INTRODUCTION TO ROBOTIC MANIPULATION

ROB498 COURSE NOTES TO COMPLEMENT CLASSROOM TEACHING

BY

Nima Fazeli

Assistant Professor of Robotics
Assistant Professor of Mechanical Engineering
University of Michigan, Ann Arbor

PUBLISHED IN THE WILD

Course Description and Grading

Description: Robotic manipulation is the art and science of the mastery of contact to purposefully change the physical world. In this course, we will study what it takes to build autonomous systems that can bring about this change. To this end, we will cover the fundamentals of contact mechanics, perception, planning, and controls for manipulation. The course blends classical and modern methods and introduces a variety of open research problems.

The material presented focuses on applied robotic manipulation in unstructured environments such as kitchens and offices where uncertainty and decision-making play a pivotal role. We will make extensive use of simulation for homework problems and real-world implementation on hardware for the course project.

Topics covered include:

1. Mechanics of Manipulation – 7 sessions
    - Preliminaries of rigid-body contact mechanics
    - Grasp analysis and synthesis
    - Patch and surface contact models – context of planar pushing
    - Rigid-body dynamical systems undergoing Coulomb frictional interactions - complementarity formulation
2. Perception and State-estimation for Manipulation – 7 sessions
    - Cameras and tactile sensors
    - Object detection, segmentation, and localization
    - Multi-modal state-estimation
    - System identification, contact formation estimation, and constraint inference
3. Planning for Manipulation – 6 sessions
    - Grasp planning
    - Planar manipulation planning
    - Trajectory optimization through contact
    - Introduction to Task and Motion Planning with Push-Grasp Synergies
    - Introduction to Learning for Manipulation Planning
4. Controls for Manipulation – 6 sessions
    - Task-space controls and Dynamic Motion Primitives
    - Hybrid position-force control in fully actuated mechanisms
    - Hybrid systems control through mode schedules
    - Impedance control for contact
    - Model predictive control through contact
    - Reinforcement learning for manipulation

Grading 7 coding assignments: 55%. Open-ended project culminating in a 4-6 page paper: 25%. Quizzes 15%. Class Participation: 5%. Assignments can be done in small groups, but each participant must submit an individual copy.

Recommended Prerequisites Linear algebra, probability theory, optimization. Homework assignments code is in Python – these requirements are generally not enforced but familiarity is strongly recommended.

Contents

Chapter 1

Introduction

Robotic manipulation is the science and art of robotic systems interacting with their physical environments. In particular, manipulation is the controlled and purposeful physical interaction of the robot with its environment to produce desired effects and changes. Robotic manipulation represents a crucial advancement in the field of robotics, encompassing a wide array of tasks that enable robots to interact with and modify their surroundings. From delicate tasks such as picking up fragile objects to heavy-duty operations like assembling industrial components, robotic manipulation is the key that unlocks the potential for machines to perform tasks that were once limited to human dexterity and strength.

In the journey towards achieving adept robotic manipulation, researchers and engineers grapple with multifaceted challenges. A fundamental challenge lies in endowing robots with the sensory perception required to comprehend their environment. This involves developing advanced tactile, visual, and proprioceptive sensors that grant the robot a real-time understanding of the objects it interacts with. By acquiring this sensory data, robots can adjust their movements, forces, and grips with an astonishing level of precision, mirroring the human capacity for adapting to different scenarios.

Furthermore, the intricacies of robotic manipulation extend beyond sensorial comprehension. Planning and control algorithms play a pivotal role in enabling robots to perform tasks with efficiency and accuracy. These algorithms must take into account not only the physical properties of the objects being manipulated but also the dynamics of the robot itself. Balancing the forces exerted during a grasp, predicting the consequences of each movement, and optimizing trajectories are just a few examples of the computational challenges that arise in this domain.

Robotic manipulation is not confined to controlled environments or structured objects. The real world is rife with uncertainties, including variations in object shape, unexpected obstacles, and changes in the environment. As a result, manipulation strategies must be adaptive and resilient, capable of handling unforeseen circumstances without succumbing to errors or collisions.

As robotics continues to integrate with diverse industries such as manufacturing, healthcare, and logistics, the significance of robotic manipulation becomes increasingly apparent. Automation of intricate tasks leads to enhanced productivity, reduced labor costs, and improved safety. In medical applications, robots can perform delicate surgeries with enhanced precision, minimizing invasiveness and recovery times. In warehouses, robots adept in manipulation can swiftly sort, pick, and pack items, streamlining supply chains and expediting deliveries.

In this journey into the realm of robotic manipulation, this exploration aims to delve into the principles, technologies, and breakthroughs that underpin this evolving field. From the mechanics of grippers to the algorithms orchestrating graceful movements, each component contributes to the grand narrative of empowering robots with the ability to not only interact with their environment, but to masterfully manipulate it. As we navigate the intricacies of this dynamic landscape, the potential for innovation is vast, and the implications for industries and society at large are transformative.

The purpose of these notes is to introduce some of the key concepts, mathematics, tools, and algorithms prevalent in manipulation. These notes are used in teaching ROB599/ROB498 “Introduction to Robotic Manipulation”, a new course taught at the University of Michigan. The field of robotic manipulation is very broad with a thriving and active research community. Our notes will touch on a small slice of this exciting

field and continue to evolve to fill the needs of the material taught at an introductory level. This course is greatly influenced by Prof. Matt Mason's "Mechanics of Robotic Manipulation" Mason (2001) and Profs. Russ Tedrake and Tomás Lozano-Pérez's "Intelligent Robotic Manipulation". Please contact Prof. Nima Fazeli at nfz@umich.edu for any feedback, suggestions, and corrections to these notes.

Chapter 2

The Mechanics of Manipulation

The central focus of Robotic Manipulation is the physical interaction between the robot and its environment. At its core, manipulation requires some form of physical contact. The question every robot must answer when reasoning about manipulation is: "what happens when I make contact with my environment?". This question hints at a foundational need for prediction. The role of contact models is exactly to address this question, i.e. predicting the outcome of the physical interactions.

The field of contact modeling is too broad for our introductory discussion on robotic manipulation. As such, we will focus our attention to rigid-body mechanics governed by Coulomb friction. In this chapter, we will cover some basic mechanics for three important manipulation problems: grasping, pushing, and bouncing. Before discussing these problems, we will first provide some common definitions and develop mathematical machinery useful for our discussions.

2.1 Preliminaries

In this section, our objective is to specify the set of assumption we will use in our analyses, define some common terms, and develop mathematical machinery useful for our discussion on the mechanics of manipulation. For this section, you'll need familiarity with linear algebra.

Rigid-Body Mechanics

Our first major assumption is rigid-body mechanics. A rigid-body is one that undergoes no deformation at any point in time and/or space. This means the object does not change shape or vibrate. This assumption is very convenient in that we can represent the configuration of a planar body with 3 numbers (2 for position and 1 for orientation), and 6 numbers in 3D space; however, we must remember that it is an assumption and in practice all objects deform to some extent.

There are a number of interesting and important implications to this assumption. First, energy is not dissipated through relative particle motion (e.g., vibration) or heat. Further, a point contact will remain a point contact. This is a relatively subtle but important point. Consider a compliant object such as a spatula being brought into contact with the environment. At first, the contact is a point or perhaps a line. As we push the spatula into the environment, it deforms, creating a contact patch. This deformation and change in contact formation is very complicated and the topic of current research. The rigid-body assumption allows us to simplify our analysis by neglecting this effect.

Rigid-body dynamics are a relatively good approximation for stiff bodies that move slowly (small velocity and acceleration). Examples of these types of objects are surprisingly common in our day to day lives (e.g., pens, cellphones, mugs, etc.). Note that there are also plenty of objects that are not "rigid" in this sense and that scale certainly has a role to play. It is unlikely that 1-Dimensional objects (e.g., bristles of a brush that have only one major length factor) or 2-Dimensional objects (e.g., shells or sheets of paper with only two major length factors) would be well-represented as rigid-objects in general.

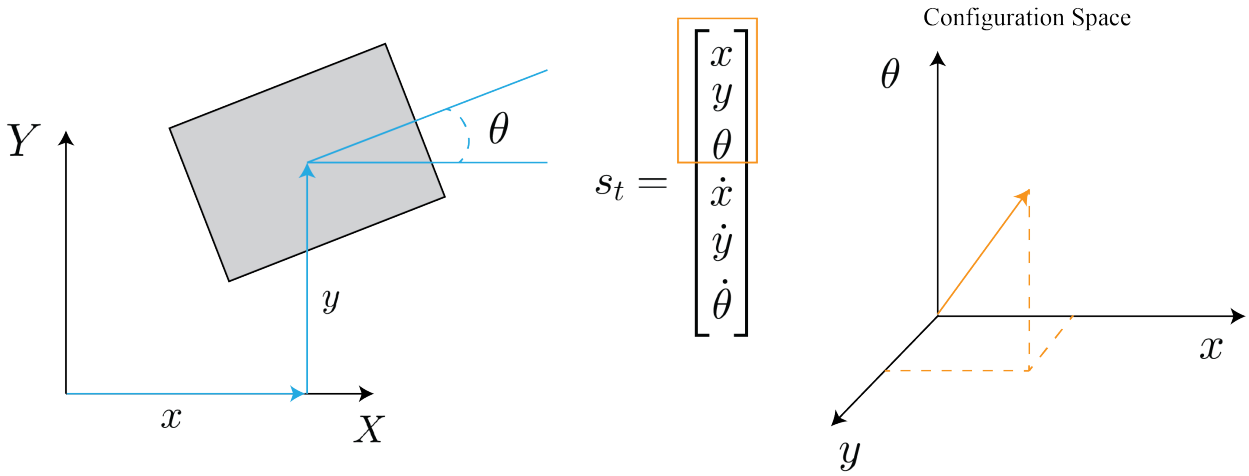

Figure 2.1. Visualization of state and configuration space for an object in 2D

Non-penetration

Our second assumption is non-penetration. We assume that no two bodies intersect each other at any point in time. This assumption implies that the velocity of the contact points between two objects cannot point towards each other. We will make extensive use of this assumption in writing constraints for contact. We will define distance functions to precisely quantify this notion in the Grasp Analysis subsection.

It is important to acknowledge that in the real world, verifying/detecting the non-penetration assumption is quite difficult due to factors such as imperfections in sensors, manufacturing tolerances, and environmental uncertainties. To address this limitation, advanced models and algorithms take into account these uncertainties to mitigate state-estimation and prediction that predict penetration and interpenetration. These models often incorporate compliance, friction, and other contact-specific parameters to realistically simulate the behavior of objects during contact and to prevent physically implausible scenarios. We will mostly be ignoring these effects in our introductory notes and refer the interested reader to [add citations].

State and Configuration Space

State is our way of describing everything we need to know about a system to predict it's evolution over time. Any single rigid-body in 3D space (far from contact) has 6 degrees of freedom, 3 linear and 3 rotational. The state vector of this body is composed of its positions and velocities, resulting in a 12 dimensional vector. This compact representation is a very nice feature of the rigid-body assumption. If we constrain the body to the plane and force it to move along a single linear axis, we'd get a system such as the classical 1 dimensional mass-spring-damper system which has 2 states: position of the mass and the velocity of the mass. This example should provide you with the following insight: for each kinematic constraint we add, we usually remove 1 degree of freedom which results in a reduction of the state space dimensionality by 2 (the velocity term is also removed). For a more complex example, consider a planar 2-link pendulum. In the absence of any constraints, this system has 24 state s (12×2) bodies in 3D space. By constraining to the plane, we have removed a total of 6 degrees of freedom (1 linear and 2 rotational per object). The revolute joints between the ground and the first body, and the first and second body also remove 2 degrees of freedom each. In total, the constraints have removed 10 degrees of freedom, leaving just 2 resulting in a 4 dimensional state-space. We usually represent state with a vector of real numbers and here denote it as $\boldsymbol{s}_t \in \mathbb{R}^n$.

Configuration space is a subset of state space that describes the positions of a system Lozano-Perez (1990). Fig. 2.1 depicts a 2D block in the world frame and its corresponding representation in configuration space. Configuration space is an important concept in robotics and is frequently used to describe not only the position of objects, but the set of admissible configurations objects are allowed to occupy. More on this later in the course.

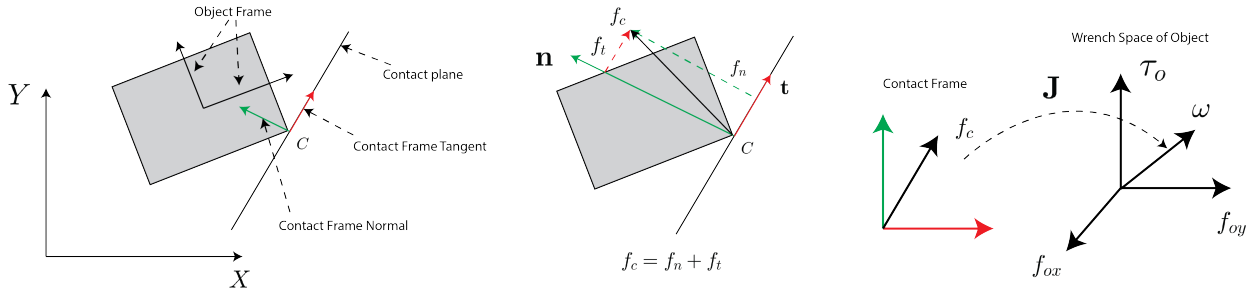

Figure 2.2. Reference frames, the contact force, and the object wrench.

Frames, forces, and Wrenches

Consider the object depicted in Fig. 2.2. We will refer to the reference frame attached to it's center of mass (COM) as the object frame. Let's assume that the object makes contact at point C. We define the contact frame at point C with the following axes:

- the contact normal $\boldsymbol{n}$ specified by the unit vector orthogonal to the plane passing through the point of contact that is tangent to the contacting bodies,
- the tangent vector $\boldsymbol{t}$ that lies within the plane passing through the point of contact and tangent to both bodies.

From their definitions, we know that $\langle \boldsymbol{t}_i, \boldsymbol{n} \rangle = 0$; i.e. that they are orthogonal. We also note that the choice of $\boldsymbol{t}$ is not unique; however, we typically choose the tangent vector that forms a right hand reference frame with the thumb pointing out of the page.

A force applied at C can be mapped into the object frame. The effective force at the COM of the object is the *wrench* applied to the object. A wrench is composed of forces and torques, while an externally applied force may or may not have torque components. In the following subsection, we relate the externally applied forces to the wrench at the object COM.

Contact Jacobian

A Jacobian is simply a mapping that projects externally applied forces to the wrench space of the object. If those forces are due to contact, then the Jacobian is referred to as the contact Jacobian. Let's derive an example of the contact Jacobian for Fig. 2.2. To start, we assume that we have applied a linear external force at point C. This force is represented in the contact frame as:

$$\boldsymbol{f}_c = f_n \boldsymbol{n} + f_t \boldsymbol{t}$$

where the first component represents the normal force along the unit normal vector and the second represents the tangential force along the unit tangent vector. Let's denote the vector pointing from the COM to the contact point with $\boldsymbol{r} = (r_x, r_y)$ in the object frame. Similarly, we can write the unit normal vector in the object frame as $\boldsymbol{n} = (n_x, n_y)$ where the subscripts specify the projection of the vector along the reference frame axes. To compute the effect of this force at the COM we write:

$$\boldsymbol{w}_n = \begin{bmatrix} \boldsymbol{n} \\ \boldsymbol{r} \times \boldsymbol{n} \end{bmatrix} = \begin{bmatrix} n_x \\ n_y \\ r_x n_y - r_y n_x \end{bmatrix} f_n$$

where the first two terms are simply the effect of the linear force translated to the COM and the last term accounts for the moment induced by the force at a distance w.r.t. to the COM. Similarly, the effect of a tangent force of magnitude f_t at the COM is:

$$\boldsymbol{w}_t = \begin{bmatrix} n_y \\ -n_x \\ -r_x n_x - r_y n_y \end{bmatrix} f_t$$

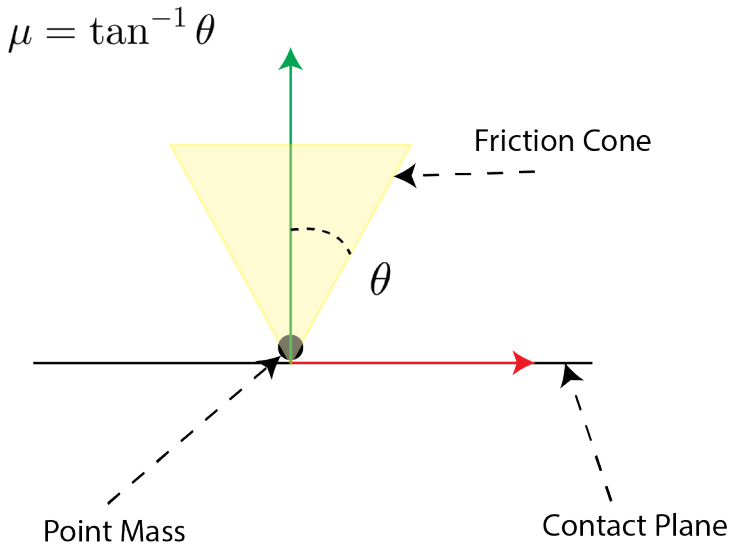

Figure 2.3. Point mass in frictional contact with a fixed surface.

where we have used the orthogonality of $\boldsymbol{n}$ and $\boldsymbol{t}$. To find the total wrench applied to the COM, we simply consolidate the previous two expressions and write:

$$\boldsymbol{w} = \begin{bmatrix} n_y & n_x \\ -n_x & n_y \\ -r_x n_x - r_y n_y & r_x n_y - r_y n_x \end{bmatrix} \boldsymbol{f}_c = \mathbf{J}_c \boldsymbol{f}_c$$

where $\mathbf{J}_c$ is called the contact Jacobian and is responsible for mapping the reaction force expressed in the contact frame to the force applied to the object in it's configuration space. The contact Jacobian is an important concept we will revisit consistently in the remainder of these notes.

An important note to make is the dual property of the contact Jacobian to map the object velocity to the contact point velocity (in the contact frame):

$$\boldsymbol{w} = \mathbf{J}_c \boldsymbol{f}_c$$

$$\boldsymbol{v}_c = \begin{bmatrix} v_t \\ v_n \end{bmatrix} = \mathbf{J}_c^T \begin{bmatrix} v_{o,x} \\ v_{o,y} \\ \omega_o \end{bmatrix} = \mathbf{J}_c^T \boldsymbol{v}_o$$

where the subscript o, x for example refers to the object velocity along the x axis of the object frame.

Coulomb Friction and the Friction Cone

We assume that all frictional interactions are governed by Coulomb friction (a.k.a. dry friction). Coulomb friction is a model of the interaction force between two bodies in contact. In this model, the frictional force at the point of contact is proportional to the normal force between the objects. The coefficient of proportionality is referred to as the coefficient of friction. The tangential force is in the plane of the surface of contact and opposes the relative motion between objects. If there is no relative motion between the objects, then the friction force is between zero and the maximum amount of the coefficient of friction multiplied by the normal force. If there is relative motion, then the frictional force is equal to the product of the normal force and the coefficient of friction. A note here, a distinction is often made between the kinetic/dynamic coefficient of friction and it's static counterpart where generally the static is larger than dynamic. Here we assume they are the same for convenience.

Consider a point mass in contact with a fixed surface, as depicted in Fig. 2.3. We can represent the set of forces allowed by the Coulomb friction model as the "friction cone". This cone specifies the set of reaction forces applies to the object from the surface due to contact. The reaction force itself is the sum of the normal and tangential forces applied to the bodies. The friction cone has the following important properties:

- The cone axis is perpendicular to the surface of contact (parallel to the surface's normal),
- The cone sides make an angle of $\theta = \tan^{-1}\mu$ with the axis.

Let's assume the point mass is stationary, if we apply an external force that lies within the friction cone then it is cancelled by the reaction force. A more precise mathematical statement regarding "lies within the friction cone" is to say that if we can find a positive combination of vectors that span the cone such that the sum of the external force and the positive span is a zero vector, then the external force lies within the friction cone. This statement means that friction due from the interaction can resist any force within the friction cone – which means that the object does not accelerate, implying that it is in sticking contact.

If we apply an external force that does not lie within the friction cone, then the reaction force can only partially resist it, and the remaining external force imparts an acceleration to the object. This implies that the object moves, either in sliding contact or separation. If the point mass has some initial velocity tangential to the surface, then the reaction force is always on the boundary of the cone and in the direction opposing the relative tangential velocity. This is due to the principle of maximal dissipation, a concept we'll cover later in the notes. Our introduction to the friction cone is for a point mass/point contact model. We will extend these ideas to patch contacts in subsequent sections of the notes.

Consider the extended body depicted in Fig. 2.4. The friction cone is depicted at the point of contact C. The friction cone is the set of all possible reaction forces that can be imparted to the object through frictional interaction. This means that we can use the contact Jacobian to project the friction cone into the wrench space of the object – referred to as the wrench space friction cone (or the wrench cone for short). We may sum up the effect of multiple contacts projected to the wrench space – referred to as the composite wrench space friction cone (composite wrench cone for short).

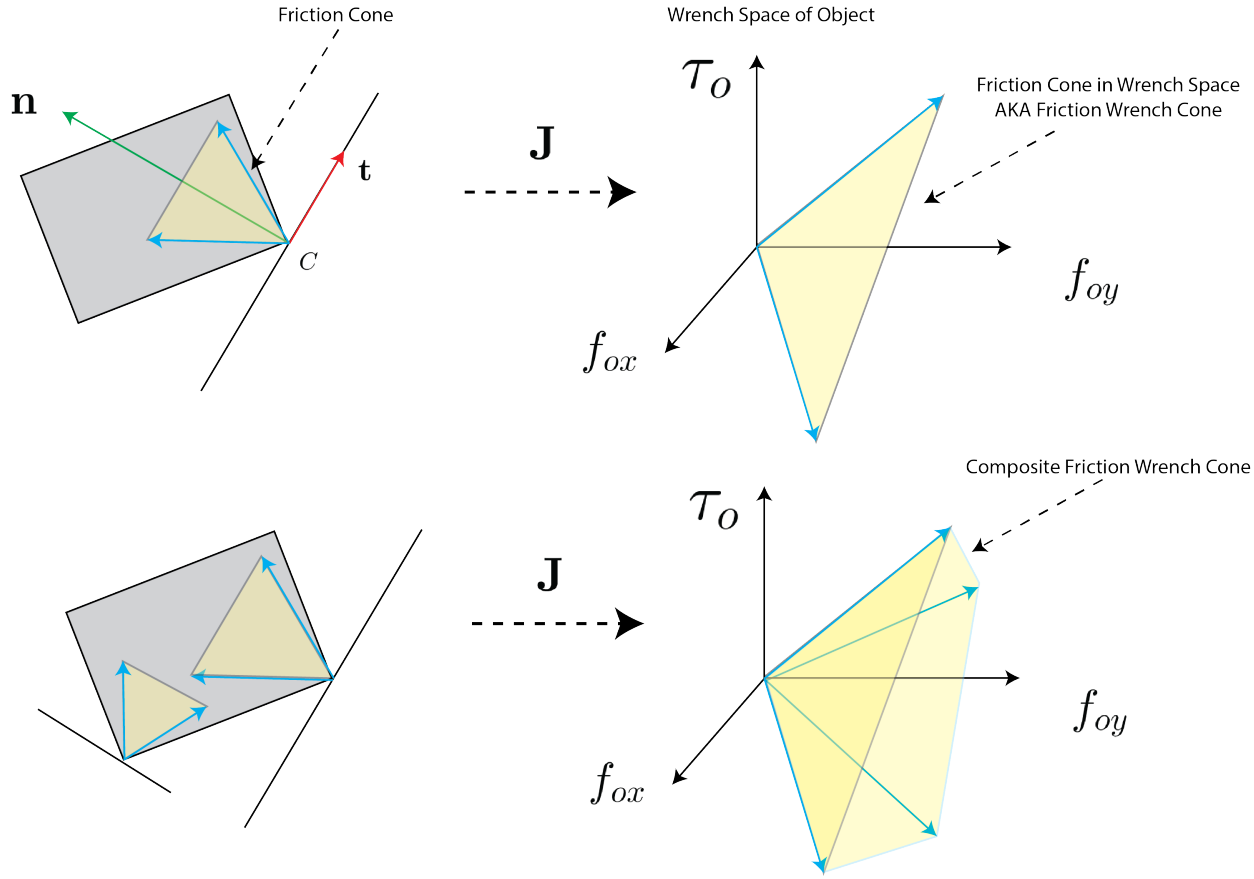

Figure 2.4. Friction cone for the extended body and it's projection into the object wrench space – resulting in the composite friction wrench cone.

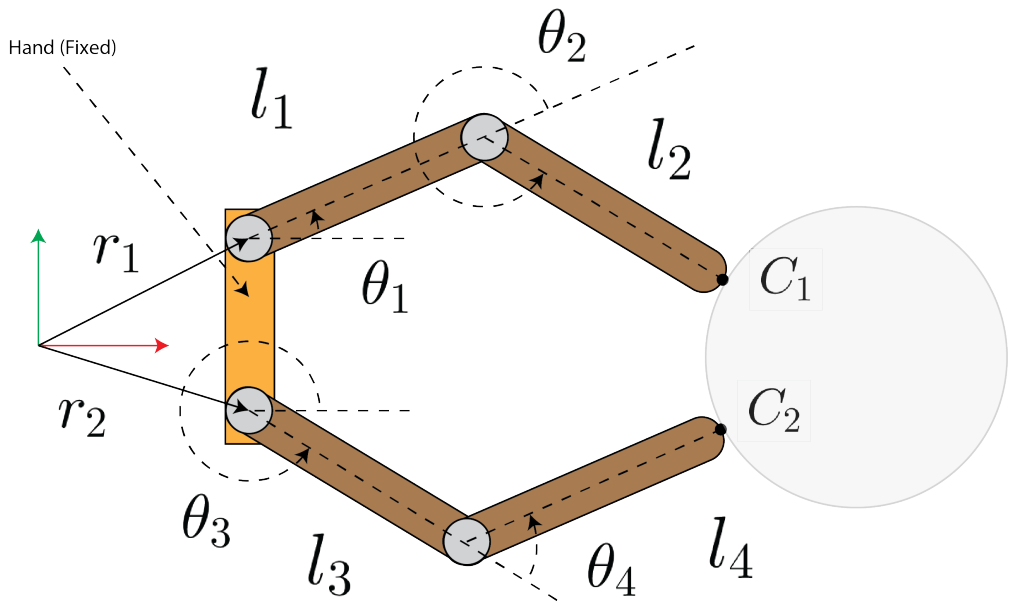

Figure 2.5. Fingers of a robot with contact points C_1 and C_2.

Robot/Finger Jacobians

The Jacobian we described in the previous subsection can be extended to handle multi-link fingers. Consider the hand depicted in Fig. 2.5. This hand has two fingers each with two links that are actuated. We assume that these fingers make contact at points C_1 and C_2. In this subsection, our objective is to find the applied contact force given the torques supplied to the joints of the fingers. Recalling from the previous subsection discussion on the Jacobians, we'll use the duality between configuration space velocity and contact points velocity to the derive this Jacobian.

Our first step is to relate the joint angles to the contact point locations C_1 and C_2 in the fixed hand reference frame. This procedure is referred to as forward kinematics. Let's denote the location of the contact points C_1 and C_2 as $\boldsymbol{q}_c = [x_{c1}, y_{c1}, x_{c2}, y_{c2}]^T$. We may write:

$$\boldsymbol{q}_c = fk(\boldsymbol{\theta}) = \begin{bmatrix} l_1 c_1 + l_2 c_{12} + r_{1,x} \\ l_1 s_1 + l_2 s_{12} + r_{1,y} \\ l_3 c_3 + l_4 c_{34} + r_{2,x} \\ l_3 s_3 + l_4 s_{34} + r_{2,y} \end{bmatrix}$$

where $\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3, \theta_4]^T$, $s_i = \sin(\theta_i)$, $c_i = \cos(\theta_i)$, $s_{ij} = \sin(\theta_i + \theta_j)$, and $c_{ij} = \cos(\theta_i + \theta_j)$. We can calculate the contact point velocity by taking the derivative of this expression w.r.t. to time:

$$\dot{\boldsymbol{q}}_c = \frac{\partial fk(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \dot{\boldsymbol{\theta}} = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} & -l_2 s_{12} & 0 & 0 \\ l_1 c_1 + l_2 c_{12} & l_2 c_{12} & 0 & 0 \\ 0 & 0 & -l_3 s_3 - l_4 s_{34} & -l_4 s_{34} \\ 0 & 0 & l_3 s_3 + l_4 s_{34} & l_4 c_{34} \end{bmatrix} \dot{\boldsymbol{\theta}} = \mathbf{J}_f^T \dot{\boldsymbol{\theta}}$$

Before applying the duality between force-wrench and velocities, we take a moment to justify it. First, we note that the total amount of power/energy in our system must be conserved. The total amount of power the fingers can exert is:

$$P_f = \boldsymbol{\tau}^T \dot{\boldsymbol{\theta}}$$

where $\boldsymbol{\tau}$ denotes the joint torques. The total power exerted at the contact points by the externally applied forces is equal to:

$$P_c = \boldsymbol{f}_c^T \dot{\boldsymbol{q}}_c$$

Since the input power has to equal the output power, we write:

$$\boldsymbol{\tau}^T \dot{\boldsymbol{\theta}} = \boldsymbol{f}_c^T \dot{\boldsymbol{q}}_c$$

now replacing the relationship we just derived for the Jacobian:

$$\boldsymbol{\tau}^T \dot{\boldsymbol{\theta}} = \boldsymbol{f}_c^T \mathbf{J}_f^T \dot{\boldsymbol{\theta}}$$

and since this relationship has to hold for all values of $\dot{\boldsymbol{\theta}}$ then we can conclude that:

$$\boldsymbol{\tau} = \mathbf{J}_f \boldsymbol{f}_c$$

This summarizes the application of the duality and derives a relationship between the joint torques and the amount of force produced at the contact points.

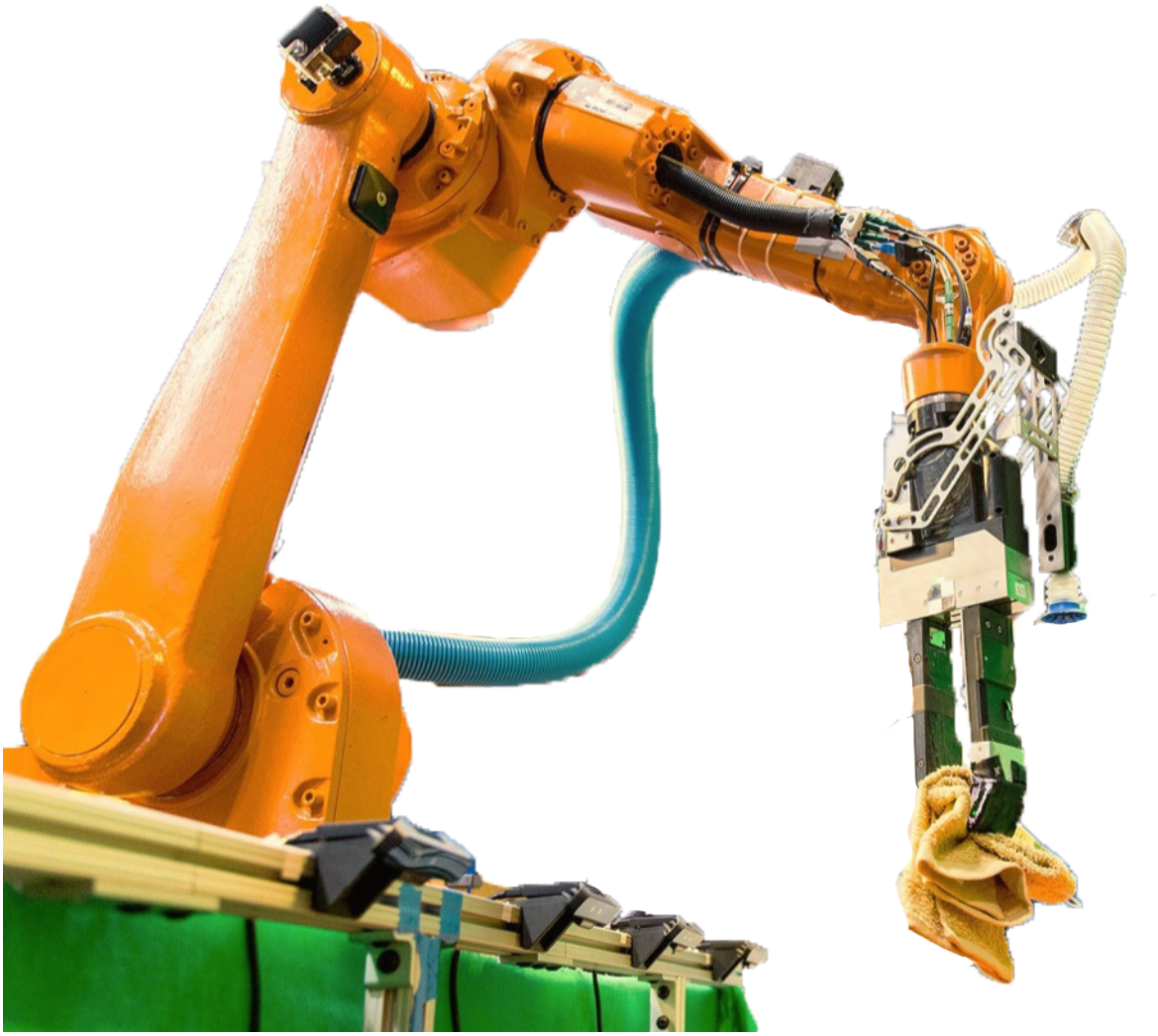

Figure 2.6. Real-world grasping system – Team MIT-Princeton's entry into the Amazon Robotics Challenge (2016-2017)

2.2 Grasping

Grasping is a fundamental skill in manipulation and has broad application to many tasks. While the term "grasping" has many interpretations and definitions, here we will specifically refer to an object rigidly held by the robot as a grasp. Specifically, the robot is permitted to make contact with the object at a known number of points and is only able to impart forces through these contacts. If the set of forces imparted to the object satisfy a set of requirements that we will study in this section, then the robot has successfully grasped the object. The robot can pick and place grasped objects and/or use them as tools to interact with its environment. A real-world example of a robotic system grasping objects autonomously is shown in Fig. 2.6.

In this section, we will assume all bodies are rigid and that Coulomb friction holds. In the remaining sections of this chapter we will develop the grasp matrix, a fundamental tool in analyzing grasp quality, and draw connections to the contact Jacobian we have studied previously. We will then use the grasp matrix to evaluate form and force closures of grasps to evaluate the stability of grasps. These properties are central to the task of planning for and controlling grasps and we will touch on these in subsequent chapters.

In the remainder, our analysis follows the excellent Grasping chapter of Prattichizzo and Trinkle (2016) and we refer the reader to this text for further details.

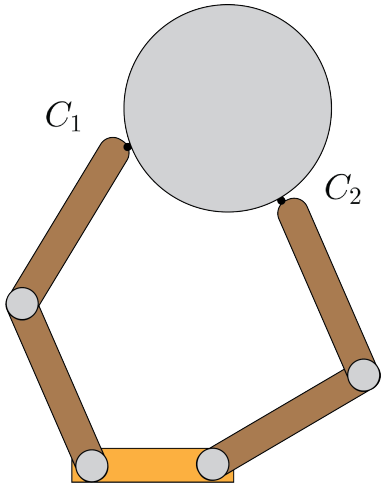

Figure 2.7. Example of a grasp with 2 fingers, each with two joints.

2.2.1 Grasp Matrix

Let's consider an object being grasped by our robot, e.g. the one depicted in Fig. 2.7. The robot makes N points of contacts with the object. The forces imparted by the robot to the object are governed by Coulomb friction. We will restrict our selves to the class of point contacts. For this class, each contact force $\boldsymbol{f}_{c,i}$ for $i = 1, ..., N$ can be written in the contact frame as:

$$\boldsymbol{f}_{c,i} = \begin{bmatrix} f_n \\ f_{t,1} \\ f_{t,2} \end{bmatrix} \text{ in 3D;} \qquad \boldsymbol{f}_{c,i} = \begin{bmatrix} f_n \\ f_t \end{bmatrix} \text{ in 2D}$$

From Section 2.1, we know that we can project this force into the reference frame of the object using the contact Jacobian:

$$\boldsymbol{w}_i = \mathbf{J}_{c,i} \boldsymbol{f}_i$$

where $\boldsymbol{w}_i$ denotes the wrench in the object frame corresponding to the force $\boldsymbol{f}_i$. Since $\boldsymbol{f}_i$ is governed by Coulomb friction, the corresponding wrench is due to the projection of the friction cone into the reference frame of the object. By summing up the individual contributions of all external forces applied by the robot to the object we have a composite friction cone, similar to the 2 point contact case in planar pushing. We may write:

$$\boldsymbol{w} = \sum_{i=1}^{N} \boldsymbol{w}_i = \sum_{i=1}^{N} \mathbf{J}_{c,i} \boldsymbol{f}_i = \begin{bmatrix} \mathbf{J}_{c,1} & ... & \mathbf{J}_{c,N} \end{bmatrix} \begin{bmatrix} \boldsymbol{f}_1 \\ ... \\ \boldsymbol{f}_N \end{bmatrix} = \mathbf{G} \boldsymbol{f}$$

where $\mathbf{G}$ denotes the grasp matrix. In general, the grasp matrix is $6 \times 3N$ dimensional for frictional point contacts with rigid bodies. Given the grasp matrix (which implicitly encodes the location of the grasps) and the coefficient of friction (to determine the friction cones), a grasp is fully identified. Note that the composite friction wrench has an identical definition to the grasp matrix.

In our derivation so far, we have implicitly assumed that the fingers of the robot are able to impart $\boldsymbol{f}_i$. In practice, the robots actuated joints can apply torques that are transmitted to the contact point through a set of contact Jacobians as well. We may write this as:

$$\boldsymbol{f}_i = \bar{\mathbf{J}}_{c,i} \boldsymbol{\tau}$$

where we have used the bar notation to differentiate between the finger Jacobian and object Jacobian, and the robot actuation is denoted as $\boldsymbol{\tau}$. With this consideration, a grasp is fully defined by the grasp matrix $\mathbf{G}$, the robot Jacobians $\bar{\mathbf{J}}_{c,i}$, and the coefficients of friction at the points of contact.

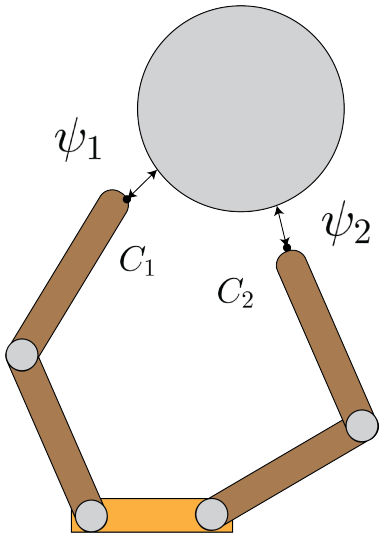

Figure 2.8. Distance functions for thee grasp depicted in Fig. 2.7.

One intuitive interpretation of the grasp matrix is that given a set of contact points and coefficients of friction, we can compute all the external wrenches we can apply to the object that can be resisted by the fingers. Another intuitive interpretation is that the grasp matrix qualifies the set of motions the object is not able to make given a grasp. In the following sections, we formalize these notions into a rigorous mathematical framework for grasp analysis.

2.2.2 Grasp Analysis - Form Closure

With the grasp matrix in hand, we are now staged to begin our discussion on useful types of grasps. We will begin with the notion of form closure. Intuitively, if the grasped object is in form closure then it is fully geometrically immobilized by the set contacts. We cannot perturb the object's configuration without violating any contact constraints; i.e. non-penetration. This simply means that any change in the configuration of the object with respect to the robot would lead to penetration of the object and is therefore impossible. This intuitive explanation is precisely how we write down and solve for form closure.

As in the previous subsection, let's assume that there are a total of N frictional contacts, each applying a force $\boldsymbol{f}_i$ at contact points $i = 1, ..., N$. For each contact point, we define a distance function $\psi_i(\boldsymbol{q}, \boldsymbol{q}_c)$ for $i = 1, ..., N$, where $\boldsymbol{q}$ denotes the object configuration and $\boldsymbol{q}_c$ denotes the configuration of the fingers. Together, these configurations specify the distance to contact for each contact point i. $\psi_i > 0$ implies separation, $\psi_i < 0$ implies penetration, and $\psi_i = 0$ implies contact. Let's assume all N contacts are active; i.e. that $\psi_i = 0$ for all i. Fig. 2.8 illustrates the distance functions for the grasp depicted in in Fig. 2.7.

For form closure to hold, we require that:

$$\boldsymbol{\psi}(\boldsymbol{q} + d\boldsymbol{q}, \boldsymbol{q}_c) \geq= 0 \implies d\boldsymbol{q} = 0$$

where the expression is to be evaluated element-wise. Intuitively, if any infinitesimal perturbation to the configuration of the object results in separation without penetration, then form-closure is violated. Conversely, we cannot find any infinitesimal perturbation to the configuration of the object that does not violate the non-penetration constraint. While this constraint provides an effective definition for form closure, we cannot use it in its current form.

A first order approximation to the definition of form closure is:

$$\frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{q}} d\boldsymbol{q} \geq 0 \implies d\boldsymbol{q} = 0$$

Intuitively, this is just like the first term in the Taylor expansion approximation of the derivative definition we have for the form closure. The interpretation is the same as before; however, to a first order approximation of perturbation. We will now relate the first order approximation to the grasp matrix.

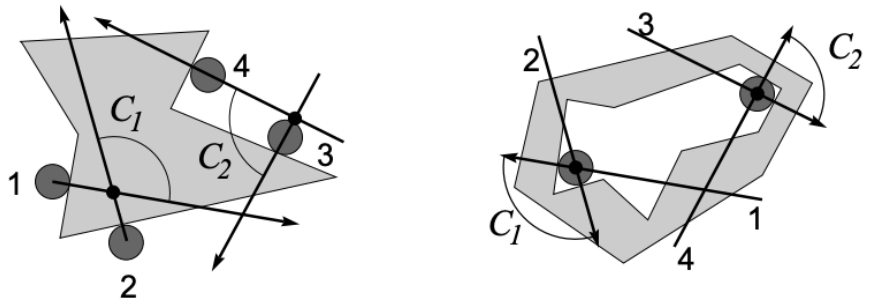

Figure 2.9. 2 examples of form closures in the plane with 4 points of contact. This image is from Prattichizzo and Trinkle (2016).

Since ψ is the distance function, it's gradient is the normal vector of the contact frames at each contact point. We know that the grasp matrix is composed of the set of normal and tangential components of contact frames. Let's denote the grasp matrix composed of only the normal contact vectors as $\mathbf{G}_n$. The condition above can equivalently be written as:

$$\mathbf{G}_n^T \boldsymbol{v} \geq 0 \implies \boldsymbol{v} = 0$$

where $\boldsymbol{v}$ denotes the instantaneous object velocity. This implication simply means that there is no set of object velocities that would lead to separation at any contact point. An equivalent formulation of this implication can be written for the set of contact forces applied to the object. Let's denote the magnitude of the normal component of the contact force as $\boldsymbol{f}_n$, then a grasp has first order form closure iff:

$$\begin{aligned} \mathbf{G}_n \boldsymbol{f}_n &= -\boldsymbol{g} \qquad \forall \boldsymbol{g} \in \mathbf{R}^6 \\ \boldsymbol{f}_n &\geq 0 \end{aligned}$$

The physical interpretation of this condition is that equilibrium can be maintained under the assumption that contacts are frictionless. We emphasize that $\boldsymbol{f}_n$ is only the magnitude of the normal component of the contact force and no other components. Since $\boldsymbol{g}$ is any vector in $\mathbf{R}^6$, for the inequality to hold we require that $\boldsymbol{g}$ be in the range of $\mathbf{G}_n$. Consequently, the rank of $\mathbf{G}_n$ must be 6 for the vector to lie in its range for all values it can take.

We can also write the condition for first order form closure as there exists $\boldsymbol{f}_n$ such that the following two conditions hold:

$$\begin{aligned} \mathbf{G}_n \boldsymbol{f}_n &= 0 \\ \boldsymbol{f}_n &> 0 \end{aligned}$$

This condition means that there exists a set of strictly compressive normal contact forces in the null space of $\mathbf{G}_n$. This also means that we can squeeze the object as tightly as we'd like while maintaining equilibrium (at no point will the object leave the grasp). For the above conditions to hold, Somov 1897 proved that at least 7 contacts are necessary for a 6 degree of freedom object and 4 are required for the planar case. Fig. 2.9 shows some example form closures in the plane (with 4 points of contact).

Geometrically, we can describe form closure using the composite friction cones we discussed in the previous sections. This idea is illustrated in Fig. 2.10.

First order form closure tests

To test whether a grasp has form closure, we'd like to check whether we can find a $\boldsymbol{f}_n$ such that the following two conditions hold:

$$\begin{aligned} \mathbf{G}_n \boldsymbol{f}_n &= 0 \\ \boldsymbol{f}_n &> 0 \end{aligned}$$

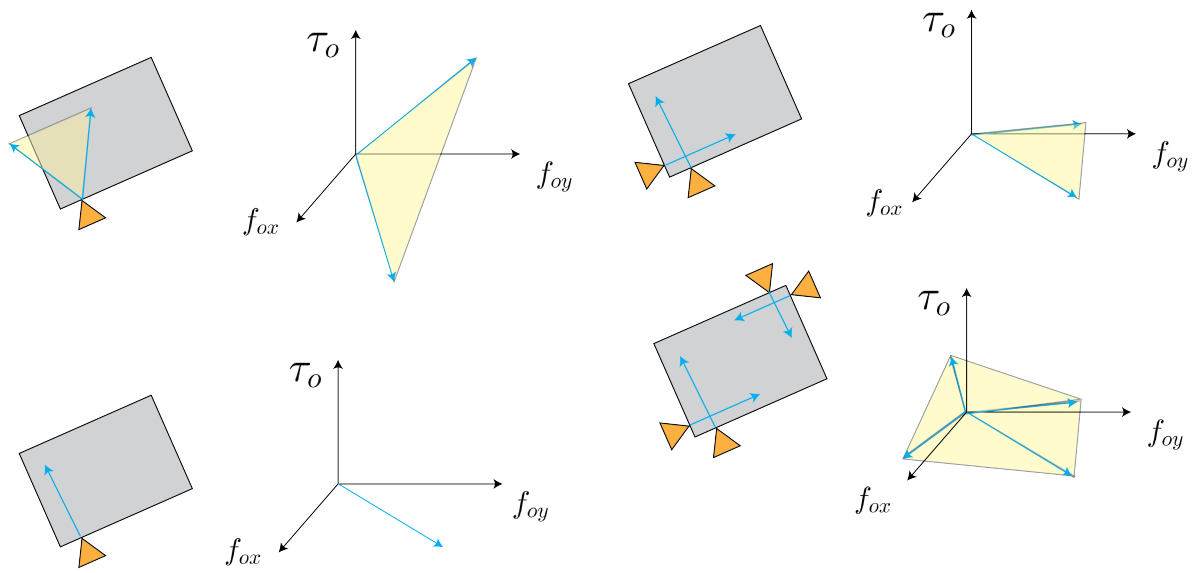

Figure 2.10. Geometric interpretation of the form closure. Each finger is permitted to only apply a force along the normal. Consequently, wrench cones can be produced by having multiple contacts with non-co-linear normals. If the resulting cone spans the entire wrench space, then form closure is possible.

Let's denote the smallest component of $\boldsymbol{f}_n$ as d. If we can find $d > 0$, it would imply that $\boldsymbol{f}_n > 0$. We can do this with the following linear program:

$$\begin{aligned} \textbf{LP1:} \quad & \text{maximize} && d \\ & \text{s.t.} && \mathbf{G}_n \boldsymbol{f}_n = 0 \\ & && \mathbf{I}\boldsymbol{f}_n - \mathbf{1}d \geq 0 \\ & && d \geq 0 \\ & && \mathbf{1}^T \boldsymbol{f}_n \leq N \end{aligned}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{1}$ is a vector with all components equal to 1. If this LP is infeasible or the optimal value d^* is zero, then the grasp is not in form closure. We can formalize this procedure as follows:
Form Closure Test:

1. Compute the rank of $\mathbf{G}_n$:
   - If this rank is less than 4 in the planar case or 7 in the 3D case, then form closure does not exist;
   - Else, the rank is adequate, proceed.
2. Solve LP1:
   - if $d^* = 0$ then form closure does not exist;
   - if $d^* > 0$ then form closure exists and d^* is a crude measure of how far the grasp is from losing form closure

Intuitively, if the smallest term in $\boldsymbol{f}_n$ is close to zero, then $\mathbf{G}_n$ is close to loosing rank which means that normal vectors of contacts are close to being dependent. This means that our grasp is close to losing form closure.

In summary, form closure is a fundamentally geometric constraint. It does not depend on frictional properties, rather it immobilizes the object using the concept of jamming. Form closure is a robust grasping strategy and a desirable one when it is possible to form. For the set of rigid-bodies, calculating form closure is equivalent to the solution of a linear program which can be done rapidly. One important drawback of form

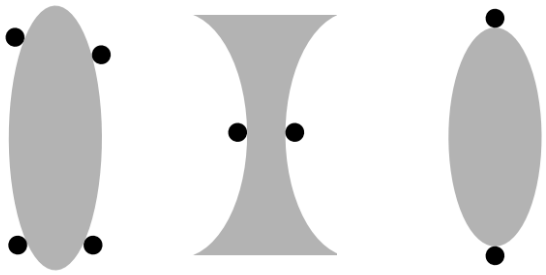

Figure 2.11. Which of the above are form closures? This image is from Prattichizzo and Trinkle (2016).

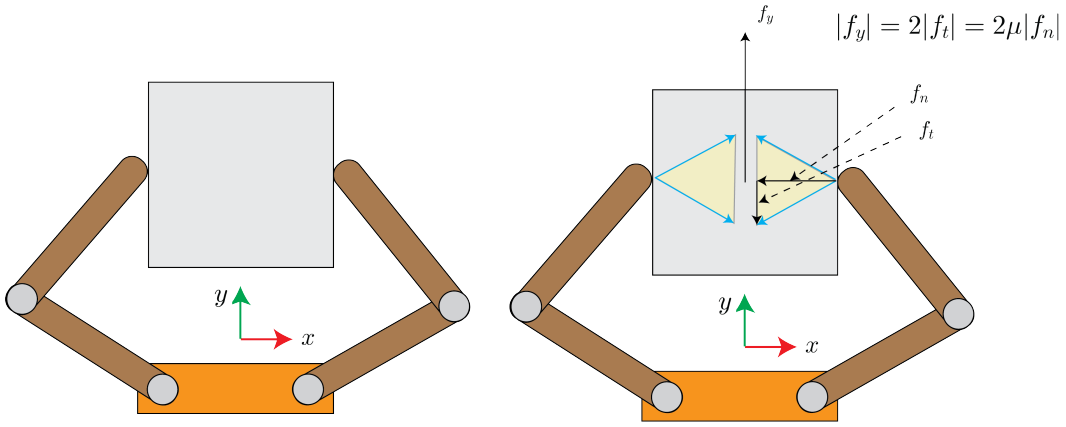

Figure 2.12. Example of a force closure induced by a "pinch grasp". The tighter the fingers pinch the object (yes that is actually a technical term), the more y force can be resisted by the frictional force at the fingers.

closure is the relatively large number of contacts required to produce it. In the following section, we study a different form of restraining that relies on contact mechanics and requires fewer contacts.

Let's go through some examples, take a look at Fig. 2.11. Which of these grasps are form closures? The first grasp is in form closure of first order. The second grasp is not in form closure of the first order, but it is in form closure of the second order. The third grasp is not in form closure at all. To understand the form closure of the second grasp, try to imagine a change in configuration of the object that does not violate the non-penetration constraint. Can you find such a change?

2.2.3 Grasp Analysis - Force Closure

In form closure, we noted that the constraints against external wrenches and motion are entirely geometric. There was no need to consider forces but we did need a rather large number of contacts. In force closure, we require that the contacts can resist any object wrench (any set of external forces and torques applied to the center of mass of the object). The key difference is that we allow frictional forces to help maintain the grasp; i.e. resist the object wrench. The key contribution is then a reduced number of contacts by virtue of including frictional forces. In fact, for a 3D object, we only need 2 soft finger contacts or 3 hard finger contacts for force closure rather than the 7 needed in form closure.

A fundamental requirement for form closure is that the hand/fingers must be able to squeeze arbitrarily tightly to account for any external wrench (no matter how large) applied to the object. For example, consider the object depicted in Fig. 2.12. The frictional force is able to resist any push we apply to the object along its y axis with a line of action passing through the center of mass – as long as there is enough normal force to produce sufficient frictional tangent force.

Since our discussion depends heavily on the finger type (and contact type), let's discuss 3 pervasive models:

- **Friction-free:** This type of finger can only apply a force along the contact normal. It is not permitted to apply tangential/frictional forces. Imagine this type of finger as made of ice! We may write the contact force in the contact frame as:

$$\boldsymbol{f}_c = \begin{bmatrix} 0, & 0, & f_n, & 0, & 0, 0, & 0 \end{bmatrix}^T$$

- **Hard finger:** The contact is well approximated by a point. It is allowed to transmit a normal force, and a tangential force in the contact plane.

$$\boldsymbol{f}_c = \begin{bmatrix} f_{t,1}, & f_{t,2}, & f_n, & 0, & 0, & 0 \end{bmatrix}^T$$

with the constraint that:

$$\sqrt{f_{t,1}^2 + f_{t,2}^2} \leq \mu f_n$$

- **Soft finger:** The contact is well approximated by a patch. The patch can transmit the same forces as a hard finger and an additional rotational torque, perpendicular to the contact plane.

$$\boldsymbol{f}_c = \begin{bmatrix} f_{t,1}, & f_{t,2}, & f_n, & 0, & 0, & \tau_p \end{bmatrix}^T$$

with the constraint that:

$$\frac{1}{\mu}\sqrt{f_{t,1}^2 + f_{t,2}^2} + \frac{1}{\alpha\nu}\sqrt{\tau_p^2} \leq f_n$$

where ν is the torsional friction coefficient and α is the characteristic length of the object to ensure consistency of units between the force and torque components.

We note that these are 3 simple models, there are more sophisticated versions if we'd like to study them. An important aspect of each of these definitions is that the contact reaction force lies in the friction cone induced by the type of contact and it's specific parameters. Specifically, for the hard finger, we can write the Coulomb friction cone $\mathcal{F}$:

$$\mathcal{F} = \{(f_n, f_{t,1}, f_{t,2}) \quad | \quad \sqrt{f_{t,1}^2 + f_{t,2}^2} \leq \mu f_n\}$$

and for the soft finger we can write the friction cone as:

$$\mathcal{F} = \{(f_n, f_{t,1}, f_{t,2}, \tau_p) \quad | \quad \frac{1}{\mu}\sqrt{f_{t,1}^2 + f_{t,2}^2} + \frac{1}{\alpha\nu}\sqrt{\tau_p^2} \leq f_n\}$$

Force Closure Definition

Recall from our form closure discussion that we defined a grasp as having form closure if:

$$\begin{aligned} \mathbf{G}_n \boldsymbol{f}_n &= -\boldsymbol{g} \\ \boldsymbol{f}_n &> 0 \end{aligned}$$

for all external wrenches applies to the object. In the definition of force closure, we still require that the grasp resists all external wrenches applied to the object. However, we have an additional constraint on the contact force: the contact force at the point of contact must be in the interior or on the boundary of the friction cone. We can express these conditions mathematically as:

$$\begin{aligned} \mathbf{G}\boldsymbol{f}_c &= -\boldsymbol{g} \\ \boldsymbol{f}_c &\in \mathcal{F} \end{aligned}$$

where $\mathcal{F}$ denotes the composite friction cone as is defined as:

$$\mathcal{F} = \mathcal{F}_1 \times ... \times \mathcal{F}_N = \{\boldsymbol{f}_{c,i} \in \mathcal{F}_i; \quad i = 1, ..., N\}$$

The key differences between the two definitions (form and force closure) are:

- we use the full grasp matrix (since we have tangential forces as well as normal forces),
- we use the full contact wrench (rather than just the normal force applied by the contact force),
- the contact wrench must belong to the friction cone.

Due to Murray et al. (1994), a grasp is said to have **frictional force closure** iff the following two conditions hold:

$$\begin{aligned} &rank(\mathbf{G}) = 3 \text{ (planar)} \quad \text{or} \quad 6 \text{ (3D)} \\ &\exists \boldsymbol{f}_c \text{ s.t. } \mathbf{G}\boldsymbol{f}_c = 0 \quad \text{and} \quad \boldsymbol{f}_c \in Interior(\mathcal{F}) \end{aligned}$$

A set of frictional contacts yields force closure if the positive span of the wrench cones is the entire wrench space. The rank condition means that the composite wrench cone spans the entire wrench space. For the second condition, since $\boldsymbol{f}_c \geq 0$ this ensures that the positive span of the wrench cones spans the entire wrench space.

Put another way, the first condition means that we want to span the space of all possible external wrenches exerted at the COM to be resisted by the contacts – if we cannot span this space, then there will be some subset of directions in which we cannot resist an external force. The second condition means there exists a set of reaction forces that span the null space of the grasp matrix and are in the interior of the composite friction cone. To understand better what this means, consider that we'd like to solve

$$\mathbf{G}\boldsymbol{f}_c = -\boldsymbol{g}$$

so we may write:

$$\boldsymbol{f}_c = -\mathbf{G}^{\dagger}\boldsymbol{g} + \bar{\mathbf{G}}\boldsymbol{f}_{c,null}$$

where the first term is the particular solution (pseudo inverse of $\mathbf{G}$ multiplied by the external wrench) and the second term $\bar{\mathbf{G}}$ (matrix with columns of the null space of $\mathbf{G}$) and $\boldsymbol{x}$ is the coefficient vector of the parameterization of the homogeneous solution. The set of internal contact forces $\boldsymbol{f}_{c,int} = \bar{\mathbf{G}}\boldsymbol{f}_{c,null}$ do not affect the solution of the equation above; however, they play a key role in determining the stability of the grasp. They specify how tightly we can grasp the object. Without the null space, the grasp can at most resist only one particular value of externally applied wrenches.

The definition we provided above has one important short-coming. Can the fingers trying to attain force closure actually provide the necessary contact forces? We say that a grasp has **force closure** (a stronger condition than frictional force closure stated above) iff:

$$\begin{aligned} &rank(\mathbf{G}) = 3 \text{ (planar)} \quad \text{or} \quad 6 \text{ (3D)} \\ &\mathcal{N}(\mathbf{G}) \cap \mathcal{N}(\mathbf{J}_f^T) = 0 \\ &\exists \boldsymbol{f}_c \text{ s.t. } \mathbf{G}\boldsymbol{f}_c = 0 \quad \text{and} \quad \boldsymbol{f}_c \in Interior(\mathcal{F}) \end{aligned}$$

where the second condition states that the null space of grasp matrix and the null space of finger Jacobians do not share any elements other than the zero vector. Intuitively, the grasp can inherently resist a set of contact forces described by the null space $\mathcal{N}(\mathbf{G})$. $\mathcal{N}(\mathbf{J}_f^T)$ specifies the set of contact forces the fingers can resist structurally but no joint torque can affect (make any changes to). If these two spaces share elements, then it means that there exists a set of contact forces that the grasp can inherently resist but that the fingers are incapable of producing. This implies the fingers are incapable of securely maintaining the grasp.

An important challenge in testing for force closure of a grasp is the quadratic/conic constraints imposed by the friction cones at the contacts. One approach to simplifying the test for force closure is to approximate the friction cone with polyhedral cone. Fig. 2.13 shows an example of a linearized friction cone.

Any of the friction cones we have discussed so far (induced by the friction free, hard finger, or soft finger) can be approximated by the non-negative span of a finite number n_g of generators $\boldsymbol{s}_{ij}$ of the friction cone. We can represent the set of applicable contact forces at contact i as:

$$\mathbf{G}_i\boldsymbol{f}_{c,i} = \mathbf{S}_i\boldsymbol{p}_i, \quad \mathbf{p}_i \geq 0$$

where $\mathbf{S}_i = [\boldsymbol{s}_{i1} \cdots \boldsymbol{s}_{in_g}]$ and $\boldsymbol{p}_i$ is a vector of non-negative generator weights. Let's write the expressions for $\mathbf{S}_i$ for each type of contact:

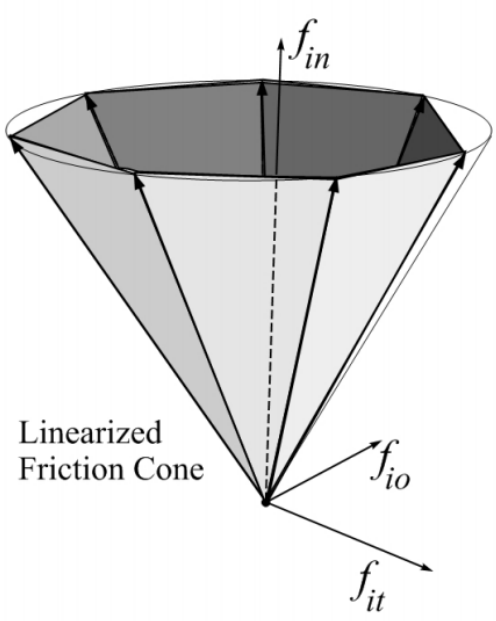

Figure 2.13. An example of the linearized friction cone for the hard finger model.

- **Friction free:** In this case the cone collapses to a line with $n_g = 1$ and $\mathbf{S}_i = [\hat{\boldsymbol{n}}_i^T \quad ((\boldsymbol{c}_i - \boldsymbol{p}) \times \hat{\boldsymbol{n}}_i)^T]^T$.
- **Hard finger:** The friction cone is represented by the non-negative sum of uniformly spaced contact force generators whose non-negative span approximates the Coulomb cone with an inscribed regular polyhedral cone. We can write:

$$\mathbf{S}_i = \begin{bmatrix} \cdots & 1 & \cdots \\ \cdots & \mu_i \cos(2k\pi/n_g) & \cdots \\ \cdots & \mu_i \sin(2k\pi/n_g) & \cdots \end{bmatrix}$$

  where $k = 1, \cdots, n_g$.
- **Soft finger:** Since the torsional friction in this model is decoupled from the tangential friction, it's generators are given by $[1\ 0\ 0\ bv_i]^T$ and we can write:

$$\mathbf{S}_i = \begin{bmatrix} \cdots & 1 & \cdots & 1 & 1 \\ \cdots & \mu_i \cos(2k\pi/n_g) & \cdots & 0 & 0 \\ \cdots & \mu_i \sin(2k\pi/n_g) & \cdots & 0 & 0 \\ \cdots & 0 & \cdots & bv_i & -bv_i \end{bmatrix}$$

  where b is a characteristic length used to unify the units and v_i is the torsional friction coefficient.

This polyhedral approximation to the friction cone allows us to efficiently represent the friction cone as a set of linear inequalities:

$$\mathbf{F}_i \boldsymbol{f}_{ci} \geq 0$$

where $\mathbf{F}_i$ is a matrix whose rows are composed of the normals to the faces formed by two adjacent generators of the approximate cone. For example, in the hard finger contact, row i of $\mathbf{F}_i$ can be computed as the cross product of $\boldsymbol{s}_i$ and $\boldsymbol{s}_{i+1}$. The intuitive interpretation of the inequalities is that we require the reaction force to be in the interior of the space created by the intersection of the set of half planes making up the sides of the friction cone. We can compose the set of all contact and their reaction forces in the compact form:

$$\mathbf{F} \boldsymbol{f}_c \geq 0$$

where $\mathbf{F} = \text{BlockDiag}(\mathbf{F}_1, \cdots, \mathbf{F}_{n_c})$.

We are now ready to test for force closure. Our procedure is as follows:

1. Compute the rank of $\mathbf{G}$:
   - if the rank is 3 in the planar case or 6 in the 3D case, continue;
   - else, force closure is not possible
2. Solve the frictional form closure linear program (LP2):

$$\begin{aligned} \textbf{LP2:} \quad & \text{maximize} && d \\ & \text{s.t.} && \mathbf{G}\boldsymbol{f}_c = 0 \\ & && \mathbf{F}\boldsymbol{f}_c - \mathbf{1}d \geq 0 \\ & && d \geq 0 \\ & && \boldsymbol{e}^T \boldsymbol{f}_n \leq N \end{aligned}$$

   where the optimal value d^* is a measure of the distance between the contact force and the boundary of the friction cone. The larger this value, the more stable the grasp. If $d^* = 0$ then force closure is not possible. Here we define $\boldsymbol{e}_i = [1\ 0\ 0\ 0\ 0\ 0]$ and $\boldsymbol{e} = [\boldsymbol{e}_1, \cdots \boldsymbol{e}_{n_c}]$, these vectors are responsible for picking out the normal component of the contact force for each contact.
3. Solve the check for $\mathcal{N}(\mathbf{G}) \cap \mathcal{N}(\mathbf{J}^T) = 0$ with LP3:

$$\begin{aligned} \textbf{LP3:} \quad & \text{maximize} && d \\ & \text{s.t.} && \mathbf{G}\boldsymbol{f}_c = 0 \\ & && \mathbf{J}^T \boldsymbol{f}_c = 0 \\ & && \mathbf{E}\boldsymbol{f}_c - \mathbf{1}d \geq 0 \\ & && d \geq 0 \\ & && \boldsymbol{e}^T \boldsymbol{f}_n \leq N \end{aligned}$$

   where $\mathbf{E} = \text{BlockDiag}(\boldsymbol{e}_1, \cdots, \boldsymbol{e}_{n_c})$. If $d^* = 0$ then force closure exists.

The polyhedral approximation to the friction cone is exact for the planar case and we can write:

$$\mathbf{F}_i = \frac{1}{\sqrt{1+\mu_i^2}} \begin{bmatrix} \mu_i & 1 \\ \mu_i & -1 \end{bmatrix}$$

Solve an example!

2.2.4 Antipodal Grasps and Parallel Jaw Grippers

Perhaps the most commonly used grippers in industry are simple two fingered parallel jaw grippers. Fig. 2.14 shows one such example (the ABB Yumi end-effector). The popularity of this design is due to its simplicity in design (leading to highly reliable mechanisms that can perform hundreds of thousands of grasps if treated well) and control (a simple single degree of freedom linear stage).

Given the popularity of this type of gripper, here we'll discuss a unique form of grasp analysis that is particularly well-suited for parallel jaw grippers: Antipodal grasps. A grasp is an "antipodal" grasp iff the line connecting the contact points lies inside both friction cones. Antipodal grasps are a special type of force closure that are particularly amenable to parallel jaw grippers. We do note that it is impossible to get antipodal form closure for hard fingers unless there are more than two points of contact.

In industrial applications, it is common to look for antipodal grasp affordances from visual feedback. We will investigate this approach more in our Perception for Manipulation section.

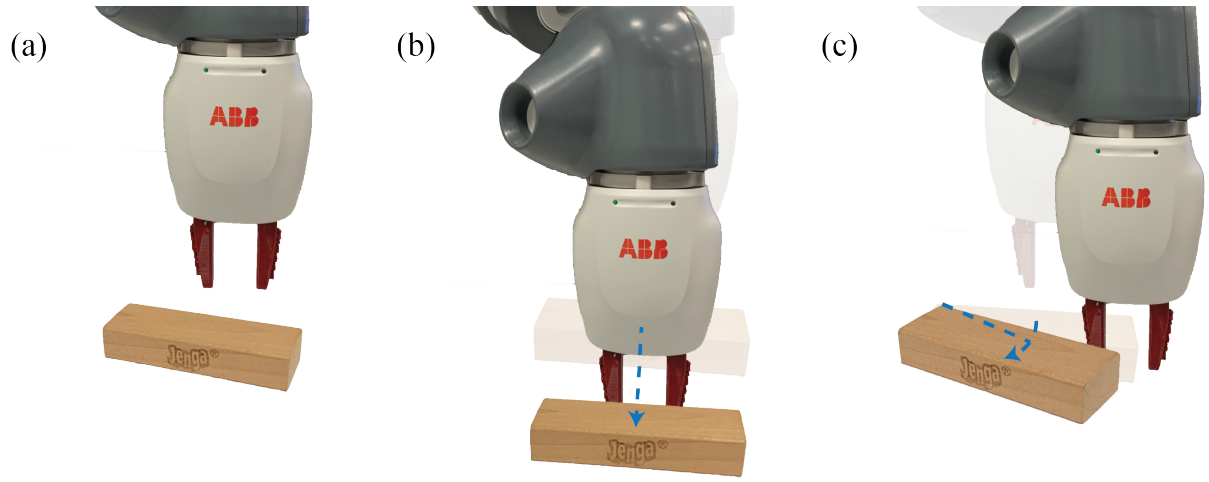

Figure 2.14. Illustrative example of planar pushing with a robot and a Jenga block.

2.3 Planar Pushing

In the previous section, we covered a dynamics analysis of rigid-bodies undergoing frictional interaction. In this section, we will cover another important type of rigid body frictional interaction where acceleration does not play an important role – the quasi-static regime. The intuition behind this regime is that motion is allowed but slowly enough such that accelerations terms such as inertial, centrifugal, and Coriolis are negligible. This assumption significantly simplifies the equations of the equations of motion and holds for many types of manipulation behaviors such as grasping and insertion.

Here, we focus on pushing, an important manipulation primitive, as a canonical example of a behavior that is well approximated by the quasi-dynamic assumption. See Fig. 2.14 for an example of this manipulation primitive. A key insight here is that if we stop pushing the object, it will stop moving because of negligible acceleration. We note that pushing can also be studied under the dynamics formulation; however, most pushing interactions are relatively slow and are well approximated by quasi-statics.

The quasi-static assumption in planar pushing has a long and rich history and we point the interested reader to some of the seminal work by Mason (1986a,b); Goyal et al. (1991); Lynch (1992); Lynch and Mason (1996); Lynch (1996); Howe and Cutkosky (1996), among many, for further details.

2.3.1 The Quasi-Static Assumption

Recall from the previous session that the dynamic equations of motion can be written as:

$$\mathbf{M}(\boldsymbol{q})\dot{\boldsymbol{v}} + \boldsymbol{c}(\boldsymbol{q}, \boldsymbol{v}) + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{f}_e + \boldsymbol{f}_c$$

where the two terms on the right hand side denote the external and contact forces. Under our quasi-static assumption we made simplify this expression to:

$$\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{f}_e + \boldsymbol{f}_c$$

where we have neglected the inertial, centrifugal, and Coriolis acceleration terms. The net change in configuration of the object is determined by the balance between external and contact forces. Intuitively, if you move something, then let it go, it stops moving, i.e. motion is possible but propensity for motion is not. **An important interpretation of the quasi-static assumption is that the change in configuration and the net applied forces must be consistent. We think less of force as causing a change in configuration and more as the change in configuration being consistent with the applied and reaction forces.**

Examples of tasks that are common in manipulation for which this assumption holds include grasping, pushing, and peg insertion. These tasks are typically executed slowly enough such that acceleration terms base safety be neglected.

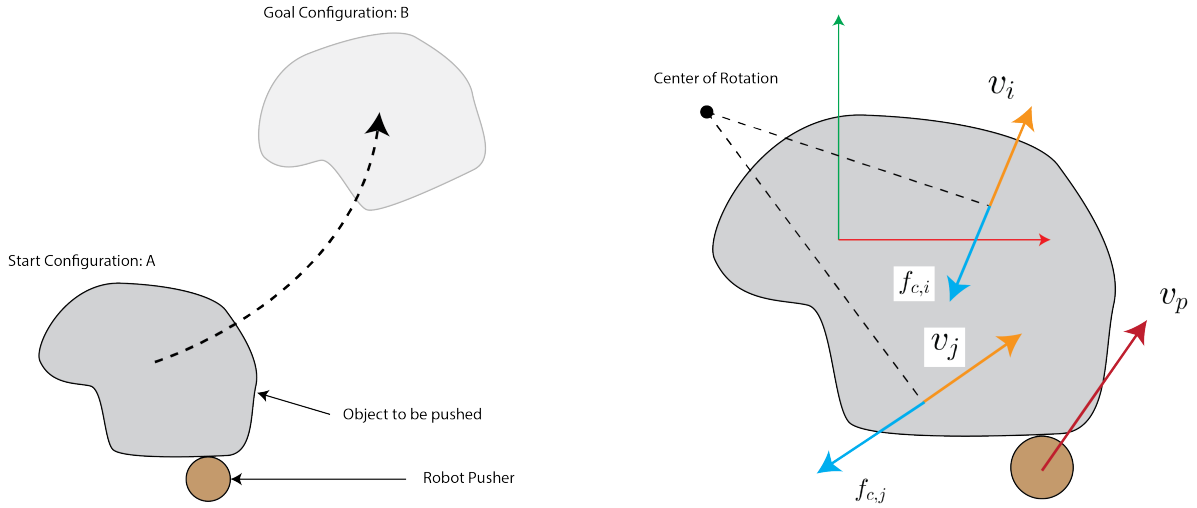

Figure 2.15. Pusher-slider system with initial and goal configurations. Here we see that the friction force at any point on the "support surface" opposes the direction of motion at that point. This is due to the principle of Maximal Dissipation – see Goyal et al. (1991) for more details.

2.3.2 Single Finger Pushing

Consider the task depicted in the left panel of Fig. 2.15. The objective is to move the object from point a to point B. One effective method of executing this task is to push the object along the trajectory connecting these two points. Pushing is particularly useful when the robot lacks the strength or is impeded by obstacles that prevent it from grasping/lift the object.

In this section, we're going to assume that the robot is only able to push the object with one finger. From the perspective of the object, there are two external forces: The first is the force applied by the robot through to finger (frictional single point contact), and the second is the frictional force from the contact between the object and the ground. The key observation here is that the frictional force exerted by the ground is of a different nature than the point contacts we've studied so far.

To handle this nuance, we will discuss a useful mathematical tool, referred to as the Limit Surface (LS). We will continue to assume that all bodies are rigid and Coulomb friction holds. We will further assume that the mass distribution across the contact patch is uniform. We also note that for the object to move it must slide across the surface; however, the contact between the finger and the block may stick or slide.

The Limit Surface and its Ellipsoidal Approximation

In this section, our objective is to find a mapping between the robot push action and the resulting object motion. From our quasi-static assumption, we know that the resultant motion of the object is due to the balance between the external force applied by the robot and the ground friction force. Here, we derive the Limit Surface (LS), a useful mathematical abstraction that we will use to relate robot action and motion to the resulting object motion by solving for this balance. Our derivation follows the work of Lynch et al. (1992) and Howe and Cutkosky (1996) closely and we refer the interested reader to these excellent texts for further details.

Consider Fig. 2.15. As the robot pushes the object, every point on the support surface (interface between the object and the surface it is moving on) resists the resulting motion due to Coulomb friction. The limit surface is the sum of all these forces on the support surface. It is the analog of the friction cone for a patch contact. We may write this sum as:

$$\boldsymbol{f}_c = \int_S \frac{\boldsymbol{v}(\boldsymbol{x})}{|\boldsymbol{v}(\boldsymbol{x}))|}\, \mu p\, d\boldsymbol{x} = \int_S \hat{\boldsymbol{v}}(\boldsymbol{x})\, \mu p\, dA$$

$$\boldsymbol{\tau}_c = \int_S \boldsymbol{x} \times \frac{\boldsymbol{v}(\boldsymbol{x})}{|\boldsymbol{v}(\boldsymbol{x}))|}\, \mu p\, d\boldsymbol{x} = \int_S \boldsymbol{x} \times \hat{\boldsymbol{v}}(\boldsymbol{x})\, \mu p\, dA$$

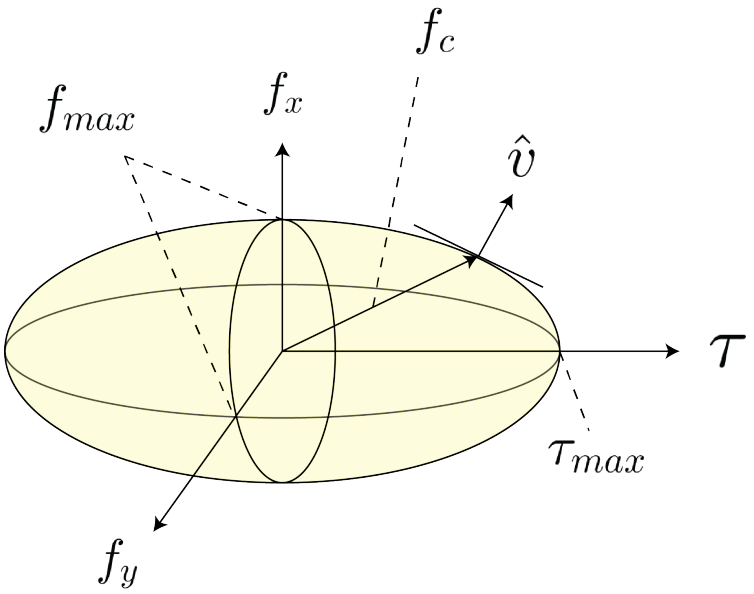

Figure 2.16. Ellipsoidal Limit Surface.

where $\boldsymbol{v}$ denotes the velocity at point $\boldsymbol{x}$ on the support surface, μ is the coefficient of friction, and p is the pressure at point $\boldsymbol{x}$ (here a constant because of our uniform pressure distribution assumption). We can calculate $\boldsymbol{v}(\boldsymbol{x})$ using rigid-body mechanics as:

$$\boldsymbol{v}(\boldsymbol{x}) = \begin{bmatrix} v_x - \omega y \\ v_y + \omega x \\ 0 \end{bmatrix}$$

where $\dot{\boldsymbol{q}} = \boldsymbol{v} = (v_x, v_y, \omega)^T$ is the velocity vector of the object and $\boldsymbol{x} = (x, y, 0)^T$ is the coordinate of every point on the surface relative to the center of mass. This function defines the Limit Surface and maps the unit velocity vector of the object to the reaction force from the support. Under the relatively easy to satisfy assumption that the pressure at every point on the support surface is finite, this function is smooth and strictly convex (one-to-one). If we solve for every unit velocity vector of the object, we have a surface (the Limit Surface) defined in force-torque space (wrench space) that maps every possible motion of the object to its corresponding frictional reaction force. Fig. 2.16 illustrates an ellipsoidal LS. Goyal et al. (1991) show that the unit vector $\hat{\boldsymbol{v}}$ is orthogonal to the Limit Surface. Intuitively, we are integrating over a gradient field $\hat{\boldsymbol{v}}$ to derive the surface, so the field must be orthogonal to the surface at all points.

To recap, if the object moves we can calculate the resistive frictional force using the LS – the reaction force lies on the boundary of the limit surface (similar to the point contact case where the force would lie on the boundary of the friction cone). The object motion is orthogonal to the surface of the LS. Further, if the net external force does not exceed the friction force for any choice of object motion, then the object does not move – the reaction force is in the interior of the LS.

It is only possible to solve for the limit surface in closed form for very few geometries. This complicates its use for controls and planning. To alleviate this challenge Howe and Cutkosky (1996); Lynch et al. (1992) proposed an ellipsoidal approximation to the surface that can easily be constructed. Lynch et al. (1992) proposes the following three steps to construct the ellipsoidal approximation of the LS:

1. Calculate the maximum torsional friction that the surface can resist using:

$$\tau_{max} = \mu \, p \int_S |x| dA$$

2. Calculate the maximum linear frictional force that the surface can resist using:

$$f_{max} = \mu |\boldsymbol{f}_n|$$

where $|\boldsymbol{f}_n|$ denotes the normal force.

3. The approximate ellipsoidal surface is given by:

$$\left(\frac{f_x}{f_{max}}\right)^2 + \left(\frac{f_y}{f_{max}}\right)^2 + \left(\frac{\tau}{\tau_{max}}\right)^2 = 1$$

Howe and Cutkosky (1996) Show that this approximation is close to the analytical solution in a variety of empirical experiments. An important note to be made is that the LS is scaled by the coefficient of friction, it grows as μ increases.

Equations of Motion

So far we have studied the effects of fresh orientation forces and object motion. We have yet to relate the robot action to the object motion. In the section, we develop procedures to relate robot actions to resulting object motion under two scenarios: first, The robot is able to impart forces to the object directly (force-controlled); and second, the robot is position controlled and can only import displacements. In both scenarios, we assume that the robot moves slowly enough such that the quasi-static assumption holds (no accelerations).

Force-control: Given an external force $\boldsymbol{f}_e = (f_x, f_y)$ applied at contact point $\boldsymbol{x}_c = (x, y)$:

1. Compute the effect of the external force at the COM. We can use the contact Jacobian we derived in previous sections to map the force at the point of contact between the robot and object:

$$\boldsymbol{f}_{e,c} = \mathbf{J}_c \boldsymbol{f}_e$$

2. This force is equal and opposite to the reaction force applied by the support surface:

$$\boldsymbol{f}_{e,c} = -\boldsymbol{f}_c$$

So find the corresponding point on the limit surface.

3. Compute the normal to the LS, denoted as $\hat{\boldsymbol{v}}$, and update the configuration of the object according to:

$$\boldsymbol{q}_{t+1} = \boldsymbol{q}_t + h\hat{\boldsymbol{v}}$$

where h denotes the step size. The smaller the value, the better the approximation.

Position-control: The central challenge in predicting the motion of the object due to changes in position of the robot (rather than force applied by the robot) is to project the robot motion into the force domain. Mason (1986a) developed a principle approach for planar pushing using the notion of *motion cones* which Lynch et al. (1992) applied to the ellipsoidal approximation of the LS. Here, we follow this approach:

1. Denote the point of contact in the object frame as $\boldsymbol{x}_c = (x_c, y_c)$, the robot pusher velocity as $\boldsymbol{v}_p = (v_{p,x}, v_{p,y})$, and the object velocity at the point as $\boldsymbol{v}_o = (v_{o,x}, v_{o,y})$ – illustrated in Fig. 2.18.

2. Compute the unit velocities $\hat{\boldsymbol{v}}_l = (v_{l,x}, v_{l,y}, v_{l,\omega})$ and $\hat{\boldsymbol{v}}_r = (v_{r,x}, v_{r,y}, v_{r,\omega})$ resulting from forces at the left and right extremes of the friction cone. The corresponding velocities at the contact point are $\hat{\boldsymbol{v}}_{c,l} = (v_{l,x} - y_c v_{l,\omega}, v_{l,y} + x_c v_{l,\omega})$ and $\hat{\boldsymbol{v}}_{c,r} = (v_{r,x} - y_c v_{r,\omega}, v_{r,y} + x_c v_{r,\omega})$. These two vectors define the *motion cone* Mason (1986a). Any force applied at the point of contact within the friction cone results in a velocity that is within the motion cone, i.e. it can be written as the positive sum of $\hat{\boldsymbol{v}}_{c,r}$ and $\hat{\boldsymbol{v}}_{c,l}$.

3. If the pusher velocity is within the motion cone, then the contact is sticking and $\boldsymbol{v}_o = \boldsymbol{v}_p$. If the pusher velocity is to the left (right) of the motion cone, then the pusher slides on the object to the left (right).

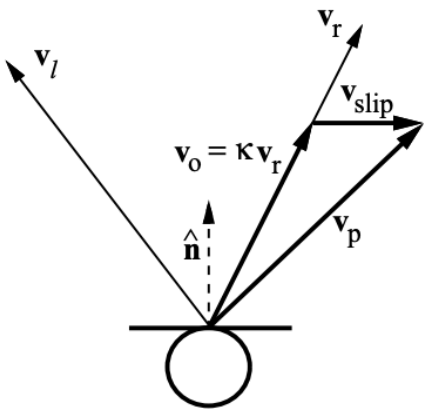

Figure 2.17. Contact between robot and object is slipping. The resulting velocities are shown. This image is from Lynch et al. (1992).

4. If the contact is sticking, we know that $\boldsymbol{v}_o = \boldsymbol{v}_p$. Since the applied force must pass through the point of contact and $\boldsymbol{\tau} = \boldsymbol{x}_c \times \boldsymbol{f}_c$, we may write:

$$\begin{aligned} v_x &= v_{p,x} + v_{p,\omega} y_c \\ v_y &= v_{p,y} - v_{p,\omega} x_c \\ \tau &= x_c f_y - y_c f_x \end{aligned}$$

Given that $\boldsymbol{f}$ is parallel to $\boldsymbol{v}$, we may write:

$$\begin{aligned} \frac{v_x}{v_\omega} &= \left(\frac{\tau_{max}}{f_{\max}}\right)^2 \frac{f_x}{\tau} = c^2 \frac{f_x}{\tau} \\ \frac{v_y}{v_\omega} &= \left(\frac{\tau_{max}}{f_{\max}}\right)^2 \frac{f_y}{\tau} = c^2 \frac{f_y}{\tau} \end{aligned}$$

and solving for the object velocity we have:

$$\begin{aligned} v_x &= \frac{(c^2 + x_c^2) v_{p,x} + x_c y_c v_{p,y}}{c^2 + x_c^2 + y_c^2} \\ v_y &= \frac{x_c y_c v_{p,x} + (c^2 + y_c^2) v_{p,y}}{c^2 + x_c^2 + y_c^2} \\ v_\omega &= \frac{x_c v_y - y_c v_x}{c^2} \end{aligned}$$

5. If the contact is sliding, i.e. when $\boldsymbol{v}_p$ is outside the motion cone, then the contact is slipping and $\boldsymbol{v}_o$ lies on the boundary of the motion cone. The slipping velocity $\boldsymbol{v}_s$ (in the plane of contact) satisfies:

$$\boldsymbol{v}_o + \boldsymbol{v}_s = \boldsymbol{v}_p$$

as shown in Fig. 2.17. To calculate the velocity of the object, we project the velocity of the pusher onto the boundary of the motion cone. Let's assume the sliding is to the right (same process holds for the left):

$$\boldsymbol{v}_o = \frac{\boldsymbol{v}_p^T \hat{\boldsymbol{n}}}{\boldsymbol{v}_r^T \hat{\boldsymbol{n}}} \boldsymbol{v}_r = \kappa \boldsymbol{v}_r$$

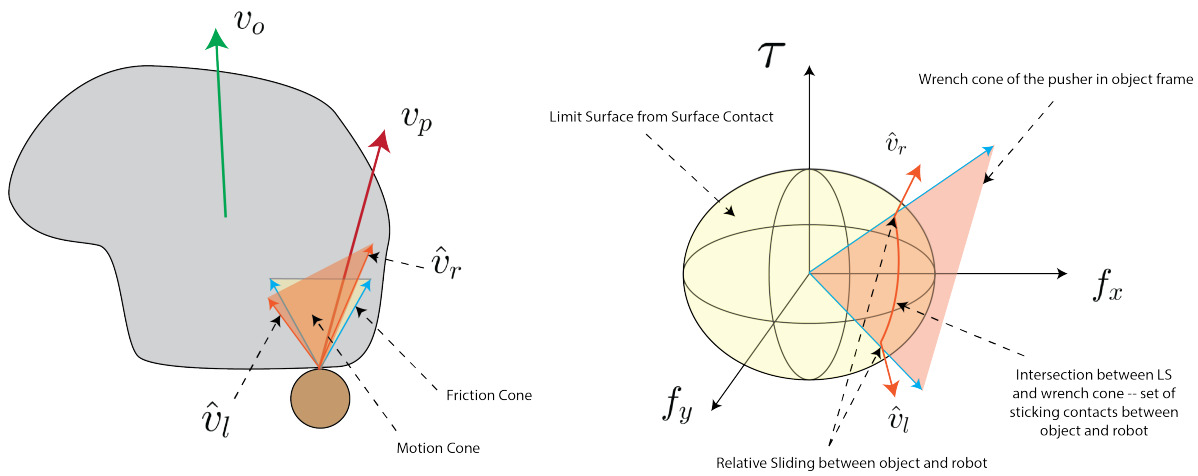

Figure 2.18. Single finger planar pushing. Left panel shows the friction cone, corresponding motion cone, and object/pusher velocities. Right panel shows the geometric interpretation of the motion mechanics.

6. Update the configuration of the object according to:

$$\boldsymbol{q}_{t+1} = \boldsymbol{q}_t + h\boldsymbol{v}_o$$

where h denotes the step size. The smaller the value, the better the approximation.

We can use this procedure to compute the next state of the object, given the motion of the robot pushing it. To provide additional insight into the mechanics, we will study the equations of motion from a geometric perspective.

Consider Fig. 2.18. The right panel shows the geometry of the interaction. Let's place ourselves at the object frame, the limit surface defines the set of all possible friction forces that the surface can apply to the object due to sliding. We know that if the object slides, the force the object feels lies on the limit surface and the unit vector orthogonal to this point defines the direction of motion of the object. When the robot pushes the object, it is only allowed to apply a force inside of the friction cone. We can project the friction cone to the wrench space to get the friction wrench cone. We know that the set of forces the robot can apply to the object has to lie inside or on the boundary of the wrench cone. The intersection of the wrench cone and the limit surface defines the force balance between the set of forces that the robot can apply to the object through frictional interaction and the set of forces the surface applies to the object due to sliding friction. The set of vectors orthogonal to this intersection is the set of object motions that the robot can induce. This intersection is a cone. In the interior of this cone, the interaction between the robot and the object is sticking, and sliding when on the boundary of the cone.

Note that $\hat{\boldsymbol{v}}_r$ and $\hat{\boldsymbol{v}}_l$ are the unit vectors of object motion in the object frame. We can project them to the contact point using the contact Jacobian:

$$\hat{\boldsymbol{v}}_{c,r} = \mathbf{J}_c\hat{\boldsymbol{v}}_r$$
$$\hat{\boldsymbol{v}}_{c,l} = \mathbf{J}_c\hat{\boldsymbol{v}}_l$$

If the pusher velocity is in the interior of the cone generated by $\hat{\boldsymbol{v}}_{c,r}$ and $\hat{\boldsymbol{v}}_{c,l}$, then the interaction is sticking, else it is sliding. If the interaction is sticking then $\boldsymbol{v}_o = \boldsymbol{v}_p$. If the interaction is sliding to the left (right), then $\boldsymbol{v}_o = \hat{\boldsymbol{v}}_l$ ($\hat{\boldsymbol{v}}_r$).

2.3.3 Stable Pushing

Naturally, controlling the motion of an object with just one finger is difficult. Applying a force at the point of contact imparts a linear force to the object. With just two control inputs, we can only control two degrees of freedom at any given time. The object has 3 degrees of freedom, so we would need to plan far into the future for a complex sequence of pushes that aim to control the linear and rotational degrees of freedom.

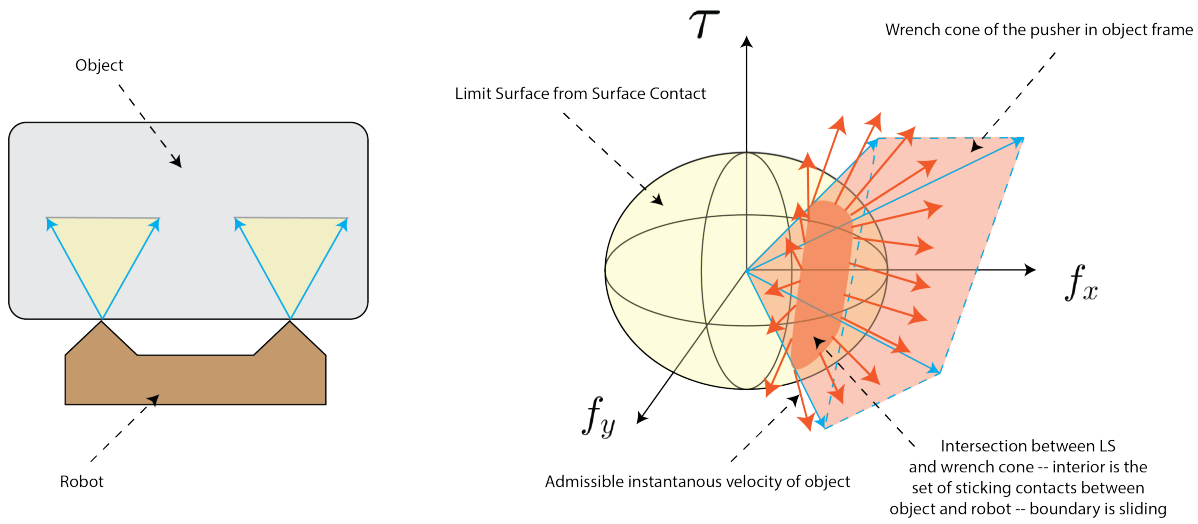

Figure 2.19. Stable pushing system and a geometric interpretation of the motion model.

An alternative strategy to single finger pushing is to use two points or a line contact. In this section, we follow the derivations of Lynch and Mason (1996) and Mason (1986b) and point the interested reader to these great texts for details.

The objective of the robot is to push the object depicted in the left panel of Fig. 2.19 to some desired pose. The robot makes a two point contact with the object and is position control. Conveniently, any line contact is equivalent to a two point contact at the ends of the line and the associated pushing problem is treated identically to our case without loss of generality. We will begin with a geometric derivation of the motion of the object induced by the motion of the robot.

We know that the set of all forces imparted by the robot to the object at the contacts must:

1. pass through the two points of contact,
2. obey the Coulomb friction constraint.

Each contact points can import a force within the friction cone at that point. We can project each friction cone into the object frame using the contact Jacobian to derive the composite wrench cone. This is an identical process to the grasp matrix derivation and the single-finger pusher cases. The set of all contact forces imparted to the object, in the object frame, is due to the sum of wrench cones in the object frame.

We recall that the limit surface maps support surface frictional forces to object motion. We know that the sum of the external forces applied by the robot through the two point contacts and the support fictional forces must be equal and opposite. To find the resulting object motion due to the applied robot motion, we can compute the intersection between the composite fiction wrench cone and the limit surface, as shown in the right panel of Fig. 2.19.

The resulting intersection is referred to as the motion cone by Mason (1986b). The motion cone characterizes the set of all possible object motions that the robot can induce, given the current contact formation. An important implication is that the object can only move along this set of instantaneous velocities – these are the only possible motions. If the robot and object slide w.r.t. to each other, the object motion is on the boundary of the cone, and if their interaction is sticking, then the object velocity is in the interior of the cone.

To derive the equations of motion, we follow the procedure of single finger pushing. We compute the motion cone at the points of contact and so long as the contact velocities lie within the motion cones, then the object and pusher will behave as a single rigid-body. By specifying the velocity of the pusher (so long as the velocities at the point of contacts obey the motion cone) then we can move the object as desired.

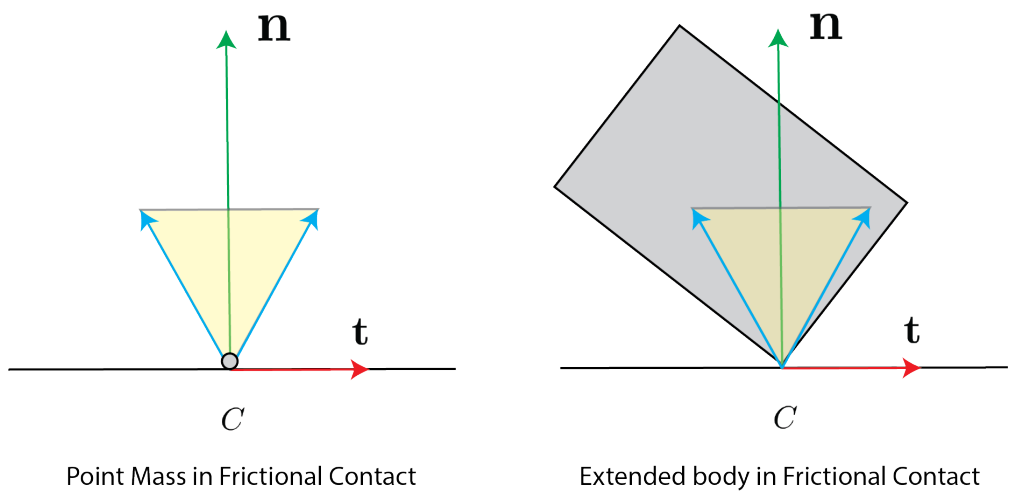

Figure 2.20. Point mass and extended body frictional interactions, now studied under the dynamics assumption.

2.4 Dynamic Rigid-Body Contact Mechanics

To start our dynamic study of contact mechanics we ask: "Given a planar rigid-body in contact with one or more immobile objects; given initial conditions consistent with the contact conditions; and a given force and torque applied to the object; what are the possible set of motions the object can undertake?" An examples of this scenario are depicted in Fig. 2.20 for a point mass and an extended body. Our derivation follows Erdmann (1994).

We will study the dynamic single rigid-body contact mechanics question in two cases: 1) the point mass, and 2) the extended body. The point mass is infinitesimally small so we do not have the worry about it's geometric properties. This simplification allows us to focus on the frictional interaction agnostic to the geometry of the object. The extended body case introduces the complexity of geometry into the problem. In contrast to our analyses in the previous sections, we will now consider the interactions with significant enough accelerations such that we have to explicitly account for them.

An important note to make is that the geometry of the friction cone and how we model friction, i.e. Coulomb friction, remains the same. These objects are not affected by the dynamic nature of the problem.

2.4.1 Point Mass Rigid-Body Contact Mechanics

Equations of Motion

The equations of motion we describe here give us a way of computing the acceleration of the object subject to the external forces applied to it and the frictional reaction forces from contact. By solving for acceleration at every point in time and integrating the information, we can compute the time evolution (a.k.a. trajectories) of the velocity and position of the object.

We recall that our object moves in the plane and that we need 3 numbers to describe its configuration (2 for position and 1 for orientation), 3 more for velocity, and 3 more for acceleration. These 9 numbers are not independent, by integration we can compute velocity and configurations from the accelerations of the object. Consider the point object depicted in Fig. 2.21. Since the object is a point, we can only apply linear external and reaction forces, there is no way to apply a torque to an object that has no physical extension (i.e. lever arm). We use the following notation:

- External force: $\boldsymbol{f}_e = (f_x, f_y)$,
- Reaction force: $\boldsymbol{f}_r = (f_t, f_n)$,
- Object acceleration: $\boldsymbol{a} = (a_x, a_y)$,
- Object velocity $\boldsymbol{v} = (v_x, v_y)$,

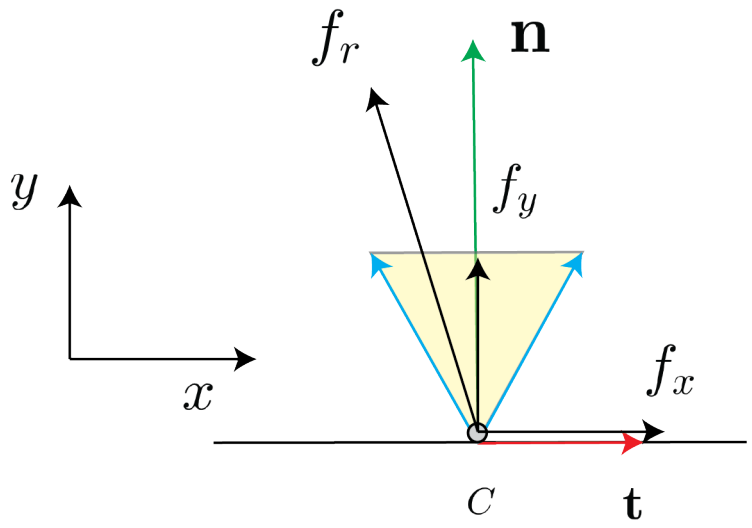

Figure 2.21. Point mass with the forces illustrated.

- Mass: m

We'll assume the object starts at rest and write down Newton's equations of motion:

$$\begin{aligned} f_x + f_t &= ma_x \\ f_y + f_n - mg &= ma_y \\ |f_t| &\leq \mu |f_n| \end{aligned}$$

Now let's do some analysis: We know that the surface does not move and the set of permissible normal accelerations is $a_y \geq 0$. If we apply a normal force (f_y) into the plane then $a_y = 0$ and we know that $mg - f_y = f_n$, i.e. the reaction force cancels the normal force applied and contact is maintained. If we apply $f_y > mg$ then the object and surface separate, i.e. $f_n = f_t = 0$. Let's focus on the case where contact is maintained.

Since $a_y = 0$ we note that $f_n = mg - f_y$. From Coulomb's friction constraint, we have one of two scenarios. First, if:

$$|f_x| \leq \mu |f_n| = \mu |f_y - mg|$$

implies sticking contact. In this scenario, $a_x = 0$ and the mass does not move (effectively rigidly attached to the surface). The second scenario is when

$$|f_x| \geq \mu |f_n| = \mu |f_y - mg|$$

then the mass moves and:

$$f_x + f_t = f_x - \mu \operatorname{sign}(v_x) |f_y| = ma_x$$

the reaction force resists the external force and tendency of motion (expressed here using the sign function) and takes on it's maximum value on the boundary of the friction cone. The tangential reaction force saturates on the boundary of the friction cone and cannot exceed this value.

In our treatment so far, we have assumed that the initial velocity of the mass is zero. This meant that we had to calculate whether the tangential force is sufficient to overcome friction and induce motion. If $v_x \neq= 0$, then the frictional force f_t always lies on the boundary of the friction cone and is equal to $\mu |f_n|$ with sign opposing that of v_x. In summary:

$$f_t = \begin{cases} -f_x & \text{if} \quad v_x = 0 \wedge |f_x| \leq \mu |f_y| \\ -\mu \operatorname{sign}(v_x) |f_y| & \text{else} \end{cases}$$

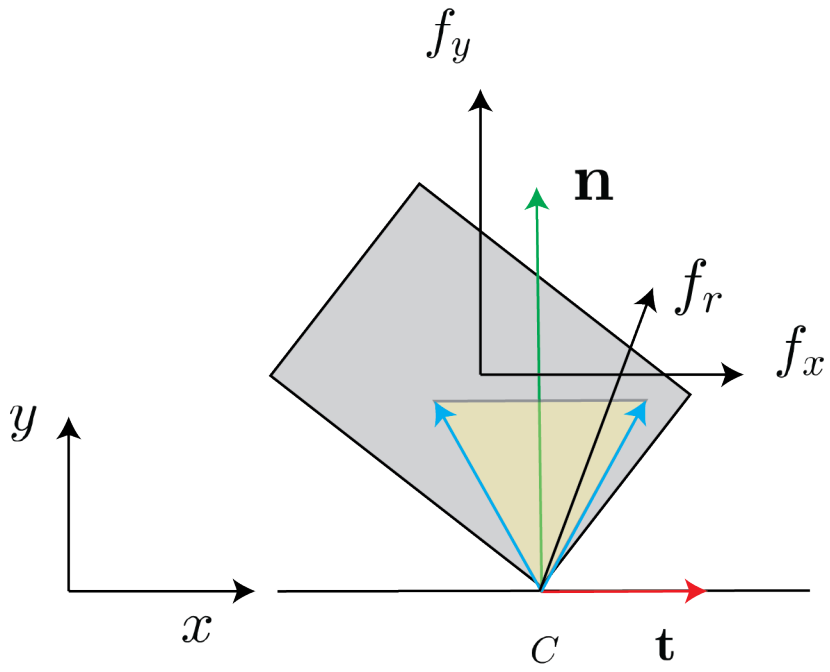

Figure 2.22. Extended body with the external forces illustrated.

Contact Modes for the Point Mass

So far, we have encountered 3 different contact "modes": 1) no contact, 2) in contact and sticking (abbreviated as sticking), and 3) in contact and sliding (abbreviated as sliding). We define contact modes (loosely) as a regimes of continuous dynamics separated by boundaries of contact. Here, boundaries of contact refer to two bodies geometrically intersecting or not (contact or no contact) and whether the friction force is on the boundary of the friction cone or in the interior (sticking vs. sliding). These boundaries are described in the joint domain of forces and configurations. Contact modes are a convenient language to describe the hybrid dynamic induced by contact and we will use them frequently in the rest of the course.

2.4.2 Extended-Body Contact Mechanics

The hypothetical point mass case we studied is instructional in understanding the basics of the friction cone. In practice bodies have physical extents and we need to handle the complexities introduces by geometry – for example, consider the body depicted in Fig. 2.22. Our assumptions are the same as those of the point mass case.

Equations of Motion

Now that we know how to project reaction forces to the COM of the object, let's write down the equations of motion to predict the accelerations of the object:

$$\boldsymbol{f}_e + \mathbf{J}_c \boldsymbol{f}_c = \mathbf{M}\boldsymbol{a}$$

where $\mathbf{M} = \text{diag}\{m,\ m,\ I\}$ denotes the inertia matrix of the object and $\boldsymbol{a} = (a_x, a_y, \alpha)$ where α is the angular acceleration of the object. Since the reaction force is due to contact with Coulomb friction, we may also write down the following constraints:

$$\begin{aligned} f_n &\geq 0 \\ \mu f_t &\leq |f_n| \end{aligned}$$

where the first constraint (a type of uni-lateral constraint) ensures that the reaction force pushes to prevent penetration and the second constraint ensures the friction cone. Can we solve for the accelerations of the object? There are 5 unknowns in the equations of motion ($\boldsymbol{a}$ and $\boldsymbol{f}_c$) but only 3 equations. The constraints also do not help here because without knowing the motion, we cannot know predict the contact mode (sticking vs sliding vs separation). There are several ways of approaching this seemingly chicken and egg-like problem, where our objective is to resolve contact for a consistent set of post-contact states and reaction forces. We focus on one particular framework, the complementarity formulation for contact.

Linear Complementarity Formulation for Contact

In this formulation for contact resolution, the constraints and dynamics are re-written as a linear complementarity problem (LCP), i.e. a problem of the form:

$$\begin{aligned} \min_{\boldsymbol{z}} \quad & \boldsymbol{z}^T(\mathbf{V}\boldsymbol{z}+\boldsymbol{p}) \\ \text{s.t.} \quad & \mathbf{V}\boldsymbol{z}+\boldsymbol{p} \geq 0, \\ & \boldsymbol{z} \geq 0 \end{aligned}$$

Let's take a moment to analyze this optimization program. We know that the minimum value of the cost function is zero and achieved if either of the rows of $\mathbf{V}\boldsymbol{z}+\boldsymbol{p}$ or $\boldsymbol{z}$ take on the value of zero. We can write this as:

$$\mathbf{V}\boldsymbol{z}+\boldsymbol{p} \geq 0 \quad \perp \quad \boldsymbol{z} \geq 0$$

where the $\perp$ notation is interpreted element-wise and implies that at least one of expressions on either side takes on the value zero. We refer to this notation as the **canonical LCP** and refer back to it later in the notes.

The objective function of the LCP is a quadratic cost term, subject to inequality constraints that ensure it is at least equal to zero. In fact, we can write the LCP problem as a quadratic program of the form:

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \frac{1}{2}\boldsymbol{x}^T\mathbf{Q}\boldsymbol{x}+\boldsymbol{c}^T\boldsymbol{x} \\ \text{s.t.} \quad & \mathbf{A}\boldsymbol{x} \geq \boldsymbol{b}, \\ & \boldsymbol{x} \geq 0 \end{aligned}$$

where:

$$\begin{aligned} \mathbf{V} &= \begin{bmatrix} \mathbf{Q} & -\mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \\ \boldsymbol{p} &= \begin{bmatrix} \boldsymbol{c} \\ -\boldsymbol{b} \end{bmatrix} \end{aligned}$$

In the remainder of this section, we follow the derivations of Stewart and Trinkle (1996); Stewart (2000) to show how we arrive at this formulation. I recommend the mathematically inclined to read through these papers for details that we may skip over in our development.

Step 1: Discretize our dynamics: You will recall from your dynamics courses that the dynamics of rigid-body motion are governed by the general expression:

$$\begin{aligned} \mathbf{M}(\boldsymbol{q})\dot{\boldsymbol{v}}+\boldsymbol{c}(\mathbf{v},\boldsymbol{q})+\boldsymbol{g}(\boldsymbol{q}) &= \boldsymbol{f}_e \\ \boldsymbol{v} &= \dot{\boldsymbol{q}} \end{aligned}$$

The first term is the inertia-acceleration product, the second is the centrifugal and Coriolis accelerations, the third is forces due to conservative forces (gravity, springs, etc.), and the right hand side is the external forces applied to the rigid-bodies.

A first-order discrete-time approximation of these dynamics is derived by assuming:

$$\dot{\boldsymbol{x}}_t \approx \frac{\boldsymbol{x}_{k+1}-\boldsymbol{x}_k}{\Delta t}$$

where x is any continuous variable (e.g. velocity). Using this approximation we may write the equations of motion at time t as:

$$\mathbf{M}(\boldsymbol{q}_t)\frac{\boldsymbol{v}_{t+1}-\boldsymbol{v}_t}{\Delta t}+\boldsymbol{c}(\boldsymbol{v}_t,\boldsymbol{q}_t)+\boldsymbol{g}(\boldsymbol{q}_t)=\boldsymbol{f}_{e,t}$$
$$\boldsymbol{v}_t=\frac{\boldsymbol{q}_{t+1}-\boldsymbol{q}_t}{\Delta t}$$

with a little rearranging of terms we have:

$$\boldsymbol{v}_{t+1}=\boldsymbol{v}_t+\Delta t\ \mathbf{M}^{-1}(\boldsymbol{q}_t)(\boldsymbol{k}(\boldsymbol{v}_t,\boldsymbol{q}_t)+\boldsymbol{f}_{e,t})$$
$$\boldsymbol{q}_{t+1}=\boldsymbol{q}_t+\Delta t\ \boldsymbol{v}_t$$

where we have collected the $\boldsymbol{c}$ and $\boldsymbol{g}$ in $\boldsymbol{k}$ for convenience.

Step 2: Contact reaction forces and constraints: Our next step is to introduce the reaction forces due to contact and their relevant constraints. Referring back to Fig. 2.22 and the contact Jacobian we covered in the previous section, we may write the equations of motion as:

$$\boldsymbol{v}_{t+1}=\boldsymbol{v}_t+\Delta t\ \mathbf{M}^{-1}(\boldsymbol{q}_t)(\boldsymbol{k}(\boldsymbol{v}_t,\boldsymbol{q}_t)+\boldsymbol{f}_{e,t}+\mathbf{J}_c(\boldsymbol{q}_t)\boldsymbol{f}_{c,t})$$
$$\boldsymbol{q}_{t+1}=\boldsymbol{q}_t+\Delta t\ \boldsymbol{v}_t$$

Note that the contact Jacobian is a function of the current configuration. Next, we will write down the constraints due to contact:

- **Frictional reaction force and distance:** Let's define the minimum distance to contact as $\phi(\boldsymbol{q}_t)$. This function measures the distance between the closest points on two bodies and is non-negative. When $\phi(\boldsymbol{q}_t)=0$, the bodies are in contact and only in contact does a normal force f_n exist between the bodies. We write this constraint as:

  $$\phi(\boldsymbol{q}_t)\geq 0 \quad \perp \quad f_n\geq 0$$

  where the perpendicularity notation is used to signify that when one variable takes on a non-zero value, then the other must be zero. The physical intuition here is that the normal reaction force is zero when distance is greater than zero and vice versa.

- **Non-penetration:** The first constraint does not specify the magnitude of the reaction force, it just implies that the reaction force is greater than zero during contact. We know that the objects must not penetrate at the next time-step. One way to enforce this constraint is to require that:

  $$f_n\geq 0 \quad \perp \quad \boldsymbol{n}^T\boldsymbol{v}_{t+1}\geq 0$$

  Physically, this constraint means that when the normal reaction force is non-zero, then the velocity of the contact point along the contact normal at the next time-step must be zero. Intuitively, we want the normal velocity of the contact point to be zero so that contact is maintained without the objects penetrating. This type of interaction where the objects do not separate post contact is referred to as plastic.

  If the objects have an elastic interaction (i.e. they bounce), we may re-write the constraint as:

  $$f_n\geq 0 \quad \perp \quad \boldsymbol{n}^T(\boldsymbol{v}_{t+1}+\epsilon\boldsymbol{v}_t)\geq 0$$

  where $\epsilon\geq 0$ is called the coefficient of restitution and loosely defines how bouncy the interaction is. The intuition here is that contact point normal velocity is flipped in direction and scaled by ϵ, implying separation of the bodies.

- **Relative motion constraint:** With the above two constraints, we can resolve for the normal contact force; however, we have not said anything about the tangential frictional reaction force or it's direction

yet. Denoting the 2 tangential vectors as $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$ with the relationship $\boldsymbol{t}_2 = -\boldsymbol{t}_1$ and corresponding tangent force magnitudes $f_{t,1}$ and $f_{t,2}$, we can write:

$$\begin{aligned} f_{t,1} \geq 0 \quad &\perp \quad \lambda + \boldsymbol{t}_1^T \boldsymbol{v}_{t+1} \geq 0 \\ f_{t,2} \geq 0 \quad &\perp \quad \lambda + \boldsymbol{t}_2^T \boldsymbol{v}_{t+1} \geq 0 \end{aligned}$$

which can conveniently be written as:

$$\boldsymbol{f}_t \geq 0 \quad \perp \quad \lambda \boldsymbol{e} + \mathbf{D}^T \boldsymbol{v}_{t+1} \geq 0$$

where $\boldsymbol{e} = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$ and we have collected the tangent vectors and the tangent force magnitudes in $\mathbf{D} = \begin{bmatrix} \boldsymbol{t}_1 & \boldsymbol{t}_2 \end{bmatrix}$ and $\boldsymbol{f}_t$ respectively. λ is a scalar variable that determines the relative tangential velocity of the contact point. If $\lambda = 0$ then the relative velocity is zero. The physical intuition behind λ will become more clear in the following constraint.

The intuition behind this constraint is that the tangential force along the direction of relative motion

- **Friction cone constraint:** We now encode Coulomb friction using the constraint:

$$\lambda \geq 0 \quad \perp \quad \mu f_n - \boldsymbol{e}^T \boldsymbol{f}_t \geq 0$$

The intuition behind this constraint is that if there is relative motion between the objects ($\lambda \geq 0$) then the friction force must lie on the friction cone, else the friction force is in the interior of the friction cone.

Step 3: Consolidate the equations of motion and constraints: With the equations of motion and constraints in hand, our task is to assemble them into the LCP formulation we outlined earlier in this section. To this end, we write:

$$\begin{aligned} \boldsymbol{v}_{t+1} &= \boldsymbol{v}_t + \Delta t \, \mathbf{M}^{-1}(\boldsymbol{q}_t) \left(\boldsymbol{k}(\boldsymbol{v}_t, \boldsymbol{q}_t) + \boldsymbol{f}_{e,t} + \mathbf{J}_c(\boldsymbol{q}_t) \boldsymbol{f}_{c,t} \right) \\ \boldsymbol{q}_{t+1} &= \boldsymbol{q}_t + \Delta t \, \boldsymbol{v}_t \\ \phi(\boldsymbol{q}_t) \geq 0 \quad &\perp \quad f_n \geq 0 \\ f_n \geq 0 \quad &\perp \quad \boldsymbol{n}^T (\boldsymbol{v}_{t+1} + \epsilon \boldsymbol{v}_t) \geq 0 \\ \boldsymbol{f}_t \geq 0 \quad &\perp \quad \lambda \boldsymbol{e} + \mathbf{D}^T \boldsymbol{v}_{t+1} \geq 0 \\ \lambda \geq 0 \quad &\perp \quad \mu f_n - \boldsymbol{e}^T \boldsymbol{f}_t \geq 0 \end{aligned}$$

where now $\mathbf{J}_c = \begin{bmatrix} \boldsymbol{n} & \mathbf{D} \end{bmatrix}$ and $\boldsymbol{f}_{c,t} = \begin{bmatrix} f_n & \boldsymbol{f}_t \end{bmatrix}^T$. The constraints are hinting at the LCP formulation we are aiming towards; however, the equations of motion are not the appropriate form. Before continuing our derivation, we will make a note about the first complementarity constraint (distance and normal force):

Contact detection and the distance function: In practice, we compute the distance function at every time-step and when its value is above a threshold ϵ_ϕ, we set $f_n = 0$ and continue to integrate the equations of motion. Only when the distance function is sufficiently close to zero do we assume contact has occurred and $f_n \geq 0$. We do this because time-stepping and the integration step mean that the distance function changes non-smoothly, which means that the objects will almost never be in contact. They will either be slightly apart or slightly penetrating. Modern simulators play a number of interesting numerical tricks to better approximate the instant of contact; however, this treatment is beyond the scope of this course.

With the note above in mind, we replace $\boldsymbol{v}_{t+1}$ from the equations of motion into the constraints and write:

$$\underbrace{\begin{bmatrix} \Delta t \, \boldsymbol{n}^T \mathbf{M}^{-1} \boldsymbol{n} & \Delta t \, \boldsymbol{n}^T \mathbf{M}^{-1} \mathbf{D} & 0 \\ \Delta t \, \boldsymbol{D}^T \mathbf{M}^{-1} \boldsymbol{n} & \Delta t \, \boldsymbol{D}^T \mathbf{M}^{-1} \mathbf{D} & \boldsymbol{e} \\ \mu & -\boldsymbol{e}^T & \end{bmatrix}}_{\mathbf{V}} \underbrace{\begin{bmatrix} f_n \\ \boldsymbol{f}_t \\ \lambda \end{bmatrix}}_{\boldsymbol{z}} + \underbrace{\begin{bmatrix} \boldsymbol{n}^T \boldsymbol{b}_n \\ \mathbf{D}^T \boldsymbol{b}_t \\ 0 \end{bmatrix}}_{\boldsymbol{p}} \geq 0 \perp \begin{bmatrix} f_n \\ \boldsymbol{f}_t \\ \lambda \end{bmatrix} \geq 0$$

$$\begin{aligned} \boldsymbol{b}_n &= (1 + \epsilon) \boldsymbol{v}_t + \Delta t \, \mathbf{M}^{-1} (\boldsymbol{k} + \boldsymbol{f}_e) \\ \boldsymbol{b}_t &= \boldsymbol{v}_t + \Delta t \, \mathbf{M}^{-1} (\boldsymbol{k} + \boldsymbol{f}_e) \end{aligned}$$

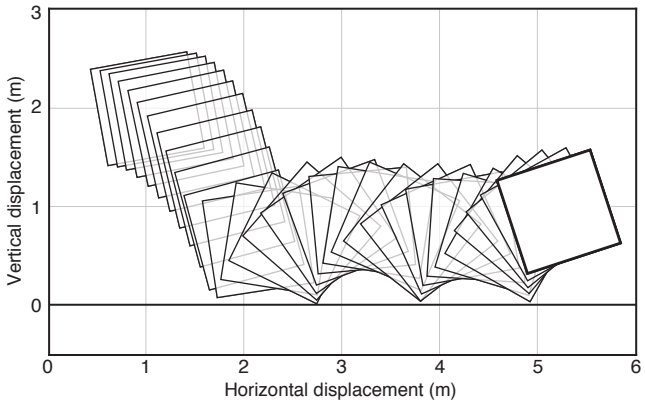

Figure 2.23. Example of dynamic bouncing with frictional contact.

where the equations of motion and contact constraints are written in the **canonical LCP** form we outlined at the beginning of this subsection.

Time-stepping and the simulation algorithm

Now that we have derived the equations of motion and the corresponding contact constraints, we would like to simulate the object undergoing frictional interaction. To do this, we use the following algorithm:

Algorithm 1: Simulate Rigid-body Basic

Data: Object Geometry, Object Inertial Properties, Friction, Restitution, Initial Conditions
Result: Object Trajectory
initialization: Set object initial conditions;
while $t < T$ **do**
  flag = checkCollision($\boldsymbol{q}_t$) ;
  if *flag* **then**
    $\boldsymbol{q}_{t+1}, \boldsymbol{v}_{t+1}$ = solveLCP($\boldsymbol{q}_t, \boldsymbol{v}_t$) ;
  else
    $\boldsymbol{v}_{t+1} = \boldsymbol{v}_t + \Delta t \mathbf{M}^{-1}(\boldsymbol{k}_t + \boldsymbol{f}_{e,t})$;
    $\boldsymbol{q}_{t+1} = \boldsymbol{q}_t + \Delta t \boldsymbol{v}_t$
  $t = t + 1$;

There are two important subroutines in this most basic form of simulation algorithm: *checkCollision* and *solveLCP*. The former routine checks for geometric intersections between bodies and the latter solves the canonical LCP we formulated in the previous subsection.

Collision checking is, typically, the most computationally expensive operation of rigid-body simulators. This is because computing the geometric intersection of complex shapes tends to be expensive. In many practical simulators, complex geometries are approximated by primitive shapes to speed up the operation. The details of implementation are beyond the scope of our study. An important role of collision detection is to identify where the contact occurs on bodies. This information is used to compute Jacobians.

Fig. 2.23 shows an example of simulating the dynamical frictional interaction using the algorithm we described.

Bibliography

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

Goyal, S., Ruina, A., and Papadopoulos, J. (1991). Planar sliding with dry friction part 1. limit surface and moment function. *Wear*, 143(2):307–330.

Howe, R. D. and Cutkosky, M. R. (1996). Practical force-motion models for sliding manipulation. *The International Journal of Robotics Research*, 15(6):557–572.

Lozano-Perez, T. (1990). *Spatial planning: A configuration space approach*. Springer.

Lynch, K. (1992). The mechanics of fine manipulation by pushing. In *Proceedings 1992 IEEE International Conference on Robotics and Automation*, pages 2269–2276. IEEE.

Lynch, K. M. (1996). *Nonprehensile robotic manipulation: Controllability and planning*. Citeseer.

Lynch, K. M., Maekawa, H., and Tanie, K. (1992). Manipulation and active sensing by pushing using tactile feedback. In *1992 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 1992*, pages 416–421. Institute of Electrical and Electronics Engineers Inc.

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

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

Mason, M. T. (1986b). On the scope of quasi-static pushing. In *Proceedings of International Symposium on Robotics Research*, pages 229–233, Cambridge, Mass. MIT Press.

Mason, M. T. (2001). *Mechanics of robotic manipulation*. MIT press.

Murray, R. M., Li, Z., Sastry, S. S., and Sastry, S. S. (1994). *A mathematical introduction to robotic manipulation*. CRC press.

Prattichizzo, D. and Trinkle, J. C. (2016). Grasping. In *Springer handbook of robotics*, pages 955–988. Springer.

Stewart, D. E. (2000). Rigid-body dynamics with friction and impact. *SIAM review*, 42(1):3–39.

Stewart, D. E. and Trinkle, J. C. (1996). 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.