\theta/a \tag{2.67}$$

$$f_y = \zeta \tag{2.68}$$

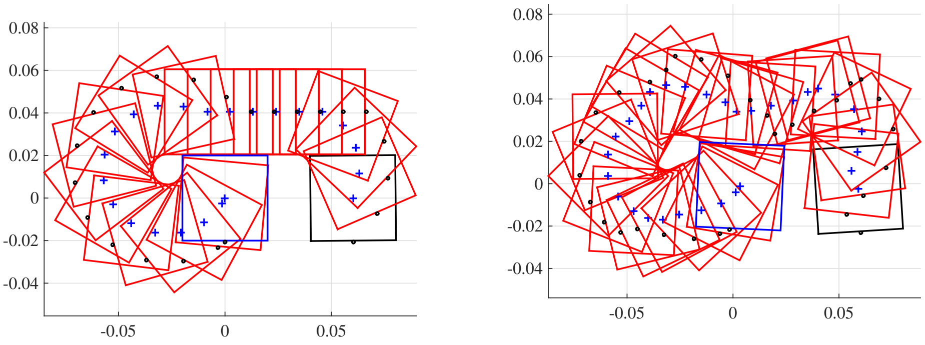
$$f_x = -w_1\cos\theta/(a\zeta br) - w_2\sin\theta/(a\zeta br) \tag{2.69}$$

We can therefore design a simple PD controller to track a planned trajectory $z_d(t)$.

$$w = \ddot{z_d} - k_p(z - z_d) - k_d(\dot{z} - \dot{z_d}). \tag{2.70}$$

This PD controller is globally exponentially stable assuming the model does not change. A robustness analysis for a changing model is beyond the scope of this paper. The manipulator velocity control input can be further determined via equations 2.37 and 2.67 to 2.69.

A simulation experiment using a high-fidelity simulator Zhou et al. (2017a) is shown in Figure 2.15b. The initial state is perturbed by -1mm in x, -2.5mm in y and 3.6 degrees in θ. The system model parameter A in equation 2.36 is perturbed from $\begin{bmatrix} 1.0537 & 0 & 0 \\ 0 & 1.0537 & 0 \\ 0 & 0 & 1.5087 \end{bmatrix}$ to $\begin{bmatrix} 1.0719 & -0.0177 & -0.1782 \\ -0.0177 & 1.0417 & 0.1599 \\ -0.1782 & 0.1599 & 1.5104 \end{bmatrix}$. The gain for position error term is $k_p = [2.0, 0.5]$ and the gain for velocity error term is $k_v = [0.1, 0.05]$. The controller runs at 60Hz for 30 seconds and the final pose error is 0.0034mm in x, 0.0012mm in y and 2.55 degrees in θ. The ABB robot is currently not suitable for closed-loop control due to low position control input frequency. In the future, we will conduct robotic experiments with recently released externally guided motion package ABB (2017b).



(a) Planned reference trajectory.

(b) Trajectory tracking with dynamic feedback linearization.

2.3.3.2 Open Loop Stabilization With Kinematic Constraints

Lynch and Mason Lynch and Mason (1996) showed that a two-points push action against an edge of the object can be stable such that the object will remain attached to the pusher without slipping or breaking contact, despite the presence of uncertainty.

This can be seen as natural mechanical feedback that tolerates model uncertainty. The object will follow a body twist motion $\mathbf{V}$ as long as the corresponding frictional wrench $\mathbf{F}$ is inside the composite wrench cone $F_c = \mathscr{K}(F_c^1, F_c^2)$ formed by the two wrench cones F_c^1, F_c^2 at the contact points, i.e., $\exists \mathbf{F}$ such that $\nabla H(\mathbf{F}) = \mathbf{V}$ and $\mathbf{F} \in \mathbf{F_c}$. The span of the composite wrench cone provides redundancy to balance uncertain frictional wrench between the object and the supporting surface.

Throughout our experiments, we use the mid point of the two points as a virtual contact point and the average normal as the contact normal to plan reference trajectory. Perception uncertainty is not addressed for this form of mechanical feedback although a sequence of designed open-loop translational pushes can reduce the initial perception uncertainty Brost (1988).

2.3.4 Experiments

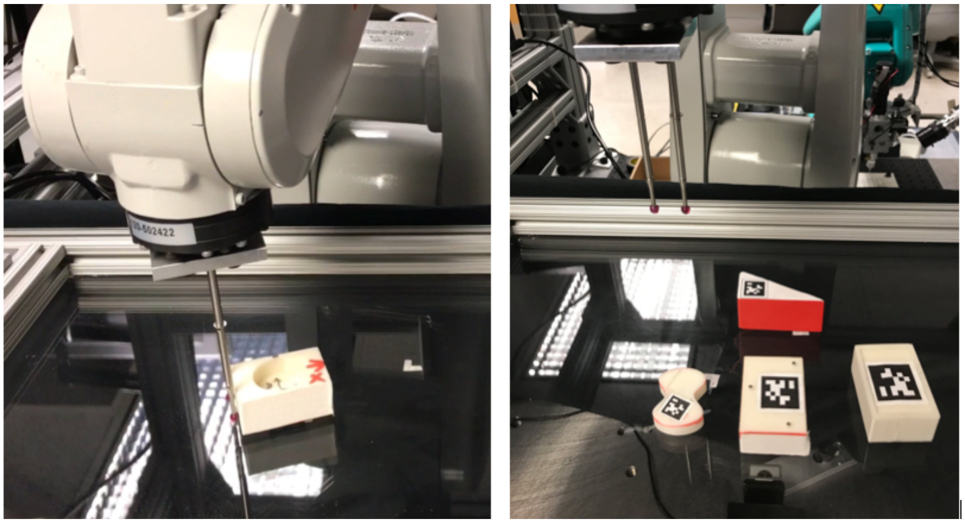


Figure 2.16: Experimental setup.

2.3.4.1 Pushing using Multiple Actions

A pushing point in the body frame defines an action associated with a Dubins car reduction. If we allow switching between multiple pushing points (actions), the object can be moved faster to the goal state. It is also natural to specify a switching cost between actions. This section presents a simple planner that gives the near-optimal path for a given query initial pose to the goal pose at the origin. We first construct a graph using the following steps:

1. Sample SE(2) poses within the boundary as graph nodes.
2. For each node, split into k copies tagged with action id, where k equals the number of actions.
3. Run Dijkstra's algorithm Dijkstra (1959) and add switching cost to the edge weight if the two nodes are tagged with different action ids.

The graph is organized as a tree structure whose paths to the goal are the shortest path subject to the sampling resolution. Then for any new query pose, we treat it as a new node to connect to the goal through either direct connection to the goal using one action or paths in the graph using multiple actions.

The experimental setup is shown in Figure 2.16. We use the ABB-120 robot mounted with a two-points pusher. The object bottoms are atached with AprilTags Olson (2011). The supporting surface is a transparent acrylic table with a camera underneath to acquire the initial and final poses. We use four objects with different pressure distributions, material and shapes. Trajectories are generated using the mid point of the two points as a single point pusher and executed open-loop. Each object is given three or four pushing points (actions). The triangular object has actions of asymmetric push point. Three different initial locations that require difficult maneuvers are chosen for each object with the same target location such that the local frame exactly aligns with table frame at the center. Each initial condition is executed five times. Trajectories generated from experimental logs are shown in Figure 2.17 to 2.20. The object initial poses (in sequence) for each action are filled with red, purple and blue colors. The final pose is filled with black color. The average error is within 1.67mm in translation and 0.5 degrees in orientation over the 60 experiments[7].

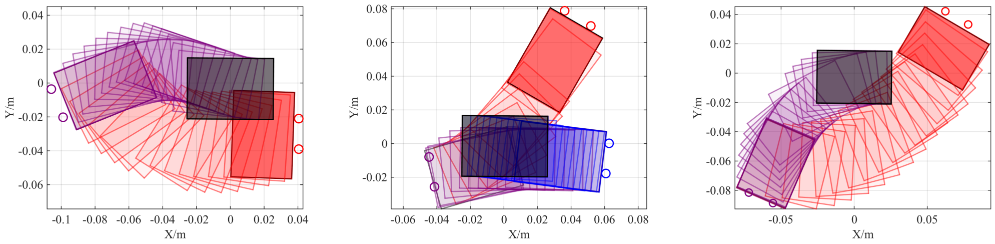


Figure 2.17: Rectangle with three-point pressure. The average error (mm, mm, degree) with 95% confidence interval from left to right are $[0.03 \pm 0.02, -3.19 \pm 0.29, 0.53 \pm 0.11], [0.50 \pm 0.13, -0.96 \pm 0.6, -0.48 \pm 0.61], [-0.23 \pm 0.11, -4.17 \pm 0.87, -1.29 \pm 0.5]$.

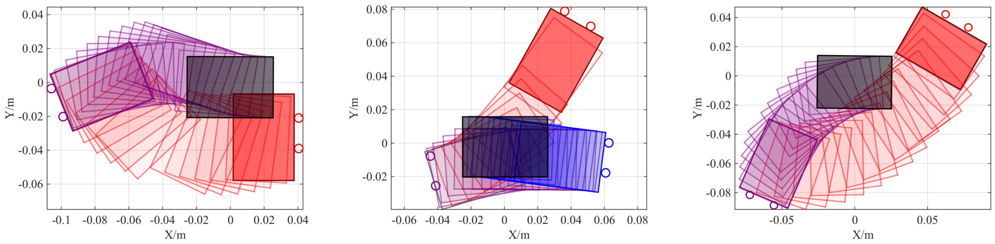


Figure 2.18: Rectangle with boundary pressure. The average error (mm, mm, degree) with 95% confidence interval from left to right are $[-0.26 \pm 0.12, -3.31 \pm 0.75, -0.46 \pm 0.19], [0.42 \pm 0.12, -1.49 \pm 1.63, -0.14 \pm 0.27], [-0.27 \pm 0.21, -4.56 \pm 0.43, -0.93 \pm 0.76]$.

2.3.4.2 Pushing Among Obstacles

The proposed reduction to Dubins curve benefits randomized motion planners since the two point boundary value problem can be solved exactly via the reduction, i.e., the steering is exact. We use a RRT LaValle and Kuffner Jr (2001) planner to generate a collision free pushing path shown in Figure 2.21 and 2.22. The triangular object and the two-points pusher are not allowed to touch the red obstacle nor the blue boundary of the map. The goal is to align the center of the triangle with the red point in an up-right orientation.

[7] All 60 runs videos are available at https://www.dropbox.com/sh/2t6cwqwv3w95iji/AABLHdlnRhSQzHKhcmg2zOT4a?dl=0.

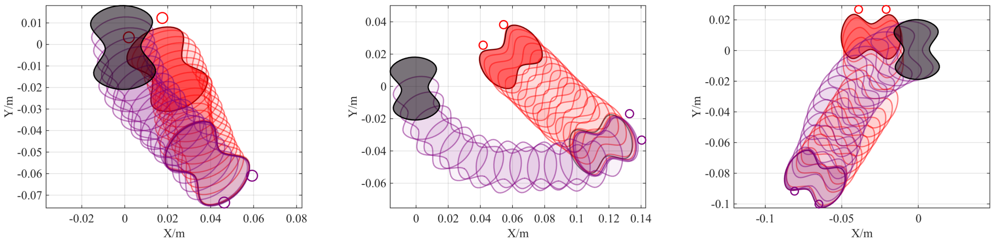


Figure 2.19: The butterfly object with boundary pressure. The average error (mm, mm, degree) with 95% confidence interval from left to right are $[-0.69 \pm 0.17, -1.46 \pm 0.06, 4.40 \pm 1.24], [-0.65 \pm 0.17, -1.38 \pm 0.07, 5.89 \pm 2.37], [-0.96 \pm 0.08, -0.09 \pm 0.73, 0.83 \pm 1.00]$.

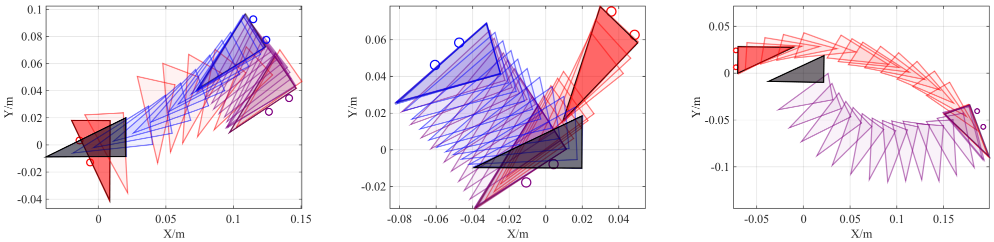


Figure 2.20: Triangle with uniform pressure. The final error (mm, mm, degree) with 95% confidence interval are $[0.64 \pm 0.05, 1.04 \pm 0.63, 0.11 \pm 0.31], [0.11 \pm 0.65, -0.50 \pm 0.30, -0.42 \pm 0.44], [2.34 \pm 0.23, 0.12 \pm 0.06, -1.06 \pm 0.42]$.

2.4 Uncertainty Reduction via Sequental Grasps

Equipped with an efficient kinematic contact model, we address the following question in this section: how can a robot design a sequence of grasping actions that will succeed despite the presence of bounded state uncertainty and an inherently stochastic system? We propose a probabilistic algorithm that generates both sensored and sensorless plans such that the post-action object pose is uniquely known (subject to symmetry). The sensored plans assume encoder feedback that gives a geometric partition of post-grasp configuration space based on contact constraints. A planning tree is generated by interleaving open-loop plan search and feedback state estimation with particle filtering. To speed up planning, we use learned approximate forward motion models, sensor models, and collision detectors.

Prior works Brost (1988); Goldberg and Mason (1990); Goldberg (1993) on planning sequential actions for uncertainty reduction focus on strategies. However, many unrealistic assumptions

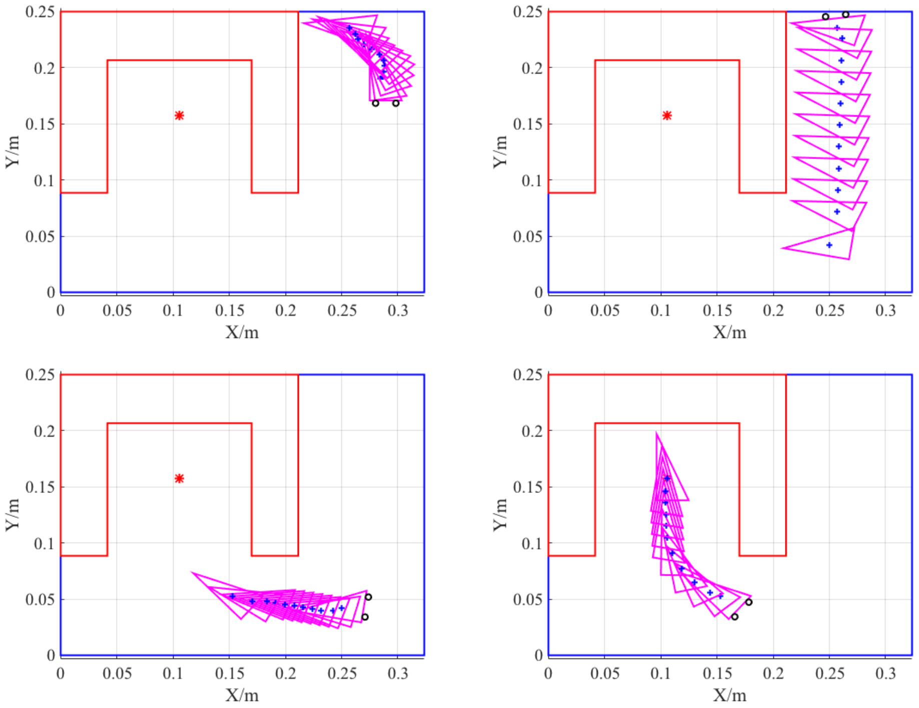


Figure 2.21: Planned pushing actions among obstacles using RRT with exact steering.

are made in order to reduce the state space and create finite discrete transitions, including infinitely long fingers approaching the object from infinitely far away. We show that with simple encoder feedback and a good model for open-loop action search, we can generate faster and more realistic strategies.

2.4.1 Configuration Space Partition and Fast Approximate Models

2.4.1.1 Post-grasp Configuration Space Partition

Contact constraints parametrized by the post-grasp distance between the fingers naturally partition the configuration space to lower dimensional subspaces. Figure 2.24 shows the partitions of the object poses based on different grasping outcomes indicated by the post-grasp finger widths. The results are generated by first simulating trajectory roll-outs with different initial poses and then performing *k*-means clustering of the final finger distances where *k* equals 4. Note that the gripper widths within the same cluster has nonzero variance which matches the nonnegligible

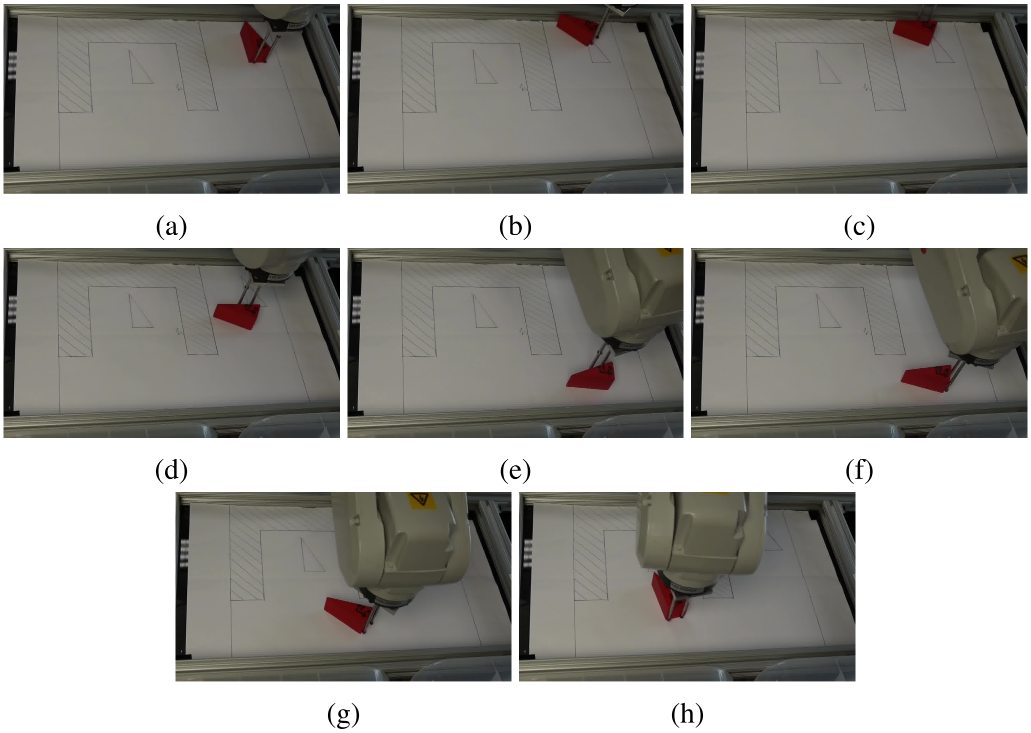


Figure 2.22: Snapshots of the robot executing the plan.

real world sensor noise.

2.4.1.2 Fast approximate models

Our planning algorithm requires three components: 1) a forward motion model that maps an initial object pose to a final object pose with respect to the hand frame; 2) a sensor model that maps a final object pose to the expected post-grasp finger width; 3) A collision detector that checks a given initial object pose with respect to the hand frame. Although they are all available through trajectory roll-out simulations, planning with particle based belief representation needs to be orders of magnitude faster.

We need to compute a forward model of planar grasping. A small deviation in initial pose can be the difference between a successful grasp and a missed one. By probabilistically modeling a planar grasp, we can capture this behavior. We will use kernel conditional density estimation (KCDE) Rosenblatt (1969) so that we can capture the inherent non-linear, multimodal nature of

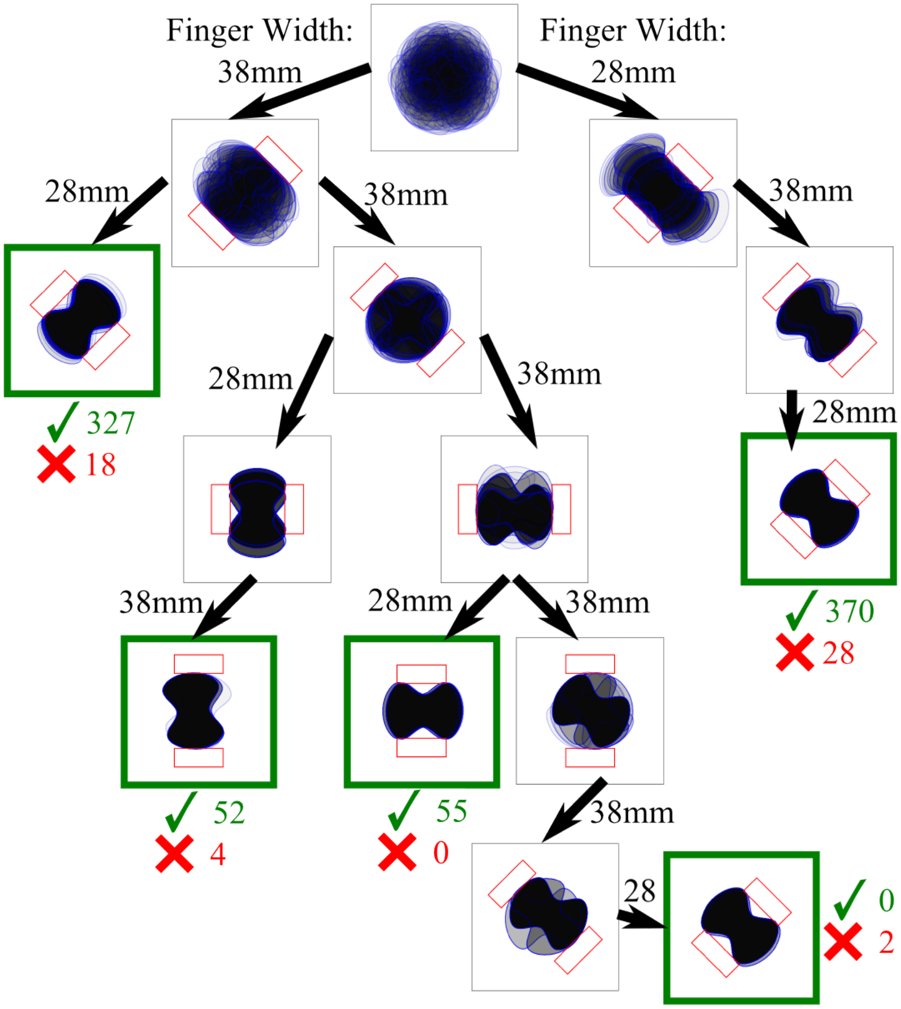
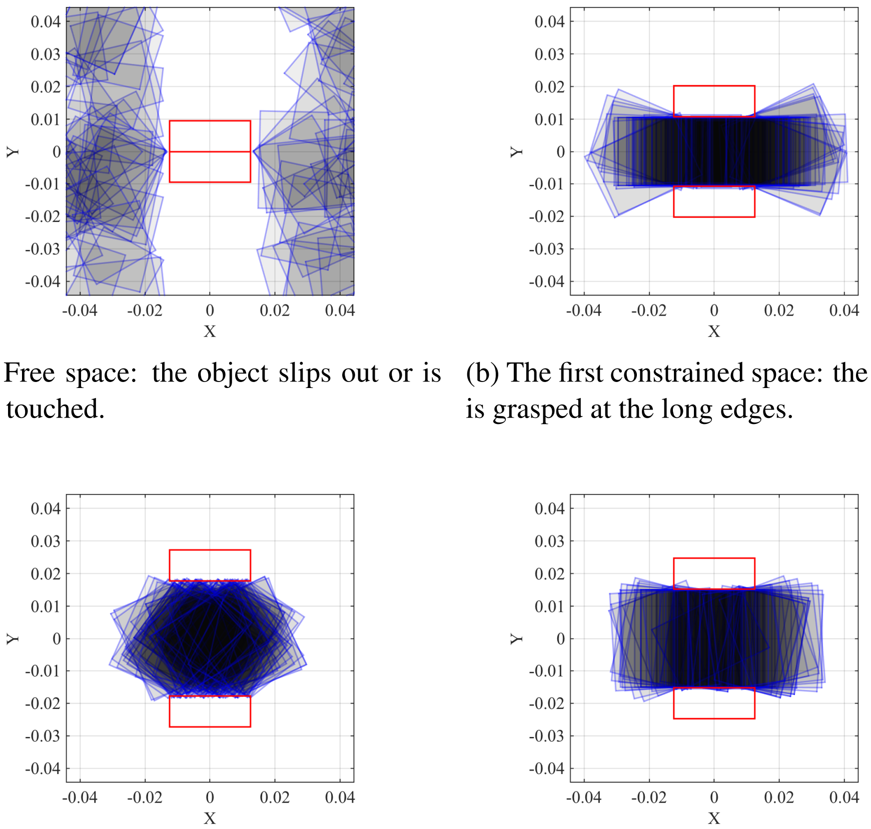


Figure 2.23: A tree-based plan generated by our algorithm for a butterfly-shaped object where the centers are uniformly distributed in a circle of radius 10mm and the frame angles are uniformly distributed from -90 to 90 degrees. In each node, red boxes indicate the expected final gripper pose and filled rectangles with blue boundary correspond to particle-based belief distribution of the object poses where density is proportional the darkness. The total number of action choices equals 4. A grasping action is applied at each non-terminal node which branches out child nodes depending on the possible post-grasp distances between fingers. At the terminal nodes (boxed in green), the robot predicts the average pose, with counts of successful and unsuccessful predictions for a total of 856 experimental grasp trials.

the distribution. Let $\mathbf{v} = [x, y, \rho\theta]^T$ be the normalized pose with respect to the hand frame where ρ is the radius of gyration (or any meaningful characteristic length) of the object. Denote the



(a) Free space: the object slips out or is not touched.

(b) The first constrained space: the object is grasped at the long edges.

(c) The second constrained space: the object is jammed between the diagonal vertices.

(d) The third constrained space: the object is grasped at the short edges.

Figure 2.24: Partition of the configuration space of a rectangle (30×21 mm) based on the post-grasp distance between the fingers. Red boxes correspond to the average gripper pose.

RBF kernel by

$$K_h(\mathbf{v}_1, \mathbf{v}_2) = \frac{1}{\eta} \exp\left(-\|\mathbf{v}_1 - \mathbf{v}_2\|^2/(2h^2)\right),$$

where η is a normalization term and h is the bandwidth. Suppose we have collected a data set $\{\mathbf{v}^i, \mathbf{v}^f\}$ of N initial and final object poses. The conditional probability of a final normalized pose $\mathbf{v}^{f*}$ to occur given an initial normalized pose $\mathbf{v}^{f*}$ is given by:

$$P(\mathbf{v}^{f*}|\mathbf{v}^{i*}) = \frac{\sum_j^N K_{h_1}(\mathbf{v}^{i*}, \mathbf{v}_j^i) K_{h_2}(\mathbf{v}^{f*}, \mathbf{v}_j^f)}{\sum_j^N K_{h_1}(\mathbf{v}^{i*}, \mathbf{v}_j^i)}. \tag{2.71}$$

The bandwidths h_1 and h_2 throughout the experiments are set as 0.5. To sample from this distribution, note that given an initial pose $\mathbf{v}^{i*}$, Equation 2.71 becomes a mixture of Gaussians:

$$P(\mathbf{v}^{f*}) = \sum_{j}^{N} w_j K_{h_2}(\mathbf{v}^{f*}.\mathbf{v}_j^f) \tag{2.72}$$

We can pick a center $\mathbf{v}_j^f$ randomly according to the weights w_j, and then sample from that chosen Gaussian. Further speed-up is achieved by only using k nearest neighbors for prediction. On a 2.4GHz i5 core, it takes about 1 second to roll out 5000 particles for a KCDE model trained with 10000 data points and using 400 nearest neighbors (sped up by kd-tree) for prediction, compared with 0.5 to 3 seconds to roll out 1 particle using the simulation.

Note that only one forward motion model is needed since we always transform the post-action object poses to the next action hand frame.

We use a regression tree to predict the post-grasp finger distance given a final pose. The collision detector is trained using a binary classification tree. Both trees achieve very fast and accurate performance.

2.4.1.3 Belief Update With Subspace Projection

The free space and constrained subspaces are represented by storing corresponding final poses through many trajectory roll-outs. These roll-outs start with randomly sampled initial poses from a large enough uncertainty area which covers all the possible poses during the grasping sequence.

To perform belief updates after rolling out the KCDE forward motion, we use the sensor model to predict the post-grasp gripper distance for each particle. For particles that belong to the same subspace based on the predicted observation, we perform projection onto the subspace using a nearest neighbor association.

2.4.2 Tree-based Planner

The planning algorithm is summarized in Algorithm 1. It first constructs the root node $\mathbf{C_0}$, and its associated belief $\mathbf{C_0}.X$, as the set of particles sampled from the initial uncertainty region. ExpandTree($\mathbf{C_0}$) will construct the tree.

For a given set of poses, *SingletonRatio* computes the ratio of poses within given tolerance values $\delta = [\delta_x, \delta_y, \delta_\theta]$ with respect to the average pose as a singleton (i.e. a unique, known final pose). To compute the average pose of a given set of poses: we mapped the poses $[x, y, \theta]$ to the augmented space $[x, y, \cos(2\theta), \sin(2\theta)]$ to take the average and then map back to the original space.

```
Input: current node C, singleton tolerance values $\delta$ and threshold ratio $r_s$
Output: Planning Tree
if SingletonRatio(C.X, $\delta$) $> r_s$ then
    MarkTerminalNode(C);
    Return;
else
    $[X, Z, A]$ = OpenLoopSearch(C.$X$, $d$, $\delta$) ;
    C.action = A;
    foreach observation subset $X_i, Z_i$ do
        $X_i$ = BeliefUpdateProjection($X_i, Z_i$);
        N = CreateNode($X_i$, C.$Z \oplus Z_i$);
        AddToTree(N) ;
        ExpandTree(N, $\delta$, $r_s$);
    end
end
```

Algorithm 1: ExpandTree

The sensorless subroutine *OpenLoopSearch* uses a breadth-first search to find the optimal sequence of actions that maximizes the ratio of singletons. Algorithm 2 describes the expansion procedure inside *OpenLoopSearch*. We use a pruning criteria that avoids expansion if the singleton ratio decreases by (1 - α) compared with its parent. α is set as 0.75 throughout the experiments. The first action in the sequence returned by the open loop search will be applied, followed by sensor updates and node expansions. Note that we can also execute the entire open

```
Input: Node element $E$, singleton tolerance values $\delta$, queue $Q$
foreach action a do
    $X_t$ = TransformToHandFrame($E.X$, $a$);
    if CollisionFree-DecisionTree($X_t$) then
        $X_p$ = ForwardMotionModel-KCDE($X_t$);
        $Z_f$ = SensorModel-RegressionTree($X_p$);
        $X_f$ = TransformToWorldFrame($X_p$);
        $r$ = SingletonRatio($X_f, \delta$);
        if $r > \alpha E.r$ then
            AddQueueElement($Q$, $X_f$, $Z_f$, $E.A \oplus a$, $r$);
        end
    end
end
```

Algorithm 2: OpenLoopSearchBFSExpand

loop action sequence and perform sensor update afterwards. At the extreme is complete sensorless planning where the open loop planner keep searching until the best singleton ratio exceeds threshold r_s or the time exceeds the limit. We have found that feasible plans are much easier to generate with feedback whereas completely open loop plans cannot find a feasible solution unless given a much larger action space often rendering computation intractable. Additionally, the length of open loop plans (if they exist) is usually much longer than the average length of grasping sequence with feedback.

The steps of the tree generation is summarized as follows:

1. Sample initial poses from a large enough uncertainty and do trajectory roll out simulations for a hand squeezing action.
2. Perform K-means clustering on the post-grasp finger distances and construct subspace partitions. Train approximate motion models, sensor models and collision detectors.
3. Sample initial poses from a query initial uncertainty and construct the root node.
4. Call Algorithm ExpandTree to construct an offline sensored tree plan. [8]

Note that different query initial uncertainty for the same object can share the approximate models

[8] Call Algorithm OpenLoopSearch only to construct a pure open loop sequence plan.

shape	butterfly	butterfly	rectangle	rectangle	rectangle	rectangle	rectangle
pressure	boundary	boundary	boundary	boundary	points	boundary	boundary
uncertainty size	15mm	10mm	12.75mm	12.75mm	25mm	25mm	25mm
gripper-object materials	felt-abs	felt-abs	foam-abs	steel-abs	felt-paper	foam-abs	steel-abs
grasp trials	327	856	682	968	193	202	95
success rate	0.835	0.939	0.972	0.872	0.917	0.896	0.768

Table 2.5: Experimental results of grasping with different combinations of object shapes, frictional materials, supporting pressure distributions and uncertainty sizes. The angular uncertainty is uniformly distributed from -90 to 90 degrees.

so the first two time-consuming steps only need to be done once.

2.4.3 Experiments

The algorithm is implemented in Matlab. Example trees in Figure 2.25 (large uncertainty) and Figure 2.23 (small uncertainty) take less than 2 hours and 5 minutes to generate offline, respectively. For all trees in the experiments, we first generate simulation data through 9000 trajectory roll-outs with initial poses sampled from a region 3 times the size of uncertainty. Convex quadratic forms of $H(\mathbf{F})$ are fitted assuming pressure distributions around the boundary of the objects. The uncertain coefficient of friction between object and gripper is uniformly sampled from 0.4 to 0.6. The gripper fingers are modeled as 27.5mm $\times$ 9.5mm rectangles. We then use the simulation data to train KCDE-based forward motion models, sensor models and collision detectors. To generate the tree plan, we uniformly sample 6000 particles from the initial uncertainty region. The tolerance values are within ± 2.5 mm in x and y and ± 5 degrees in angle. Additionally we construct a kd tree and use 400 nearest neighbors for KCDE-based forward motion model prediction.

2.4.3.1 Setup

The experimental setup for planar grasping is shown in Figure 2.26a. We use a 6 degree-of-freedom ABB industrial robot ABB (2017a). We attach to the arm a Robotiq C-85 2-fingered parallel jaw gripper ROBOTIQ (2017). The two fingers open and close at the same time. The

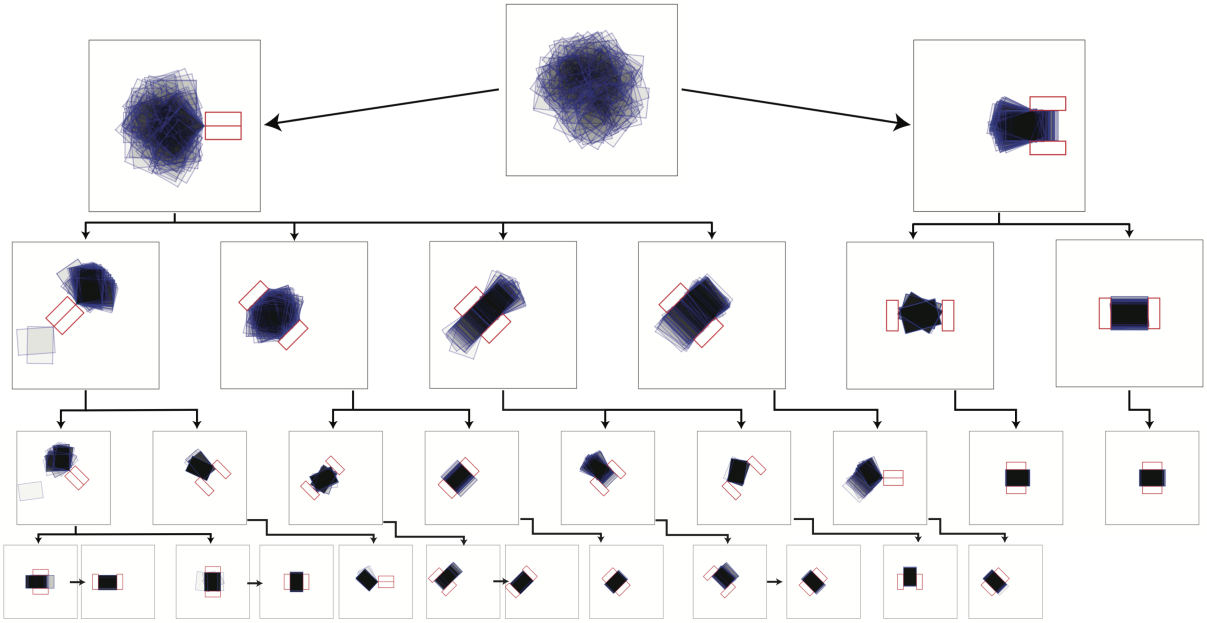
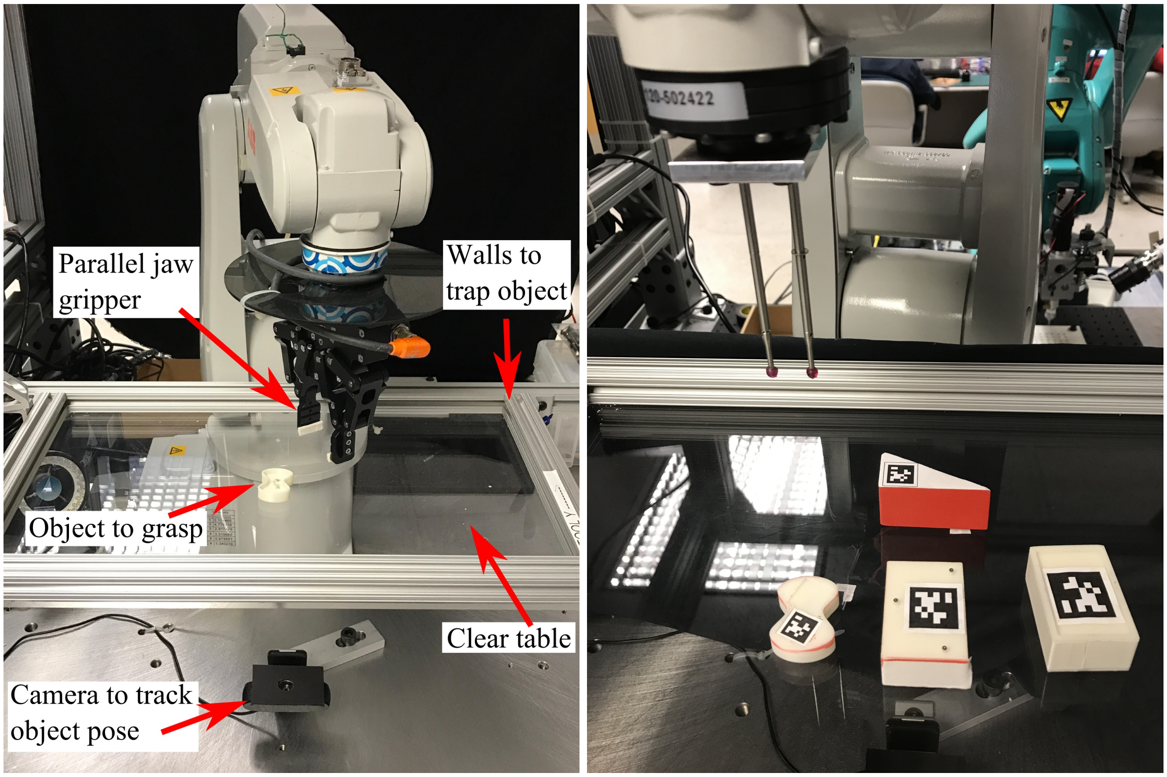


Figure 2.25: A tree-based plan generated by our algorithm for a rectangular object (30mm * 21mm) where the centers are uniformly distributed in a circle of radius 25mm and the frame angles are uniformly distributed from -90 to 90 degrees. The total number of action choices equals 20.

robot wrist is synchoronized with the finger motion to maintain the tip at fixed height throughout the grasp. The vision system consists of a Logitech c930e webcam looking through a clear acrylic table. We attach an an AprilTag Olson (2011) to the bottom of the object, and its pose is found by the camera and projected onto the table. The camera's intrinsics are calibrated using the standard ROS camera calibration tool, and the camera's frame is calibrated to the robot by having the robot move a larger AprilTag around. Then, a further calibration is done by having the robot grasp the object and move it around on the surface of the table. This allows us to obtain a 0.3mm object pose estimation accuracy near the center of the table, and a 1.5mm accuracy near the edge of the table. If the object or robot during a grasp would be too close to the edge of the table, the robot drags the object back to the center before starting its next grasp.



(a) Grasping platform showing ABB arm, Robotiq gripper, object to grasp, transparent table, and underneath camera for verification.

(b) 3D printed trial objects: (top) rectangular with boundary pressure distribution, (middle) rectangular (glued with hard paper) with discrete point pressure distribution, (bottom) butterfly with boundary pressure distribution.

Figure 2.26: Experimental Setup

2.4.3.2 Results Analysis

During execution, we read the finger distance from the Robotiq hand encoders and descend to the child node with the closest expected observation. The robot stops execution when terminal node is reached and reads the ground truth object pose value from the vision system. Table 2.5 shows the experimental results of over 3000 experiments on the robot. A grasp sequence is considered successful if the combined metric $d(q_1,q_2) = \sqrt{(x_1-x_2)^2+(y_1-y_2)^2} + \rho \cdot \min(|\theta_1-\theta_2|, 2\pi - |\theta_1-\theta_2|)$ between the predicted pose $q_1 = [x_1, y_1, \theta_1]$ at the terminal node and the ground truth pose $q_2 = [x_2, y_2, \theta_2]$ from the camera is smaller than 3mm. The uncertainty for orientation is uniformly distributed from -90 to 90 degrees. The centers of the objects are uniformly distributed in a circle of radius specified in the third row. For low uncertainty settings (columns 2 to 5), the action space consists of 4 different angles with one fixed hand frame center. For high uncertainty settings (columns 6 to 8), the action space consists of 4 different angles with 5 different hand frame centers along a line. The best 4 step open loop plans with the same action choices (increasing search depth does not improve performance) for the high friction low uncertainty settings (columns 1 to 3) do not exceed 70 percent success due to unrecognizable repeated jamming.

The most frequent failures are due to unexpected dynamic behavior when objects are jammed in an unstable equilibrium: the large force applied by the stiff gripper on the light objects cause "fly away" phenomenon not captured by the quasi-static simulation, particularly for the case of low friction steel gripper material for large uncertainty in column 8 of Table 2.5. Other failure patterns include missing multi-modal patterns during node expansion and cascading small amount of objects movement when the gripper looses each grasp, both of which can cause unexpected collision. In our future, we plan to improve the models by incorporating experimental data which will generate more stable plans.

2.4.4 Summary

We present a tree-based probabilistic planning algorithm for planar grasping under uncertainty capable of generating action sequences with or without sensor feedback. To improve the planning speed, the forward motion model is approximated by kernel conditional density estimation. Regression and classification trees are trained to approximate the sensor model and the collision detector.

Learning from experimental data is crucial to capture high variance input-output mappings that deviate from the underlying physics model assumptions, e.g., initial object poses that can "fly away" later during the grasp action. To avoid these issues in the future, the planner could prioritize more stable actions while making uncertainty reduction progress. We have not addressed the optimality of the plans. The trees generated are not optimal: the sequences can be longer than necessary and some terminal nodes are not visited. The system would also benefit from automatic tuning of the bandwidth for the KCDE-based forward motion model. The plan suffers from slow convergence speed and generates redundant actions if the bandwidth is too large, whereas low probability outcomes are missed during node expansion if the bandwidth is too small. Using a larger set of manipulation actions including pure pushing and push-grasps can significantly increase the planner's capability to deal with more complicated object geometry and larger uncertainty region. We plan to use Monte Carlo Tree Search techniques Kocsis and Szepesvári (2006) to deal with the computational challenges brought by a larger branching factor.

Although our planner is capable of generating open-loop plans with fewer assumptions compared with existing literature, we note that planning is easier, faster, and more robust with even the simplest feedback – finger encoder readings (available for most off-the-shelf robot hands). This sets up a promising framework of integrating basic proprioceptive feedback on an industrial robot arm and gripper that produces more robust plans without the need for high cost external sensors Canny and Goldberg (1994).

Chapter 3

Force Controlled Pulling and Grasping with External Contact

This chapter presents some extensions beyond position-controlled planar manipulation, where we demonstrate grasping a large-size book that resembles how a human strategy. The robot needs to perform a sequence of non-prehensile contact-rich actions to finish the task while obeying kinematic and hardware limitations. Such actions include reorienting the book by force-controlled twisting and pulling, and push-grasping in the gravity plane treating the table as a third supporting finger.

3.1 Case Study: Book Grasping

Grasping a book with both width and height dimensions larger than the gripper's maximum finger openning distance requires moving the book to the edge of the table and picking from the side. We decompose the involved action sequences as follows:

1. Estimate the pose of the book from point cloud data.
2. Move the gripper to a position above and align the gripper orientation with the book. Executes a guarded move to ensure contact with the book by pressing with required normal

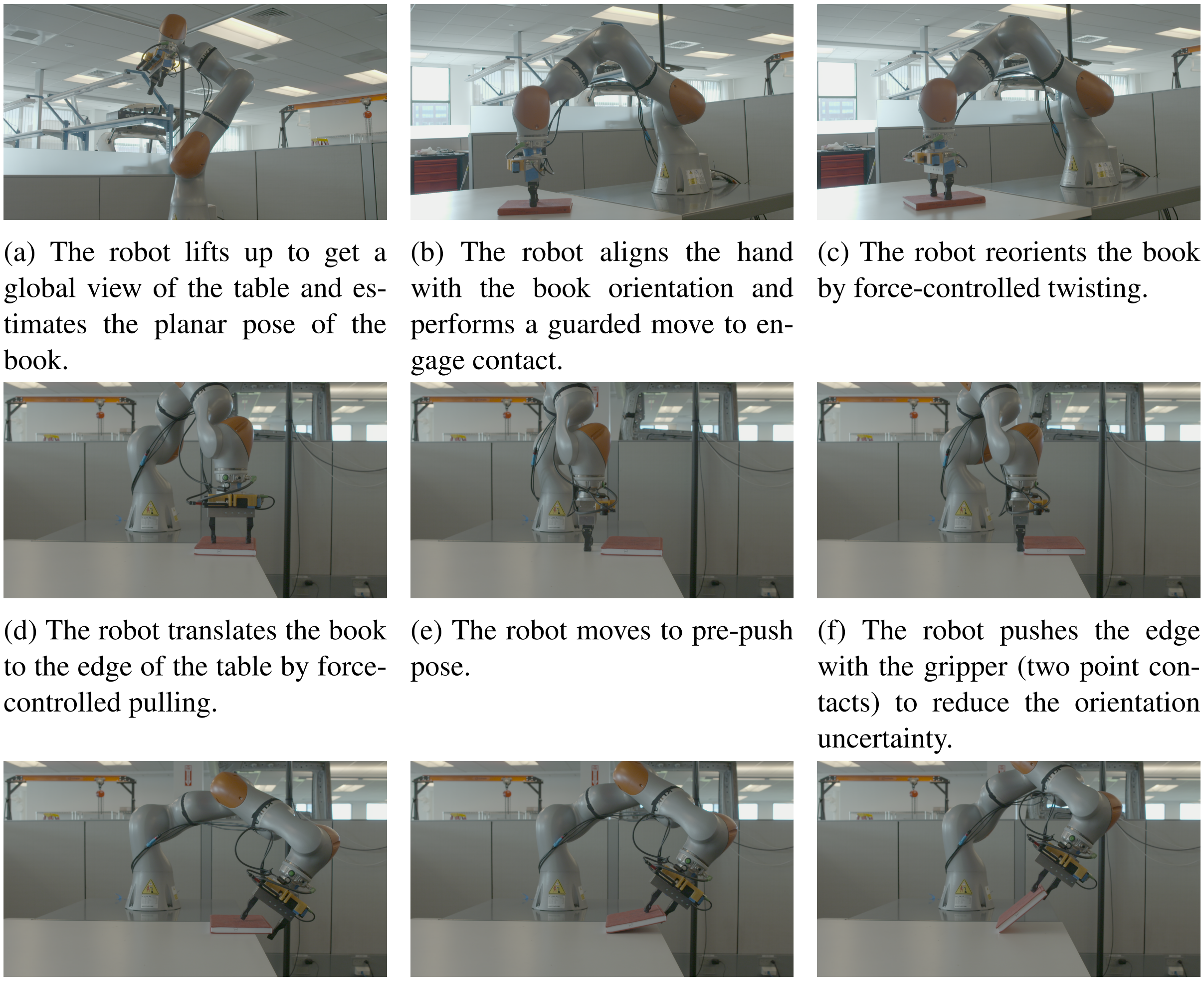
force.

3. Reorient the book and move it to the edge of the table using force-controlled pulling.

4. Perform straightline two-point pushing to reduce orientation uncertainty.

5. Push-grasp in the gravity plane while obeying the robot's workspace limit.

Figure 3.1 shows the several stages of the entire manipulation process.



(a) The robot lifts up to get a global view of the table and estimates the planar pose of the book.

(b) The robot aligns the hand with the book orientation and performs a guarded move to engage contact.

(c) The robot reorients the book by force-controlled twisting.

(d) The robot translates the book to the edge of the table by force-controlled pulling.

(e) The robot moves to pre-push pose.

(f) The robot pushes the edge with the gripper (two point contacts) to reduce the orientation uncertainty.

(g) The robot reaches to a pre-grasp position that is about to reach its kinematic limit.

(h) The robot performs a grasping with synchronized wrist motion while using the table as an external supporting contact.

(i) The robot grasps the book in hand and lifts it up from the table.

Figure 3.1: Book grasping demo

3.1.1 Pose Estimation from RGBD Point Cloud

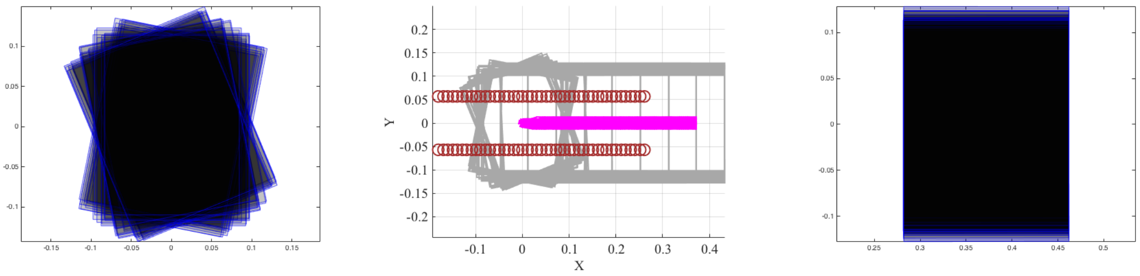
Before each execution, we lift up the wrist mount camera (Asus Xtion Pro) such that its field of view covers all the reachable space on the table by the robot. The estimation steps are summarized as follows:

1. Use Ransac algorithm Fischler and Bolles (1987) to fit the table plane and subtract it from the point cloud.
2. Find a bounding box that covers the closest 95% of the points to the centroid of the remaining point cloud.
3. Use the principle component vectors as the book frame with respect to the robot base frame.

Due to imperfect calibration, sensing noise and changing ambient light conditions, the pose estimation errors are about 1 centimeter in translation and 5 degrees in orientation.

3.1.2 Force-controlled Twisting and Pulling

The Kuka Iiwa robot is under joint impedance control mode. The position gains for each joint is $[2000, 1500, 1500, 1500, 1000, 500, 500]$ (Nm per radian). The damping ratio is 1 (critically damped). During the twisting and pulling, we maintain a normal force of 40N and the resultant friction force due to sliding is compensated by the position gain as the robot needs to move in the horizontal plane. During the twisting motion as shown in Figure 3.1c, slip can occur between the robot fingers and the book, which will introduce around 5 to 10 degrees of uncertainty. During the pulling motion, the contacts between the fingers and the book usually stick, and hence will not introduce much additional uncertainty.



(a) The initial book poses before the pushing action.

(b) Sample trajectories for different initial poses.

(c) The final book poses after the pushing action.

Figure 3.2: The pushing action generates a mechanical funnel robust to initial uncertainty.

3.1.3 Pushing Funnels

The uncertainty induced in initial perception and the slip during the twisting motion needs to be removed before the grasp. We employ a simple two point straightline pushing. Figure 3.2 shows the simulation results for the final aligning by pushing steps from Figure 3.1g to Figure 3.1i. The stochasticity modeling in the pressure distribution is similar to section 2.2.3 with degree of freedom equals 200. The book is modeled as a rectangle of 18 centimeters in width and 23 centimeters in height. The opening width of the gripper width is 10 centimeters. The initial three-dimensional pose uncertainty (x, y, θ) in Figure 3.2a is reduced to the one-dimensional uncertainty (y) in Figure 3.2c. Figure 3.2b shows some sample trajectories that correspond to different initial poses, where the book contours are shown as grey boxes, the two fingers are shown as red circles and the center of masses are shown as magenta dots.

3.1.4 Push-Grasp in the Gravity Plane

In this section, we analyze the kinematics for the grasping process as shown in Figure 3.1g to Figure 3.1i. Due to limited workspace, the Kuka Iiwa robot cannot reach into a horizontal grasp pose. Hence we embrace the idea of exploiting external contacts where the table can act as a third finger to support the book while the robot executes a push-grasp in the gravity plane to re-orient the book in hand. Figure 3.3 and 3.4 illustrate the contact conditions and kinematic reduction of the grasping process.

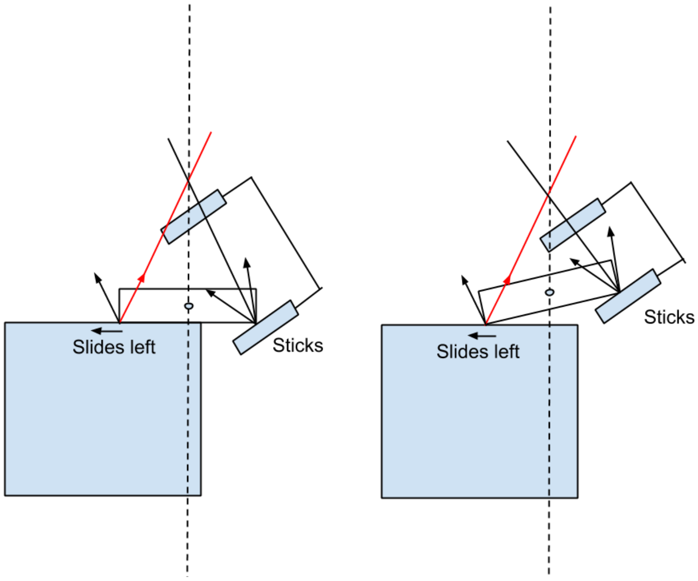


Figure 3.3: Illustration of the quasi-static balance. The book slides inward and hence the friction force is on the right edge of the friction cone. The contact between the gripper and the book sticks and the friction force inside the friction cone. The two friction forces and the line of gravity intersects at a point and hence quasi-static balance can be achieved.

The following equations describe the kinematics of the book-finger (slider-pin joint) system:

$$x_A(t) = x_A(0) - vt\cos\alpha \tag{3.1}$$

$$y_A(t) = y_A(0) - vt\sin\alpha \tag{3.2}$$

$$x_B(t) = x_A(t) - L\cos\theta(t) \tag{3.3}$$

$$y_B(t) = y_A(t) - L\sin\theta(t) = 0 \tag{3.4}$$

The pose of the book $(x_B(t), y_B(t), \theta(t))$ follows:

$$x_B(t) = x_A(0) - vt\cos\alpha - L\cos(\arcsin((y_A(0) - vt\sin\alpha)/L)) \tag{3.5}$$

$$y_B(t) = 0 \tag{3.6}$$

$$\theta(t) = \arcsin((y_A(0) - vt\sin\alpha)/L) \tag{3.7}$$

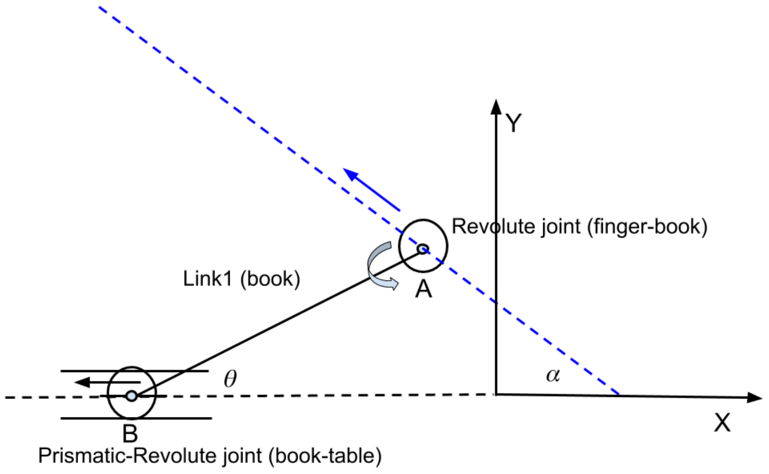

Y
Revolute joint (finger-book)
Link1 (book)
A
θ
α
X
B
Prismatic-Revolute joint (book-table)

Figure 3.4: The sticking contact is equivalent to a pin joint between the book and the finger while the sliding contact is equivalent to a prismatic joint. The pin joint is connected to the robot finger link as a floating link not shown in the figure. We command the finger to move in a straight blue line.

3.1.5 Experimental Result

https://www.youtube.com/watch?v=k_mpCreIhDE provides the video recording of executive eight success runs. For each run, a human (including the author and a random lab visitor) places the book with different initial poses on the table. The robot perceives the book and executes the sequential strategy described above.

The robot starts to fail when enough experiments are performed and the wear on the book cover due to force-controlled pulling and twisting results a decrease in the friction coefficient, and hence more slippage occurs that exceeds the funnel tolerance in the subsequent straightline two point pushing.

Chapter 4

Conclusions and Future Work

4.1 Harnessing Task Mechanics

Robot actuators can be made faster, stronger and more durable than human muscles and tendons. Accurate joint encoders, torque and current sensors are readily available to equip robots with better proprieceptive sensing than humans. Many color and depth sensors capture higher resolution images and finer three dimensional measurements than human vision. Additionally, human feedback system have much lower control frequency and higher delay than typical eletromechanical systems. Yet, humans beat robots in almost every fine manipulation tasks. Why so? I conjecture that humans achieve astounishing manipulation capabilities via intelligent exploitation of "intuitive" physics (feed-forward) models, together with a lower frequency but intelligent sensorimotor controller based on vision and touch feedback. Since such physics models are quickly learned from small observations and trials, it may suggest some strong prior knowledge built into the brains.

A combination of analytical physics modeling and data-driven techniques could enable robots to learn equally fast. This thesis partly explores such idea in the context of planar sliding manipulation. However, the limitation is also evident: building detailed models may not be tractable in the context of cluttered three dimensional world where humans effortlessly deal with. Thus, building approximate models where only a small number of free parameters that are online identified from visual data remains as important future work.

4.2 Preliminary Results on Improving Perception for Clutter

In the future, robots in the home need to recognize millions of different daily objects and pick up the right one in a highly cluttered enviroment. Rather than tackling this task, we restrict the setting to just clearing a random pile of objects from a bin or a drawer. The robot only needs to pick up objects one by one without any specific order. The task is similar to the stowing task in 2017 Amazon Robotics Challenge Zeng et al. (2018).

The perception capability to segment each object instance is crucial for manipulation in clutter. Recent advances in deep learning based segmentation systems Ren et al. (2015); Girshick (2015); Li et al. (2017); He et al. (2017) have achieved impressive results. We trained an instance segmentation system based on the Mask-RCNN architecture He et al. (2017) using only photorealistic simulation data given high-quality textured mesh models from the YCB dataset Calli et al. (2015). In this section, we demonstrate some prelimary qualitative results.

Figure 4.1 shows the pipeline for synthetic data generation. A sample of 20 YCB objects that cover a range of object geometries and textures is shown in Figure 4.2.

For each scene, we first choose a subset of objects from the database with random initial pose above the bin. Then the Bullet Coumans (2010) physics engine simulates the objects falling into the bin by gravity before settling down into a stable equilibrium state. To create possible shadows and over exposures, the point lighting is varied with respect to location, intensity for each rendering. The projective camera has a resolution of 640×480, focal length of 35mm and sensor size of 32mm. To generate images from different perspective views, the camera position is also sampled from a set of predefined viewpoints around the bin. Figure 4.3 shows some rendered sample scenes. These randomly generated synthetic scenes are then used to train a Mask-RCNN network pretrained using the resnet-50 He et al. (2016) architecture. Figure 4.4 shows some

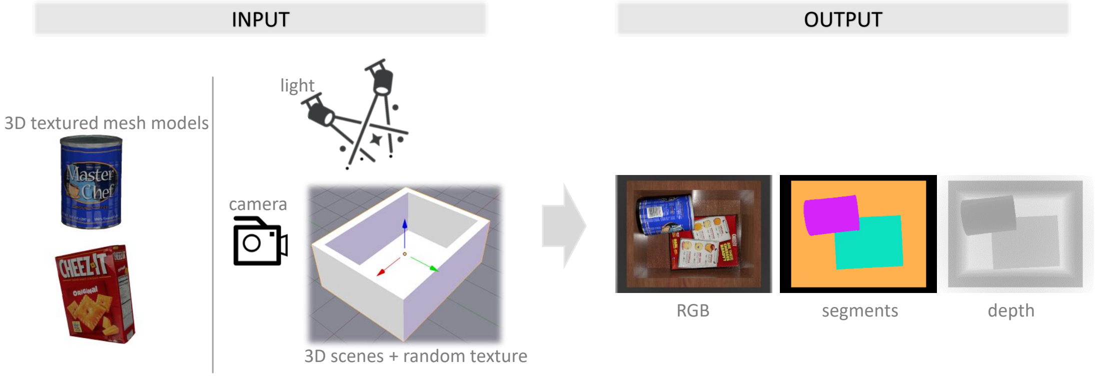


Figure 4.1: Our pipeline for generating synthetic data based on textured object models. When the physics simulation stabilizes, a frame is rendered with a textured background, specific lighting condition and camera position. These ingredients are randomly sampled to increase the diversity of the data. The output includes colored images, depth images and ground truth instance segmentation results.

qualitative results on transferring to real images collected from the wrist mount camera. We demonstrate some generation capabilities of the network by adding some novel objects in the scene including Red Bull bottles, real bananas and Colgate toothpaste boxes.

4.3 Vision Guided Compliant Manipulators

From the Darpa ARM-S challenge Bagnell et al. (2012), the Amazon picking challenge Correll et al. (2016) to home robot tasks including clearing a random pile of objects on the table and loading dishes into a dish washer, we have seen an increasing amount of efforts in tackling manipulation challenges arising in semi-structured and unstructured environments. For these sensor-rich manipulation tasks, how accurate does the manipulator needs to be? Perhaps the sub-millimeter accuracy is not as important and the extreme rigidity and high position gain common in industrial manipulators do not fit sensor-rich applications.

We need fast, compliant and reliable manipulators that are accurate enough to match the sensor accuracy but not two orders of magnitude better. Compliant manipulators can be made either intrinsically soft with springs and dampers Robotics; Pratt and Williamson (1995) or with

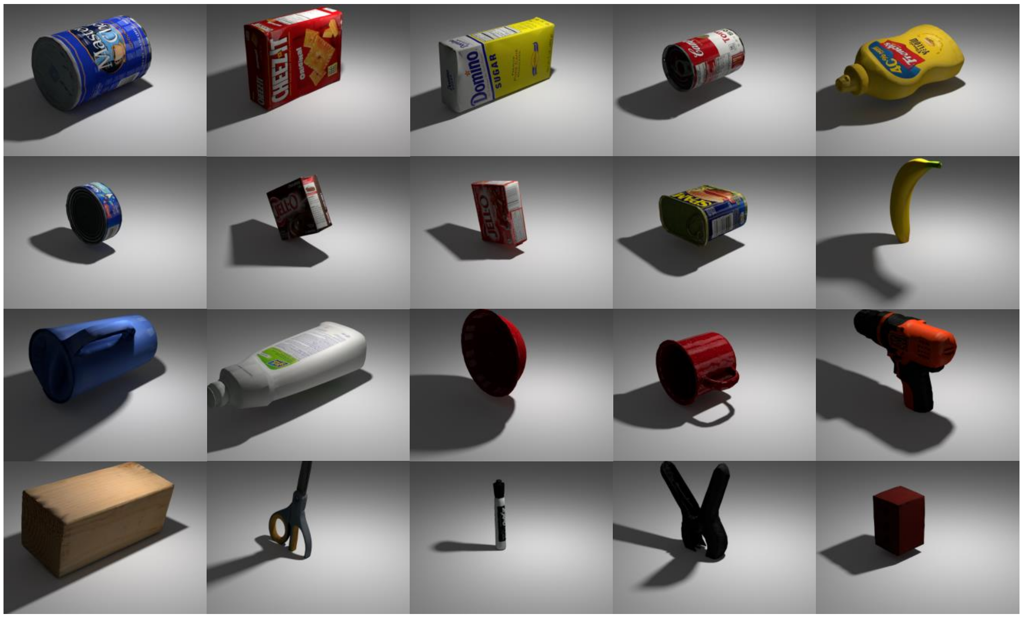


Figure 4.2: YCB object set used for our rendering process. The above 20 objects cover common shapes in manipulation scenarios. The images are rendered by Blender Blender.

high-bandwidth force control and accurate well calibrated sensors Albu-Schäffer et al. (2007). It's unclear that which one is better. For many applications, compliance in the wrist is sufficient and the rest of the arm can still operate in high-gain position mode. We can design either passive or active wrist to attach to the robot arm.

Of particular interest is putting both the arm and object pose estimation in a common fixed camera frame Klingensmith et al. (2013). We may treat a successful engagement with the object of interest as servoing for a target image. In many practical applications where operational speeds are critical, the motion control scheme can start with open-loop motions to reach roughly correct regions and followed by visual servoing to adjust for the positioning error.

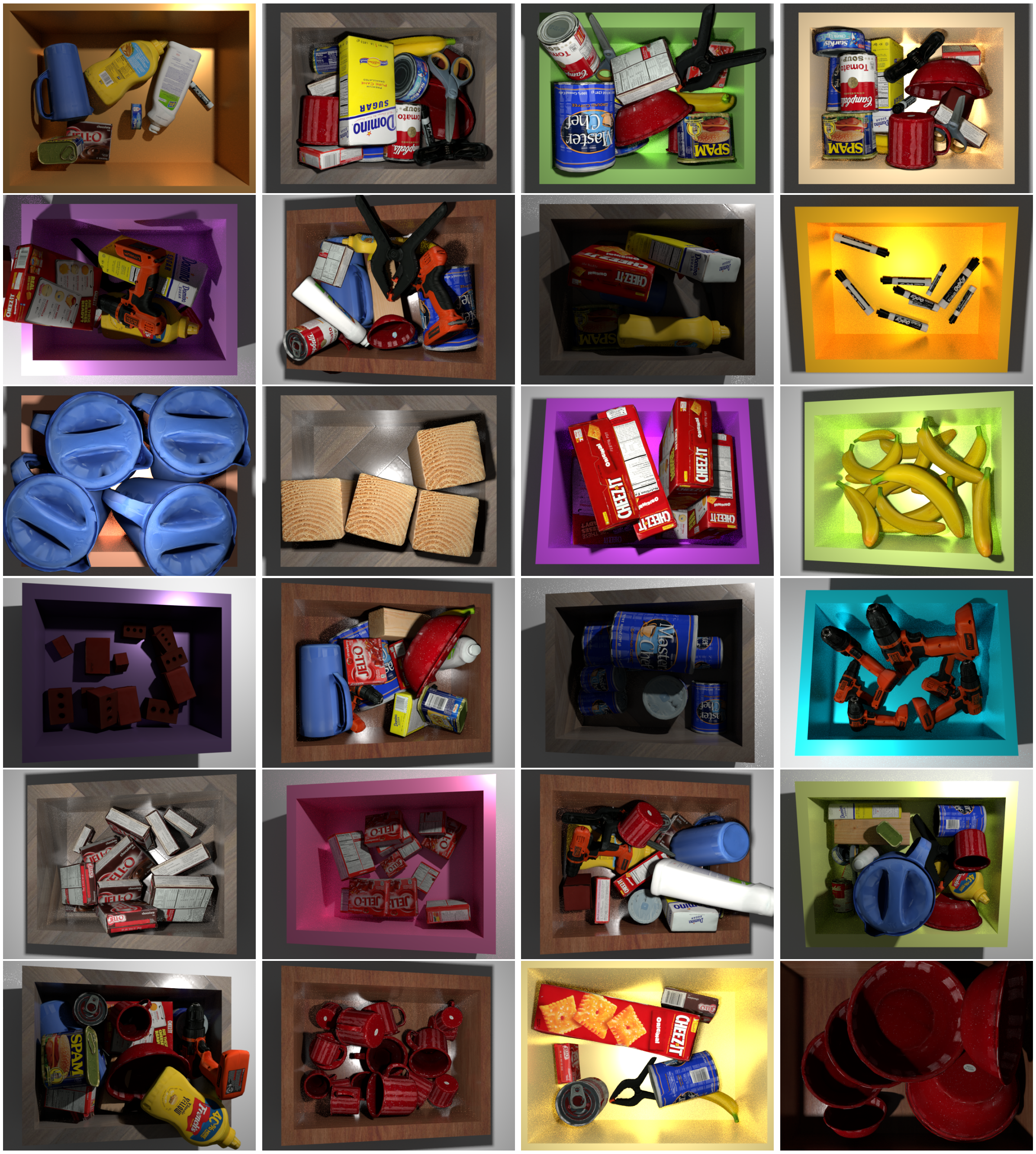


Figure 4.3: Sample RGB images from our generated dataset with a large variety of object combinations, lighting conditions, viewpoints and background textures. Both homogeneous objects and challenging cluttered scenes are presented.

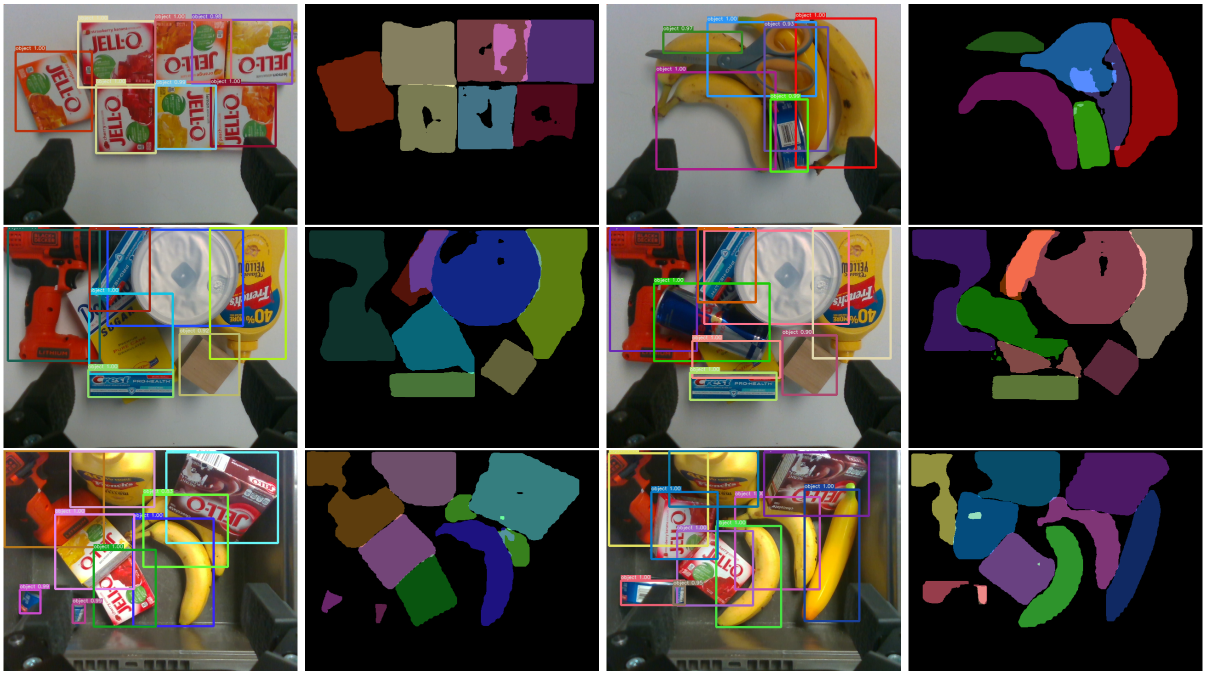


Figure 4.4: Detection and instance segmentation results on real-world data. The Red Bull bottle, real bananas and Colgate toothpaste boxes are novel objects not in the object database for training.

Bibliography

ABB. Abb irb-120. http://new.abb.com/products/robotics/industrial-robots/irb-120, 2017a. [Online; accessed 15-Feb-2017].

ABB. Abb externally guided motion (egm). http://new.abb.com/products/ABB.PARTS.SEROP3HAC054376-001, 2017b. [Online; accessed 14-June-2017].

Pulkit Agrawal, Ashvin V Nair, Pieter Abbeel, Jitendra Malik, and Sergey Levine. Learning to poke by poking: Experiential learning of intuitive physics. In *Advances in Neural Information Processing Systems*, pages 5074–5082, 2016.

Amir Ali Ahmadi and Pablo A Parrilo. A convex polynomial that is not sos-convex. *Mathematical Programming*, pages 1–18, 2012.

Alin Albu-Schäffer, Sami Haddadin, Ch Ott, Andreas Stemmer, Thomas Wimböck, and Gerhard Hirzinger. The dlr lightweight robot: design and control concepts for robots in human environments. *Industrial Robot: an international journal*, 34(5):376–385, 2007.

Mihai Anitescu and Florian A Potra. Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems. *Nonlinear Dynamics*, 14(3):231–247, 1997.

Suguru Arimoto, Sadao Kawamura, and Fumio Miyazaki. Bettering operation of robots by learning. *Journal of Field Robotics*, 1(2):123–140, 1984.

Christopher G Atkeson, Andrew W Moore, and Stefan Schaal. Locally weighted learning for control. In *Lazy learning*, pages 75–113. Springer, 1997.

J Andrew Bagnell, Felipe Cavalcanti, Lei Cui, Thomas Galluzzo, Martial Hebert, Moslem Kazemi, Matthew Klingensmith, Jacqueline Libby, Tian Yu Liu, Nancy Pollard, et al. An integrated system for autonomous robotics manipulation. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 2955–2962. IEEE, 2012.

Blender. Blender - a 3d modeling and rendering package. URL http://www.blender.org/.

Robert Bolles and Richard Paul. The use of sensory feedback in a programmable assembly system. Technical report, STANFORD UNIV CA DEPT OF COMPUTER SCIENCE, 1973.

Abdeslam Boularias, Oliver Kroemer, and Jan Peters. Learning robot grasping from 3-d images with markov random fields. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 1548–1553. IEEE, 2011.

Abdeslam Boularias, James Andrew Bagnell, and Anthony Stentz. Efficient optimization for autonomous robotic manipulation of natural objects. In *AAAI*, pages 2520–2526, 2014.

Abdeslam Boularias, J Andrew Bagnell, and Anthony Stentz. Learning to manipulate unknown objects in clutter by reinforcement. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 1336–1342. AAAI Press, 2015.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge university press, 2004.

Randy C Brost. Automatic grasp planning in the presence of uncertainty. *The International Journal of Robotics Research*, 7(1):3–17, 1988.

Randy C Brost and Matthew T Mason. Graphical analysis of planar rigid-body dynamics with multiple frictional contacts. In *The fifth international symposium on Robotics research*, pages 293–300. MIT Press, 1991.

Martin Buehler, Daniel E Koditschek, and Peter J Kindlmann. Planning and control of robotic juggling and catching tasks. *The International Journal of Robotics Research*, 13(2):101–118, 1994.

Robert R Burridge, Alfred A Rizzi, and Daniel E Koditschek. Sequential composition of dynamically dexterous robot behaviors. *The International Journal of Robotics Research*, 18(6): 534–555, 1999.

Arunkumar Byravan and Dieter Fox. Se3-nets: Learning rigid body motion using deep neural networks. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 173–180. IEEE, 2017.

Berk Calli, Arjun Singh, Aaron Walsman, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M Dollar. The ycb object and model set: Towards common benchmarks for manipulation research. In *Advanced Robotics (ICAR), 2015 International Conference on*, pages 510–517. IEEE, 2015.

John F Canny and Kenneth Y Goldberg. " risc" industrial robotics: recent results and open problems. In *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pages 1951–1958. IEEE, 1994.

Nikhil Chavan-Dafle and Alberto Rodriguez. Prehensile pushing: In-hand manipulation with push-primitives. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 6215–6222. IEEE, 2015.

Ignasi Clavera, David Held, and Pieter Abbeel. Policy transfer via modularity and reward guiding. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pages 1537–1544. IEEE, 2017.

Nikolaus Correll, Kostas E Bekris, Dmitry Berenson, Oliver Brock, Albert Causo, Kris Hauser, Kei Okada, Alberto Rodriguez, Joseph M Romano, and Peter R Wurman. Lessons from the amazon picking challenge. *arXiv preprint arXiv:1601.05484*, 2016.

Erwin Coumans. Bullet physics engine. *Open Source Software: http://bulletphysics. org*, 1:3, 2010.

Nikhil Chavan Dafle, Alberto Rodriguez, Robert Paolini, Bowei Tang, Siddhartha S Srinivasa, Michael Erdmann, Matthew T Mason, Ivan Lundberg, Harald Staab, and Thomas Fuhlbrigge. Extrinsic dexterity: In-hand manipulation with external forces. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 1578–1585. IEEE, 2014.

Raphael Deimel, Clemens Eppner, José Álvarez-Ruiz, Marianne Maertens, and Oliver Brock. Exploitation of environmental constraints in human and robotic grasping. In *Robotics Research*, pages 393–409. Springer, 2016.

Edsger W Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.

M Dogar, Kaijen Hsiao, Matei Ciocarlie, and Siddhartha Srinivasa. Physics-based grasp planning through clutter. 2012.

Mehmet Dogar and Siddhartha Srinivasa. A framework for push-grasping in clutter. *Robotics: Science and systems VII*, 1, 2011.

Mehmet R Dogar and Siddhartha S Srinivasa. Push-grasping with dexterous hands: Mechanics and a method. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 2123–2130. IEEE, 2010.

Aaron M Dollar and Robert D Howe. The highly adaptive sdm hand: Design and performance evaluation. *The international journal of robotics research*, 29(5):585–597, 2010.

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.

M. Erdmann. Using backprojections for fine motion planning with uncertainty. *The International Journal of Robotics Research*, 5(1):19, 1986.

Michael Erdmann. On a representation of friction in configuration space. *The International Journal of Robotics Research*, 13(3):240–271, 1994.

Michael A Erdmann and Matthew T Mason. An exploration of sensorless manipulation. *IEEE Journal on Robotics and Automation*, 4(4):369–379, 1988.

Carlo Ferrari and John Canny. Planning optimal grasps. In *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, pages 2290–2295. IEEE, 1992.

Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. In *Readings in computer vision*, pages 726–740. Elsevier, 1987.

Noriatsu Furukawa, Akio Namiki, Senoo Taku, and Masatoshi Ishikawa. Dynamic regrasping using a high-speed multifingered hand and a high-speed vision system. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 181–187. IEEE, 2006.

Ross Girshick. Fast r-cnn. In *Computer Vision (ICCV), 2015 IEEE International Conference on*, pages 1440–1448. IEEE, 2015.

Kenneth Y Goldberg. Orienting polygonal parts without sensors. *Algorithmica*, 10(2-4):201–225, 1993.

Kennneth Goldberg and Matthew T. Mason. Bayesian grasping. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1264–1269, 1990.

Suresh Goyal. *Planar Sliding of a Rigid Body With Dry Friction: Limit Surfaces and Dynamics of Motion*. PhD thesis, Cornell University, Dept. of Mechanical Engineering, 1989.

Suresh Goyal, Andy Ruina, and Jim Papadopoulos. Planar sliding with dry friction. Part 1. Limit surface and moment function. *Wear*, 143:307–330, 1991.

Marcus Gualtieri, Andreas ten Pas, Kate Saenko, and Robert Platt. High precision grasp pose detection in dense clutter. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ Interna-*

tional Conference on, pages 598–605. IEEE, 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017.

Tucker Hermans, Fuxin Li, James M Rehg, and Aaron F Bobick. Learning stable pushing locations. In *Development and Learning and Epigenetic Robotics (ICDL), 2013 IEEE Third Joint International Conference on*, pages 1–7. IEEE, 2013.

François Robert Hogan and Alberto Rodriguez. Feedback control of the pusher-slider system: A story of hybrid and underactuated contact dynamics. *arXiv preprint arXiv:1611.08268*, 2016.

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.

Moslem Kazemi, Jean-Sebastien Valois, J Andrew Bagnell, and Nancy Pollard. Robust object grasping using force compliant motion primitives. *Robotics: Science and Systems VIII*, page 177, 2013.

Matthew Klingensmith, Thomas Galluzzo, Christopher M Dellin, Moslem Kazemi, J Andrew Bagnell, and Nancy Pollard. Closed-loop servoing using real-time markerless arm tracking. 2013.

Jens Kober, J Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013.

Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *European conference on machine learning*, pages 282–293. Springer, 2006.

Marek Kopicki, Jeremy Wyatt, and Rustam Stolkin. Prediction learning in robotic pushing manipulation. In *Advanced Robotics, 2009. ICAR 2009. International Conference on*, pages 1–6.

IEEE, 2009.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

Steven M LaValle. *Planning algorithms*. Cambridge university press, 2006.

Steven M LaValle and James J Kuffner Jr. Randomized kinodynamic planning. *The international journal of robotics research*, 20(5):378–400, 2001.

Sergey Levine, Peter Pastor, Alex Krizhevsky, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with large-scale data collection. In *International Symposium on Experimental Robotics*, pages 173–184. Springer, 2016.

Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. Fully convolutional instance-aware semantic segmentation. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2359–2367, 2017.

Tomas Lozano-Perez, Matthew T. Mason, and Russell H. Taylor. Automatic synthesis of fine-motion strategies for robots. *International Journal of Robotics Research*, 3(1), 1984.

Kevin M Lynch. Estimating the friction parameters of pushed objects. In *Intelligent Robots and Systems' 93, IROS'93. Proceedings of the 1993 IEEE/RSJ International Conference on*, volume 1, pages 186–193. IEEE, 1993.

Kevin M Lynch. *Nonprehensile robotic manipulation: controllability and planning*. Carnegie Mellon University Pittsburgh, 1996.

Kevin M Lynch and Matthew T Mason. Stable pushing: Mechanics, controllability, and planning. *The International Journal of Robotics Research*, 15(6):533–556, 1996.

Kevin M Lynch and Matthew T Mason. Dynamic nonprehensile manipulation: Controllability, planning, and experiments. *The International Journal of Robotics Research*, 18(1):64–92, 1999.

KM Lynch, H Maekawa, and K Tanie. Manipulation and active sensing by pushing using tactile feedback. In *Intelligent Robots and Systems, 1992., Proceedings of the 1992 lEEE/RSJ International Conference on*, volume 1, pages 416–421. IEEE, 1992.

Alessandro Magnani, Sanjay Lall, and Stephen Boyd. Tractable fitting with convex polynomials via sum-of-squares. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*, pages 1672–1677. IEEE, 2005.

Jeffrey Mahler, Jacky Liang, Sherdil Niyaz, Michael Laskey, Richard Doan, Xinyu Liu, Juan Aparicio Ojea, and Ken Goldberg. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. *arXiv preprint arXiv:1703.09312*, 2017.

Matthew T. Mason. On the scope of quasi-static pushing. In *International Symposium on Robotics Research*, pages 229–233. Cambridge, Mass: MIT Press, 1986a.

Matthew T. Mason. Mechanics and planning of manipulator pushing operations. *International Journal of Robotics Research*, 5(3):53–71, Fall 1986b.

Matthew T Mason. *Mechanics of robotic manipulation*. MIT press, 2001.

Matthew T Mason. Toward robotic manipulation. *Annual Review of Control, Robotics, and Autonomous Systems*, (0), 2018.

Matthew T Mason, Alberto Rodriguez, Siddhartha S Srinivasa, and Andrés S Vazquez. Autonomous manipulation with a general-purpose simple hand. *The International Journal of Robotics Research*, 31(5):688–703, 2012.

Matthew T Mason et al. Compliant motion. *Robot Motion: planning and control*, pages 305–322, 1982.

Tad McGeer et al. Passive dynamic walking. *I. J. Robotic Res.*, 9(2):62–82, 1990.

Daniel Mellinger and Vijay Kumar. Minimum snap trajectory generation and control for quadrotors. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 2520–2525. IEEE, 2011.

Andrew T Miller, Steffen Knoop, Henrik I Christensen, and Peter K Allen. Automatic grasp planning using shape primitives. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, volume 2, pages 1824–1829. IEEE, 2003.

Tom Michael Mitchell, Matthew T Mason, and Alan D Christiansen. Toward a learning robot. 1989.

Igor Mordatch, Emanuel Todorov, and Zoran Popović. Discovery of complex behaviors through contact-invariant optimization. *ACM Transactions on Graphics (TOG)*, 31(4):43, 2012.

Jean J Moreau. Unilateral contact and dry friction in finite freedom dynamics. In *Nonsmooth Mechanics and Applications*, pages 1–82. Springer, 1988.

Richard M Murray, Muruhan Rathinam, and Willem Sluis. Differential flatness of mechanical control systems: A catalog of prototype systems. In *ASME international mechanical engineering congress and exposition*, 1995.

Nils J Nilsson. Shakey the robot. Technical report, SRI INTERNATIONAL MENLO PARK CA, 1984.

Lael U Odhner, Leif P Jentoft, Mark R Claffee, Nicholas Corson, Yaroslav Tenzer, Raymond R Ma, Martin Buehler, Robert Kohout, Robert D Howe, and Aaron M Dollar. A compliant, underactuated hand for robust manipulation. *The International Journal of Robotics Research*, 33(5):736–752, 2014.

Edwin Olson. Apriltag: A robust and flexible visual fiducial system. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 3400–3407. IEEE, 2011.

Giuseppe Oriolo, Alessandro De Luca, and Marilena Vendittelli. Wmr control via dynamic feedback linearization: design, implementation, and experimental validation. *IEEE Transactions on control systems technology*, 10(6):835–852, 2002.

Pablo A Parrilo. *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. PhD thesis, California Institute of Technology, 2000.

Michael A Peshkin and Arthur C Sanderson. Planning robotic manipulation strategies for workpieces that slide. *IEEE Journal on Robotics and Automation*, 4(5):524–531, 1988a.

Michael A Peshkin and Arthur C Sanderson. The motion of a pushed, sliding workpiece. *IEEE Journal on Robotics and Automation*, 4(6):569–598, 1988b.

Lerrel Pinto and Abhinav Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 3406–3413. IEEE, 2016.

Michael Posa, Cecilia Cantu, and Russ Tedrake. A direct method for trajectory optimization of rigid bodies through contact. *The International Journal of Robotics Research*, 33(1):69–81, 2014.

Gill A Pratt and Matthew M Williamson. Series elastic actuators. In *Intelligent Robots and Systems 95.'Human Robot Interaction and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference on*, volume 1, pages 399–406. IEEE, 1995.

Marc H Raibert. *Legged robots that balance*. MIT press, 1986.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

Rethink Robotics. Sawyer - high performance collaborative robot. URL http://www.rethinkrobotics.com/sawyer/.

ROBOTIQ. Two-finger adaptive robot gripper. http://robotiq.com/products/adaptive-robot-gripper/, 2017. [Online; accessed 15-Feb-2017].

Murray Rosenblatt. Conditional probability density and regression estimators. *Multivariate analysis II*, 25:31, 1969.

Stefan Schaal and Christopher G Atkeson. Open loop stable control strategies for robot juggling.

In *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, pages 913–918. IEEE, 1993.

Stefan Schaal and Christopher G Atkeson. Learning control in robotics. *IEEE Robotics & Automation Magazine*, 17(2):20–29, 2010.

David E Stewart and Jeffrey C Trinkle. An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction. *International Journal for Numerical Methods in Engineering*, 39(15):2673–2691, 1996.

Yuval Tassa, Tom Erez, and Emanuel Todorov. Synthesis and stabilization of complex behaviors through online trajectory optimization. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 4906–4913. IEEE, 2012.

Russ Tedrake, Ian R Manchester, Mark Tobenkin, and John W Roberts. Lqr-trees: Feedback motion planning via sums-of-squares verification. *The International Journal of Robotics Research*, 29(8):1038–1052, 2010.

Emanuel Todorov. A convex, smooth and invertible contact model for trajectory optimization. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1071–1076. IEEE, 2011.

Sean Walker and J Kenneth Salisbury. Pushing using learned manipulation maps. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 3808–3813. IEEE, 2008.

D. E. Whitney. Quasi-static assembly of compliantly supported rigid parts. *ASME Journal of Dynamic Systems, Measurement, and Control*, 104:65–77, March 1983.

John Wishart. The generalised product moment distribution in samples from a normal multivariate population. *Biometrika*, pages 32–52, 1928.

Tsuneo Yoshikawa and Masamitsu Kurisu. Identification of the center of friction from pushing an object by a mobile robot. In *Intelligent Robots and Systems' 91.'Intelligence for Mechanical*

Systems, Proceedings IROS'91. IEEE/RSJ International Workshop on, pages 449–454. IEEE, 1991.

Kuan-Ting Yu, Maria Bauza, Nima Fazeli, and Alberto Rodriguez. More than a million ways to be pushed. a high-fidelity experimental dataset of planar pushing. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 30–37. IEEE, 2016.

Andy Zeng, Shuran Song, Kuan-Ting Yu, Elliott Donlon, Francois R Hogan, Maria Bauza, Daolin Ma, Orion Taylor, Melody Liu, Eudald Romo, et al. Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching. In *ICRA*, 2018.

Jiaji Zhou and Matthew T Mason. Pushing revisited: Differential flatness, trajectory planning and stabilization. In *Proceedings of the International Symposium on Robotics Research (ISRR)*, 2017.

Jiaji Zhou, R. Paolini, J. A. Bagnell, and M. T. Mason. A convex polynomial force-motion model for planar sliding: identification and application. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 372–377, May 2016.

Jiaji Zhou, J Andrew Bagnell, and Matthew T Mason. A fast stochastic contact model for planar pushing and grasping: Theory and experimental validation. *Robotics: Science and systems XIII*, 2017a.

Jiaji Zhou, R. Paolini, A. Johnson, J. A. Bagnell, and M. T. Mason. A probabilistic planning framework for planar grasping under uncertainty. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, 2017b.