Iowa State University
Digital Repository

Graduate Theses and Dissertations

Iowa State University Capstones, Theses and Dissertations

2014

Displacement-based grasping of deformable objects

Feng Guo
Iowa State University

Follow this and additional works at: https://lib.dr.iastate.edu/etd

Part of the Robotics Commons

Recommended Citation

Guo, Feng, "Displacement-based grasping of deformable objects" (2014). *Graduate Theses and Dissertations*. 14136.
https://lib.dr.iastate.edu/etd/14136

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.

Displacement-based grasping of deformable objects

by

Feng Guo

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Major: Computer Science

Program of Study Committee:

Yan-Bin Jia, Major Professor

David Fernández-Baca

Greg R. Luecke

James Oliver

Guang Song

Iowa State University

Ames, Iowa

2014

Copyright © Feng Guo, 2014. All rights reserved.

DEDICATION

I would like to dedicate this dissertation to my wife Yueran Yang and my parents without whose support I would not have been able to complete this work.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my thanks to people who gave me all kinds of help in my study.

First and foremost, I would like to thank Dr. Yan-Bin Jia, who has given me tremendous support, inspiring guidance and valuable help on my research. It was such a pleasure to work with him during the Ph.D career. He has always been there when I have difficulties in research and guide me towards a thinker.

I am also greatly thankful to my committee members, Dr. David Fernández-Baca, Dr. Greg R. Luecke, Dr. James Oliver and Dr. Guang Song, for their help and contribution to this work. It is such an honor to have them as my committee member and learn from them.

The Robotics Lab has been a great place for study and research. All the lab members, Jiang Tian, HyunTae Na, Theresa Driscoll, Huan Lin, Rex Fernando, Feifei Wang, Sean Strickland, Trenton Anagnostopoulos, Sheng Bi and Jiale Feng have made it a lovely place. I am grateful to them all.

I would also thank my friends Yetian Chen, Rui Han, Ru He, Wangyuejue Hong, Xiang Huang, Chuan Jiang, Tsing-yi Jiang, Zi Li, Qiuyan Liao, Yu Liu, Lisen Peng, Yang Peng, Hua Qin, Chuang Wang, Wanwu Wang, Ting Wu, Liyuan Xiao, Jinsheng Zhang, Wei Zhang, Wen Zhao, Ming Zhou. Their presences make Ames a great home to me.

Support for this research has been provided in part by Iowa State University, and in part by the National Science Foundation through the grant IIS-0915876. Any opinions, findings and conclusions or recommendations in this dissertation are those of the author and do not necessarily reflect the views of the National Science Foundation

ABSTRACT

Robotic grasping of deformable objects is inherently different from that of rigid objects, and is an under-researched area. Difficulties arise not only from expensive deformable modeling, but also from the changing object geometry under grasping force.

This dissertation studies strategies of grasping deformable objects using two robotic fingers. Discovering the inapplicability of the traditional force-centered grasping strategies for rigid objects, I have designed an approach for grasping deformable objects that specifies finger displacements. This not only ensures equilibrium under the elasticity theory, but also enhances stability and simplifies finger control in the implementation.

Deformable modeling is carried out using the Finite Element Method (FEM), for which our analysis establishes uniqueness of the shape of a grasped object given the finger displacements. Meanwhile, preprocessing based on the Singular Value Decomposition greatly reduces the complexity of computation. Grasping strategies have been investigated on a range of objects, including 2D hollow and solid, and 3D ones. With a hollow 2D object, the grasping fingers make point contacts. The condition of a successful grasp is that the friction cones at the two contacts must contain the line segment through them before and after the deformation, indicating equilibrium throughout the grasping process. With a solid planar object, the fingers make area contacts. Grasp computation is carried out by an event-driven algorithm, which has been validated by our robot experiments. For 3D objects, a simple squeeze-and-test strategy has been designed to lift them off the supporting table against gravity with a method that predicts the squeeze amounts.

In reality, objects' shapes are affected to various degrees by gravity, but such a effect has been ignored in the FEM-based modeling. For accuracy, the gravity-free shape of an object is sometimes needed. I have introduced an iterative algorithm that will converge to such shape as a "fixed point".

In the last part of my thesis, I study planning of the finger squeeze paths, not to limited by straight movements. The objective is to not only enlarge the range of finger placements for successful grasps, but also improve stability and energy efficiency. I have designed a path planning algorithm based on the Rapidly-exploring Random Trees (RRT) that is able to achieve certain optimalities.

CHAPTER 1. INTRODUCTION

Grasping deformable objects is quite different from grasping rigid ones. Two types of analysis have been developed for the latter. Form closure [37] means the object cannot move given the fingers are fixed, while force closure [32] grasps resist any arbitrary wrench and keep the object in equilibrium. However, deformable objects have infinite degrees of freedom, which makes form closure impossible. On the other hand, the grasp wrench space changes as the object deforms, which makes it impossible to conduct any conventional force closure analysis as well.

Grasping deformable objects is an under-researched area, primarily due to the reasons from both mechanics and computation. More complicated than rigid body grasping, deformable object grasping must keep the grasped object in equilibrium at every grasping configuration rather than just one scenario. The changing geometry of the object under grasp makes it necessary to track the contact configuration. Physics-based deformation modeling usually depends on FEM or BEM, which introduces big linear systems that usually takes cubic time to solve in terms of the object's resolution.

When the deformation is small, the stress-strain relationship of every node of the deformable objects can be modeled using the linear elasticity theory, in which the applied force and the displacement of the contact are strictly related and thus cannot be both specified at the same time. In contrary to the tradition of specifying the grasping force, this dissertation chooses to specify desired displacements of the fingers for the following reasons:

1. In practice, it is much easier to control the finger's displacement than the force it exerted. Controlling the robotic hands' movement or locations is by far the most common, direct, easy way of manipulation.

2. The exact grasping force, especially that in deformation process, is not very much concerned, as long as the object can be grasped.

3. Specifying the displacement gives rise to certain constraints that are sufficient for determining the deformation and corresponding force.

Computation of deformed shape based on linear elasticity comes down to solving either a system of fourth order differential equations, which has no closed-form solution in general, or practically, a large linear system using FEM or BEM. The latter takes subcubic time in the number of discretization nodes, which is typically high for accurate modeling. A large deformation can only be modeled by nonlinear elasticity and computed using the even more expensive nonlinear FEM.

The lack of a closed form description of the deformed shape implies that (part of) the shape needs to be computed repeatedly with hypothesized finger placements in order to compute a single grasp. Computational efficiency has thus become a bottleneck, even more so for grasp optimization and real-time implementation. Whether a finger placement with certain finger displacement can form a grasp without slip depends on the local geometry of the contacts. Therefore global deformation is not needed. The stiffness matrix of the object stays the same for different grasp tests, although the boundary conditions may vary. An improvement in computation is possible by preprocessing the stiffness matrix.

Grasping deformable 3D objects differs from the 2D version of the task in several aspects beyond just adding one more dimension. Gravity can no longer be ignored given the volume (and thus the weight) of a 3D object. Firstly, it directly affects the shape of the object as it rests on the supporting surface. Secondly, when the object is being lifted, complex interactions take place in the contact regions inside which every contact could slide in a continuum of directions.

Like our handling of the 2D grasping task, we sequence the entire operation into periods within each of which the contact configuration under a finite element discretization does not change. During a single period, the displacements of the contact nodes are either known or estimable from the finger movements. From them we can uniquely determine the object's deformation, which in turn causes new change in the contact configuration for the next period.

The finite element method (FEM) [8] often constructs the stiffness matrix based on the body's shape, neglecting the fact that it is already deformed under gravity, and then applies the matrix to deformable modeling with or without accounting for the gravitational force. However, nonlinearity of the stiffness matrix in the body geometry suggests that accurate modeling of the effect of gravity on deformation, more prominent over a 3D solid, needs to resort to the stiffness matrix under zero gravity. A fixed point iteration method was used to restore the gravity-free shape out of the observed shape, deformed under gravity.

When a human hand grasps an object, the movements of the fingers trace out trajectories, which, more often than not, are curved rather than straight. This observation indicates that curved grasping paths may be superior to straight ones, and planning such trajectories help improve the grasping power. The problem of finding out such curves is solved in two steps. Firstly, a goal finger displacement is calculated. Secondly, a piecewise straight path is planned in a high dimensional configuration space from one point, indicating zero finger displacement, to another, indicating the goal displacement. An algorithm using Rapidly-exploring Random Trees (RRT) was introduced to plan such trajectories. Compared with performing straight squeeze, the grasping fingers following the planned trajectories may not only achieve a grasp at the places where straight squeezing cannot, but also grasp the object with certain optimalities.

1.1 Assumptions

In this dissertation, I deal with only objects that can be described by linear elasticity theory with small deformations. The grasping process is treated as a quasi-static process and no dynamics is considered. More specifically, we make the following assumptions.

(A1) The object to be grasped is isotropic and can be described by linear elasticity theory.

(A2) The deformation is small enough to be handled by linear elasticity theory.

(A3) The grasping process is quasi-static and no dynamics is considered.

1.2 Dissertation Outline

This dissertation describes several deformable object grasping strategies. Chapter 2 reviews the related research that has been done. Chapter 3 will briefly review linear elasticity and FEM with derivations of some basic results to be used later. In Chapter 4, we will show that it is possible to grasp a deformable object by squeezing it with two fingers moving toward each other along a straight line, as long as the connection line of the two contacts stay inside the contact friction cones before and after deformation. We will also show that actions other than pure squeeze can also result in grasps if so does the corresponding pure squeeze. An $O(n)$ time algorithm for grasp testing is presented, where n is the number of vertices in FEM, after obtaining the singular value decomposition (SVD) of the object's stiffness matrix. The cost of finding all grasps reduces to $O(n^2)$. It turns out that the pre-process using Singular Value Decomposition (SVD), which takes $O(n^3)$ time, dominates the overall computation. Chapter 5 introduces the contact mode analysis and event driven algorithm to grasp the 2D solid objects. Chapter 6 gives an algorithm to squeeze and lift 3D objects deformed under gravity. Chapter 7 studies the problem of obtaining the original shape from gravity-influenced shape observed for precise modeling. Chapter 8 introduces algorithms that plans the grasping fingers' trajectories to enhance the grasping ability and efficiency. Discussion will follow in Chapter 9 to conclude the dissertation.

CHAPTER 2. REVIEW OF LITERATURE

Rigid body grasping is an extensively studied area rich with theoretical analyses, algorithmic syntheses, and implementations with robotic hands [2]. Form closure grasp eliminates the degree of freedoms of the grasped objct [37]. The upper bounds of the sufficient number of contacts is given in [30] [26]. Algorithms were developed for computing form closure grasps [4] [44].

Two-finger force-closure grasps of 2-D objects are well understood and efficiently computable for polygons [32] and piecewise-smooth curved shapes [34]. Algorithms for 3D objects are given in [35].

The task ellipsoid [22] notion formalized the idea that the choice of a grasp ought to be based on its capacity to generate wrenches that were relevant to the task. Grasp quality measures for robotic hands with multiple fingers considered selection of internal grasping forces that were furthest from violating any closure, friction, or mechanical constraints [18], or were directly derived from the grasp matrix which characterized the wrench space of a grasp [22]. Grasp metrics for polygons and polyhedra often try to maximize the worst-case external force that could be resisted by a unit grasping force [25] [28]. A summary was offered in [29] on various grasp metrics, addressing the trade-offs among grasp goodness, object geometry, the number of fingers, and the computational complexity for grasp synthesis. Some recent work [5] [3] focused on minimizing the maximum magnitude of the applied force at any frictional contact of a grasp in order to maintain equilibrium against a known adversary wrench, via employing semidefinite programming techniques.

Much fewer work exists on grasping of deformable objects, which needs to deal with accurate modeling of deformations caused by the grasping fingers. The concept of bounded force-closure was proposed in [47]. Visual and tactile information was effective on controlling the motion of a grasped deformable object [14]. The deformation-space (D-space) approach [12] characterized the optimal grasp of a deformable part as one where the potential energy needed to release the part equals the amount needed to squeeze it to its elastic limit– hence the object could not escape.

It has been a very active area on manipulation of flexible linear objects such as wires or ropes. There are works on static modeling [46], knotting and unknotting [40], [27], [19], [45], pickup [36], and path planning [31]. The operations, however, can be carried out without a serious need for deformable modeling.

In [41], a model for deformable contact regions under a grasp was introduced to predict normal and tangential contact forces with no concern of grasp computation or modeling of global deformation. Simulation accuracy and efficiency could be improved based on derived geometric properties at deformable contact [24]. Deformation modeling of shell-like objects that have been grasped is studied in [43].

More thorough investigations were conducted by the mechanics community on the elastic contact problem concerned with two deformable bodies under a known applied load. The gradual nature of the physical process suggests iterative updates of the growing contact region(s). Solutions based on FEM were given to objects with [33] or without [9] friction.

Robotic path planning is an extensively studied area. We refer to [6] for a survey on this topic. Particularly the Rapidly-exploring Random Trees (RRT) [20] algorithm was introduced to find a path between known origin and goal.

This dissertation includes parts or all of several published works including the author. The grasping force, instead of finger displacements, was specified in [16]. Extra constraints, which lead to unrealistic requirements, had to be imposed for computing the deformed shape. The corresponding grasp space (i.e., the set of feasible finger placements) was 1-D, and the synthesis algorithm was too inefficient to be applicable to solid 2-D objects. In [13], the idea of displacement-based grasp was proposed and an event-based algorithm was introduced to calculate the grasping of solid 2D objects. The extended version of this work is [15]. The optimality of the grasping was studied based on an energy criteria [17]. The concept of stable squeeze and pure squeeze were introduced.

CHAPTER 3. FINITE ELEMENT METHOD

The first part of this chapter reviews the 2D linear elasticity. The displacement field which does not generate strain energy is characterized. The second part describes the Finite Element Method used to model the deformation. The null space of the stiffness matrix is shown. A Singular Value Decomposition analysis is performed on the stiffness matrix in the third part. The result will later be used in our design of a grasping strategy.

3.1 Linear Plane Elasticity

Consider a thin flat object as is shown in Figure 3.1, the thickness h of which is dominated by the other two dimensions. The object is bounded by a generalized cylinder. Here we consider the *plane stress* [10] parallel to the xy-plane, which assumes zero normal stress σ_z and shear stresses τ_{xz} and τ_{yz} in the xz and yz planes.

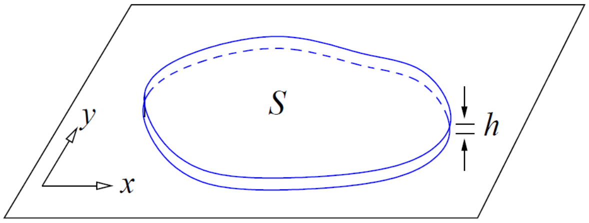


Figure 3.1: Planar object

Under a displacement field $(u(x,y), v(x,y))$, every point of the object moves to $(x+u, y+v)$. The normal strains ε_x, ε_y and the shearing strain γ_{xy} within every cross section are given below:

$$\begin{aligned}\varepsilon_x &= \frac{\partial u}{\partial x},\\ \varepsilon_y &= \frac{\partial v}{\partial y},\end{aligned} \tag{3.1}$$

$$\gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}.$$

The strain energy can be derived as [7]:

$$U = \frac{h}{2} \iint\limits_{S} (\frac{E}{1-v^2}(\varepsilon_x^2 + 2v\varepsilon_x\varepsilon_y + \varepsilon_y^2) + \frac{E}{2(1+v)}\gamma_{xy}^2)\,\mathrm{d}x\mathrm{d}y, \tag{3.2}$$

where E and v are Young's modulus and Poisson's ratio of the material, resp., with $E > 0$ and $-1 \leq v \leq \frac{1}{2}$.

Theorem 1. *Under linear elasticity, any displacement field $(u(x,y), v(x,y))$ that yields zero strain energy is linearly spanned by three fields: (1, 0), (0, 1) and $(-y, x)$.*

Proof. Suppose $U = 0$ under a displacement field (u, v). From (3.2) we see that the strains ε_x, ε_y and γ_{xy} must vanish everywhere. From (3.1),

$$\begin{aligned} u &= \int \varepsilon_x \,\mathrm{d}x + f(y) = \int 0 \,\mathrm{d}x + f(y) = f(y), \\ v &= \int \varepsilon_y \,\mathrm{d}y + g(x) = \int 0 \,\mathrm{d}y + g(x) = g(x), \end{aligned}$$

where f and g are arbitrary single variable functions. Since $\gamma_{xy} = 0$, $\mathrm{d}u/\mathrm{d}y + \mathrm{d}v/\mathrm{d}x = f'(y) + g'(x) = 0$ for all (x, y) in the body. Given f and g do not share variable, $f'(y) = -g'(x) = c$ for some constant c. Integration of the two derivatives gives

$$(u, v) = c(-y, x) + d(1, 0) + e(0, 1),$$

for some constants d and e. □

Displacement fields that generate no strain energy are essentially rigid body transformations. The fields $(1, 0)$ and $(0, 1)$ represent translation in x- and y- directions resp. The field $(-y, x)$, which displays every point (x, y) in the direction orthogonal to (x, y), corresponds to rotation around origin. Note that, as is shown in Figure 3.2, it approximates rotation well only when the rotation is small enough. When it is not, such field also inflates the original shape. When such field is large enough, the change of the orientation of the object approaches $\pi/2$. Such deviation from the real rotation indicates certain limit of linear elasticity in modeling the real world.

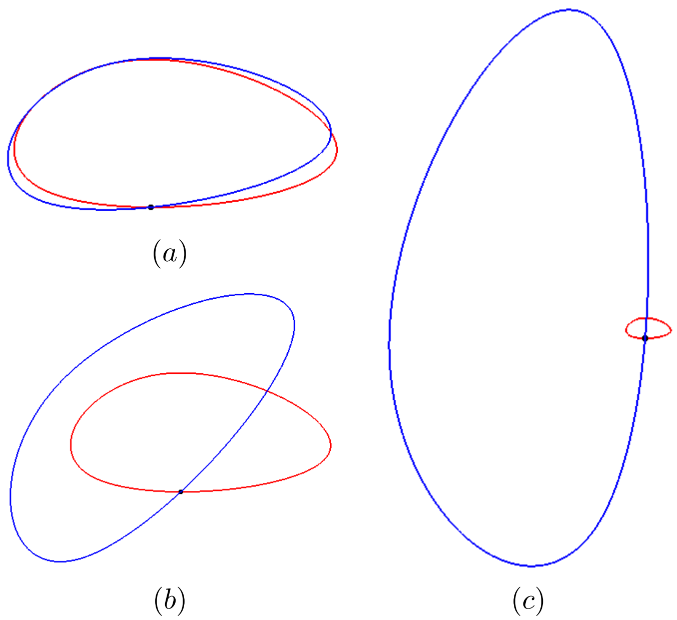


Figure 3.2: The rotation field under linear elasticity. The red shape is original shape, while the blue one shows the shape under certain rotation field. Denote the original shape as S and the rotation field as r, the blue shapes can be expressed as $S + \lambda r$, where λ is a real number. The figures (a),(b) and (c) show the resulting shape when λ is very small, certain value that is note very small and approaching infinity, resp. (b) and (c) are shrunk to fit.

3.2 Stiffness Matrix

For the rest part of the chapter, all vectors are column vectors and all indices in a vector or matrix start at 0.

Closed forms of the strain energy integrals do not exist for most objects. The Finite Element Method (FEM) is widely used to compute it (and the deformation). The object's cross section is discretized into a finite number of elements(e.g. triangles) with vertices $\mathbf{p}_0, \cdots, \mathbf{p}_{n-1}$, where $\mathbf{p}_k = (p_{kx}, p_{ky})^T$, for $0 \le k \le n-1$. Among these vertices, $\mathbf{p}_0, \cdots, \mathbf{p}_{m-1}$ where $m \le n$, are on the boundary in counterclockwise order. One example is shown in Figure 3.3.

Let $\Delta = (\delta_0^T, \cdots, \delta_{n-1}^T)^T$, where $\delta_k = (\delta_{kx}, \delta_{ky})^T$, be the displacement of $\mathbf{p}_k$, for $0 \le k \le n-1$, the displacement of any interior point of an element can be linearly interpolated over those of the vertices of the element. The displacement field and the deformed shape are thus uniquely determined by Δ. We first obtain the strain energies of individual elements, and then

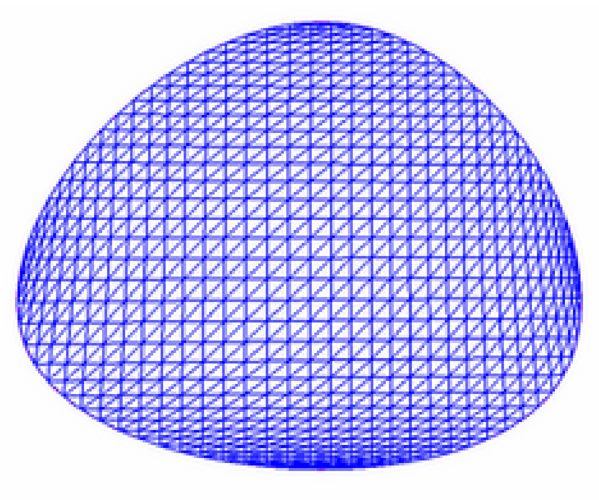


Figure 3.3: Triangular mesh with 3,120 vertices, 156 of which are on the boundary.

assemble them into the total strain energy:

$$U = \frac{1}{2}\Delta^T K \Delta, \tag{3.3}$$

where K is the $2n \times 2n$ *stiffness matrix*. The fact that K is quadratic form indicates the symmetry of K, and the non-negativeness of strain energy ensures that K is positive semi-definite.

The strain energy U is zero if and only if $K\Delta = 0$, that is, Δ is in the null space of K. Such a vector Δ gives the form of a rigid body displacement [11, pp. 48]. Meanwhile, by Theorem 1, the displacement field generating zero strain energy is spanned by $(-y, x)$, $(0, 1)$ and $(1, 0)$. Under linear interpolation, it indicates that the null space of K, where lies Δ, is spanned by the following three vectors:

$$\mathbf{v}_x = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 1 \\ 0 \end{pmatrix}, \ \mathbf{v}_y = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \\ 1 \end{pmatrix}, \ \mathbf{v}_r = \begin{pmatrix} -p_{0y} \\ p_{0x} \\ \vdots \\ -p_{n-1,y} \\ p_{n-1,x} \end{pmatrix}. \tag{3.4}$$

Here $\mathbf{v}_x$ and $\mathbf{v}_y$ translate all vertices by unit distance in the x- and y-directions, resp., while $\mathbf{v}_r$ rotates them about the origin. Note that $\mathbf{v}_r$ is orthogonal to $\mathbf{v}_x$ and $\mathbf{v}_y$ if the geometric center of the object is placed at origin.

Theorem 2. *The stiffness matrix K of an (unconstrained) object with n discretization vertices has rank $2n-3$.*

Following from Lemma 2, the matrix K has $2n-3$ positive eigenvalues $\lambda_0, \cdots, \lambda_{2n-4}$. Let $\mathbf{u}_0, \cdots, \mathbf{u}_{2n-4}$ be the corresponding unit eigenvectors, and

$$\begin{aligned} \mathbf{u}_{2n-3} &= \frac{\mathbf{v}_x}{||\mathbf{v}_x||}, \\ \mathbf{u}_{2n-2} &= \frac{\mathbf{v}_y}{||\mathbf{v}_y||}, \\ \mathbf{u}_{2n-1} &= \frac{\mathbf{z}}{||\mathbf{z}||}. \end{aligned} \tag{3.5}$$

where $\mathbf{z} = \mathbf{v}_r - (\mathbf{v}_r \cdot \mathbf{u}_{2n-3})\mathbf{u}_{2n-3} - (\mathbf{v}_r \cdot \mathbf{u}_{2n-2})\mathbf{u}_{2n-2}$, correspond to the zero eigenvalues. It follows from the Spectral Theorem [42] that

$$K = U \Lambda U^T, \tag{3.6}$$

where $U = (\mathbf{u}_0, \cdots, \mathbf{u}_{2n-1})$ is orthonormal, and $\Lambda = \text{diag}(\lambda_0, \cdots, \lambda_{2n-4}, 0, 0, 0)$.

Suppose the object is in equilibrium with the configuration $(\Delta, \mathbf{F})$. Since only boundary vertices take external force, $\mathbf{F} = (\mathbf{f}_0^T, \cdots, \mathbf{f}_{m-1}^T, 0, \cdots, 0)^T$. According to Virtual Work Principle [11, pp. 136], the *virtual work* done by the equilibrium force $\mathbf{F}$ through a *virtual displacement*[1] is equal to the change of potential energy of the object under such virtual displacement, which leads to

$$K\Delta = \mathbf{F}. \tag{3.7}$$

In equation (3.7), we have $4n$ variables, $2n$ from Δ and $2n$ from $\mathbf{F}$, and we need half of them to be known to solve for the other half. Note that since K is singular, if improper variables are picked as known, for example, the $2n$ variables of $\mathbf{F}$, we will get a space of the unknown variables rather than a specific solution. In the next chapter, constraints generated by the grasping strategy will be imposed so that the solution to the system is unique.

3.3 BEM Formulation

The Finite Element Method provides a way of formulating the problem by discretizing the object. Such method can approach the true solution when the resolution is big enough.

[1] The virtual displacement is an admissible imaginary infinitesimal displacement that is superposed to the equilibrium deformation.

However, when dealing with solid objects, FEM creates a linear system with the size quadratic to the resolution, which could be very big. In this section, another way of dealing with solid objects, Boundary Element Method (BEM), is introduced. Such method introduces a linear system only quadratic to the boundary resolution of the object, which greatly increased the computational efficiency. However, such method may still be biased even with high resolution.

3.3.1 Navier's Equations and Kelvin's Foundamental Solution

Consider an infinite elastic thin 2D plate, which is only able to deform in x- and y- direction. According to linear elasticity theory , when such deformed object achieves equilibrium, we may get *Navier's equations of equilibrium*:

$$\begin{cases} \nabla^2 u + \frac{1+\nu}{1-\nu}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 v}{\partial x \partial y}\right) + \frac{1}{G} b_x = 0, \\ \nabla^2 v + \frac{1+\nu}{1-\nu}\left(\frac{\partial^2 u}{\partial x \partial y} + \frac{\partial^2 v}{\partial y^2}\right) + \frac{1}{G} b_y = 0, \end{cases} \tag{3.8}$$

where u, v are displacements in x, y directions respectively; ν is the Poisson ratio of the object; Here $G = \frac{E}{2(1+\nu)}$ is shear modulus with E being Young's Modulus; b_i is the body force, or exerted traction on the $i-$ direction.

Consider a concentrated unit force $\mathbf{F}(F_\xi, F_\eta)$ applied at point $Q(\xi, \eta)$ of the plane. According to equation (3.8), the resulting displacement and traction at a field point $P(x, y)$ are

$$\begin{cases} U^*_{ij}(Q,P) = \frac{1}{8\pi(1-\nu)G}\left[(4\nu - 3)\delta_{ij}\ln r + r_{,i} r_{,j})\right], \\ T^*_{ij}(Q,P) = \frac{-1}{4\pi(1-\nu)r}\{[(1-2\nu)\delta_{ij} + 2r_{,i}r_{,j}]r_{,n} - (1-2\nu)(r_{,i}n_j - r_{,j}n_i)\}, \end{cases} \tag{3.9}$$

where $i, j = 1, 2$; $U^*_{ij}(Q,P), T^*_{ij}(Q,P)$ denotes the displacement and traction resp. in j direction at field point P, due to a unit load applied at the source point Q in the i direction; δ_{ij} is the *Kronecker delta*, with value 1 when $i = j$ and 0 when $i \neq j$; with $\mathbf{r} = P - Q$, r is $|\mathbf{r}|$ and $r_{,k}$ denotes the kth component of $\mathbf{r}$; $r_{,n} = r_{,x}n_x + r_{,y}n_y$ expresses the derivative of r in the direction of the outward normal to the curve (boundary) passing through the point Q.

3.3.2 Integral Representation of the Problem

According to *Betti's reciprocal theorem* of structural analysis, we have

$$\int_\Gamma t_i \bar{w}_i d\Gamma + \int_\Omega F_i \bar{w}_i d\Omega = \int_\Gamma \bar{t_i} u_i d\Gamma + \int_\Omega \bar{F_i} u_i d\Omega, \tag{3.10}$$

where Ω represents the 2D elastic region considered with Γ as its boundary curve; $\bar{w_i}$ is an arbitrary weighting function which can be viewed as displacement field; The corresponding traction and body force fields are $\bar{t_i}$ and $\bar{F}$. Substitute in the fundamental solution, we get the *Somigliana's identity*:

$$\int_\Gamma t_i(P) U^*_{ij}(Q,P) d\Gamma + \int_\Omega F_i(P) U^*_{ij}(Q,P) d\Omega = \int_\Gamma u_i(P) T^*_{ij}(Q,P) d\Gamma + \int_\Omega u_i(P) \Delta(Q,P) \delta_{ij} d\Omega. \tag{3.11}$$

With body force being neglected, the second term vanishes. By some derivation on the last term, equation (3.11) becomes

$$\epsilon \delta_{ij} u_i(Q) = \int_\Gamma [u_i(P) T^*_{ij}(Q,P) - t_i(P) U^*_{ij}(Q,P)] d\Gamma(P), \tag{3.12}$$

where ϵ is associated with the location of point Q and smoothness of the boundary Γ. When Q is on smooth boundary, $\epsilon = 0.5$. equation (3.12) is the basic equation needed in BEM formulation.

3.3.3 Discretization and BEM Matrix

Discretize the boundary into N segments. Denote the position, displacement and traction by p_i, u_i and t_i resp. Interpolate the inner points using linear interpolation:

$$\begin{aligned} p(\xi) &= N_1(\xi) p_1 + N_2(\xi) p_2, \\ u(\xi) &= N_1(\xi) u_1 + N_2(\xi) u_2, \\ t(\xi) &= N_1(\xi) t_1 + N_2(\xi) t_2, \end{aligned} \tag{3.13}$$

where

$$\begin{aligned} N_1(\xi) &= 0.5(1-\xi), \\ N_2(\xi) &= 0.5(1+\xi). \end{aligned} \tag{3.14}$$

After discretization, equation (3.12) can be written as sum of piecewise integration. Locate the source point at point i, we have

$$\frac{1}{2}\{u\}^i + \sum_{j=1}^{N} \int_{\Gamma_j} [\hat{H}]^{ij}[N]d\Gamma_j\{u\}^j = \sum_{j=1}^{N} \int_{\Gamma_j} [G]^{ij}[N]d\Gamma_j\{t\}^j, \tag{3.15}$$

where $\{u\}^i = \{u^i \; v^i \; u^{i+1} \; v^{i+1}\}^T$ and $\{t\}^i = \{t_x^i \; t_y^i \; t_x^{i+1} \; t_y^{i+1}\}^T$ denote the displacement and traction at the i-th and $i+1$-th nodes; and

$$[G]^{ij} = \begin{bmatrix} U^*_{\xi x}(q,p_i) & U^*_{\eta x}(q,p_i) \\ U^*_{\xi y}(q,p_i) & U^*_{\eta y}(q,p_i) \end{bmatrix}, \tag{3.16}$$

$$[\hat{H}]^{ij} = \begin{bmatrix} T^*_{\xi x}(q,p_i) & T^*_{\eta x}(q,p_i) \\ T^*_{\xi y}(q,p_i) & T^*_{\eta y}(q,p_i) \end{bmatrix}, \tag{3.17}$$

$$[N] = \begin{bmatrix} N_1 & 0 & N_2 & 0 \\ 0 & N_1 & 0 & N_2 \end{bmatrix}. \tag{3.18}$$

Let $[H_i] = \frac{1}{2}\{u\}^i + \sum_{j=1}^{N} \int_{\Gamma_j} [\hat{H}]^{ij}[N]d\Gamma_j$ and $[G_i] = \sum_{j=1}^{N} [G]^{ij}[N]d\Gamma_j$, we have

$$[H_i]\{u\} = [G_i]\{t\}.$$

By placing the source point at every node of discretization, we get N such $2 \times 2N$ submatrices. Forming them into big matrices, we have

$$[H]\{u\} = [G]\{t\}. \tag{3.19}$$

3.3.4 Singular Integration

Note that equation (3.15) contains 2 types of singularity. The singularity happens when the source point lies on the segment under integration, either on the first nodal point or on the second one of the segment. The one associated with G is *Logarithmic singularity*,

$$g_1 = \int_{\Gamma_j} N_1 \ln r \, ds_q = \frac{l_j}{4} \int_{-1}^{1} (1-\xi)\ln 0.5 l_j + (1-\xi)\ln|\xi_J - \xi| \, d\xi, \tag{3.20}$$

$$g_2 = \int_{\Gamma_j} N_2 \ln r \, ds_q = \frac{l_j}{4} \int_{-1}^{1} (1+\xi)\ln 0.5 l_j + (1+\xi)\ln|\xi_J - \xi| \, d\xi, \tag{3.21}$$

where $\xi_J = \pm 1$ denotes the position of the source point. Logarithmic singularity can still be calculated in theory. Take g_1 for example, when $\xi_J = -1$,

$$\begin{aligned} g_1 &= \frac{l_j}{2} \ln \frac{l_j}{2} + \frac{l_j}{2} \int_{-1}^{1} (1-\xi) \ln(1+\xi)\, d(\xi+1) \\ &= \frac{l_j}{2} \ln \frac{l_j}{2} + \frac{l_j}{4} (1-\xi^2) \ln(1+\xi)|_{-1}^{1} - \frac{l_j}{4} \int_{-1}^{1} (1+\xi)(\frac{1-\xi}{\xi+1} - \ln(1+\xi))\, d\xi \\ &= \frac{l_j}{2} \ln \frac{l_j}{2} - \frac{l_j}{4} \int_{-1}^{1} (1-\xi)\, d\xi + \frac{l_j}{4} \int_{-1}^{1} (1+\xi) \ln(1+\xi)\, d\xi \\ &= \frac{l_j}{2} \ln \frac{l_j}{2} - \frac{l_j}{2} + \frac{l_j}{16} \int_{0}^{4} \ln t\, dt, \text{ with } t = (\xi+1)^2 \\ &= \frac{l_j}{2} (\ln l_j - 1.5). \end{aligned} \tag{3.22}$$

Similarly, we get

$$g_2 = \frac{l_j}{2} (\ln l_j - 0.5). \tag{3.23}$$

And when $\xi_J = 1$,

$$g_1 = \frac{l_j}{2} (\ln l_j - 0.5) g_2 = \frac{l_j}{2} (\ln l_j - 1.5). \tag{3.24}$$

The singularity associated with H is strong singularity,

$$h_1 = \int_{\Gamma_j} \frac{N_1}{r} ds_q = \frac{1}{2} \int_{-1}^{1} \frac{1-\xi}{|\xi_J - \xi|} d\xi, \tag{3.25}$$

$$h_2 = \int_{\Gamma_j} \frac{N_2}{r} ds_q = \frac{1}{2} \int_{-1}^{1} \frac{1+\xi}{|\xi_J - \xi|} d\xi. \tag{3.26}$$

It is easy to see that

1. when $\xi_J = -1$, h_1 is singular while $h_2 = 1$;

2. when $\xi_J = 1$, h_2 is singular while $h_1 = 1$.

The result of this singularity is that the 2×2 matrices on the diagonal of H can not be computed. Luckily we can avoid direct evaluation using rigid body motion.

Consider equation (3.19). Let u be the vector of unit displacement vector along x-, y- axis respectively. Such movement is rigid body motion and should not give rise to any traction on the boundary. So if we let $\{t\} = 0$. Then $[H]\{u\} = 0$. According to this, it is easy to obtain the diagonal entries,

$$\begin{bmatrix} h_{2i,2i} & h_{2i,2i+1} \\ h_{2i+1,2i} & h_{2i+1,2i+1} \end{bmatrix} = \begin{bmatrix} -\sum_{j \neq i}^{N} h_{2i,2j} & -\sum_{j \neq i}^{N} h_{2i,2j+1} \\ -\sum_{j \neq i}^{N} h_{2i+1,2j} & -\sum_{j \neq i}^{N} h_{2i+1,2j+1} \end{bmatrix}. \tag{3.27}$$

For the part where integration is not singular, a self-adjustable *Gauss quadrature* integration scheme is employed.

3.3.5 Boundary Conditions

Generally, the boundary conditions can be classified into the following four types:

1. u, v on Γ_1
2. u, t_y on Γ_2
3. t_x, v on Γ_3
4. t_x, t_y on Γ_4

where $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3 \cup \Gamma_4$. Of course any combination of the 4 types of boundary conditions may be exerted on the object. In this case, we want to maintain the tangent no rotation condition on both finger contacts. In a direct method as BEM, Lagrange multiplier as a way of exerting boundary conditions is not desired. As a compromise, we manipulate the deformation of the nodal points which define the segment where contact finger is located. With bottom and top finger locating on segment b and t resp., let the boundary condition be:

$$(u_b, v_b) = (u_{b+1}, v_{b+1}) = 0, \tag{3.28}$$

$$(u_t, v_t) = (u_{t+1}, v_{t+1}) = c\mathbf{d}, \tag{3.29}$$

$$t_{xi} = t_{yi} = 0, \tag{3.30}$$

where $\mathbf{d}$ is the unit vector from center of segment t to that of segment b; c is a constant; i is the index of nodal points that are not associated with segment b or t.

CHAPTER 4. DISPLACEMENT-BASED GRASPS

The following two Chapters 4 and 5 focus on how to grasp planar objects without concerning any body force, e.g. gravity. The contacts between fingers and objects are frictional. So the following assumptions are made.

(A1) The object to be grasped is either planar or thin $2\frac{1}{2}$D.

(A2) Gravity is ignored as the object lies in a horizontal supporting plane.

(A3) Two grasping fingers are in the same plane, and make point contacts with the object in the presence of friction.

In this Chapter, I will focus on hollow objects with point contact with the fingers.

4.1 The Grasping Problem

Figure 4.1 shows a grasp achieved by squeezing an object. The action is equivalent to keeping one finger still and stuck to its contact point, say, $\mathbf{q}$, while translating the other finger toward $\mathbf{q}$ without slip at its contact point $\mathbf{p}$.

To grasp the deformable object in Figure 4.1, the finger placement $\mathcal{G}(\mathbf{p}, \mathbf{q})$ should prevent any Euclidean motion such that the only possible displacement is deformation. In presence of friction, this requires the grasp to be force closure if the object were rigid. By Nguyen's result [32], the segment $\overline{\mathbf{pq}}$ in Figure 4.1 must lie inside the friction cones at $\mathbf{p}$ and $\mathbf{q}$ on the object's original shape.

If no contact slips, the same finger placement exerting the same forces also needs to maintain equilibrium over the deformed shape of the object. Suppose under the deformation the contact

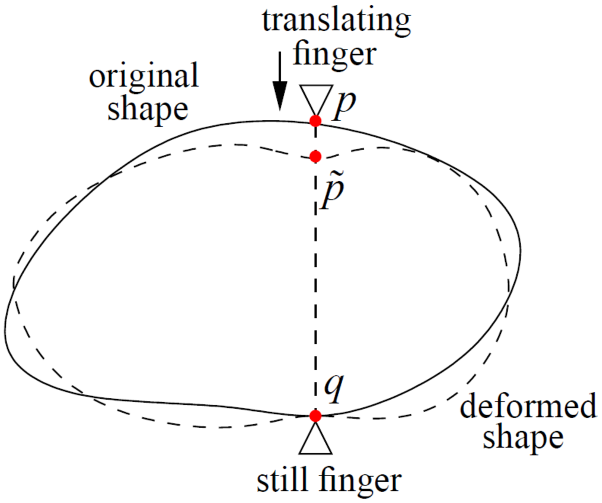


Figure 4.1: Squeeze grasp

points $\mathbf{p}$ and $\mathbf{q}$ have moved to $\tilde{\mathbf{p}}$ and $\tilde{\mathbf{q}}$, resp. The segment $\overline{\tilde{\mathbf{p}}\tilde{\mathbf{q}}}$ must lie inside the friction cones at $\tilde{\mathbf{p}}$ and $\tilde{\mathbf{q}}$ on the object's post-deformation shape.

In rigid body grasping synthesis and analysis, people tend to specify the grasping force. However, specifying force is no longer viable in grasping deformable objects. Not only does such approach fail to constrain the object from free homogeneous transformations, but also introduce instability in the grasp due to similarity to reverse perpendulum.

We choose to specify the displacement of the grasping fingers over the exerted force, based on mainly, but not limit to, the following reasons:

1. in practice, it is much easier to command a robot finger to move to certain locations, than to exert a certain amount of force;

2. the object is fully constrained under certain specified finger displacements, and hence the grasping stability is guaranteed;

3. the object under grasp will alway be in equilibrium;

4. the magnitude of the grasping force is not much of our concern as long as the object is grasped.

In this section, the idea of displacement based grasp will be illustrated. The first part pictured the squeeze grasp of an object. The second part computes the deformation induced by the displacement of grasping fingers. The third part analyzes and defines the squeeze grasp. The last part extended the squeeze grasp to be general two-finger displacement based grasp.

4.1.1 Scenario of Squeeze Grasp

In the 2D case, two fingers suffice grasping an object. The simplest finger displacements in grasping objects is two fingers moving towards each other.

As shown in Figure 4.2, we place two fingers at $\mathbf{p}_0$ and $\mathbf{p}_i$. The finger at $\mathbf{p}_0$ is kept still, while the other finger at $\mathbf{p}_i$ squeezes the object for a grip. Without loss of generality, we place $\mathbf{p}_0$ at the origin and align the positive y-axis with $\overrightarrow{\mathbf{p}_0\mathbf{p}_i}$. The remaining boundary points are not in contact with anything, thus no forces are applied. So

$$\mathbf{f}_k = 0, \tag{4.1}$$

for $1 \leq k \leq n-1$ with $k \neq i$. The force vector is now

$$\mathbf{F} = \begin{pmatrix} \mathbf{f}_0 \\ 0 \\ \vdots \\ 0 \\ \mathbf{f}_i \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{4.2}$$

Proposition 1. *The forces exerted by the two fingers are opposite to each other, that is,* $\mathbf{f}_0 + \mathbf{f}_i = 0$.

Proof. Since $\mathbf{v}_x$ and $\mathbf{v}_y$ are in the null space of K, they are orthogonal to the eigenvectors corresponding to non-zero eigenvalues. Substitute equation (3.6) into (3.7),

$$U\Lambda U^T\Delta = \mathbf{F}. \tag{4.3}$$

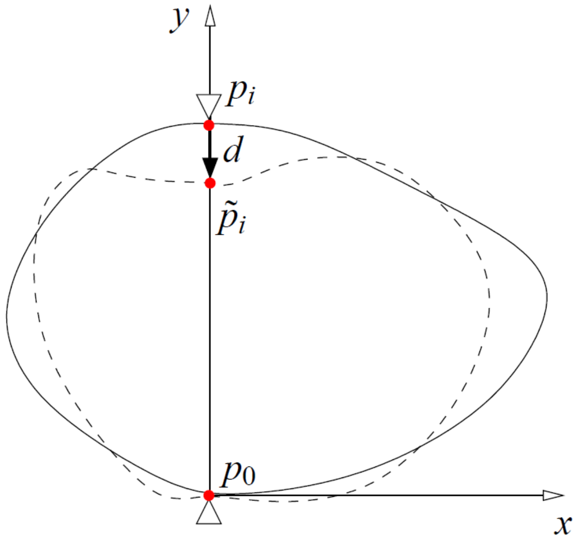


Figure 4.2: Translation of $\mathbf{p}_i$ towards $\mathbf{p}_0$.

Left multiply $\mathbf{v}_x^T$ on both sides of the above equation and substitute equation (4.2) in, the left side vanishes, yielding

$$0 = (1, 0, \cdots, 1, 0) \begin{pmatrix} \mathbf{f}_0 \\ 0 \\ \vdots \\ 0 \\ \mathbf{f}_i \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

or equivalently, $(1, 0) \cdot (\mathbf{f}_0 + \mathbf{f}_i) = 0$. Similarly, multiplications of $\mathbf{v}_y^T$ on equation (4.3) lead to $(0, 1) \cdot (\mathbf{f}_0 + \mathbf{f}_i) = 0$. Thus we have $\mathbf{f}_0 + \mathbf{f}_i = 0$. □

From now on, we will write $\mathbf{f}_0 = -\mathbf{f}$ and $\mathbf{f}_i = \mathbf{f}$.

4.2 Deformation Due to Contact Displacement

It is not always possible to squeeze an object. Two necessary conditions for squeezing the object are

1) the two fingers can maintain its equilibrium before and after the deformation that would result from such a squeeze, and

2) no slip happens at either finger contact.

Our strategy is to first look at how the object deforms under constraints that assume condition 2, and then verify the consistency between both conditions and the computed deformation under them.

The stationary finger in contact with the object at $\mathbf{p}_0$ indicates

$$\delta_0 = 0. \tag{4.4}$$

This eliminates $\mathbf{v}_x$ and $\mathbf{v}_y$ from the solution space of equation (3.7) because translations are now prohibited. The vector, now with $p_{0x} = p_{0y} = 0$, represents a rotation about $\mathbf{p}_0$—the only rigid body motion left. In equation (3.7), we eliminate the first two rows and columns from K, and the first two elements each from Δ and $\mathbf{F}$, obtaining

$$K'\Delta' = F', \tag{4.5}$$

where $\Delta' = (\delta_1^T, \cdots, \delta_{n-1}^T)$ and $F' = (0, \cdots, 0, \mathbf{f}^T, 0, \cdots, 0)^T$. The null space of K' is spanned by the vector[1]

$$\mathbf{v}_r = \begin{pmatrix} -p_{1y} \\ p_{1x} \\ \vdots \\ -p_{n-1,y} \\ p_{n-1,x} \end{pmatrix}. \tag{4.6}$$

The $(2n-2) \times (2n-2)$ matrix K' is symmetric and positive semi-definite, with rank $2n-3$, and can be spectrum-decomposed as:

$$K' = \sum_{i=0}^{2n-3} \lambda_i' \mathbf{u}_i' \mathbf{u}_i'^T, \tag{4.7}$$

where λ_i''s are eigenvalues of K' with $\lambda_{2n-3}' = 0$, and $\mathbf{u}_i'$'s are corresponding eigenvectors, with $\mathbf{u}_{2n-3}' = \mathbf{v}_r'/||\mathbf{v}_r'||$.

[1] Note that $p_{ix} = 0$ in the coordinate system in Figure 4.2.

Proposition 2. *The contact force* $\mathbf{f}$ *exerted at* $\mathbf{p}_i$ *under constraint (4.4) is collinear with the segment* $\overline{\mathbf{p}_0\mathbf{p}_i}$*.*

Proof. Like what we do in proving Proposition 1, we substitute equation (4.7) into (4.5), and multiply both sides of the resulting equation with $\mathbf{u}_{2n-3}^{\prime T}$, obtaining

$$0 = \mathbf{u}_{2n-3}^{\prime T}\mathbf{F}',$$

or equivalently, $\mathbf{v}_r^{\prime T}\mathbf{F}' = 0$, which by equation (4.6) reduces to $(-p_{iy}, p_{ix})\mathbf{f} = 0$. Thus $\mathbf{f}$ and $\overline{\mathbf{p}_0\mathbf{p}_i}$ are colinear. □

Under Proposition 2, we conveniently represent the squeezing force exerted by the moving finger as $\mathbf{f} = (0, -f)^T$ with f being its magnitude. So

$$\mathbf{F}' = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -f \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

where the entry $-f$ has index $2i-1$.

As an important part of our strategy, we specify the finger displacement. Such specification gives us another boundary condition:

$$\delta_i = \mathbf{d} = \begin{pmatrix} d_x \\ d_y \end{pmatrix}. \tag{4.8}$$

From the above notation, the displacement vector Δ' can be re-written in the form of each point's displacement:

$$\Delta' = \begin{pmatrix} \delta_1 \\ \vdots \\ \delta_{i-1} \\ \mathbf{d} \\ \delta_{i+1} \\ \vdots \\ \delta_{n-1} \end{pmatrix}. \tag{4.9}$$

We are essentially solving a version of system (4.5) in $2n-3$ variables: $\delta_1^T, \cdots, \delta_{i-1}^T, \delta_{i+1}^T, \cdots, \delta_{n-1}^T$, each with two coordinates, and f.

Theorem 3. *Given a displacement* $\mathbf{d} = (d_x, d_y)^T$ *of the moving finger, the displacement field* Δ' *of the object and the squeezing force* $\mathbf{F}'$ *are uniquely determined.*[2]

Proof. Denote $\mathbf{u}'_j = (u'_{0,j}, \cdots, u'_{2n-3,j})^T$, for $0 \leq j \leq 2n-3$. Left multiply both sides of equation (4.5), after substitution of equation (4.7), by $\mathbf{u}_0'^T, \cdots, \mathbf{u}_{2n-4}'^T$ sequentially, utilizing the orthogonality of these vectors:

$$\begin{pmatrix} \lambda'_0 \mathbf{u}_0'^T \\ \vdots \\ \lambda'_{2n-4} \mathbf{u}_{2n-4}'^T \end{pmatrix} \Delta' = -f \begin{pmatrix} u'_{2i-1,0} \\ \vdots \\ u'_{2i-1,2n-4} \end{pmatrix}.$$

With the above, we project Δ' onto $\mathbf{u}_0'^T, \cdots, \mathbf{u}_{2n-3}'^T$, denoting $g = \mathbf{u}_{2n-3}'^T \Delta'$,

$$\Delta' = -f \sum_{j=0}^{2n-4} \frac{1}{\lambda'_j} u'_{2i-1,j} \mathbf{u}'_j + g\mathbf{u}'_{2n-3}. \tag{4.10}$$

Since $\mathbf{u}'_{2n-3} = \mathbf{v}'_r / ||\mathbf{v}'_r||$, we have $u'_{2i-1,2n-3} = p_{ix} = 0$. Hence $||\mathbf{u}'_{2i-1}||^2 = \sum_{j=0}^{2n-4} u'_{wi-1,j} = 1$. Now, we look at the two equations in equation (4.10) that involve $\mathbf{d}$:

$$d_x \quad = \quad -f \sum_{j=0}^{2n-4} \frac{1}{\lambda'_j} u'_{2i-1,j} u'_{2i-2,j} + g u'_{2i-2,2n-3}, \tag{4.11}$$

[2] Theorem given and proven by Yan-Bin Jia.

$$d_y \quad = \quad -f \sum_{j=0}^{2n-4} \frac{1}{\lambda'_j} u'^2_{2i-1,j}. \tag{4.12}$$

The sum in equation (4.12) is positive because $\lambda'_j > 0$ for $0 \leq j \leq 2n-4$, and some $u'_{2i-1,j} \neq 0$. We solve the above two equations:

$$f \quad = \quad -d_y/(\sum_{j=0}^{2n-4} \frac{1}{\lambda'_j} u'^2_{2i-1,j}), \tag{4.13}$$

$$g \quad = \quad \frac{1}{u'_{2i-2,2n-3}}(d_x - d_y(\sum_{j=0}^{2n-4} \frac{1}{\lambda'_j} u'_{2i-1,j} u'_{2i-2,j})/(\sum_{j=0}^{2n-4} \frac{1}{\lambda'_j} u'^2_{2i-1,j})). \tag{4.14}$$

Finally, plug f and g into equation (4.10) to obtain $\delta_1^T, \cdots, \delta_{i-1}^T, \delta_{i+1}^T, \cdots, \delta_{n-1}^T$. □

Or equivalently, with the boundary conditions given by equation (4.1), (4.4) and (4.8), the system (3.7) is uniquely solvable.

In the special case $d_y = 0$, the finger in contact with $\mathbf{p}_i$ moves in the x-direction. It follows from equation (4.13) and (4.14) that $f = 0$ and $g = d_x/u'_{2i-2,2n-3}$. Plugging them into equation (4.7), we can show that $\Delta = (d_x/u'_{wi-2,2n-3})\mathbf{u}_{2n-3}$. Consequently, the object undergoes a pure rotation with no deformation.

4.3 Squeeze Grasp

To squeeze the object, one finger moves towards the other, or in our scenario, $d_x = 0$. We refer to $d = -d_y > 0$ as the *squeezing distance.*

Corollary 1. *Under a squeeze grasp, the contact forces and displacements of all vertices scale with the squeezing distance d.*

Proof. This follows directly from substitutions of $d_x = 0$ and $d_y = -d$ into equation (4.13) and (4.14), and from subsequent substitutions of the obtained f and g into equation (4.10). □

The next corollary states that a squeeze makes no difference in the resulting deformation if the moving and still fingers switch their roles.

Corollary 2. *Squeezing with $\mathbf{p}_i$ fixed while $\mathbf{p}_0$ moving toward $\mathbf{p}_i$ by a distance of d yields the same shape except under a translation of $(0, d)^T$.*

Proof. Suppose that the original squeeze with $\mathbf{p}_0$ fixed and $\mathbf{p}_i$ moving by $(0, -d)^T$ under force $\mathbf{F}$ results in a displacement field δ. System (3.7) is satisfied by $\mathbf{F}$ and δ under the constraints $\delta_0 = 0$ and $\delta_i = (0, -d)^T$. It must also be satisfied by $\mathbf{F}$ and a new displacement $\delta' = \delta + d \cdot \mathbf{v}_y$ since $\mathbf{v}_y$ is in the null space of K. A result analogous to Theorem 3 can be easily established to ensure that $\mathbf{F}$ and δ' are the unique solution under the new constraints $\delta'_0 = (0, d)^T$ and $\delta'_i = 0$. The deformed shape is the same as the one constrained by $\delta_0 = 0$ and $\delta_i = (0, -d)^T$, except it is translated by $(0, d)^T$. □

In a squeeze grasp, two finger contacts stay on the same line all the time. According to Proposition 1 and 2, the equilibrium of the body is guaranteed. We yet have to see that whether slip will happen or not. One simple fact is that, no slip happens between two contact objects if the contact force stays inside the friction cone. If the force is right on the edge of the friction cone, it depends on the initial state of the two objects. If they are relatively still in the initial state, then still no slip happens. Since Proposition 2 says that the direction of the squeezing force is parallel to $\overline{\mathbf{p}_i\mathbf{p}_j}$, then if $\overline{\mathbf{p}_i\mathbf{p}_j}$ stays inside the friction cones all the time, no slip will happen. It follows directly that no rotation about $\mathbf{p}_j$ may happen either, because if it does, slip must happen at $\mathbf{p}_i$.

The orientation of the friction cone can be represented by the orientation of the contact tangent, which is decided by its neighbor vertices. For example, the tangent at $\mathbf{p}_i$ is along the direction of $\mathbf{p}_{i+1} - \mathbf{p}_{i-1}$ before deformation and $\mathbf{p}_{i+1} - \mathbf{p}_{i-1} + \delta_{i+1} - \delta_{i-1}$ after deformation. According to Corollary 1, $\delta_{i+1} - \delta_{i-1}$ scales with d. Thus the orientation of the tangent changes monotonically with d. So if no slip happens on original shape and the deformed shape with squeezing distance d, then no slip may happen for any deformed shape with squeezing distance that is less than d. This agrees with our experience that hard squeezes are more likely to cause slips on soft objects.

The range of d is bounded by some factors. Since the squeezing force scales with d, it should not be too small that the squeezing force fails requirements for certain tasks, for example, picking the object up from the supporting plane, in which case, enough friction is necessary. Meanwhile, d cannot be too large in order for the squeezing forces to stay in their respective contact friction cones, and for the resulting deformation to be small enough so that it is well described by the linear elasticity.

How large can d be? Linear elasticity theory does not tell us. It depends on the material, the global shape of the object, the contact locations, etc. For simplification, we introduce a factor $\phi \in (0, 1)$ and consider all squeezing distances $d = \rho||\mathbf{p}_i - \mathbf{p}_j||$, where $\rho \in (0, \phi]$, to cause small deformations[3]. We call ρ the relative squeezing depth.

Definition 1. *A finger placement $\mathcal{G}(\mathbf{p}, \mathbf{q})$ on an object is a ρ-*squeeze grasp *if*

1) the line segment $\overline{\mathbf{pq}}$ is inside the friction cones at $\mathbf{p}$ and $\mathbf{q}$, and

2) after deformation due to the displacement of $\mathbf{p}$ to $\tilde{\mathbf{p}} = \mathbf{p} + \rho(\mathbf{q} - \mathbf{p})$, the line segment $\overline{\tilde{\mathbf{p}}\mathbf{q}}$ lies inside the friction cones at $\tilde{\mathbf{p}}$ and $\mathbf{q}$ of the deformed shape.

4.4 Generalized Squeeze Grasp

Now consider the case where $d_x \neq 0$. $\tilde{\mathbf{p}}_i$ is now off the line $\mathbf{p}_i\mathbf{p}_j$. Although the total force sum up to 0 according to Proposition 1, the two squeezing forces do not point to each other according to Proposition 2, which means that there is a torque and equilibrium is broken.

Is a grasp only possible when the action is exactly squeeze? Of course not. The above illusion is due to the limit of linear elasticity in modeling the real world, as shown in Figure 3.2.

Let us leave linear elasticity theory aside for the moment, and look at Figure 4.3. Shape 1 and Shape 2 are undeformed shapes only different by an angle α in orientation. Shape 3 are deformed shape of shape 1 generated by displacing point a to point b, a pure squeeze. Shape 4 are deformed shape of shape 1 generated by displacing point a to point d, where $|dq| = |bq|$ and d is on $\overline{cq}$. The four shapes share a point p. Now consider the deformed shape s of shape 2 generated by displacing point c to point d, which is a pure squeeze. Shape s is the same as shape 4 because shape 1 and 2 are the same, and the corresponding points of them are fixed at same locations. Now, since shape 3 and shape 4 are both deformed shape generated by a pure squeeze of the same distance from the same shape, they are also the same, except for a difference α in orientation. So the displacement $\overrightarrow{ab}$ and $\overrightarrow{ad}$ generates the same resulting

[3] We usually take ϕ to be less than 20%.

shape. On the other hand, the force exerted at d is along $\overrightarrow{dq}$ since shape 4 can be generated from shape 2 by the same pure squeezing. So both the deformed shape 3 and 4, and the grasping forces exerted on them, are the same, except for an angle α in orientation. Thus if displacement $\overrightarrow{ab}$ could result in a grasp, so could $\overrightarrow{ad}$.

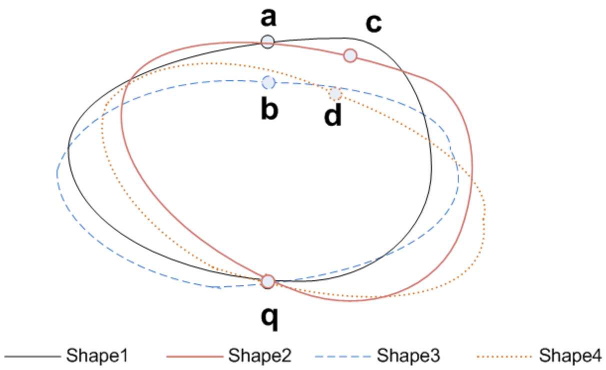


Figure 4.3: Generalized squeeze grasp. Shape 1 and Shape 2 are same undeformed shapes different by α orientation. a and c are corresponding points. Shape 3 and Shape 4 are deformed shapes generated from shape 1 by displacement $\vec{ab}$ and $\vec{ad}$ resp. All 4 shapes share point q. a,b,q are collinear and c,d,q are also collinear. $||\vec{ab}|| = ||\vec{cd}||$.

Proposition 3. *If a finger placement $\mathcal{G}(\mathbf{p}, \mathbf{q})$ can achieve a grasp by pure squeeze $\mathbf{d}$ of ρ-depth, then any displacement $\mathbf{d}'$ of the same squeeze depth can achieve a grasp. The set of all displacement vectors can be grouped into equivalent classes according to their squeeze depth at same finger placement.*

With above Proposition, given any finger placement and the displacement vector, we can test the grasp by testing the pure squeezing grasp at the same finger placement.

4.5 Grasp Computation

Algorithm 1 tests whether a general finger placement $\mathcal{G}(\mathbf{p}_i, \mathbf{p}_j)$, $0 \leq i < j \leq m-1$ is a ρ-squeeze grasp. Step 4 of the algorithm is the most expensive one. A brute force method would fix one contact, say at $\mathbf{p}_i$, and solve system (4.5), where K' is the $(2n-2) \times (2n-2)$ stiffness matrix generated after removing the $2i$-th and $(2i+1)$-st rows and columns from K. Inversion of the matrix is necessary in order to check for different locations $\mathbf{p}_j$ of the moving finger. This operation can be carried out in $O(n^{2.807})$ time using Stassen's algorithm, or in $O(n^{2.376})$ time using the Coppersmith-Winograd algorithm.[4]

Algorithm 1 Test of $\mathcal{G}(\mathbf{p}_i, \mathbf{p}_j)$ for a ρ-squeeze grasp

```
if $\overline{\mathbf{p}_i\mathbf{p}_j}$ does not lie inside the friction cone at $\mathbf{p}_i$ or $\mathbf{p}_j$ then
    return no
else
    evaluate the tangents at $\mathbf{p}_i$ and the displaced location $\tilde{\mathbf{p}}_j$, using $\tilde{\mathbf{p}}_{i-1}$, $\tilde{\mathbf{p}}_{i+1}$, $\tilde{\mathbf{p}}_{j-1}$ and $\tilde{\mathbf{p}}_{j+1}$
    if $\overline{\mathbf{p}_i\tilde{\mathbf{p}}_j}$ does not lie inside the friction cone at $\mathbf{p}_i$ or $\tilde{\mathbf{p}}_j$ then
        return no
    else
        return yes
    end if
end if
```

Nevertheless, the matrix K' changes whenever $\mathbf{p}_i$ does, that is, whenever the still finger is relocated. Thus matrix inversion needs to be performed every time. The number of matrix inversions equals m, the number of boundary vertices that are possible locations of $\mathbf{p}_i$. For a brute force iteration, the running time is $O(m^2 n^{2.807})$ or $O(n^{3.807})$ since $m = O(\sqrt{n})$ for a solid object.

This chapter describes fast grasp testing that works on the stiffness matrix K only. We make use of its spectral decomposition (3.6), and obtain the matrices U and Λ via singular value decomposition (SVD) in $O(n^3)$ time. Below we show that the placement $\mathcal{G}(\mathbf{p}_i, \mathbf{p}_j)$ can be checked for a squeeze grasp in $O(n)$ time.[5]

4.5.1 An Efficient Algorithm

We first perform the standard Singular Value Decomposition procedure to the symmetric matrix K

$$K = U\Sigma U^T, \tag{4.15}$$

where $\Sigma = \text{diag}(\rho_0, \cdots, \rho_{2n-4}, 0, 0, 0)$ with ρ_k's being non-zero eigenvalues of K, and $U = (\mathbf{w}_0, \cdots, \mathbf{w}_{2n-1})^T$ is the orthonormal matrix consisting eigenvectors of K with $\mathbf{w}_k$'s being its

[4] The latter algorithm is mainly useful for proving theoretical time bounds.

[5] The efficient algorithm was mainly developed by Yan-Bin Jia

row vectors written in column vector form. The sequence of the eigenvectors are corresponding to that of the eigenvalues.

Now apply the coordinate transformation. We project Δ on to the eigenvectors, and denote the vector of projection to be $\mathbf{y}$

$$\mathbf{y} = \begin{pmatrix} y_0 \\ \vdots \\ y_{2n-1} \end{pmatrix} = U^T \Delta. \tag{4.16}$$

Since U is orthonormal, $U^T = U^{-1}$. Then

$$\Delta = U\mathbf{y}. \tag{4.17}$$

Substitute it into equation (3.7):

$$K\Delta = U\Sigma U^T U\mathbf{y} = U\Sigma\mathbf{y} = \mathbf{F},$$

Left multiply U^T on both sides of the last equal sign, we get

$$\begin{aligned}(\rho_0 y_0, \cdots, \rho_{2n-4} y_{2n-4}, 0, 0, 0)^T &= U^T\mathbf{F} \\ &= U^T(0, \cdots, 0, f_{2i}, f_{2i+1}, 0, \cdots, 0, f_{2j}, f_{2j+1}, 0, \cdots)^T \\ &= f_{2i}\mathbf{w}_{2i} + f_{2i+1}\mathbf{w}_{2i+1} + f_{2j}\mathbf{w}_{2j} + f_{2j+1}\mathbf{w}_{2j+1}. \qquad (4.18)\end{aligned}$$

Let $\mathbf{r} = (r_x, r_y)^T = (\mathbf{p}_j - \mathbf{p}_i)/||\mathbf{p}_j - \mathbf{p}_i||$. According to Proposition 1 and 2,

$$\begin{pmatrix} f_{2i} \\ f_{2i+1} \\ f_{2j} \\ f_{2j+1} \end{pmatrix} = f \begin{pmatrix} \mathbf{r} \\ -\mathbf{r} \end{pmatrix},$$

where f is the magnitude of the force. Let $\mathbf{a} = (\mathbf{r}^T, -\mathbf{r}^T)^T$ and $W = (\mathbf{w}_{2i}, \mathbf{w}_{2i+1}, \mathbf{w}_{2j}, \mathbf{w}_{2j+1})$, equation (4.18) becomes

$$(\rho_0 y_0, \cdots, \rho_{2n-4} y_{2n-4}, 0, 0, 0)^T = fW\mathbf{a}.$$

Divide the i-th entry of both sides of the above equation by ρ_i, for $0 \le i \le 2n-4$, resp., and denote the resulting vector of the right side as P, we get:

$$\begin{pmatrix} y_0 \\ \vdots \\ y_{2n-4} \\ 0 \\ 0 \\ 0 \end{pmatrix} = fP. \tag{4.19}$$

Since $\Delta = U\mathbf{y}$,

$$\begin{aligned} (\delta_i^T, \delta_j^T)^T &= W^T \mathbf{y} \\ &= W^T \left[\begin{pmatrix} y_0 \\ \vdots \\ y_{2n-4} \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ y_{2n-3} \\ y_{2n-2} \\ y_{2n-1} \end{pmatrix} \right] \\ &= fW^T P + W_{s3}^T \begin{pmatrix} y_{2n-3} \\ y_{2n-2} \\ y_{2n-1} \end{pmatrix}. \end{aligned} \tag{4.20}$$

where W_{s3} is a 3×4 submatrix of W taking its last 3 rows. Given $(\delta_i^T, \delta_j^T)^T = (0, 0, \mathbf{d}^T)^T$, we then have

$$\begin{pmatrix} W^T P & W_{s3}^T \end{pmatrix} \begin{pmatrix} f \\ y_{2n-3} \\ y_{2n-2} \\ y_{2n-1} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \mathbf{d} \end{pmatrix}. \tag{4.21}$$

Once we solve the above 4 by 4 system, we can calculate $\mathbf{y}$ using equation (4.19). And then Δ is solved using equation (4.17).

4.5.2 Algorithm Analysis

The preprocessing, SVD, takes $O(n^3)$ time. After the that, obtaining W^TP takes $O(n)$ time, and obtaining W_{s3}^T takes constant time. Solving the system (4.21) takes constant time. Obtaining $\mathbf{y}$ takes $O(n)$ time. With $\mathbf{y}$, we can evaluate any value of Δ in $O(n)$ time.

Get back to step 4 of Algorithm 1, evaluating $\tilde{\mathbf{p}}_{i-1}$, $\tilde{\mathbf{p}}_{i+1}$, $\tilde{\mathbf{p}}_{j-1}$ and $\tilde{\mathbf{p}}_{j+1}$, which takes $O(n)$ time, gives us the result of one grasp testing.

Theorem 4. *After SVD of the stiffness matrix K, which takes $O(n^3)$ time, the grasp test on a finger placement $\mathcal{G}(\mathbf{p}_i, \mathbf{p}_j)$ takes $O(n)$ time.*

To compute the global deformation resulted by one grasp, we need to evaluate Δ, which takes $O(n^2)$ time, or, if we only need the contour of the shape, $2m-4$ entries of Δ are evaluated, which takes $O(n^{1.5})$ time. To find all grasps, we exhaustively test the $\binom{m}{2}$ pairs of boundary points, which can be done in $O(m^2n)$, or $O(n^2)$ time. The overall running time is dominated by SVD.

Table 4.1 shows the running time of the naive algorithm and the efficient algorithm for different tasks on solid and hollow objects. We can see that almost for every task, the efficient algorithm reduces the time by order of 2.

	Naive		Efficient(after SVD $O(n^3)$)	
	Solid	Hollow	Solid	Hollow
Single Grasp Test	$O(n^3)$	$O(n^3)$	$O(n)$	$O(n)$
Compute global deformed contour	$O(n^3)$	$O(n^3)$	$O(n^{1.5})$	$O(n^2)$
Find 2nd finger location given 1st	$O(n^{3.5})$	$O(n^4)$	$O(n^{1.5})$	$O(n^2)$
Find all grasps	$O(n^4)$	$O(n^5)$	$O(n^2)$	$O(n^3)$

Table 4.1: Algorithm Comparison. Running time of naive and efficient algorithms on different tasks and different types of object.

4.6 Experiment

Figure 4.4 shows the experimental setup in which grasping was carried out by two fingers of a Barrett Hand. Every finger had a strain gauge sensor mounted at its lower joint to measure contact force.

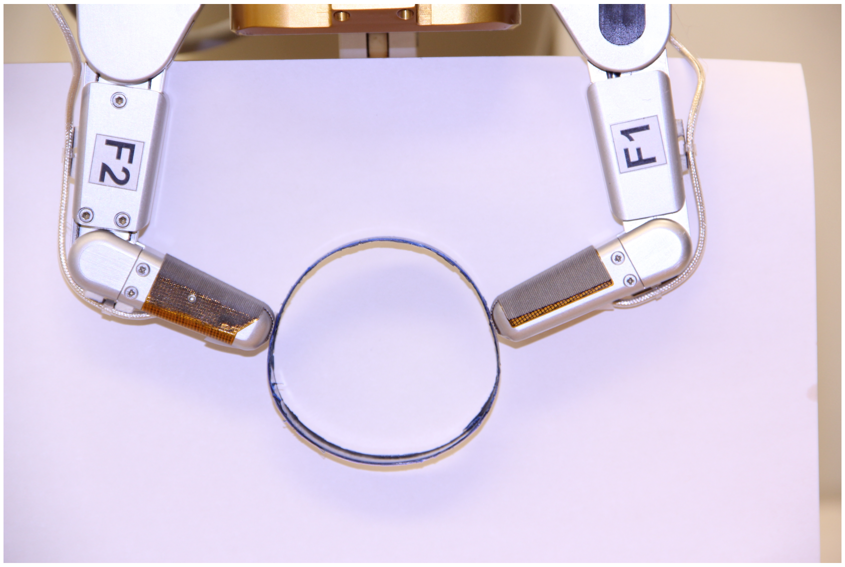


Figure 4.4: Grasping with a Barrett Hand.

Table 4.2 shows an example of successful grasp using Barret Hand[6] that agreed with the prediction by the algorithm. Our algorithm took each shape as input and predicted finger placements that would result in grasps. The Barrett Hand then execute grasps under some placements randomly selected from the predictions.

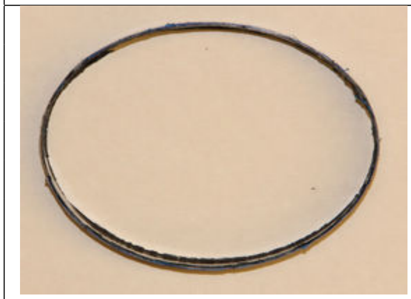
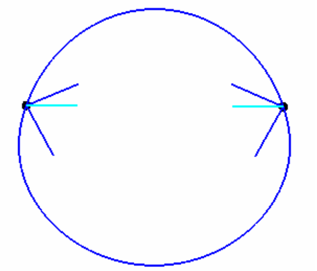
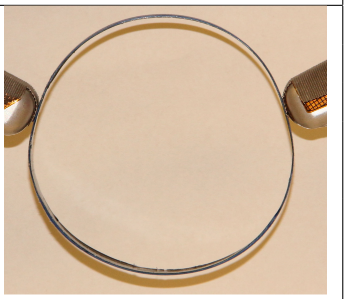
Shape	Simulation	Experiment
		

Table 4.2: Grasping a ring-like object (column 1): simulation (column 2) and experimental (column 3).

The hand initially grasped the hollow ellipse displayed in the first column of Table 4.2, which was cut from a food container. The elliptic object was made of high-density polyethylene

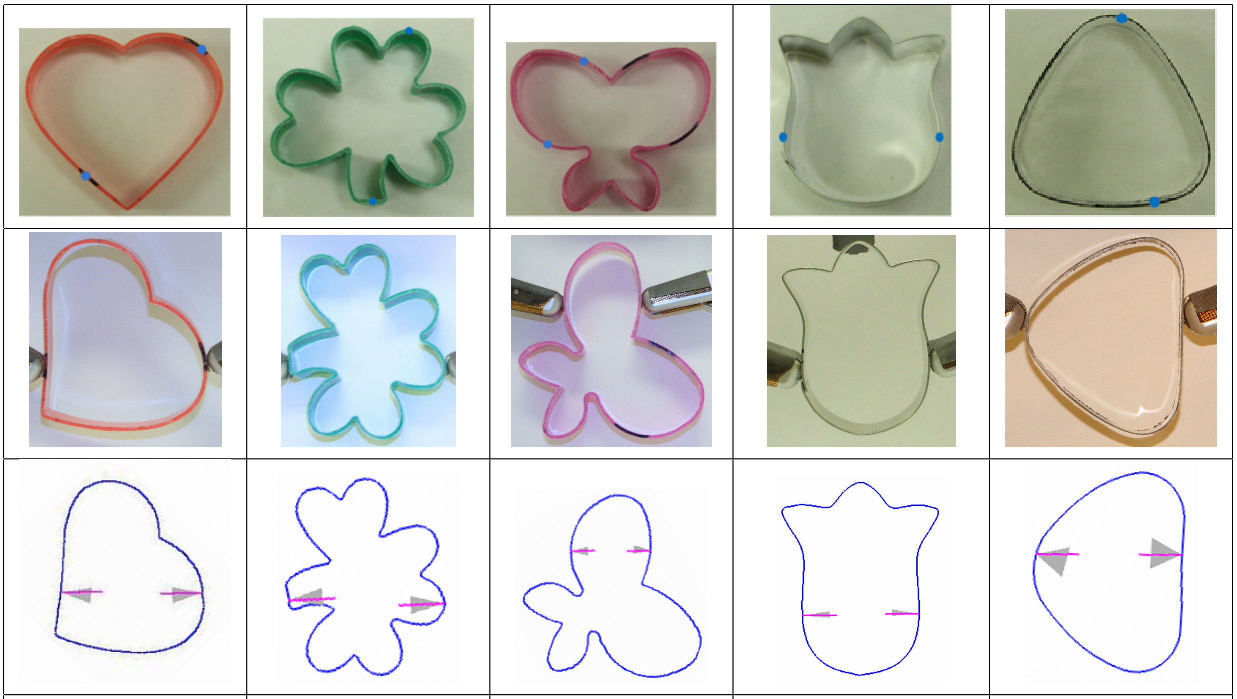
[6] Although we could not keep one finger still and move the other towards it due to lack of degrees of freedom, the relative movement of the fingers was still pure squeeze.

(Young's modulus $E = 0.8$GPa). The cross section had size 11mm × 0.6mm. The surfaces of the objects were filed to increase friction with fingers. The coefficients of friction was measured by determining the max slope of the phalange on which the objects could be placed without slip. The values were 0.9. The second column of Table 4.2 shows the simulation results of the grasp, and the third column shows the outcome of an experiment (which matched the simulation results well), at 10%-squeeze depth.

The experiment was repeated on a variety of objects, including plastic and metal cookie cutters, and cross sections of medicine and food bottles. The shape of each object was either obtained from a scanner, or reconstructed from sampled points or a geometric specification. The first row of Figure 4.5 shows five objects, each with two blue dots indicating a finger placement. The second row displays the objects when grasped by two fingers of the Barrett Hand. Under the finger placements in the first row, the third row shows simulation results of the grasps, with contact friction cones indicated by gray triangles and forces by short purple lines. At every contact, the friction was enough to keep the contact force pointing at the opposite contact, as required for equilibrium. Each entry in the last row lists the relevant physical parameters of the corresponding grasp: the objects' Young's Modulus (GPa), coefficient of friction, thickness (mm), and the relative squeeze depth.

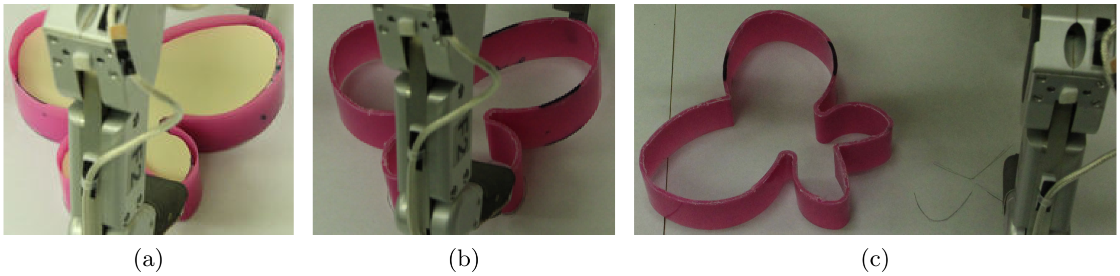
Figure 4.6 shows an example in which the same finger placement succeeded on a "rigid" object but failed on a deformable one with the identical boundary and boundary material. The "rigid" object was made from stuffing the deformable one with fairly stiff plastic foam. The object was initially constrained by the fingers, since the line connecting two contacts was inside their friction cones. As the squeeze went on, both friction cones rotated, until one edge of a friction cone went pass the connection line of the two contacts. As the contact forces no longer balance each other, the object escaped entirely from the grasp, releasing its stain energy.

Rotations of the two contact friction cones do not always lead to a grasp failure. On some occasions, they rotate toward each other, so that the applied forces become more aligned with the cone axes, resulting in a more stable grasp. Figure 4.7 shows two scenarios, one in which the cones rotate toward each other, strengthening the grasp, and the other in which they rotate away from each other, breaking the grasp.



$(2, 0.26, 5, 10\%)$	$(2, 0.26, 5, 11\%)$	$(2, 0.26, 5, 9\%)$	$(200, 0.12, 3, 15\%)$	$(3, 0.5, 3, 15\%)$

Figure 4.5: Successful grasps of five deformable objects (row 1) from experiment (row 2) and simulation (row 3), given physical parameters (row 4).



(a) (b) (c)

Figure 4.6: Grasp failure due to deformation: (a) grasp of a "rigid" shape; (b) attempt at a deformable shape with the same boundary at the same finger placement; (c) escape of the object at 15% squeeze depth.

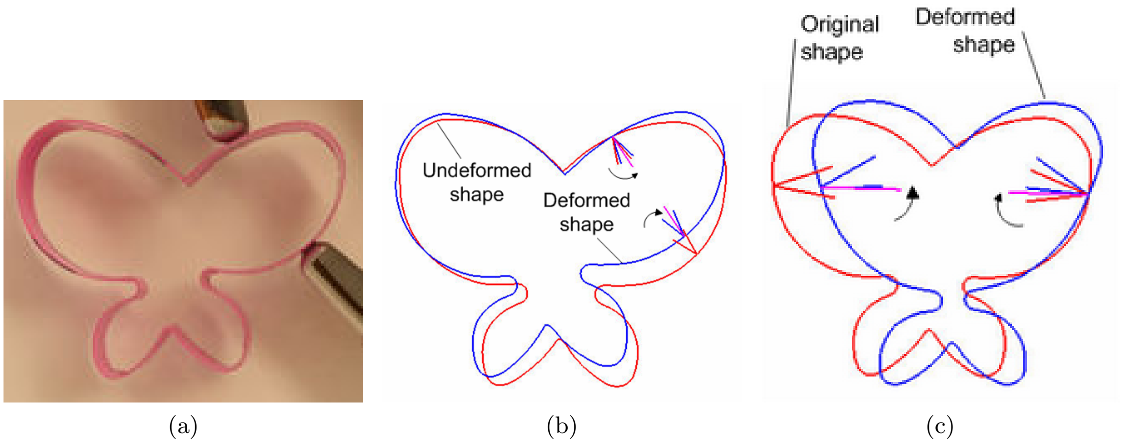
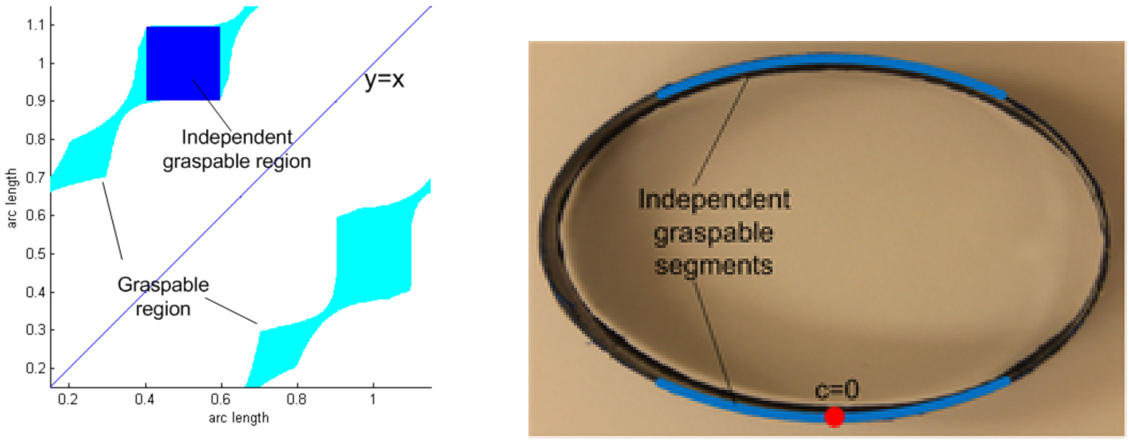


Figure 4.7: Rotations of two contact friction cones toward each other can achieve a grasp in (a) experiment and (b) corresponding simulation, and rotations away from each other can fail one (c). The arrows indicate the rotation directions.

A finger placement can be represented by a point in the plane, when the boundary is parameterized by arc length, which allows us to plot grasps as regions. Figure 4.8(a) shows all pairs of finger locations on the shape in Figure 4.8(b) that can achieve grasps at the 15% squeeze depth. Note that the regions are symmetric about the axis $y = x$.

The rectangle in Figure 4.8(a) corresponds to the pair of blue segments in Figure 4.8(b), on which two fingers can be placed anywhere independently to form a grasp at 15% squeeze depth. They are called *independent graspable segments*. Given a matrix M, where $M(i, j)$ contains the grasp test result of $\mathcal{G}_{ij}$, a dynamic program algorithm is developed to find the maximum independent graspable region in $O(n^2)$ time, which is also the lower bound of such problem.

Figure 4.9 shows the graspable regions of 3 objects. The graspable regions grow significantly with the friction coefficient. However, they decrease very little while w increases from 1% to 10%. Note that the regions are symmetric about line $y = x$, as implied by Corollary 2.



(a) (b)

Figure 4.8: (a) Set of finger placements that yield successful grasps, with the rectangular region corresponding to two independent graspable segments shown in (b).

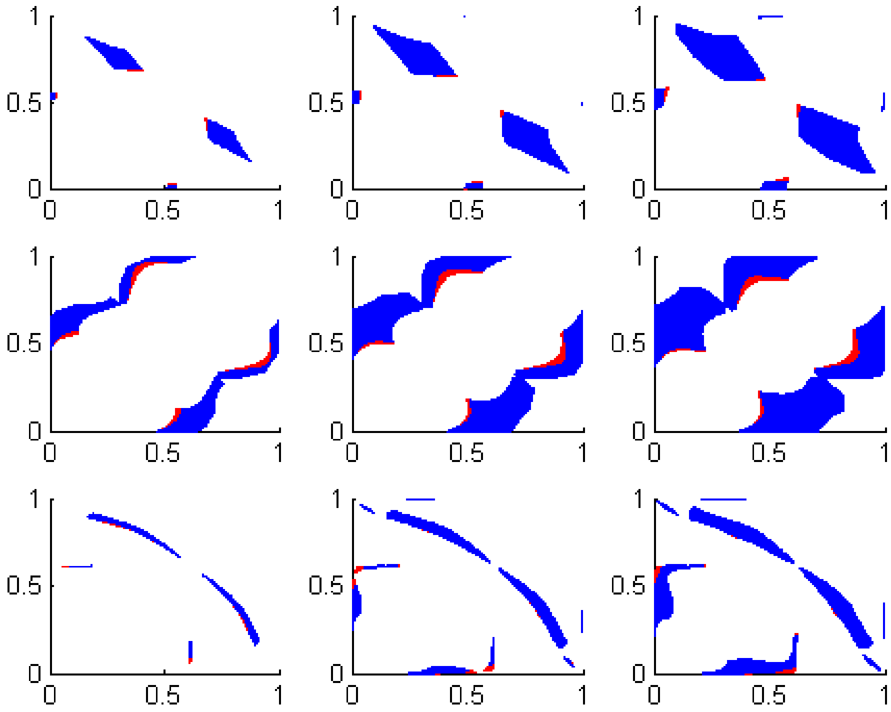


Figure 4.9: Graspable regions. Cell $(i,\, j)$, $i,\, j$ =1, 2, 3, shows the graspable regions on object i with friction coefficient μ_j =0.3, 0.5, 0.7, resp. The blue regions are 10%-graspable and also 1%-graspable, and the red regions are only 1%-graspable.

CHAPTER 5. GRASPING PLANAR OBJECTS WITH AREA CONTACT

Two fingers $\mathcal{F}_1$ and $\mathcal{F}_2$, with identical semicircular tips of radius r and centers $\mathbf{o}_1$ and $\mathbf{o}_2$, are placed on the object at boundary nodal points $\mathbf{p}_i$ and $\mathbf{p}_j$, respectively. The finger orientations are irrelevant assuming that only the tips will be in contact. Also, $\mathbf{o}_1$ and $\mathbf{o}_2$ must lie on the object's normal at $\mathbf{p}_i$ and $\mathbf{p}_j$. The finger placement is thus fully specified.

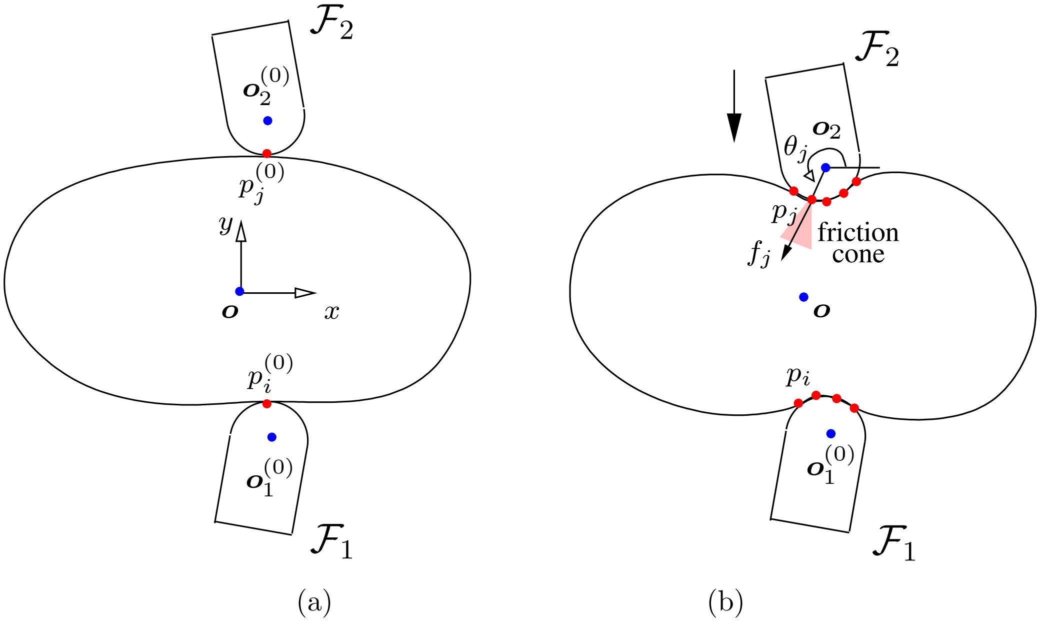


Figure 5.1: Object (a) before and (b) after a squeeze grasp.

As shown in Fig. 5.1(a), we place the origin at the center of the object, and let the y-axis point toward $\mathbf{p}_j$. Finger $\mathcal{F}_1$ is motionless, while finger $\mathcal{F}_2$ translates in the direction $\mathbf{p}_i - \mathbf{p}_j$ by a distance $d > 0$, which is referred to as the *squeeze depth.* As the squeeze continues, some boundary nodal points may come into contact with the fingers, as illustrated in Fig. 5.1(b), while others may break contact with them. A node in contact may be sticking to a fingertip or sliding on it. The *contact configuration* at the squeeze depth d describes which nodal points are in contact, and among them, which are sticking or sliding. We will incrementally track the contact configuration as d increases.

5.1 Contact Configuration

The squeeze depth will be sequenced into $d_0 = 0 < d_1 < \cdots$, such that at $d = d_l$ some event happens to trigger a change in the contact configuration. For $d > d_l$, we use the new configuration and evaluate the changes in $\mathbf{f}$ and $\boldsymbol{\Delta}$ using the FEM, and predict d_{l+1}. At d_l, we maintain two sets: T of nodes sticking with a finger, and P of nodes sliding on a finger. Translate $\mathcal{F}_2$ down by a small extra distance $\epsilon > 0$. Suppose T and P do not change as d varies within $[d_l, d_l + \epsilon)$.

For each node $\mathbf{p}_k \in T \cup P$, denote by θ_k its polar angle with respect to the center of the contact fingertip. See Fig. 5.1(b) for an illustration on $\mathbf{p}_j$. Denote $\delta_k = \delta_k^{(l)}$ and $\theta_k = \theta_k^{(l)}$ when $d = d_l$. We can determine the displacement $\delta_k = \delta_k^{(l)} + \mathring{\delta}_k$ when $d = d_l + \epsilon$ as follows. If $\mathbf{p}_k$ is on $\mathcal{F}_2$,

$$\mathring{\delta}_k = -\epsilon\hat{\mathbf{y}} + r\begin{pmatrix}\cos\theta_k - \cos\theta_k^{(l)} \\ \sin\theta_k - \sin\theta_k^{(l)}\end{pmatrix}, \tag{5.1}$$

where $\hat{\mathbf{y}} = \binom{0}{1}$. If $\mathbf{p}_k$ is on $\mathcal{F}_1$, the term $-\epsilon\hat{\mathbf{y}}$ in (5.1) vanishes.

A sticking contact at $\mathbf{p}_k$ imposes a position constraint $\theta_k = \theta_k^{(l)}$ on deformation. If $\mathbf{p}_k$ slips, the contact force $\mathbf{f}_k = \mathbf{f}_k^{(l)} + \mathring{\mathbf{f}}_k$ must stay on one edge of the friction cone at $\mathbf{p}_k$ as the node moves. Let $\phi = \tan^{-1}\mu$, where μ is the coefficient of contact friction. This imposes a force constraint:

$$\left(\mathbf{f}_k^{(l)} + \mathring{\mathbf{f}}_k\right) \times \begin{pmatrix}\cos(\theta_k \pm \phi) \\ \sin(\theta_k \pm \phi)\end{pmatrix} = 0, \tag{5.2}$$

where the sign '+' or '−' can be determined either from the previous step or using hypothesis-and-test.

We can thus solve for $\mathring{\mathbf{f}}$ and $\mathring{\mathbf{g}}$. And consequently, we obtain $\mathring{\delta}_l$ and thus δ_l, for $1 \leq l \leq n$, which are linear in terms of ϵ, $\cos\theta_t$, and $\sin\theta_t$, $\forall \mathbf{p}_t \in P$. Substitute the expression for $\mathring{\mathbf{f}}_k$ in (5.2). This yields an equation linear in ϵ and quadratic in $\cos\theta_t$ and $\sin\theta_t$, for every $\mathbf{p}_t \in P$.

There are a total of $|P|$ such equations that form a system S in the same number of variables θ_t. Given a value of ϵ, we can solve for these θ_ts. Since ϵ is small, Newton's method converges fast with the initial values $\theta_t^{(l)}$. Hence $\mathbf{\Delta}$ and $\mathbf{f}$ are updated.

With θ_k known for every sliding contact $\mathbf{p}_k$, we can also determine the derivative $\frac{d\theta_k}{d\epsilon}$, which will be used for checking whether a node $\mathbf{p}_k$ switches from slip to stick. Differentiate both sides of every equation in the system S with respect to ϵ. This yields a new linear system of $|P|$ equations in $|P|$ derivatives $\frac{d\theta_t}{d\epsilon}$, $p_t \in P$. Simply solve the system.

5.2 Contact Event Detection

Now we look at how to predict the value of ϵ such that the next event occurs at squeeze distance $d^{(l+1)} = d^{(l)} + \epsilon$ to trigger a change in one or both of the contact sets T and P. There are four types of events described as follows.

Event A — New Contact A boundary node $\mathbf{p}_k$ comes into contact with one of the two fingers. This happens when its distance to the center of the contacting fingertip reduces to r, or equivalently, when the following condition holds (assuming the moving finger $\mathcal{F}_2$ to be in contact):

$$||\tilde{\mathbf{p}}_k - \tilde{\mathbf{o}}_2 + \begin{pmatrix} 0 \\ \epsilon \end{pmatrix}|| = r. \tag{5.3}$$

Here we denote $\tilde{\mathbf{q}}$ as the displaced position of a point $\mathbf{q}$. For completeness, we ought to check every boundary node that is currently not in contact with any finger. Often a new contact node is adjacent to an outermost contact node.

To determine the mode of contact for $\mathbf{p}_k$, we first hypothesize that it sticks, apply a small extra squeeze, and check if the resulting contact force $\mathbf{f}_k$ stays inside the friction cone. If not, the node slips. Add $\mathbf{p}_k$ to T or P accordingly.

Event B — Contact Break

As ϵ increases, the force $\mathbf{f}_k$ at a node $\mathbf{p}_k$ varies inside or on one edge of the contact friction cone. When its derivative is pointing inside the finger and its magnitude reduces to zero, it is about to become a sticking force, which is unrealistic. This implies that the contact breaks when $\|\mathbf{f}_k\| = 0$. Remove $\mathbf{p}_k$ from P or T that contains it.

Event C — Stick to Slip. When the contact force $\mathbf{f}_k$ applied on a sticking node $\mathbf{p}_k$ is rotating out of the inward friction cone (cf. Fig. 5.1(b)) as d increases, the contact starts to slip. The rotation of the force $\mathbf{f}_k$ at the moment is indicated by its derivative with respect to ϵ. We need to check the conditions:

$$\mathbf{f}_k \times \begin{pmatrix} \cos(\theta_k \mp \phi) \\ \sin(\theta_k \mp \phi) \end{pmatrix} = 0 \text{ and } \mp \frac{d\mathbf{f}_k}{d\epsilon} \times \begin{pmatrix} \cos(\theta_k \mp \phi) \\ \sin(\theta_k \mp \phi) \end{pmatrix} > 0 \tag{5.4}$$

for reaching the left (sign '$-$') or right ('$+$') edge, respectively. Remove $\mathbf{p}_k$ from T and add it to P.

Event D — Slip to Stick. As ϵ increases, the contact node $\mathbf{p}_k$ slides, and its polar angle θ_k with respect to the corresponding fingertip's center varies. Slip changes to stick when $d\theta_k/d\epsilon = 0$. In this case, move $\mathbf{p}_k$ from P to T.

5.3 The Squeeze Algorithm.

The algorithm starts at $d = 0$. At step l, it hypothesizes each of the four events for every possible node, and computes the extra squeeze distance ϵ for the first hypothesized event to happen. Let $d_{l+1} = d_l + \epsilon$. Event testing involves solving for ϵ and polar angles θ_t of all sliding contacts $\mathbf{p}_t$ from the event condition and the corresponding $|P|$ equations. Analytical solution is difficult if not impossible. We increment the squeeze depth d by a small step size h and use Newton's method in computing θ_t for $\mathbf{p}_t \in P$. Checking whether an event happens becomes testing either an inequality or whether an expression changes sign. If no event happens for the current increment h, the algorithm simply continues.

The algorithm terminates if the grasp succeeds when the specified d is reached, or if the grasp fails when all contacts with some finger slip before d is reached.

5.4 Finger Kinematics

In reality, the robotic fingers may have to rotate while squeezing. Let the changes in orientation of $\mathcal{F}_1$ and $\mathcal{F}_2$ be $\alpha_1(d)$ and $\alpha_2(d)$, respectively, according to hand kinematics. Some of the above derivations need adaption. Redefine θ_k as $\mathbf{p}_k$'s polar angle with respect to the center of the fingertip in the finger's local frame. In equations (5.1) and (5.2), and the conditions for Event C and D, θ_k should be replaced by $\theta_k + \alpha_i$ if $\mathbf{p}_k$ is on $\mathcal{F}_i$. An analysis of the kinematics of the fingers mounted with rounded fingertips can be found in Appenix C.

5.5 Experiment

A 0.1m × 0.1m square made of rubber foam (thickness 0.0254m) was grasped by the Barrett Hand. Figure 5.2 compares the actual grasp configuration with one from simulation by the squeeze grasp algorithm. In the center, the deformed mesh from simulation is superposed onto the real shape with an almost perfect alignment (average error is 1.3mm while the edge length is 0.1m) after data matching. Such matching aims at minimizing the difference between real experiment data and simulation data introduced by the choices of coordinate systems in simulation and experiment. The details of such matching is described in Appendix A, and the physical parameters are measured as described in Appendix B. Columns 1 and 3 give close-up comparisons between the contact regions, from experiment and simulation, on the two fingers.

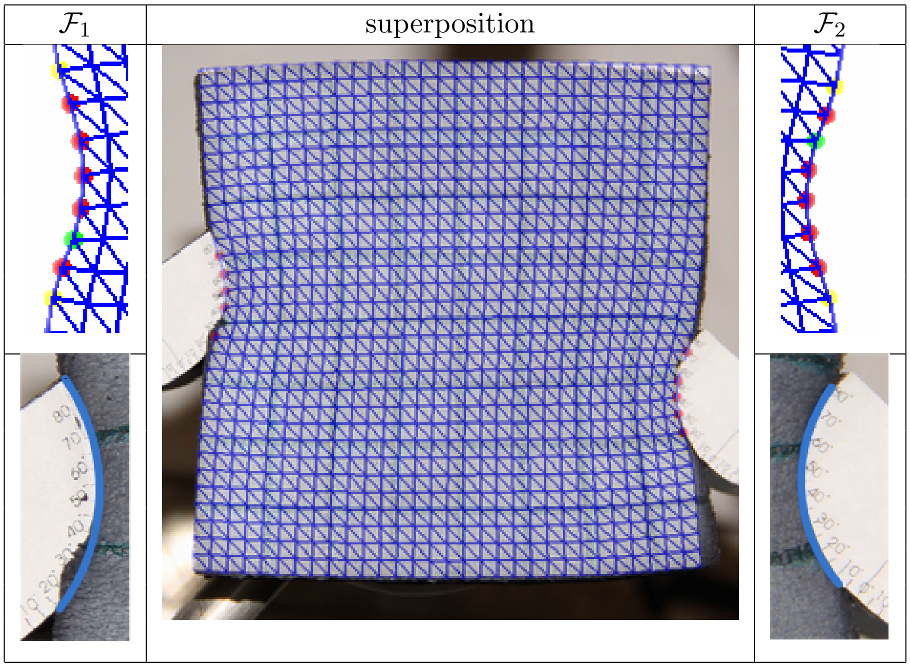


Figure 5.2: Grasp configuration with contact regions: simulation vs. experiment.

The contact forces evolve as the squeeze deepens, as shown in Figure 5.3. At different relative squeeze depth d, the forces exerted by finger 1 (left column) and 2 (right column) were decomposed to force density (upper row) and polar angle (lower row). The nodal values were then interpolated with cubic splines. The magnitude of the contact force at each nodal point calculated from simulation was transformed to the force density over curve length, while the direction was represented as its polar angle. The density in the center of the contact is generally bigger than that on the edge. Also, The force on each segment, and the total force, increases with deeper squeeze. The directions of the contact forces show a general trend of decreasing along curve length. As the squeeze continues, the curve spans over more nodes, indicating a growing contact area.

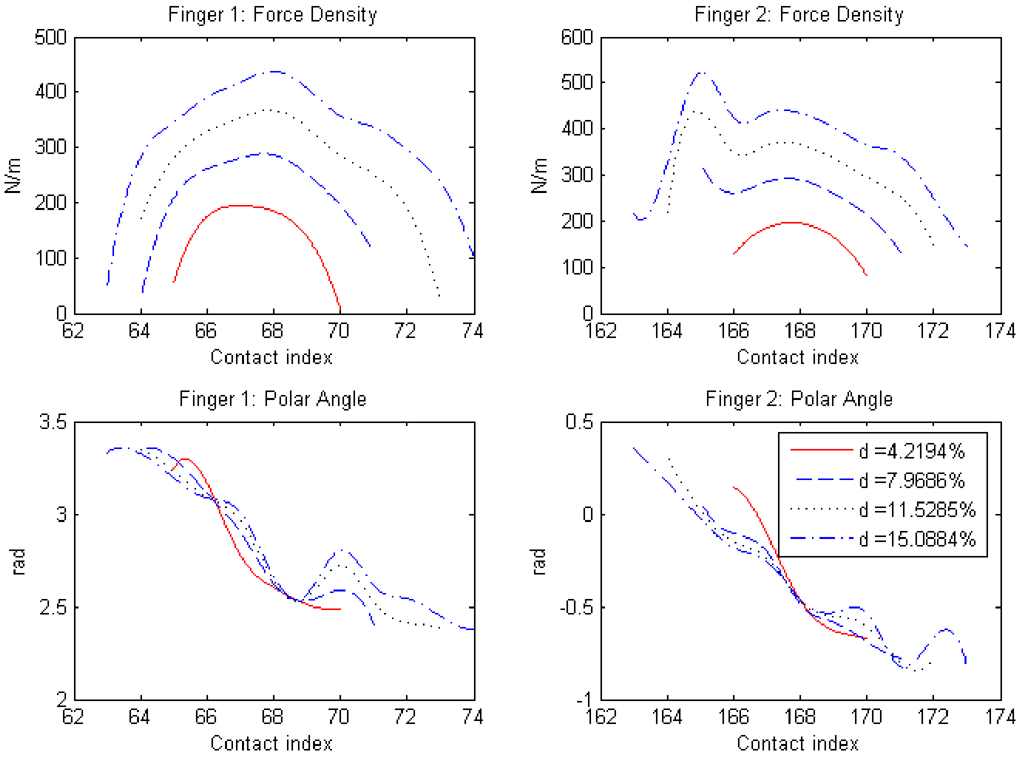


Figure 5.3: Force profile.

Among the four types of events introduced in Section 5.2, Event A is easy to picture with common sense. Event B was so rare that it was not observed in our experiment. Event C was widely observed in both simulation and experiment wherever friction is insufficient. Figure 5.4 shows some grasps in which Event C happened. Each of the yellow and red arrows in the middle and lower rows emphasizes one point on the objects and fingertips respectively. Note the changes in their relative positions when squeeze deepened, indicating an Event C.

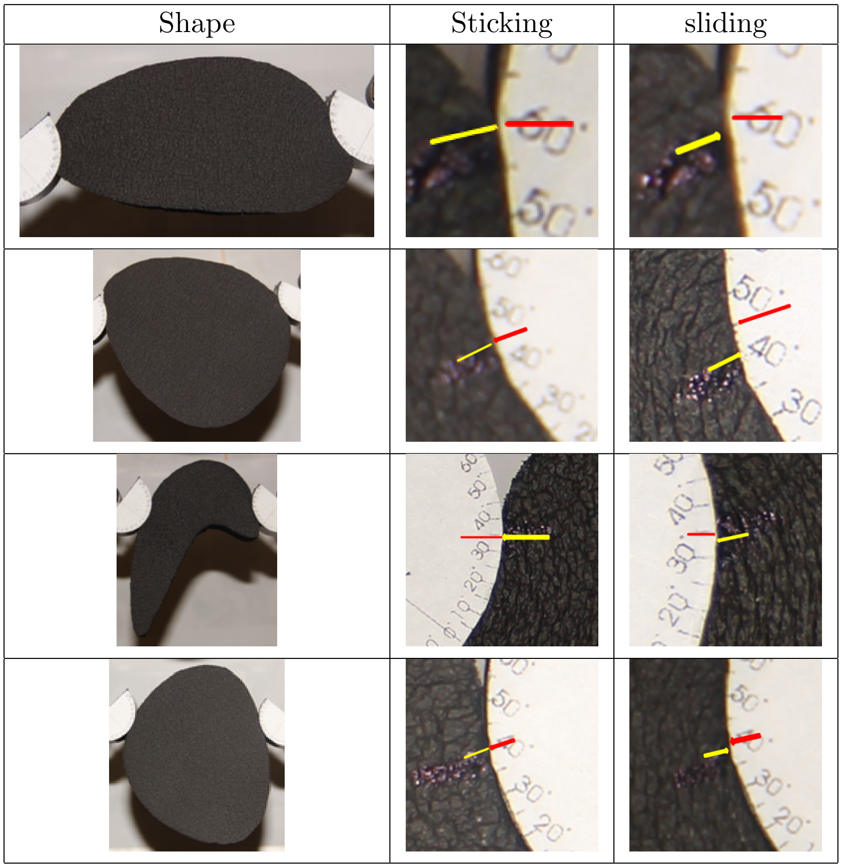


Figure 5.4: Event C.

Each contact point is in contact with one point on the cylindrical finger, and thus its position can be described by the polar angle with respect to the center of the finger. In existence of event three, its position changes with the squeeze depth. Figure 5.5 shows the evolution of sliding contacts with the growing squeeze depth. The curves indicating contacts' sliding distance are also cubic splines interpolated from nodal displacements obtained from simulation. As d increases, we can see that the sliding distance of the contacts increases.

The fourth event, i.e. sliding to sticking, happened a lot less than the previous one. In Figure 5.6, (a) and (b) show simulation results that correspond to the experiment images in

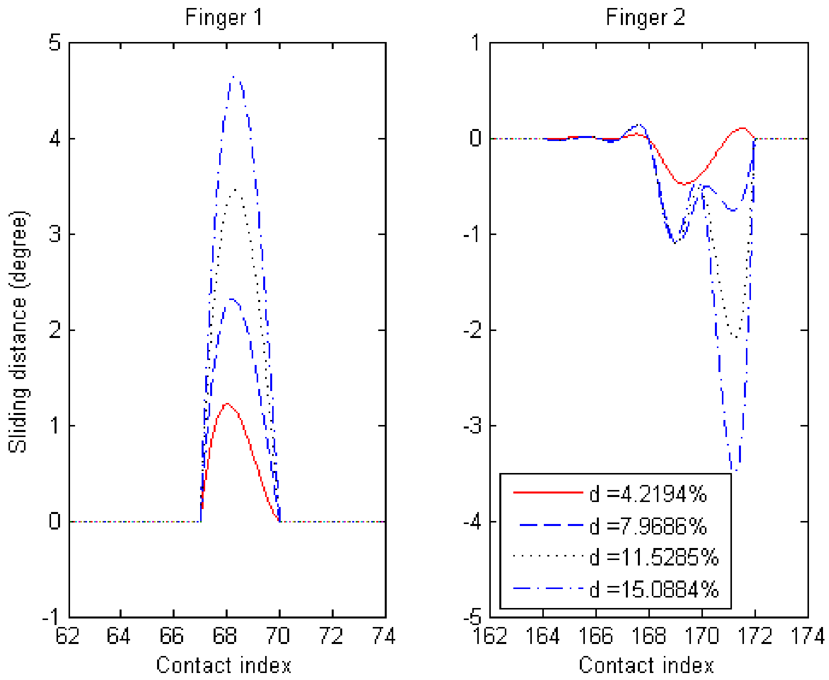
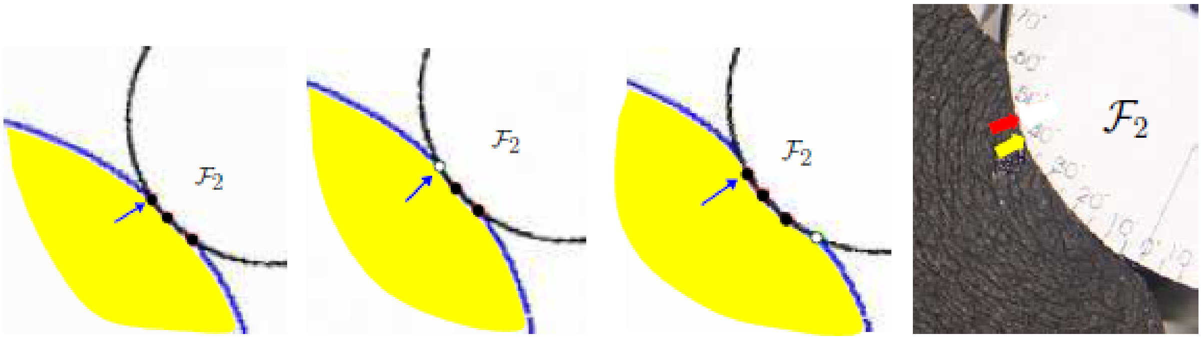


Figure 5.5: Sliding profile.

the entries (2, 3) and (3, 3) in Table 2. The blue arrows mark the same node on the object that started out sticking in (a), transitioned into sliding in (b), and switched back to sticking in (c). Shown in (d) is an experiment image that displays the distance of sliding (about five degrees) on the fingertip by the same node identified with arrows in (a)(c) from the fingertip contact location in (b), pointed at by the red arrow in (d), to the location of the new fingertip contact in (c), pointed at by the yellow arrow in (d).



(a)
(b)
(c)
(d)

Figure 5.6: Transitions of a contact from (a) stick to (b) slip to (c) slick. Here (a) and (b) are the simulation results over the second object in Table 2 that correspond to its entries (2, 3) and (3, 3), respectively. In (c), the contact has stopped sliding, which is also observed in the experiment in (d).

CHAPTER 6. GRASPING 3D OBJECTS

Having studied grasping the 2D objects, we move on to consider the task of using two fingers to pick up a deformable 3D solid on a table, which is a(horizontal) plane $\mathcal{P}$.

6.1 Stiffness Matrix

In a 3D body under a displacement field (u, v, w), the strain energy density is given as

$$U_0 = \frac{1}{2}(\sigma_x \epsilon_x + \sigma_y \epsilon_y + \sigma_z \epsilon_z + \tau_{xy}\gamma_{xy} + \tau_{yz}\gamma_{yz} + \tau_{zx}\gamma_{zx}), \tag{6.1}$$

where

$$\begin{aligned}
\epsilon_x &= \partial u/\partial x,\\
\epsilon_y &= \partial v/\partial y,\\
\epsilon_z &= \partial w/\partial z,\\
\gamma_{xy} &= \partial u/\partial y + \partial v/\partial x,\\
\gamma_{yz} &= \partial v/\partial z + \partial w/\partial y,\\
\gamma_{zx} &= \partial w/\partial x + \partial u/\partial z,
\end{aligned}$$

are strains, and

$$\sigma_x = \frac{E}{(1+\nu)(1-2\nu)}[(1-\nu)\epsilon_x + \nu\epsilon_y + \nu\epsilon_z], \tag{6.2}$$

$$\sigma_y = \frac{E}{(1+\nu)(1-2\nu)}[\nu\epsilon_x + (1-\nu)\epsilon_y + \nu\epsilon_z], \tag{6.3}$$

$$\sigma_z = \frac{E}{(1+\nu)(1-2\nu)}[\nu\epsilon_x + \nu\epsilon_y + (1-\nu)\epsilon_z], \tag{6.4}$$

$$\tau_{xy} = \frac{E}{2(1+\nu)}\gamma_{xy}, \tag{6.5}$$

$$\tau_{yz} = \frac{E}{2(1+\nu)}\gamma_{yz}, \tag{6.6}$$

$$\tau_{zx} = \frac{E}{2(1+\nu)}\gamma_{zx}, \tag{6.7}$$

are stresses, and E and v are Young's Modulus and Poisson's ratio.

Substitute equations (6.2) to (6.7) in equation (6.1),

$$U_0 = \frac{E}{4(1+\nu)}\left[\frac{2(1-\nu)}{1-2\nu}(\epsilon_x^2+\epsilon_y^2+\epsilon_z^2)+\frac{4\nu}{1-2\nu}(\epsilon_x\epsilon_y+\epsilon_y\epsilon_z+\epsilon_z\epsilon_x)+(\gamma_{xy}^2+\gamma_{yz}^2+\gamma_{zx}^2)\right]. \tag{6.8}$$

Suppose $V_i = (x_i, y_i, z_i)^T$, and $d_i = (u_i, v_i, w_i)^T$, $i = 1, 2, 3$ and 4 are the position and displacements of the vertices of a Tetrahedron respectively. Denote P the position, and d the displacement of a point inside the tetrahedron. They can be interpolated using Barycentric interpolation:

$$P = \sum_{i=1}^{4} c_i(x_i, y_i, z_i)^T, \tag{6.9}$$

$$d = \sum_{i=1}^{4} c_i(u_i, v_i, w_i)^T. \tag{6.10}$$

where c_i's are positive and $\sum_{i=1}^{4} c_i = 1$. Substitute $c_4 = 1 - \sum_{i=1}^{3} c_i$ in equations (6.9) and (6.10):

$$P = (x, y, z)^T = (x_4, y_4, z_4)^T + \sum_{i=1}^{3} c_i(x_i - x_4, y_i - y_4, z_i - z_4)^T, \tag{6.11}$$

$$d = (u, v, w)^T = (u_4, v_4, w_4)^T + \sum_{i=1}^{3} c_i(u_i - u_4, v_i - v_4, w_i - w_4)^T. \tag{6.12}$$

Taking partial derivatives with respect to c_1, c_2 and c_3, we get

$$\begin{bmatrix} \frac{\partial x}{\partial c_1} & \frac{\partial x}{\partial c_2} & \frac{\partial x}{\partial c_3} \\ \frac{\partial y}{\partial c_1} & \frac{\partial y}{\partial c_2} & \frac{\partial y}{\partial c_3} \\ \frac{\partial z}{\partial c_1} & \frac{\partial z}{\partial c_2} & \frac{\partial z}{\partial c_3} \end{bmatrix} = \begin{bmatrix} x_1 - x_4 & x_2 - x_4 & x_3 - x_4 \\ y_1 - y_4 & y_2 - y_4 & y_3 - y_4 \\ z_1 - z_4 & z_2 - z_4 & z_3 - z_4 \end{bmatrix}, \tag{6.13}$$

$$\begin{bmatrix} \frac{\partial u}{\partial c_1} & \frac{\partial u}{\partial c_2} & \frac{\partial u}{\partial c_3} \\ \frac{\partial v}{\partial c_1} & \frac{\partial v}{\partial c_2} & \frac{\partial v}{\partial c_3} \\ \frac{\partial w}{\partial c_1} & \frac{\partial w}{\partial c_2} & \frac{\partial w}{\partial c_3} \end{bmatrix} = \begin{bmatrix} u_1 - u_4 & u_2 - u_4 & u_3 - u_4 \\ v_1 - v_4 & v_2 - v_4 & v_3 - v_4 \\ w_1 - w_4 & w_2 - w_4 & w_3 - w_4 \end{bmatrix}. \tag{6.14}$$

The absolute value of the determinant of the matrix in the right hand side of equation (6.13) is 6 times the volume of the tetrahedron. Since the tetrahedrons have positive volume in this case, the matrix has to be fully ranked, and thus its inverse exists. From equation (6.13), we obtain

$$\begin{bmatrix} \frac{\partial c_1}{\partial x} & \frac{\partial c_1}{\partial y} & \frac{\partial c_1}{\partial z} \\ \frac{\partial c_2}{\partial x} & \frac{\partial c_2}{\partial y} & \frac{\partial c_2}{\partial z} \\ \frac{\partial c_3}{\partial x} & \frac{\partial c_3}{\partial y} & \frac{\partial c_3}{\partial z} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial c_1} & \frac{\partial x}{\partial c_2} & \frac{\partial x}{\partial c_3} \\ \frac{\partial y}{\partial c_1} & \frac{\partial y}{\partial c_2} & \frac{\partial y}{\partial c_3} \\ \frac{\partial z}{\partial c_1} & \frac{\partial z}{\partial c_2} & \frac{\partial z}{\partial c_3} \end{bmatrix}^{-1} = \begin{bmatrix} x_1 - x_4 & x_2 - x_4 & x_3 - x_4 \\ y_1 - y_4 & y_2 - y_4 & y_3 - y_4 \\ z_1 - z_4 & z_2 - z_4 & z_3 - z_4 \end{bmatrix}^{-1} . \tag{6.15}$$

Let us go back to the strains,

$$\epsilon_x = \frac{\partial u}{\partial x} = \sum_{i=1}^{3} \frac{\partial u}{\partial c_i}\frac{\partial c_i}{\partial x} = \sum_{i=1}^{3} \frac{\partial c_i}{\partial x}(u_i - u_4) = \sum_{i=1}^{3} \frac{\partial c_i}{\partial x} u_i - \sum_{i=1}^{3} \frac{\partial c_i}{\partial x} u_4. \tag{6.16}$$

Now ϵ_x is represented by the displacements of the 4 vertices. Similarly, we represent all the (parts of) strain terms by the displacements of vertices as

$$\begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} & \frac{\partial w}{\partial x} \\ \frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} & \frac{\partial w}{\partial y} \\ \frac{\partial u}{\partial z} & \frac{\partial v}{\partial z} & \frac{\partial w}{\partial z} \end{bmatrix} = \begin{bmatrix} \frac{\partial c_1}{\partial x} & \frac{\partial c_2}{\partial x} & \frac{\partial c_3}{\partial x} & -\sum_{i=1}^{3} \frac{\partial c_i}{\partial x} \\ \frac{\partial c_1}{\partial y} & \frac{\partial c_2}{\partial y} & \frac{\partial c_3}{\partial y} & -\sum_{i=1}^{3} \frac{\partial c_i}{\partial y} \\ \frac{\partial c_1}{\partial z} & \frac{\partial c_2}{\partial z} & \frac{\partial c_3}{\partial z} & -\sum_{i=1}^{3} \frac{\partial c_i}{\partial z} \end{bmatrix} \begin{bmatrix} u_1 & v_1 & w_1 \\ u_2 & v_2 & w_2 \\ u_3 & v_3 & w_3 \\ u_4 & v_4 & w_4 \end{bmatrix} . \tag{6.17}$$

With equation (6.17), we can assemble the stiffness matrix.

Denote Q the first matrix on the right-hand side of (6.17), and Q_{ij} its entry in row i and column j. Then

$$\epsilon_x = \frac{\partial u}{\partial x} = \sum_{i=1}^{4} Q_{1i} u_i, \quad \epsilon_y = \frac{\partial v}{\partial y} = \sum_{i=1}^{4} Q_{2i} v_i, \text{ and } \epsilon_z = \frac{\partial w}{\partial z} = \sum_{i=1}^{4} Q_{3i} w_i.$$

So

$$\epsilon_x^2 = (u_1, u_2, u_3, u_4) \begin{pmatrix} Q_{11}^2 & Q_{11}Q_{12} & Q_{11}Q_{13} & Q_{11}Q_{14} \\ Q_{12}Q_{11} & Q_{12}^2 & Q_{12}Q_{13} & Q_{12}Q_{14} \\ Q_{13}Q_{11} & Q_{13}Q_{12} & Q_{13}^2 & Q_{13}Q_{14} \\ Q_{14}Q_{11} & Q_{14}Q_{12} & Q_{14}Q_{13} & Q_{14}^2 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} , \tag{6.18}$$

$$\epsilon_y^2 = (v_1, v_2, v_3, v_4) \begin{pmatrix} Q_{21}^2 & Q_{21}Q_{22} & Q_{21}Q_{23} & Q_{21}Q_{24} \\ Q_{22}Q_{21} & Q_{22}^2 & Q_{22}Q_{23} & Q_{22}Q_{24} \\ Q_{23}Q_{21} & Q_{23}Q_{22} & Q_{23}^2 & Q_{23}Q_{24} \\ Q_{24}Q_{21} & Q_{24}Q_{22} & Q_{24}Q_{23} & Q_{24}^2 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{pmatrix} , \tag{6.19}$$

$$\epsilon_z^2 \quad = \quad (w_1, w_2, w_3, w_4) \begin{pmatrix} Q_{31}^2 & Q_{31}Q_{32} & Q_{31}Q_{33} & Q_{31}Q_{34} \\ Q_{32}Q_{31} & Q_{32}^2 & Q_{32}Q_{33} & Q_{32}Q_{34} \\ Q_{33}Q_{31} & Q_{33}Q_{32} & Q_{33}^2 & Q_{33}Q_{34} \\ Q_{34}Q_{31} & Q_{34}Q_{32} & Q_{34}Q_{33} & Q_{34}^2 \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{pmatrix}. \qquad (6.20)$$

On the other hand,

$$\begin{aligned} \gamma_{xy} &= \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} = \sum_{i=1}^{4} (Q_{1i}v_i + Q_{2i}u_i), \\ \gamma_{yz} &= \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} = \sum_{i=1}^{4} (Q_{3i}v_i + Q_{2i}w_i), \\ \gamma_{zx} &= \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} = \sum_{i=1}^{4} (Q_{3i}u_i + Q_{1i}w_i). \end{aligned}$$

Let $d_t = (u_1, v_1, w_1, \cdots, u_4, v_4, w_4)^T$. Then,

$$\gamma_{xy}^2 = (\sum_{i=1}^{4} Q_{1i}v_i)^2 + (\sum_{i=1}^{4} Q_{2i}u_i)^2 + 2(\sum_{i=1}^{4} Q_{1i}v_i \cdot \sum_{i=1}^{4} Q_{2i}u_i)$$

$$= d_t^T \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q_{11}^2 & 0 & 0 & Q_{11}Q_{12} & 0 & 0 & Q_{11}Q_{13} & 0 & 0 & Q_{11}Q_{14} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q_{12}Q_{11} & 0 & 0 & Q_{12}^2 & 0 & 0 & Q_{12}Q_{13} & 0 & 0 & Q_{12}Q_{14} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q_{13}Q_{11} & 0 & 0 & Q_{13}Q_{12} & 0 & 0 & Q_{13}^2 & 0 & 0 & Q_{13}Q_{14} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q_{14}Q_{11} & 0 & 0 & Q_{14}Q_{12} & 0 & 0 & Q_{14}Q_{13} & 0 & 0 & Q_{14}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} d_t$$

$$+ d_t^T \begin{pmatrix} Q_{21}^2 & 0 & 0 & Q_{21}Q_{22} & 0 & 0 & Q_{21}Q_{23} & 0 & 0 & Q_{21}Q_{24} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ Q_{22}Q_{21} & 0 & 0 & Q_{22}^2 & 0 & 0 & Q_{22}Q_{23} & 0 & 0 & Q_{22}Q_{24} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ Q_{23}Q_{21} & 0 & 0 & Q_{23}Q_{22} & 0 & 0 & Q_{23}^2 & 0 & 0 & Q_{23}Q_{24} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ Q_{24}Q_{21} & 0 & 0 & Q_{24}Q_{22} & 0 & 0 & Q_{24}Q_{23} & 0 & 0 & Q_{24}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} d_t$$

$$+ d_t^T$$

$$\begin{pmatrix} 0 & Q_{11}Q_{21} & 0 & 0 & Q_{12}Q_{21} & 0 & 0 & Q_{13}Q_{21} & 0 & 0 & Q_{14}Q_{21} & 0 \\ Q_{11}Q_{21} & 0 & 0 & Q_{11}Q_{22} & 0 & 0 & Q_{11}Q_{23} & 0 & 0 & Q_{11}Q_{24} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q_{11}Q_{22} & 0 & 0 & Q_{12}Q_{22} & 0 & 0 & Q_{13}Q_{22} & 0 & 0 & Q_{14}Q_{22} & 0 \\ Q_{12}Q_{21} & 0 & 0 & Q_{12}Q_{22} & 0 & 0 & Q_{12}Q_{23} & 0 & 0 & Q_{12}Q_{24} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q_{11}Q_{21} & 0 & 0 & Q_{12}Q_{23} & 0 & 0 & Q_{13}Q_{23} & 0 & 0 & Q_{14}Q_{23} & 0 \\ Q_{13}Q_{21} & 0 & 0 & Q_{13}Q_{22} & 0 & 0 & Q_{13}Q_{23} & 0 & 0 & Q_{13}Q_{24} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q_{11}Q_{23} & 0 & 0 & Q_{12}Q_{24} & 0 & 0 & Q_{13}Q_{24} & 0 & 0 & Q_{14}Q_{24} & 0 \\ Q_{14}Q_{21} & 0 & 0 & Q_{14}Q_{22} & 0 & 0 & Q_{14}Q_{23} & 0 & 0 & Q_{14}Q_{24} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$d_t$$

$$= d_t^T$$

$$\begin{pmatrix}
Q_{21}^2 & Q_{11}Q_{21} & 0 & Q_{21}Q_{22} & Q_{12}Q_{21} & 0 & Q_{21}Q_{23} & Q_{13}Q_{21} & 0 & Q_{21}Q_{24} & Q_{14}Q_{21} & 0 \\
Q_{11}Q_{21} & Q_{11}^2 & 0 & Q_{11}Q_{22} & Q_{11}Q_{12} & 0 & Q_{11}Q_{23} & Q_{11}Q_{13} & 0 & Q_{11}Q_{24} & Q_{11}Q_{14} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
Q_{22}Q_{21} & Q_{11}Q_{22} & 0 & Q_{22}^2 & Q_{12}Q_{22} & 0 & Q_{22}Q_{23} & Q_{13}Q_{22} & 0 & Q_{22}Q_{24} & Q_{14}Q_{22} & 0 \\
Q_{12}Q_{21} & Q_{12}Q_{11} & 0 & Q_{12}Q_{22} & Q_{12}^2 & 0 & Q_{12}Q_{23} & Q_{12}Q_{13} & 0 & Q_{12}Q_{24} & Q_{12}Q_{14} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
Q_{23}Q_{21} & Q_{11}Q_{21} & 0 & Q_{23}Q_{22} & Q_{12}Q_{23} & 0 & Q_{23}^2 & Q_{13}Q_{23} & 0 & Q_{23}Q_{24} & Q_{14}Q_{23} & 0 \\
Q_{13}Q_{21} & Q_{13}Q_{11} & 0 & Q_{13}Q_{22} & Q_{13}Q_{12} & 0 & Q_{13}Q_{23} & Q_{13}^2 & 0 & Q_{13}Q_{24} & Q_{13}Q_{14} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
Q_{24}Q_{21} & Q_{11}Q_{23} & 0 & Q_{24}Q_{22} & Q_{12}Q_{24} & 0 & Q_{24}Q_{23} & Q_{13}Q_{24} & 0 & Q_{24}^2 & Q_{14}Q_{24} & 0 \\
Q_{14}Q_{21} & Q_{14}Q_{11} & 0 & Q_{14}Q_{22} & Q_{14}Q_{12} & 0 & Q_{14}Q_{23} & Q_{14}Q_{13} & 0 & Q_{14}Q_{24} & Q_{14}^2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}$$

d_t.

Or in other format:

$$\begin{aligned}
\gamma_{xy}^2 &= d_t^T[(Q_{21}, Q_{11}, 0, Q_{22}, Q_{12}, 0, Q_{23}, Q_{13}, 0, Q_{24}, Q_{14}, 0)^T \\
&\quad (Q_{21}, Q_{11}, 0, Q_{22}, Q_{12}, 0, Q_{23}, Q_{13}, 0, Q_{24}, Q_{14}, 0)]d_t. \qquad (6.21)
\end{aligned}$$

Similarly,

$$\begin{aligned}
\gamma_{yz}^2 &= d_t^T[(0, Q_{31}, Q_{21}, 0, Q_{32}, Q_{22}, 0, Q_{33}, Q_{23}, 0, Q_{34}, Q_{24})^T \\
&\quad (0, Q_{31}, Q_{21}, 0, Q_{32}, Q_{22}, 0, Q_{33}, Q_{23}, 0, Q_{34}, Q_{24})]d_t, \qquad (6.22)
\end{aligned}$$

$$\begin{aligned}
\gamma_{zx}^2 &= d_t^T[(Q_{31}, 0, Q_{11}, Q_{32}, 0, Q_{12}, Q_{33}, 0, Q_{13}, Q_{34}, 0, Q_{14})^T \\
&\quad (Q_{31}, 0, Q_{11}, Q_{32}, 0, Q_{12}, Q_{33}, 0, Q_{13}, Q_{34}, 0, Q_{14})]d_t. \qquad (6.23)
\end{aligned}$$

6.2 Null Space of Stiffness Matrix

Note that equation (6.8) can be written as

$$U_0 = \frac{E}{2(1+\nu)(1-2\nu)}\left[(1-2\nu)(\epsilon_x^2+\epsilon_y^2+\epsilon_z^2)+\nu(\epsilon_x+\epsilon_y+\epsilon_z)^2+\frac{1-2\nu}{2}(\gamma_{xy}^2+\gamma_{yz}^2+\gamma_{zx}^2)\right]. \tag{6.24}$$

Since $-1 < \nu < 0.5$, we will show that $U_0 = 0$ implies that all the six terms are 0. When ν is positive, it is obvious that $U_0 = 0$ implies

$$\begin{aligned}\epsilon_x^2+\epsilon_y^2+\epsilon_z^2 &= 0,\\ (\epsilon_x+\epsilon_y+\epsilon_z)^2 &= 0,\\ \gamma_{xy}^2+\gamma_{yz}^2+\gamma_{zx}^2 &= 0,\end{aligned}$$

and thus all the six terms must be 0.

In case $-1 < \nu \leq 0$. Since

$$(\epsilon_x+\epsilon_y+\epsilon_z)^2 \leq 3(\epsilon_x^2+\epsilon_y^2+\epsilon_z^2),$$

then

$$\begin{aligned}& (1-2\nu)(\epsilon_x^2+\epsilon_y^2+\epsilon_z^2)+\nu(\epsilon_x+\epsilon_y+\epsilon_z)^2\\ \geq\ & (1-2\nu)(\epsilon_x^2+\epsilon_y^2+\epsilon_z^2)+3\nu(\epsilon_x^2+\epsilon_y^2+\epsilon_z^2)\\ =\ & (1+\nu)(\epsilon_x^2+\epsilon_y^2+\epsilon_z^2).\end{aligned}$$

The equality is achieved only when $\epsilon_x = \epsilon_y = \epsilon_z$. So

$$U_0 \geq \frac{E}{2(1+\nu)(1-2\nu)}\left[(1+\nu)(\epsilon_x^2+\epsilon_y^2+\epsilon_z^2)+\frac{1-2\nu}{2}(\gamma_{xy}^2+\gamma_{yz}^2+\gamma_{zx}^2)\right].$$

Then in this case, $U_0 = 0$ indicates that the six terms are 0. So in summary, $U_0 = 0$ implies all the six terms must all be 0, for $-1 < \nu < 0.5$.

From that

$$\begin{aligned}\epsilon_x &= \frac{\partial u}{\partial x} = 0,\\ \epsilon_y &= \frac{\partial v}{\partial y} = 0,\end{aligned}$$

$$\epsilon_z = \frac{\partial w}{\partial z} = 0,$$

we see that u, v, w are independent of x, y, z respectively.

$$\begin{aligned} u &= u(y,z), \\ v &= v(x,z),, \\ w &= w(x,y). \end{aligned}$$

From

$$\gamma_{xy} = \frac{\partial u(y,z)}{\partial y} + \frac{\partial v(x,z)}{\partial x} = 0,$$

we see that $\frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x}$ are functions of only one variable, namely z. Because otherwise, given the independence of x and y, their sum can not always be 0. Denote

$$\frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x} = F(z) + az + A, \tag{6.25}$$

where $F(z)$ is a function of only one variable z that has no first order and constant terms, and a and A are constants. Similarly, we have

$$\frac{\partial u}{\partial z} = -\frac{\partial w}{\partial x} = G(y) + by + B, \tag{6.26}$$

$$\frac{\partial v}{\partial z} = -\frac{\partial w}{\partial y} = H(x) + cx + C, \tag{6.27}$$

where G and H are single-variable functions without first order or constant terms, and b, B, c, C are constants.

From equation (6.25) and (6.26), we know that u must be in the following form:

$$u = F(z) + azy + Ay + f_1(z) + c_1 z + d_x, \tag{6.28}$$

$$= G(y) + byz + Bz + f_2(y) + c_2 y + d_x', \tag{6.29}$$

where f_1 and f_2 are single-variable functions that have no first order or constant terms. And c_1, c_2, d_x, d_x' are constants. Due to the equality of equation (6.28) and (6.29), we see the following facts:

$$F(z) = f_1(z) = 0, \quad \text{(no second order term of } z \text{ in (6.29))} \tag{6.30}$$

$$\begin{aligned} G(y) &= f_2(y) = 0, \quad \text{(no second order term of } y \text{ in (6.28))} \qquad (6.31) \\ d_x &= d'_x, \\ c_2 &= A, \\ c_1 &= B, \\ a &= b. \end{aligned}$$

So equation (6.28) and (6.29) can be written as

$$u = ayz + Ay + Bz + d_x. \qquad (6.32)$$

Following similar logic, we have the following equations:

$$\begin{aligned} v &= -axz - Ax + Cz + d_y, \\ a &= -c, \\ w &= -bxy - Bx - Cy + d_z, \\ b &= c, \end{aligned}$$

where d_y, d_z are constants. Note that $a = b = c = -a$, so $a = b = c = 0$. So we obtain

$$\begin{aligned} u &= d_x \qquad\qquad + Ay + Bz, \\ v &= \qquad d_y \qquad - Ax \qquad + Cz, \\ w &= \qquad\qquad d_z \qquad - Bx - Cy,. \end{aligned}$$

And from that, we see that $U_0 = 0$ implies that

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = d_x \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + d_y \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + d_z \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + A \begin{pmatrix} y \\ -x \\ 0 \end{pmatrix} + B \begin{pmatrix} z \\ 0 \\ -x \end{pmatrix} + C \begin{pmatrix} 0 \\ z \\ -y \end{pmatrix}. \qquad (6.33)$$

So there are 6 dimensional null space in the stiffness matrix. The first three represent translation in x-,y-,z-directions, and the latter three represent rotations about each of the 3 axis.

6.3 Constraining the Object

In this section, I will establish that at least 3 contacts, which are not co-linear, are needed to constrain the 3D object.

Lemma 1. *Let line l pass through the origin and point $P_1 = (x_1, y_1, z_1)$. Then the vector representing the object's rotation about l is a linear combination of the three rotation vectors.*

Proof. Let p_1, p_2 be two arbitrary points on an object, and their corresponding displacement d_1, d_2 represent the rotation about an axis $l = s\mathbf{a}$, where $s \in \mathcal{R}$ is a scaler and $\mathbf{a} = (x_1, y_1, z_1)$, then their angular velocity must be the same. Hence the following conditions must be satisfied:

1) $\mathbf{a} \cdot d_1 = \mathbf{a} \cdot d_2 = 0$, i.e. d_1 and d_2 are perpendicular to l;

2) $|d_1|h_2 = |d_2|h_1$, where $h_i = p_1 \times \mathbf{a}/|\mathbf{a}|$, $i = 1, 2$ is the distance from p_i to l. This means that the magnitude of the displacement is proportional to the distance;

3) $(d_1 \times \mathbf{a}) \cdot (d_2 \times \mathbf{a}) \geq 0$, i.e. any two points are rotating in the same clockwise or counter-clockwise direction.

Thus the vector R_l representing the rotation of a point $P = (x, y, z)$ about line l is collinear with

$$R_l' = (x_1, y_1, z_1) \times (x, y, z) = x_1 \begin{pmatrix} 0 \\ -z \\ y \end{pmatrix} + y_1 \begin{pmatrix} z \\ 0 \\ -x \end{pmatrix} + z_1 \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix}, \tag{6.34}$$

which is a linear combination of the three rotation vectors. □

When l is not passing through the origin, R_l is a linear combination of all the six null vectors instead of only three rotational null vectors.

From the equilibrium equation

$$K\Delta = F, \tag{6.35}$$

follows the SVD procedure as presented in [15], we have

$$M \begin{pmatrix} \bar{\mathbf{f}} \\ \mathbf{g} \end{pmatrix} = \begin{pmatrix} \bar{\delta} \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{6.36}$$

where the small matrix M with dimension $(2m+3) \times (2m+3)$, with m being the number of points in contact, i.e. with specified displacement, is

$$M = \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix}, \tag{6.37}$$

Here $\bar{\mathbf{f}}$ is the force for contact points, $\mathbf{g}$ is the projection of the deformation onto the null space, and $\bar{\delta}$ is the displacement for contact points.

Lemma 2. *If $m \leq 2$, M is singular.*

Proof. Given M is a square matrix, (6.36) has at least one solution. Suppose $(\bar{\mathbf{f}}_0^T \mathbf{g}_0^T)^T$ is a solution to it. Then according to it, we can obtain unique Δ_0 and F_0 such that $K\Delta_0 = F_0$.

Without loosing generality, we place the origin at one of the contact(s). If $m = 2$, let l be the line defined by the contacts, and if $m = 1$, let l be any line that passes through the origin. Let R_l be the vector that represents the rotation of the object about l. Note that R_l is a linear combination of the three rotational null vectors. So $KR_l = 0$. Adding with $K\Delta_0 = F_0$, we get $K(hR_l + \Delta_0) = F_0$, where $h \in \mathcal{R}$ is some scalor. Note that the displacement of contacts is still $\bar{\delta}$. So equation (6.36) must have another solution $(\bar{\mathbf{f}}_1^T \mathbf{g}_1^T)^T$ that corresponds to $\Delta_1 = \Delta_0 + hR_l$ and F_0. So M is singular. □

Note that $\bar{\mathbf{f}}_1 = \bar{\mathbf{f}}_0$. In the solution, only $\mathbf{g}$, the projection to the null space is different.

The fingers have identical hemispherical tips $\mathcal{F}_1$ and $\mathcal{F}_2$ for simplicity. Making contact with the resting object at the nodes $\mathbf{p}_i$ and $\mathbf{p}_j$ in their current locations under gravity, the fingertips first squeeze the object and later lift it via an upward translation, breaking its contact with the plane.

6.4 An Algorithm for Picking up 3D Objects

The fingers are assumed to be translating during the squeeze in constant directions, denoted by unit vectors $\hat{\mathbf{d}}_1$ and $\hat{\mathbf{d}}_2$, respectively. Without loss of generality, let $\mathcal{F}_1$ be the moving fingertip. For every unit distance $\mathcal{F}_1$ translates in $\hat{\mathbf{d}}_1$, $\mathcal{F}_2$ translates in $\hat{\mathbf{d}}_2$ by $s \geq 0$. Thus, the squeeze action can be represented by $\rho(\hat{\mathbf{d}}_1, s\hat{\mathbf{d}}_2)$, where $\rho \geq 0$ is referred to as the *squeeze depth.*

Algorithm 2 Two-finger pickup of a 3D solid

```
Input: tetrahedral mesh, table contact $\triangle \mathbf{p}_q\mathbf{p}_r\mathbf{p}_s$, finger contacts $\mathbf{p}_i, \mathbf{p}_j$, squeeze $(\hat{\mathbf{d}}_1, s\hat{\mathbf{d}}_2)$
1: Check if $\mathbf{p}_i$ and $\mathbf{p}_j$ form force closure with $\triangle \mathbf{p}_q\mathbf{p}_r\mathbf{p}_s$. If not, return failure.
2: Compute the object's initial resting configuration.
3: Squeeze the object by translating $\mathcal{F}_1$ and $\mathcal{F}_2$.
4: During the squeeze, test if the object can be lifted.
5: If so, stop squeezing to lift the object. Return success.
6: If enough squeeze has been applied and the object still cannot be picked up, report failure.
   Otherwise, go back to step 3.
```

As ρ increases, the contacts will grow from $\mathbf{p}_i$ and $\mathbf{p}_j$ into regions on the fingertips. The contact region with the plane will also change and later shrink during the lift. Denote by $\mathbb{I}, \mathbb{J}, \mathbb{K}$ the (varying) sets of the indices of the nodes that are in contact with $\mathcal{F}_1, \mathcal{F}_2, \mathcal{P}$, respectively. Their union $\mathbb{C} = \mathbb{I} \cup \mathbb{J} \cup \mathbb{K}$ consists of the indices of all contact nodes.

Algorithm 2 describes how to pick up the object. Step 1 uses a procedure from [21] to check if $\mathbf{p}_i$, $\mathbf{p}_j$, and the center of $\triangle \mathbf{p}_q\mathbf{p}_r\mathbf{p}_s$ would be force closure on a rigid body with the same shape of the resting object.[1] If no force closure, the object will not deform and the algorithm will fail.

Below we will first describe step 2 on modeling of the object's resting configuration (when it is in contact with $\mathcal{P}$ only), then step 4 on testing whether the object can be lifted after some squeeze (by considering its contacts with $\mathcal{F}_1$ and $\mathcal{F}_2$ only), and finally step 3 on squeezing the object (when it has active contacts with all of $\mathcal{F}_1$, $\mathcal{F}_2$, and $\mathcal{P}$).

6.5 Initial Resting Configuration

Before grasping the object, we need to estimate its resting configuration on the table. The object's initial resting configuration is determined from a surface triangle $\triangle \mathbf{p}_q\mathbf{p}_r\mathbf{p}_s$ in contact with $\mathcal{P}$. The vertical projection of the object's center of mass lies in the interior of the triangle. The object deforms under gravity over the triangle, causing the contact region to grow from $\triangle \mathbf{p}_q\mathbf{p}_r\mathbf{p}_s$.

[1] Since the triangle is small, the contact is approximated by its center.

Clearly, $\mathbb{I} = \mathbb{J} = \emptyset$ and $\mathbb{C} = \mathbb{K}$ since the object has not yet made contact with the fingertips. We describe an iterative procedure that computes the initial resting configuration. At the start of each iteration, a node $\mathbf{p}_k$, $1 \leq k \leq n$, has been displaced to the location $\tilde{\mathbf{p}}_k$ and receives force $\mathbf{f}_k$ if it is in contact with the plane. Let the set $\mathbb{P}$ collect the indices of sliding nodes. The following steps are carried out in the iteration.

1: $\mathbb{P} \leftarrow \emptyset$.

2: Compute the change $\mathbf{\Delta}'$ from displacement vector $\mathbf{\Delta}$, and the change $\bar{\mathbf{F}}'$ from contact force vector $\bar{\mathbf{F}}$.

3: $\mathbf{\Delta} \leftarrow \mathbf{\Delta} + \mathbf{\Delta}'$ and $\bar{\mathbf{F}} \leftarrow \bar{\mathbf{F}} + \bar{\mathbf{F}}'$.

4: If no new contact node is found and, for every $k \in \mathbb{K}$, $\mathbf{f}_k$ is inside the friction cone, terminate the algorithm.

5: Otherwise, handle new contact if any.

6: Identify sliding nodes and add their indices to $\mathbb{P}$.

7: Recompute the displacements of all sliding nodes.

In step 2, we fix every contact node $\mathbf{p}_k$ at $\tilde{\mathbf{p}}_k$, $k \in \mathbb{C}$. by setting its change in displacement $\delta'_k \leftarrow 0$. The change $\bar{d}'$ in the contact displacement vector, which consists of all δ'_k, $k \in \mathbb{C}$, is thus zero. The terms $\bar{\mathbf{F}}'$ and $\mathbf{\Delta}'$ can be evaluated by

$$\Delta = \sum_{k=1}^{3n-6} \frac{1}{\lambda}(\bar{v}_k^T \bar{\mathbf{F}}) v_k + (v_{3n-5}, \cdots, v_{3n})\mathbf{b} + \mathbf{D}, \tag{6.38}$$

where G is the gravity vector, v_k is the k-th eigenvector of the stiffness matrix, the term $\mathbf{b}$ is the vector that gathers the projections of Δ on to the null vectors, and $D = \sum_{k=1}^{3n-6} \frac{1}{\lambda}(v_k^T G) v_k$. Here $\bar{a}$ means the vector composed by the elements of vector a that are corresponding to the contact nodes. The contact force

$$\bar{\mathbf{F}} = C(\bar{\Delta} - \bar{D}) - \frac{mg}{\sqrt{n}}, \tag{6.39}$$

where C is the reduced stiffness matrix and . Set $\tilde{\mathbf{p}}_k \leftarrow \tilde{\mathbf{p}}_k + \delta'_k$, for $1 \leq k \leq n$, as described in step 3.

In step 4, a new contact exists if some node on the object would be displaced below the plane. Among all such nodes, let $\mathbf{p}_t$ be the one that is the furthermost below $\mathcal{P}$. Step 5 adds $\mathbf{p}_t$ as the new contact; namely, $\mathbb{C} \leftarrow \mathbb{C} \cup \{t\}$, and scale down δ'_t such that $\tilde{\mathbf{p}}_t + \delta'_t$ lies in $\mathcal{P}$.

Step 6 determines if any existing contact $\mathbf{p}_k$, $k \in \mathbb{C}$ and $k \neq t$, has slid by checking if $\mathbf{f}_k$ is outside the friction cone. If the condition holds, add k to $\mathbb{P}$. The sliding direction of a sliding node is opposite to the projection of f_k onto the plane. The extra displacement δ'_k due to sliding is thus determined by the sliding distance d_k.

Step 7 evaluates $\bar{\mathbf{F}}'$ and $\boldsymbol{\Delta}'$ again as in step 2, this time from the updated δ'_k for $k \in \mathbb{P}$, δ'_t, and $\delta'_l = 0$ for $l \in \mathbb{C} \setminus (\mathbb{P} \cup \{t\})$. Each updated $\mathbf{f}_k$, $k \in \mathbb{P}$, is a linear function of d_l for all $l \in \mathbb{P}$. That $\mathbf{f}_k$ is on the boundary of the friction cone yields a system of quadratic equations:

$$(1 + \mu_{\mathcal{P}}^2)(\mathbf{f}_k \cdot \hat{\mathbf{z}})^2 = \mathbf{f}_k \cdot \mathbf{f}_k, \qquad \text{for } k \in \mathbb{P}, \tag{6.40}$$

where $\mu_{\mathcal{P}}$ is the coefficient of friction between the object and the table. There are $|\mathbb{P}|$ equations in $|\mathbb{P}|$ variables d_k, $k \in \mathbb{P}$. Solve the system using the homotopy continuation method[2] [1]. Move on to the next iteration.

6.6 Squeezing and Lifting

The squeeze depth has to be large enough so that the friction between the object and the two fingertips is enough to hold the object if it is picked up. We need to perform a quick lift test in order to check if the object can be picked up. This corresponds to step 4 in Algorithm 2.

We first set $\mathbb{C} \leftarrow \mathbb{I} \cup \mathbb{J}$, as if the supporting plane $\mathcal{P}$ were suddenly removed. Then, the contact force vector and the displacement field are recomputed by specifying the displacements of only the nodes in contact with the fingertips. Finally, we identify the sliding nodes. If some finger slides or the portion of the sliding nodes is beyond some threshold, we consider the test as a failure.

Imagine we hypothetically increase the weight of the object from zero. Beyond certain value w, one finger will slide on the object. We call w as *liftable weight*. Here we assume that w increases continuously with the squeeze depth ρ. As the squeeze goes on and ρ increases, we keep track of $w(\rho)$. Initially, $w(0) = 0$. At each step l update $w(\rho_l)$ based on the current contact configuration and $w(\rho_{l-1})$. We iterate to test $w(\rho_{l-1}) + h$, $w(\rho_{l-1}) + 2h$, ..., until the object is

[2] This method solves a system $\mathbf{P}(\mathbf{x}) = 0$ by tracking the solutions of "nearby" systems starting with another system $\mathbf{Q}(\mathbf{x}) = 0$ that has a known solution.

no longer liftable at some $w(\rho_{l-1}) + kh$, where h is the step size. Set $w(\rho_l) = w(\rho_{l-1}) + (k-1)h$. Once w reaches or exceeds the actual weight of the object, the lift test is passed. Otherwise, more squeeze is needed. And the lift test fails.

In step 3 of Algorithm 2, the object is being squeezed by $\mathcal{F}_1$ and $\mathcal{F}_2$ with ρ increasing. The squeeze continues until either at some instant the lift test is passed, or the amount of squeeze becomes too large that the object is deemed impossible to pick up.

Sequence ρ into $\rho_0 = 0 < \rho_1 < \cdots$ such that at $\rho = \rho_l$ some event happens to trigger a change in the contact configuration. Within the interval $[\rho_l, \rho_{l+1})$ the changes $\bar{\mathbf{F}}'$ in the contact forces and $\mathbf{\Delta}'$ in the displacement vector are updated.

The 4 types of events are considered as is described in Section 5.2. Between two events, the movements of all sliding nodes need to be tracked. Their indices are collected in the set $\mathbb{P}$. If $\mathbf{p}_k$ slides in the plane $\mathcal{P}$, its sliding distance d_k is a variable as described in Section 6.5.

The situation that $\mathbf{p}_k$ slides on a fingertip is illustrated in Fig. 6.1. Let $\tilde{\mathbf{p}}_k$ be the sliding point, then the friction $\mathbf{f}_k$ must lie on the edge of the friction cone,

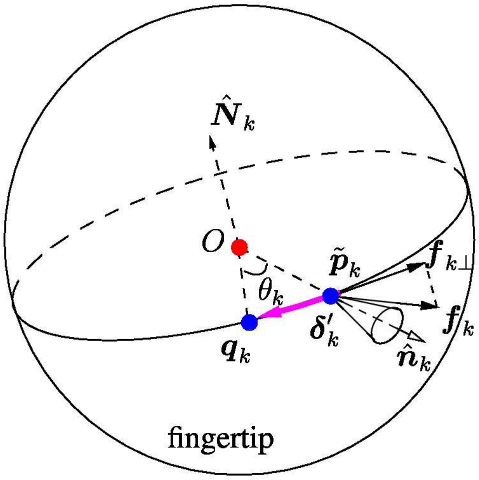


Figure 6.1: Sliding of a node on a hemispherical fingertip (with the entire sphere shown).

$$(1 + \mu_{\mathcal{F}}^2)(\mathbf{f}_k \cdot \hat{\mathbf{n}}_k)^2 = \mathbf{f}_k \cdot \mathbf{f}_k, \tag{6.41}$$

where $\mu_{\mathcal{F}}$ is the coefficient of friction and $\hat{\mathbf{n_k}}$ is the normal of the fingertip as $\tilde{\mathbf{p}}_k$. Also

$$c_k^2 + s_k^2 = 1. \tag{6.42}$$

where c_k and s_k are the cos and sin values of the angle $\tilde{\mathbf{p}}_k$ slides on the fingertip respect to the finger center.

With the updated δ_k's, whether p_k slides in the plane or on a fingertip, the change $\bar{d}'$ in the contact displacement vector has been reset. We can obtain the changes $\mathbf{\Delta}'$ in the displacement vector and $\bar{\mathbf{F}}'$ in the contact force with $\mathbf{\Delta}', \mathbf{F}', \bar{d}'$. They depend on all d_k, $k \in \mathbb{P} \cap \mathbb{K}$, and all c_l and s_l, $l \in \mathbb{P} \cap (\mathbb{I} \cup \mathbb{J})$. We end up with a system of $2|\mathbb{P} \cap (\mathbb{I} \cup \mathbb{J})| + |\mathbb{P} \cap \mathbb{K}|$ quadratic equations in the forms of (6.40), (6.41), or (6.42) in the same number of variables. Solve the system to update all contact slips using the homotopy continuation method.

We take small increments in the squeeze depth and check after each increment if any event happens. Upon such an occurrence, Newton's method is used to polish the corresponding squeeze depth value. The contact index sets $\mathbb{I}, \mathbb{J}, \mathbb{K}, \mathbb{P}$ are updated accordingly.

In step 5 of Algorithm 2, once the lift test is passed, the two fingers translate upward. During the lift, the nodal contacts with the plane $\mathcal{P}$ will break gradually, and some contacts with the fingertips could also break under gravity. Modeling, however, is no different from that of squeezing. If all the contact nodes on one finger are sliding, the object slides on the finger and the pickup fails. Otherwise, the pickup is a success when the object leaves the plane.

CHAPTER 7. RESTORATION OF GRAVITY-FREE 3D SHAPES

In the previous FEM formulation, the stiffness matrix is constructed over the (observed) shape of the object, which has already deformed under gravity. The entries of the matrix encode the effects of gravity in a nonlinear manner. Treatment of the gravitational force still as a body force in the constitutive equation is similar to taking into account the gravitational effect the second time. On the other hand, simply excluding the gravitational force from the equation will not resolve the issue, due to the nonlinearity of the encoding of geometry in the stiffness matrix. Such influence posed by gravity can be large when the density-to-Young's-Modulus ratio is not negligible. Particularly, the local contact geometry of the object and the supporting plane may be dramatically affected.

To address the gravity issue more precisely and to achieve higher accuracy in deformable modeling with non-negligible gravitational effects, the idea is to restore the gravity-free shape and stiffness matrix of the solid. The obtained matrix will not be correlated with any gravitational effect. It will be readily applied to model deformation under all body forces and external forces.

7.1 Gravity-Free Shape Restoration

When gravity is involved, the constitutive equation (3.7) becomes

$$K\Delta = F + G, \tag{7.1}$$

where G is the gravity force vector. Our task of restoring the gravity-free shape comes down to determining $\mathbf{\Delta}$ from the known deformed shape $\tilde{\mathbf{P}}$, so that $\mathbf{P} + \Delta = \tilde{\mathbf{P}}$. We introduce a fixed-point iteration algorithm. At the beginning, we initialize $\mathbf{p}_i^{(0)} = \tilde{\mathbf{p}}_i$, $i = 1, \ldots, m$. At the lth iteration, we first construct the object's stiffness matrix based on the current shape estimation $\mathbf{P}^{(l)} = \tilde{\mathbf{P}} - \mathbf{\Delta}^{(l)}$, apply a negative gravity on the object, and then obtain the new displacement estimate:

$$\mathbf{\Delta}^{(l+1)} = (1-r)\mathbf{\Delta}^{(l)} + rK^{-1}(\mathbf{\Delta}^{(l)})\mathbf{G}, \tag{7.2}$$

where r with $0 < r \leq 1$ is a control parameter. This is essentially a fixed point iteration with the iteration function given as

$$\alpha(\mathbf{\Delta}) = (1-r)\mathbf{\Delta} + rK^{-1}(\mathbf{\Delta})\mathbf{G}. \tag{7.3}$$

We also let

$$\beta(\mathbf{\Delta}) = K^{-1}(\mathbf{\Delta})\mathbf{G}. \tag{7.4}$$

The vector β represents the displacement that would be caused by gravity based on the current estimate $\tilde{\mathbf{P}} - \mathbf{\Delta}$ of the free shape. Clearly, $\alpha = (1-r)\mathbf{\Delta} + r\beta$, and $\beta = \alpha$ when $r = 1$. The iteration function (7.4) could result in large changes in $\mathbf{\Delta}$ and eventually lead to divergence. The iteration function (7.3) with a small r value avoids dramatic changes and reduces the possibility of divergence.

Under the Contraction Mapping Theorem [38, pp. 237–242], iterations (7.2) converge if the following two conditions hold:

1) α maps a convex set $D \subset \mathbb{R}^m$ onto itself;

2) the Jacobian $J_\alpha = \partial\alpha/\partial\mathbf{\Delta}$ of α satisfies $\|J_\alpha\| \leq c < 1$, for some c, over D.

Let us consider condition b) for convergence for a general solid. It follows from (7.3) that the Jacobian of α takes the form $J_\alpha = (1-r)I_m + rJ_\beta$, where I_m is the $m \times m$ identity matrix and J_β is the Jacobian of β. We have that

$$\begin{aligned} \|J_\alpha\| &\leq (1-r)\|I_m\| + r\|J_\beta\| \\ &= (1-r) + r\|J_\beta\|. \end{aligned}$$

If $\|J_\beta\| \leq c < 1$, then $\|J_\alpha\| = 1 - r + rc < 1$.

Given that α is parametrized by r, we also denote it as $\alpha(r)$ with $\alpha(1) = \beta$, and its Jacobian as $J_\alpha(r)$. Using the ∞-norm, we can obtain a stronger result. It is intuitive that when r is smaller, more damping is involved, and it is easier to converge. In terms of the norm of the Jacobian, we have the following theorem.

Theorem 5. *Suppose* $0 < r_1 < r_2 \leq 1$. *If* $\|J_\alpha(r_2)\|_\infty < 1$ *then* $\|J_\alpha(r_1)\|_\infty < 1$.

Proof. The ∞-norm of a matrix, say, $J_\beta = (b_{ij})$, is its maximum absolute row sum, i.e. $\max_{1\leq i\leq n} \sum_{1\leq j\leq n} |b_{ij}|$. For $1 \leq i \leq m$ let

$$s_i = \sum_{\substack{1\leq j\leq m \\ j\neq i}} |b_{ij}| \quad \text{and} \quad t_i = ||1 - r + rb_{ii}| + rs_i|.$$

Then, $\|J_\alpha\|_\infty = \max_i t_i$. We have

$$t_i = \begin{cases} r(s_i + b_{ii} - 1) + 1, & \text{if } b_{ii} \geq 0 \text{ or } (b_{ii} < 0 \quad \text{and } 0 < r \leq \frac{1}{1-b_{ii}}); \\ r(s_i - b_{ii} + 1) - 1, & \text{otherwise.} \end{cases} \tag{7.5}$$

Let us make an observation about the case $b_{ii} < 0$. From the above definition of t_i, when $0 < r \leq 1/(1 - b_i)$ its value $r(s_i + b_{ii} - 1) + 1$ is no less than the alternative value $r(s_i - b_{ii} + 1) - 1$; and when $1/(1 - b_{ii}) < r \leq 1$ its value $r(s_i - b_{ii} + 1) - 1$ is always greater than the alternative.

The above observation allows us to first get the maximum of the values in the first form in (7.5) for all i, and the maximum of the values in the second form for all j with $b_{jj} < 0$; and then simply compare these two maxima to obtain $\|J_\alpha\|$. More specifically, we introduce

$$c = \max_{1\leq i\leq n} (s_i + b_{ii}) \quad \text{and} \quad d = \max_{\substack{1\leq i\leq n \\ b_{ii}<0}} (s_i - b_{ii}).$$

Then, $\|J_\alpha\|_\infty = \max\{r(c - 1) + 1, r(d + 1) - 1\}$. There exist 3 cases based on c and d:

1) $c \geq 1$. Clearly, $\|J_\alpha\|_\infty \geq 1$ for all $0 < r \leq 1$.

2) $c < 1$ and $d \leq c$. Then $\|J_\alpha\|_\infty = r(c - 1) + 1 < 1$.

3) $c < 1$ and $d > c$. There are two subcases:

   (a) $0 < r \leq \frac{2}{d-c+2}$. $\|J_\alpha\|_\infty = r(c - 1) + 1 < 1$ as in case 2).

   (b) $\frac{2}{d-c+2} \leq r \leq 1$. $\|J_\alpha\|_\infty = r(d + 1) - 1$. We see that $\|J_\alpha(r_1)\|_\infty < \|J_\alpha(r_2)\|_\infty$.

In all the above cases mentioned, the condition $\|J_\alpha(r_2)\| < 1$ implies $\|J_\alpha(r_1)\| < 1$. So the theorem holds. □

Next, we take a closer look at the Jacobians J_α and J_β to find out their physical meanings, starting with J_β. Rewriting (7.4) as $K\beta = \mathbf{G}$ and differentiating both sides, we obtain $J_\beta = -K^{-1}Q$, where

$$Q\mathbf{x} = \begin{pmatrix} \beta^T(\partial \mathbf{k}_1/\partial \mathbf{\Delta})\mathbf{x} \\ \vdots \\ \beta^T(\partial \mathbf{k}_m/\partial \mathbf{\Delta})\mathbf{x} \end{pmatrix} = (\Delta K)\beta,$$

where $\Delta K = (\Delta\mathbf{k}_1, \ldots, \Delta\mathbf{k}_m)$. Let $\mathbf{x}$ be a small enough m-vector. For $1 \le j \le m$ denote $\Delta\mathbf{k}_j = (\partial \mathbf{k}_j/\partial \mathbf{\Delta})\mathbf{x}$, which represents the first order change in the jth column of the stiffness matrix K due to a change from the (predicted) shape $\tilde{\mathbf{P}} - \mathbf{\Delta}$ to the (predicted) shape $\tilde{\mathbf{P}} - (\mathbf{\Delta} + \mathbf{x})$. The matrix ΔK can be approximated as

$$\Delta K \approx K(\mathbf{\Delta} + \mathbf{x}) - K(\mathbf{\Delta}).$$

Subsequently, $Q\mathbf{x} \approx (K(\mathbf{\Delta} + \mathbf{x}) - K(\mathbf{\Delta}))\beta$ and

$$\begin{aligned} J_\beta \mathbf{x} &= -K^{-1}(\mathbf{\Delta})\Delta K\beta \\ &\approx \beta - K^{-1}(\mathbf{\Delta})K(\mathbf{\Delta} + \mathbf{x})\beta. \end{aligned} \tag{7.6}$$

The last expression above represents the opposite of the extra nodal displacement that would be generated by the force to realize the displacement β for a shape change of $-\mathbf{x}$ from its current estimate $\tilde{\mathbf{P}} - \mathbf{\Delta}$. The norms of J_β and J_α are then approximated below, based on the definition of the matrix norm:

$$\|J_\beta\| \approx \max_{\text{small}\, x} \frac{\|\beta - K^{-1}K(\mathbf{\Delta} + \mathbf{x})\beta\|}{\|\mathbf{x}\|}, \tag{7.7}$$

$$\|J_\alpha\| \approx \max_{\text{small}\, x} \frac{\|(1-r)\mathbf{x} + r(\beta - K^{-1}K(\mathbf{\Delta} + \mathbf{x})\beta)\|}{\|\mathbf{x}\|}. \tag{7.8}$$

Given the complex form of the stiffness matrix K, accurate approximations of $\|J_\alpha\|$ and $\|J_\beta\|$ would require significant sampling of $\mathbf{x}$, yielding very high computational cost. Instead, we examine the ratios in (7.7) and (7.8) at only those $\mathbf{x}$ values that will occur in the iterations. More specifically, consider the iteration function (7.4). We have

$$\begin{aligned} \|\mathbf{\Delta}^{(l+1)} - \mathbf{\Delta}^{(l)}\| &= \|\beta(\mathbf{\Delta}^{(l)}) - \beta(\mathbf{\Delta}^{(l-1)})\| \\ &\approx \|J_\beta(\mathbf{\Delta}^{(l-1)})(\mathbf{\Delta}^{(l)} - \mathbf{\Delta}^{(l-1)})\|, \end{aligned}$$

by applying the Taylor series. For $l = 1, \ldots$, evaluate the norm $\|J_\beta(\mathbf{\Delta})\mathbf{x}\|$, with $\mathbf{\Delta} = \mathbf{\Delta}^{(l-1)}$ and $\mathbf{x} = \mathbf{\Delta}^{(l)} - \mathbf{\Delta}^{(l-1)}$, according to (7.6). The obtained ratios $\|J_\beta \mathbf{x}\|/\|\mathbf{x}\|$ could give us a good idea about whether convergence happens.

We use the iteration function α with $r = 0.5$ and $r = 1$ (i.e., β for the second value) separately in attempts to restore the zero-gravity shape of a sphere shown in Fig. 7.1 with radius 0.05 and density 700. Its Young's modulus is set to 2.5×10^5 and Poisson's ratio to 0.3. The mesh representing the sphere consists of 367 vertices (328 on the surface), 1144 tetrahedrons, and 2613 triangular facets. The sphere makes contact with the plane at one triangular facet, whose three vertices are fixed. The computed $\mathbf{\Delta}$ has 2-norm 0.239769.

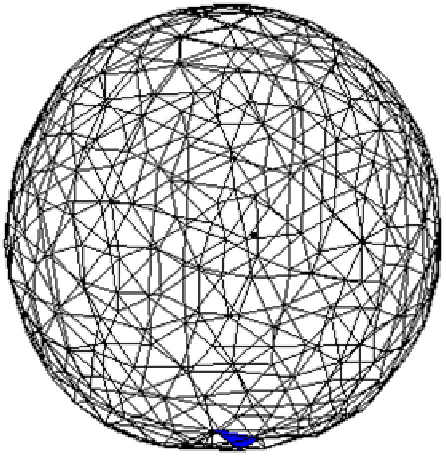


Figure 7.1: Sphere deformed by its gravity sitting on a table.

Fig. 7.2 plots, during the two executions ($r = 1$ and $r = 0.5$), the values of $\|J_\alpha \mathbf{x}\|/\|\mathbf{x}\|$, where $\mathbf{x} = \mathbf{\Delta}^{(l)} - \mathbf{\Delta}^{(l-1)}$ and $J_\alpha \mathbf{x}$ evaluated at $\mathbf{\Delta}^{(l-1)}$ approximately according to (7.6). Note that each curve describes iterations during a separate execution. The curves are lower bounds respectively for the trajectories of the norm $\|J_\alpha\|$ with $r = 0.5$ and $r = 1$. The execution with $r = 0.5$ succeeded while the one with $r = 1$ failed. In the figure we see that the plot for $r = 1$ goes up dramatically, while that for $r = 0.5$ quickly approaches 0.

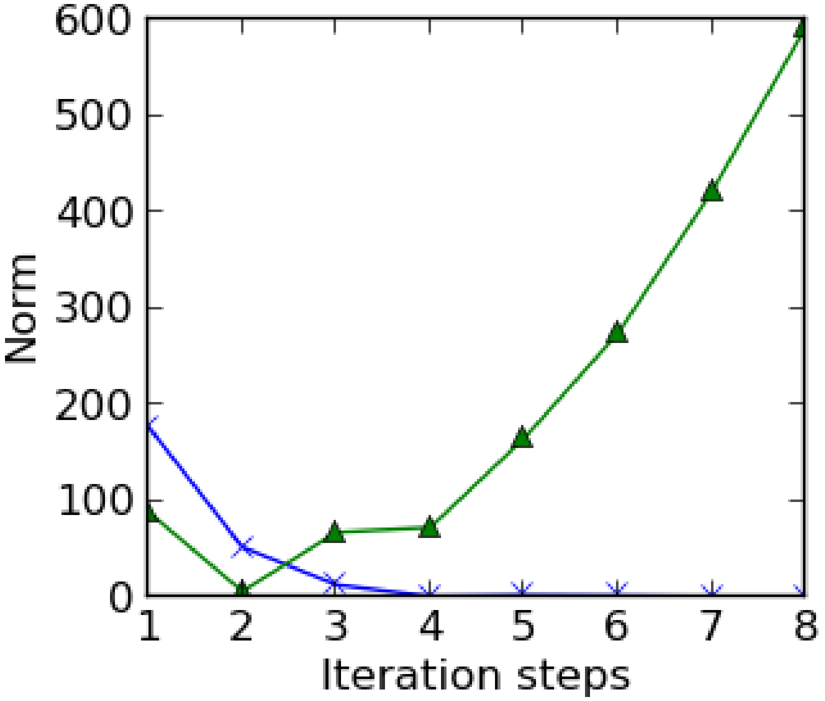


Figure 7.2: Lower bounding curves for $\|J_\alpha\|$ during separate executions ($r = 0.5$, $r = 1$), where l is the number of iterations.

7.2 Experiment

Experiment was carried out to validate the use of the restored gravity-free shape and stiffness matrix. A jelly with the brand name “Jell-O”, shown in Fig. 7.3(a), was used. Its shape was roughly a frustum of a cone. The jelly was sitting on a platform, with its base (bottom) making the contact. Its shape shown in (b) (deformed under gravity) was first acquired by a 3D scanner from NextEngine, Inc., and then simplified using the open source project MeshLab (`http://meshlab.sourceforge.net/`). Measurements of the shape are shown in the second row of Table 7.1.

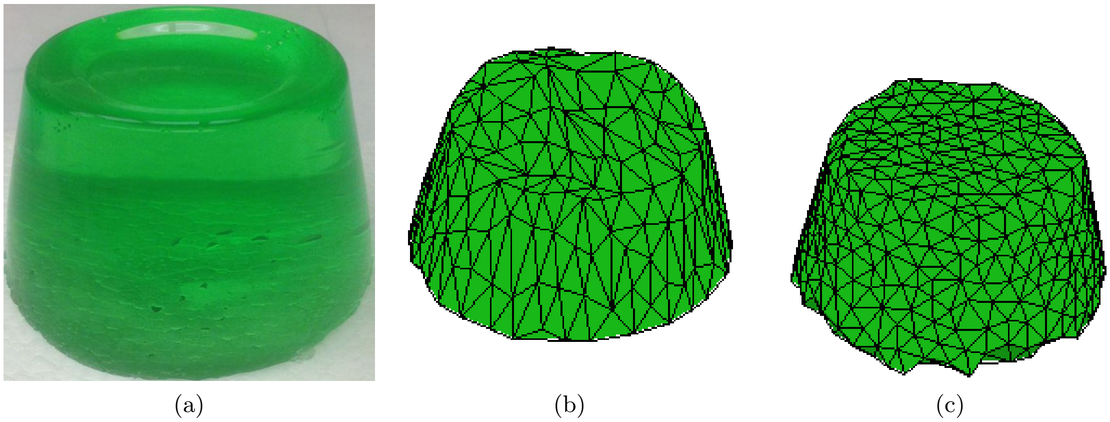
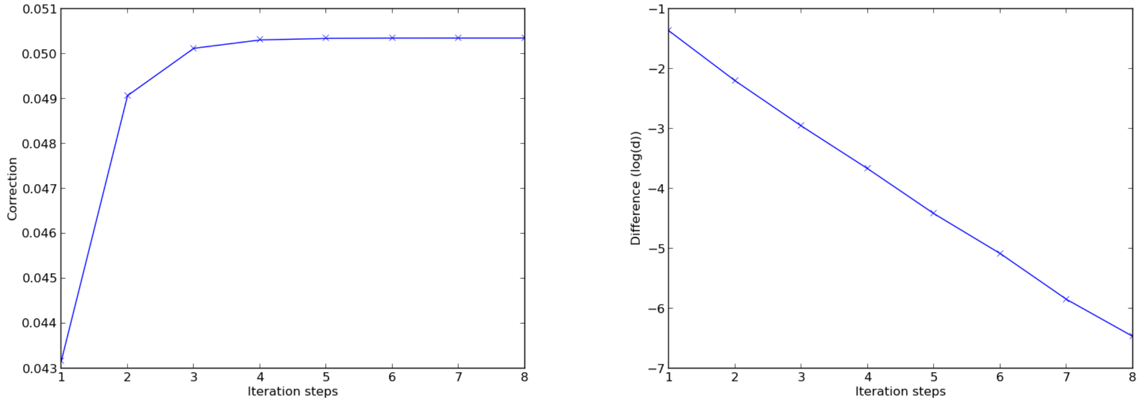


Figure 7.3: Jelly pudding: (a) image; (b) mesh output from a scanner; and (c) mesh under zero gravity.

	height	top diam.	bot. diam.	volume
original	0.031	0.050	0.062	7.68×10^{-5}
original (gravity-free)	0.034	0.050	0.060	7.85×10^{-5}
flipped	0.031	0.057	0.056	7.6×10^{-5}
flipped (gravity)	0.028	0.062	0.051	7.4×10^{-5}
flipped (gravity-free)	0.030	0.060	0.050	7.5×10^{-5}

Table 7.1: Measurements of the height, top diameter, bottom diameter, and volume of the jelly in its original, gravity-free, and flipped shapes, and the predictions of the flipped shape using the gravity-influenced and gravity-free stiffness matrices.



(a)

(b)

Figure 7.4: Convergence during restoration of the gravity-free shape in Fig. 7.3(c). (a) Norm of predicted nodal displacements from the gravity-free shape. (b) Logarithm of the norm of the difference between two consecutive predictions.

The measured Young's modulus was 3×10^3, Poisson's ratio was 0.4,[1], and density was 9.6×10^2 The mesh (b) was transformed using the code from the Computational Geometry Algorithm Library (`http://www.cgal.org/`) into a tetrahedral mesh $\mathcal{T}$ with 1119 vertices, 1012 surface facets, and 5152 tetrahedra.[2]

Assuming all the vertices on the base were fixed, we restored the gravity-free shape of the jelly. The parameter r in the iteration function α in (7.3) was set to be 1, so the iteration function β in (7.4) was used. The termination condition $\|\mathbf{\Delta}^{(k+1)} - \mathbf{\Delta}^{(k)}\| < 10^{-6}$ was met after 8 iterations. The restored shape of the jelly is shown in Fig. 7.3(c). As seen in the third row of Table 7.1, the jelly's height and volume increased by 9.68% and 2.21%, respectively. Note that the mesh $\mathcal{T}$ and the restored mesh in (c) have the same set of nodes at different locations, and different stiffness matrices.

Fig. 7.4 (a) plots the 2-norm of $\mathbf{\Delta}$, the estimated node-wise difference between the observed shape of the jelly from its gravity-free shape. It can be seen that the first iteration step made the largest shape correction. Corrections in the following steps decreased monotonically and fell within the preset tolerance of 10^{-6} at the 8th iteration step. Part (b) of the figure

[1] The measurement method for these two constants of elasticity was described in [13].

[2] The library code resampled the mesh (b) at a higher resolution.

plots the logarithm curve of the 2-norm of the difference between the shape corrections in two adjacent steps. The curve approximates a line with its slope indicating the exponential rate of error reduction.

Next, we flipped the jelly upside down as shown in Fig. 7.5(a) alongside its scanned image (b). When flipped over, the jelly's height remained the same, its original top face grew due to contact, and its original bottom face shrunk. See the fourth row of Table 7.1. Again, assuming all the vertices on the contact faces were fixed, we simulated the deformed shapes by flipping over the mesh $\mathcal{T}$ and the gravity-free mesh in Fig. 7.3 (c), respectively. The results are displayed in Fig. 7.5(c) and (d), respectively. We see that the deformation result (last row in the table) computed using the gravity-free mesh and stiffness matrix are visibly closer to the measurements of the real object (fourth row) except the bottom diameter.

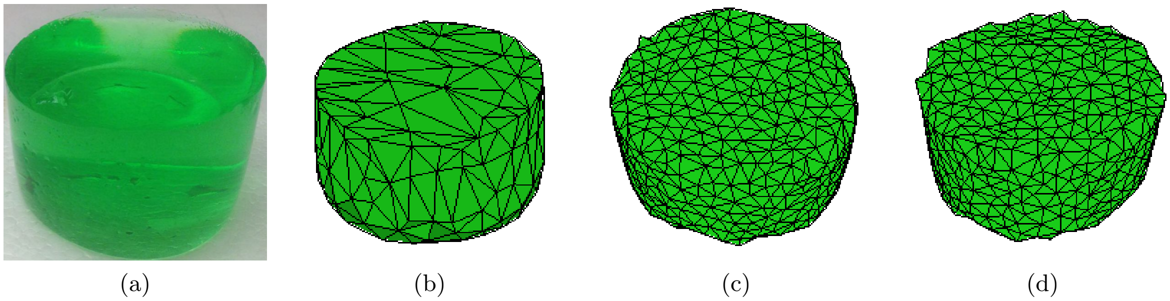


Figure 7.5: Jelly pudding from Fig. 7.3. Flipped upside down: (a) image; (b) meshed output from the scanner; (c) deformed shape predicted using the gravity-influenced mesh $\mathcal{T}$; and (d) deformed shape predicted using the gravity-free mesh in Fig. 7.3(c).

CHAPTER 8. GRASP PLANNING

This chapter describes a strategy to lift up a deformable 2D object resting on a supporting plane. Inspired by human hands, the strategy plans grasping trajectories of two fingertips. The problem can be formulated as a standard path planning problem once the goal is obtained, and is solved using modified Rapidly-exploring Random Trees (RRT). Compared with a straight squeeze, the resulting finger movement not only enlarges the graspable region, but also tries to minimize the amount of finger work. The strategy also tries to reduce the control complexity and increase robustness. The strategy is applicable to both hollow and solid 2D objects, and is extendable to 3D ones.

It is usually an easy task for a human hand to pick up a deformable object sitting on a table. Placed at certain positions on the object, the fingertips move along some trajectories in coordination until the object is off the table. The fingertips usually squeeze the object first in order to form a firm grip, and gradually turn upward. They may even apply a downward initial squeeze, in order to form the firm grip that cannot be achieved otherwise. Such movements, employed by human beings using visual and tactile information, as well as experience, are often efficient and successful. However, such amount of sensor data, or grasping skills, are not possessed by robots.

The degenerate case of a straight squeeze was studied in the previous work. While such a strategy is simple, it has some drawbacks. Firstly, there are situations where straight squeezing does not work well, while moving along a curved trajectory, may succeed. Secondly, no optimality was considered. The work done in a straight squeeze may be more than necessary.

Inspired by human hand grasping, I introduce a strategy to grasp deformable objects by specifying a trajectory for each fingertip. More precisely, given an object sitting on a table, and two finger placements on it, we want to find out a set of curves, following which, the fingertips can grasp and lift the object off of the table against gravity. With such a strategy, we will not only succeed at the locations where straight squeezes would fail, but also achieve several levels of optimality. The most important optimization is to minimize the work done by the fingers. Based on that, the control complexity is minimized. It is usually harder for a robotic finger to follow an arbitrary curve, than a piecewise straight one. It is also more complex to follow a curve with many turns, than one with less turns. In this chapter, the trajectories we planned are piecewise straight lines. The grasp stability maximization is also under consideration.

The contact between the finger and the 2D hollow object is point contact. The displacement of the fingers can be described by a point in a four dimensional space – the configuration space, which is described by a set of fourth order four dimensional polynomial equations obtained from physical constraints. Through manipulation of the stiffness matrix, the contact force, contact local geometry, and the work done by the fingers are expressed by the finger displacement in explicit form. The goal state, which is the finger displacement that lifts up the object and results in the global minimum work, is then obtained. The problem is then reduced to be a typical path planning problem with known starting and goal state in the four dimensional space.

Later we extend the above strategy to solid objects. Due to Flamant effect [39], the contact between fingertips and the object is area contact. Besides, the contact configuration changes during the grasp [15]. Such difference introduced extra difficulty. For instance, there is no clear polynomial relationship between the contact force and the finger displacement. We handle such problem by approximating the contact configuration.

In the following part of this chapter, Section 8.1 introduces the background on deformable computing given finger placement and gravity, Section 8.2 describes the path planning algorithm on hollow objects, Section 8.3 extends the strategy to solid objects, Section 8.4 shows the simulation results, and Section 8.5 concludes the work and discusses possible future works.

8.1 Deformed Shape under Contact and Gravity

This section will review linear elasticity, Finite Element Methods (FEM), and the formulation of computing deformation under finger displacement and gravity.

A ring-like object is swept out by a rectangular cross section along a closed two dimensional curve $\gamma(s)$ parameterized by arc length. Denote the displacement at $\gamma(s)$ to be $\delta(s) = \alpha(s)\mathbf{t} + \beta(s)\mathbf{n}$, where $\mathbf{t}$ is the curve's unit tangent and $\mathbf{n}$ is its unit normal. The object's strain energy is

$$U = \frac{1}{2}\int_0^L \left(h\epsilon^2 + \frac{h^3}{12}\zeta^2 \right) ds, \tag{8.1}$$

where E is the object's Young's Modulus, w is the width, h is the thickness, L is the length of the center curve, $\epsilon = d\alpha/ds - \kappa\beta$ is the *elongation,* and $\zeta = -d^2\beta/ds^2 - \alpha d\kappa/ds - \kappa d\alpha/ds$ is the *change in curvature*, with κ being the curvature of γ.

When external force $\mathbf{f}(s)$ is applied on the object and caused deformation $\delta(s)$, the load potential is

$$W = -\int_L \delta^T(s)\mathbf{f}(s)ds. \tag{8.2}$$

The deformation at equilibrium is obtained by minimizing the total potential energy $\Pi = U + W$.

Generally without a close form solution, it is solved by Finite Element Method(FEM)[11]. The object is represented as a ring of small segments with n vertices $\mathbf{p}_1, \cdots, \mathbf{p}_n$, where $\mathbf{p}_i = (x_i, y_i)^T$, for $1 \leq i \leq n$. When the object is deformed, $\mathbf{p}_i$ is deformed by δ_i to the location $\tilde{\mathbf{p}} = \mathbf{p}_i + \delta_i$. The displacement of any point can be interpolated by that of vertices. Thus the deformed shape is uniquely described by $\Delta = (\delta_1^T, \cdots, \delta_n^T)^T$, which is referred to as displacement vector.

The strain energy of every segment can be written as a quadratic form in terms of the displacements of its adjacent vertices. Assembling over all the segments gives the strain energy of the object $U = \frac{1}{2}\Delta^T K \Delta$, where K is the stiffness matrix of the object. It is symmetric and positive semi-definite with three null vectors representing the rigid body motion of translation in x-, y-direction, and rotation. Let $\mathbf{F} = (\mathbf{f}_1^T, \cdots, \mathbf{f}_n^T)^T$ be the force vector exerted to the object through contact, and $\mathbf{G} = (\mathbf{g}_1^T, \cdots, \mathbf{g}_n^T)^T$ be the gravity force, the system's total potential energy is

$$\Pi = \frac{1}{2}\Delta^T K \Delta - \Delta^T(\mathbf{F} + \mathbf{G}).$$

Minimizing Π gives the equilibrium equation:

$$K\Delta = \mathbf{F} + \mathbf{G}. \tag{8.3}$$

Similarly as in [23], we can construct the Δ vector from projections, together with the force projection on the null vectors

$$\begin{aligned} \Delta &= \sum_{i=1}^{2n-3} \frac{1}{\lambda_i}(\bar{\mathbf{v}}_i^T \bar{\mathbf{F}})\mathbf{v}_i + (\mathbf{v}_{2n-2}, \mathbf{v}_{2n-1}, \mathbf{v}_{2n})\mathbf{b} + \mathbf{g}, & (8.4) \\ 0 &= \mathbf{v}_i^T(\mathbf{F}+\mathbf{G}), \quad i = 2n-2, 2n-1, 2n, & (8.5) \end{aligned}$$

where $\mathbf{v}_i$ is the i-th eigenvector of K, $\mathbf{b}$ gathers the projections of Δ on to the three null vectors $\mathbf{v}_{2n-2}, \mathbf{v}_{2n-1}, \mathbf{v}_{2n}$, $\mathbf{g} = \sum_{i=1}^{2n-3} \frac{1}{\lambda}(\mathbf{v}_i^T \mathbf{G})\mathbf{v}_i$ is a constant vector. The notation $\bar{\mathbf{v}}$ of a vector $\mathbf{v}$ means the vector composed by elements of $\mathbf{v}$ that are corresponding to contact nodes. For example, $\bar{\mathbf{F}}$ consists the entries of $\mathbf{F}$ that are corresponding to the nodes in contact.

Extracting the rows that correspond to nodes in contact from equation (8.4), together with (8.5), a linear system for the unknown $\bar{\mathbf{F}}$ and $\mathbf{b}$ is built.

$$M \begin{pmatrix} \bar{\mathbf{F}} \\ \\ \mathbf{b} \end{pmatrix} = \begin{pmatrix} \bar{\Delta} \\ \\ 0 \end{pmatrix} - \begin{pmatrix} \bar{\mathbf{g}} \\ \\ (\mathbf{v}_{2n-2}, \mathbf{v}_{2n-1}, \mathbf{v}_{2n})^T \mathbf{G} \end{pmatrix}, \qquad (8.6)$$

where M is a constant symmetric matrix and is fully ranked if at least three terms of Δ are specified [15]. In other words, the size m of $\bar{\Delta}$ is at least 3. Let

$$M^{-1} = \begin{pmatrix} C & E \\ \\ E^T & H \end{pmatrix}.$$

where C, a matrix of dimension $m \times m$, is called reduced stiffness matrix. E and H are matrices of dimensions $m \times 3$ and 3×3 respectively.Thus

$$\begin{aligned} \begin{pmatrix} \bar{\mathbf{F}} \\ \\ \mathbf{b} \end{pmatrix} = & \begin{pmatrix} C \\ \\ E^T \end{pmatrix} \bar{\Delta} \\ & -M^{-1} \begin{pmatrix} \bar{\mathbf{g}} \\ \\ (\mathbf{v}_{2n-2}, \mathbf{v}_{2n-1}, \mathbf{v}_{2n})^T \mathbf{G} \end{pmatrix}. \qquad (8.7) \end{aligned}$$

With $\bar{\mathbf{F}}$ and $\mathbf{b}$ known, we can substitute them back into equation (8.4), and solve for any unknowns.

8.2 Grasp Planning for Hollow Objects

This section studies the grasp planning problem on a hollow object. We will first describe the problem setup, then analyze the problem properties. Using the properties, we find out the goal state of grasping that results in minimum amount of work. Finally we introduce Rapid Reaching Tree to plan the path.

8.2.1 Problem Setup

Consider a hollow object sitting vertically on a smooth supporting plane with no friction. We assume there are two points $\mathbf{p}_1$ and $\mathbf{p}_2$ in contact with the supporting plane. Without losing generality we assume the plane is horizontal. Thus

$$\mathbf{p}_{1y} = \mathbf{p}_{2y}, \quad \mathbf{p}_{1x} \neq \mathbf{p}_{2x}. \tag{8.8}$$

Since the two points stay on the table, we have the constraint

$$\delta_{iy} = 0, \tag{8.9}$$

where $\delta_{iy}, i = 1, 2$ are the displacement of $\mathbf{p}_1$ and $\mathbf{p}_2$ in the y direction. Since we assume the supporting plane has no friction,

$$f_{ix} = 0, \tag{8.10}$$

where $f_{ix}, i = 1, 2$ are contact force of $\mathbf{p}_1$ and $\mathbf{p}_2$ in the x direction.

Initially, every node $\mathbf{p}_i$ of the object displaces δ'_i under gravity $\mathbf{G}$[1]. Let $\mathbf{f}'_1 = (0, f'_1)^T$ and $\mathbf{f}'_2 = (0, f'_2)^T$ be the supporting force at $\mathbf{p}_1$ and $\mathbf{p}_2$.

Two robotic fingers then make point, frictional contact with the object at $\mathbf{p}_3$ and $\mathbf{p}_4$ respectively. According to previous derivation,

$$\begin{pmatrix} F_T \\ \\ F_{S_y} \end{pmatrix} = C \begin{pmatrix} \Psi \\ \\ \Delta_{S_y} \end{pmatrix} + \begin{pmatrix} \mathbf{0} \\ \\ f'_1 \\ \\ f'_2 \end{pmatrix}, \tag{8.11}$$

[1] When computing the deformation due to gravity, we let $\delta_{1x} = 0$ in order to let the system solvable. Such constraint is later removed and will not impact anything other than making δ' unique.

where $F_T = (f_{3x}, f_{3y}, f_{4x}, f_{4y})^T$ is the force vector of finger contacts and $F_{S_y} = (f_{1y}, f_{2y})^T$ is the y-directional force vector of supporting points. $\Psi = (\psi_{3x}, \psi_{3y}, \psi_{4x}, \psi_{4y})^T$ is the finger displacement. We do not allow slip between object and fingers, so the displacement of $\mathbf{p}_3$ and $\mathbf{p}_4$, $\Delta_T = (\delta_3^T, \delta_4^T)^T = \Psi + (\delta_3'^T, \delta_4'^T)^T$. $\Delta_{S_y} = (\delta_{1y}, \delta_{2y})^T = 0$ is the y-displacement of supporting nodes. The 6×6 matrix C is the reduced stiffness matrix.

We define the object is lifted up at the moment of the supporting force becoming 0.

$$F_{S_y} = 0. \tag{8.12}$$

The planning problem is to find a sequence of $\Psi_0, \cdots, \Psi_l$, such that $\Psi_0 = 0$, and $F_{s_y}(\Psi_l) = 0$, subject to some constraints. We will look into the necessary constraints later.

8.2.2 Reduced Stiffness Matrix

Before solving the problem, we need to analyze the property of C, the conclusion of which will be used later.

Since C is independent from gravity, in this subsection, we set $\mathbf{G} = 0$. Thus $\mathbf{f}_i' = \delta_i' = 0$, for $i = 1, \cdots, n$. Thus $\psi_i = \delta_i$ for $i = 3, 4$,

The object's strain energy

$$E = \frac{1}{2}\begin{pmatrix} \Psi \\ \\ \Delta_{S_y} \end{pmatrix}^T C \begin{pmatrix} \Psi \\ \\ \Delta_{S_y} \end{pmatrix}. \tag{8.13}$$

Theorem 6. *C is symmetric, positive semidefinite and rank(C) = null(C) = 3.*

Proof. It is easy to see C is symmetric because M is. Given (8.13), it is also easy to see C is positive semidefinite because the strain energy can never be negative.

Let

$$\bar{\Delta}_1 = \begin{pmatrix} \Psi \\ \\ \Delta_{S_y} \end{pmatrix} = (1, 0, 1, 0, 0, 0)^T, \tag{8.14}$$

Note that $\bar{\Delta}_1$ corresponds to the rigid body motion of translating the whole object in x-direction. Let the two vectors of size $2n$, $\Delta_1 = (1, 0, \cdots, 1, 0)^T$ and $F_1 = \mathbf{0}$, then Δ_1 and F_1 form an equilibrium configuration of the object satisfying the specification of $\bar{\Delta}_1$. According to [15], since $\bar{\Delta}_1$ specifies the displacements of at least two points, the configuration is the unique configuration. Thus

$$\bar{F}_1 = C\bar{\Delta}_1 = \mathbf{0}. \tag{8.15}$$

Similarly, let $\bar{\Delta}_2 = (0,1,0,1,1,1)^T$ and $\bar{\Delta}_3 = (-\mathbf{p}_{3y}, \mathbf{p}_{3x}, -\mathbf{p}_{4y}, \mathbf{p}_{4x}, \mathbf{p}_{1x}, \mathbf{p}_{2x})^T$, which result in Δ_2 and Δ_3 as global deformation respectively.

$$C\bar{\Delta}_2 = C\bar{\Delta}_3 = \mathbf{0}. \tag{8.16}$$

Note $\bar{\Delta}_3$ cannot be linearly combined by $\bar{\Delta}_1$ and $\bar{\Delta}_2$ because $\mathbf{p}_{1x} \neq \mathbf{p}_{2x}$ given $\mathbf{p}_{1y} = \mathbf{p}_{2y}$. So $\bar{\Delta}_1$, $\bar{\Delta}_2$ and $\bar{\Delta}_3$ are linearly independent and span C's 3 dimensional null space. Rank(C)$\leq$3.

Suppose $\bar{\Delta}_4$, perpendicular to $\bar{\Delta}_1$, $\bar{\Delta}_2$ and $\bar{\Delta}_3$, is another null vector of C. Then its corresponding global deformation vector Δ_4 is linearly independent of Δ_1, Δ_2 and Δ_3 since $\bar{\Delta}_i$ is part of Δ_i, $i = 1,2,3,4$. Also Δ_4 is a null vector of the global stiffness matrix K, which is a contradiction to the known result that null$(K) = 3$.

Thus rank$(C) = 3$. □

Since C is symmetric, we can write

$$C = \begin{pmatrix} C_1, C_2^T \\ \\ C_2, C_3 \end{pmatrix}, \tag{8.17}$$

where C_1, C_2, C_3 are 4×4, 2×4, and 2×2 submatrices of C, C_1 and C_3 are also symmetric since they are on the diagonal positions of C.

Theorem 7. *C_1 is positive semi-definite, and rank(C_1) $= 3$.*

Proof. Let $\bar{\Delta} = (X^T, 0)^T$, where $X \in \mathbb{R}^4$ is an arbitrary vector. So the strain energy of the object $E = \frac{1}{2}X^T C_1 X \geq 0$. It indicates that C_1 is positive semi-definite.

When $X = (1,0,1,0)^T$, $\bar{\Delta}$ lies in the null space of C. So the energy of the object $E = \frac{1}{2}X^T C_1 X = \frac{1}{2}\bar{\Delta}^T C \bar{\Delta} = 0$. Therefore $(1,0,1,0)^T$ lies in the null space of C_1 and rank$(C_1) \leq 3$.

Suppose for contradiction that rank$(C_1) < 3$. Then rank(null$(C_1)) \geq 2$. As a result, there must be another null vector of C_1, i.e. $\exists X' \perp (1,0,1,0)^T$, $|X'| \neq 0$ such that $C_1 X' = C\bar{\Delta}' = 0$, where $\bar{\Delta}' = (X'^T, 0)^T$. Then $\bar{\Delta}'$ must also lie in C's null space and be a linear combination of C's the other two null vectors:

$$\begin{pmatrix} X \\ 0 \end{pmatrix} = x_1 \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{pmatrix} + x_2 \begin{pmatrix} -\mathbf{p}_{3y} \\ \mathbf{p}_{3x} \\ -\mathbf{p}_{4y} \\ \mathbf{p}_{4x} \\ \mathbf{p}_{1x} \\ \mathbf{p}_{2x} \end{pmatrix}, \tag{8.18}$$

where $x_1, x_2 \in \mathbb{R}$ and $x_1^2 + x_2^2 \neq 0$. Take the last two equations out from (8.18)

$$\begin{pmatrix} 1 & \mathbf{p}_{1x} \\ 1 & \mathbf{p}_{2x} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{8.19}$$

Given (8.8), $\mathbf{p}_{1x} \neq \mathbf{p}_{2x}$. Thus the coefficient matrix in (8.19) is fully ranked. Therefore $x_1 = x_2 = 0$, which is a contradiction. Thus it implies $\text{rank}(C_1) = 3$. □

Theorem 8. *Rank(C_2) = 2.*

Proof. It is obvious that $\text{rank}(C_2) \leq 2$ given its dimension. It is also obvious that $\text{rank}(C_2) > 0$, because otherwise $\forall \Psi \in \mathbb{R}^4$, when $\Delta_{S_y} = 0$, $F_S = C_2 \Psi = 0$, which immediately violates equilibrium condition and is impossible.

As an overview of the proof of $\text{rank}(C_2) = 2$, we prove by contradiction that suppose otherwise, then F_T can be written as a linear combination of at most 2 constant vectors, indicating that C_1 has only rank 2, contradicting Theorem 7.

Suppose for contradiction that $\text{rank}(C_2) = 1$. Then

$$C_2 = \begin{pmatrix} a \\ b \end{pmatrix} \mathbf{v}^T, \tag{8.20}$$

where $a^2 + b^2 \neq 0$, $\mathbf{v} \in \mathbb{R}^4$, $\mathbf{v} \neq \mathbf{0}$, $a, b, \mathbf{v}$ are constants determined by C. Ψ could be any non-zero vector, and set $\Delta_{S_y} = 0$. According to (8.11)

$$\begin{pmatrix} F_T \\ F_{S_y} \end{pmatrix} = \begin{pmatrix} C_1\Psi \\ C_2\Psi \end{pmatrix} = \begin{pmatrix} C_1\Psi \\ ac \\ bc \end{pmatrix}, \tag{8.21}$$

where $c = \mathbf{v}^T\Psi$. Denote $\mathbf{f}_3 = (f_{3x}, f_{3y})^T$ and $\mathbf{f}_4 = (f_{4x}, f_{4y})^T$, so $F_T = (\mathbf{f}_3^T, \mathbf{f}_4^T)^T$. Since only the 4 points in contact take non-zero force, the object must be in equilibrium under the contact forces (which is also a direct result from that $(F_T^T, F_{S_y}^T)^T$ must be perpendicular to the null space of C since it is a linear combination of C's columns),

$$\sum_{i=3}^{4} f_{ix} = 0, \tag{8.22}$$

$$\sum_{i=1}^{4} f_{iy} = 0, \tag{8.23}$$

$$\sum_{i=1}^{4} \mathbf{p}_i \times \mathbf{f}_i = 0, \tag{8.24}$$

where (8.22), (8.23) and (8.24) represent x- and y-directional force equilibrium, and torque equilibrium, respectively. Let $d = f_{3x}$, then (8.22) leads to

$$f_{3x} = -f_{4x} = d. \tag{8.25}$$

Rewrite (8.24) as

$$\mathbf{p}_{3x} f_{3y} + \mathbf{p}_{4x} f_{4y} = d(\mathbf{p}_{3y} - \mathbf{p}_{4y}) - (a\mathbf{p}_{1x} + b\mathbf{p}_{2x})c. \tag{8.26}$$

The equations (8.23) and (8.26) contain the same variables. Let us write the linear system in the matrix form,

$$\begin{pmatrix} 1 & 1 \\ \mathbf{p}_{3x} & \mathbf{p}_{4x} \end{pmatrix} \begin{pmatrix} f_{3y} \\ f_{4y} \end{pmatrix} = \begin{pmatrix} 0 \\ \mathbf{p}_{3y} - \mathbf{p}_{4y} \end{pmatrix} d - \begin{pmatrix} a + b \\ a\mathbf{p}_{1x} + b\mathbf{p}_{2x} \end{pmatrix} c. \tag{8.27}$$

In case $\mathbf{p}_{3x} \neq \mathbf{p}_{4x}$, the coefficient matrix on the left side is fully ranked, so

$$\begin{pmatrix} f_{3y} \\ f_{4y} \end{pmatrix} = \frac{c}{\mathbf{p}_{4x} - \mathbf{p}_{3x}} \begin{pmatrix} -(a+b)\mathbf{p}_{4x} + (a\mathbf{p}_{1x} + b\mathbf{p}_{2x}) \\ (a+b)\mathbf{p}_{3x} - (a\mathbf{p}_{1x} + b\mathbf{p}_{2x}) \end{pmatrix}$$

$$+\frac{d(\mathbf{p}_{4y}-\mathbf{p}_{3y})}{\mathbf{p}_{4x}-\mathbf{p}_{3x}}\begin{pmatrix} 1 \\ \\ -1 \end{pmatrix}. \tag{8.28}$$

Combine (8.25) and (8.28), we have

$$\begin{aligned} F_T &= \frac{c}{\mathbf{p}_{4x}-\mathbf{p}_{3x}}\begin{pmatrix} 0 \\ \\ -(a+b)\mathbf{p}_{4x}-(a\mathbf{p}_{1x}+b\mathbf{p}_{2x}) \\ 0 \\ \\ (a+b)\mathbf{p}_{3x}+(a\mathbf{p}_{1x}+b\mathbf{p}_{2x}) \end{pmatrix} \\ &\quad +\frac{d}{\mathbf{p}_{4x}-\mathbf{p}_{3x}}\begin{pmatrix} \mathbf{p}_{4x}-\mathbf{p}_{3x} \\ \\ \mathbf{p}_{4y}-\mathbf{p}_{3y} \\ \\ \mathbf{p}_{3x}-\mathbf{p}_{txx} \\ \\ \mathbf{p}_{3y}-\mathbf{p}_{4y} \end{pmatrix}. \end{aligned} \tag{8.29}$$

In case $\mathbf{p}_{3x}=\mathbf{p}_{4x}$, the first row of the right hand side of (8.27) times $\mathbf{p}_{3x}$ is equal to its second row:

$$(a\mathbf{p}_{1x}+b\mathbf{p}_{2x}-(a+b)\mathbf{p}_{3x})c=(\mathbf{p}_{4y}-\mathbf{p}_{3y})d. \tag{8.30}$$

Take the first equation of (8.28),

$$f_{3y}+f_{4y}=-(a+b)c. \tag{8.31}$$

We can use only one linear equation to represent the results shown in equation (8.25),(8.30) and (8.28),

$$F_T=\left(o\begin{pmatrix} 1 \\ \\ 0 \\ \\ -1 \\ \\ 0 \end{pmatrix}-\begin{pmatrix} 0 \\ \\ a+b \\ \\ 0 \\ \\ 0 \end{pmatrix}\right)c+\begin{pmatrix} 0 \\ \\ -1 \\ \\ 0 \\ \\ 1 \end{pmatrix}e, \tag{8.32}$$

for some $e\in\mathbb{R}$, where

$$o=\frac{a\mathbf{p}_{1x}+b\mathbf{p}_{2x}-(a+b)\mathbf{p}_{3x}}{\mathbf{p}_{3y}-\mathbf{p}_{4y}},$$

is a constant. Note $\mathbf{p}_{3y} - \mathbf{p}_{4y} \neq 0$ because otherwise $\mathbf{p}_3$ and $\mathbf{p}_4$ coincide with each other given $\mathbf{p}_{3x} = \mathbf{p}_{4x}$.

Note that in both equation (8.29) and (8.32), all terms on the right hand side, other than c, d and e on the right hand side, depend only on C and contact locations. In other words, they are constant to Ψ. This means F_T can always be linearly spanned by at most 2 vectors, which implies that rank$(C_1) = 2$ given Ψ is an arbitrary vector. Thus, we found a contradiction to Theorem 7. So rank$(C_2) = 2$. □

8.2.3 Constraints

In the process of grasping, the object should be fully constraint by the fingers (together with the supporting plane). So the contact force at $\mathbf{f}_3$ and $\mathbf{f}_4$ should always stay inside the friction cone.

$$\frac{\mathbf{f}_i \cdot \mathbf{N}_i}{|\mathbf{f}_i|} \geq \frac{1}{\sqrt{1 + (\kappa\mu)^2}}, \tag{8.33}$$

where $i = 3, 4$, μ is the friction coefficient and $\kappa \in (0, 1]$ is the *safety parameter.* Since there is control and modeling errors, we hope the force has a certain margin with the friction cone. The bigger κ is, the smaller the margin is, the less "harsh" the constraint is, and the less safe it is for the grasping. According to the FEM interpolation scheme, The inward normal $\mathbf{N}_i$ of $\mathbf{p}_i$ is given as

$$\mathbf{N}_i = \frac{\mathbf{n}_i + L_i\Psi}{|\mathbf{n}_i + L_i\Psi|}, \tag{8.34}$$

where the vector $\mathbf{n}_i$ is the initial normal under gravity with i^+ and i^- being the index of $\mathbf{p}_i$'s two neighbors,

$$\mathbf{n}_i = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} ((\mathbf{p}_{i^+} + \delta'_{i^+}) - (\mathbf{p}_{i^-} + \delta'_{i^-})),$$

and

$$\begin{aligned} L_i &= \frac{\partial(\delta_{i^+}^T, \delta_{i^-}^T)^T}{\partial\Psi} \\ &= \sum_{j=1}^{2n-3} (\check{\mathbf{v}}_j \bar{\mathbf{v}}_j^T C) + (\check{\mathbf{v}}_{2n-2}, \check{\mathbf{v}}_{2n-1}, \check{\mathbf{v}}_{2n})E^T, \end{aligned}$$

where $\check{\mathbf{v}}_j = (\mathbf{v}_{j,2i^+}, \mathbf{v}_{j,2i^++1}, \mathbf{v}_{j,2i^-}, \mathbf{v}_{j,2i^+-1})$, $j = 1, \cdots, 2n$, is the influence matrix of Ψ on δ_{i+} and δ_{i^-}.

Write $C_1 = (R_3^T, R_4^T)$, where R_3 and R_4 are both of dimension 2×4. Then

$$\mathbf{f}_i = R_i \Psi, \tag{8.35}$$

Substitute (8.34) and (8.35) to (8.33), we have the constraints for the set of possible Ψ:

$$\begin{aligned} \Psi^T R_i^T L_i \Psi + \Psi^T R_i^T \mathbf{n}_i &\geq 0, & (8.36)\\ (L_i\Psi + \mathbf{n}_i)^T (R_i\Psi(R_i\Psi)^T - \frac{(R_i\Psi)^T R_i\Psi}{1+(\kappa\mu)^2} I)(L_i\Psi + \mathbf{n}_i) &\geq 0, & (8.37) \end{aligned}$$

for $i = 3, 4$. In the four constraint equations, all parts other than Ψ are constant. So the constraints are (up to) fourth order, four dimensional polynomials.

8.2.4 Goal State

The system under consideration is conservative. Thus the work done by the finger depends only on the final value of Ψ.

$$W = \frac{1}{2}\Psi^T C_1 \Psi.$$

We naturally want to minimize the work at the final state.

Let $J = (f_{1y}, f_{2y})^T$ denote the supporting force vector under only gravity.

At the moment of lifting up the object,

$$F_{S_y} = C_2\Psi + J = 0. \tag{8.38}$$

Since C_2 is fully ranked, the inverse of $C_2C_2^T$ exists. Thus the general solution of the linear system can be calculated:

$$\Psi = -C_2^T(C_2C_2^T)^{-1}J + \alpha V_1 + \beta V_2,$$

where $V_1 = (1, 0, 1, 0)^T$ and $V_2 \perp V_1$, which can be obtained through Gram-Schmidt procedure, are two vectors that span the null space of C_2, $\alpha, \beta \in \mathbb{R}$ are projections of Ψ with V_1 and V_2 respectively. Since V_1 is pure translation in the x-direction, and result in no change in deformation, force or energy, we can choose to set $\alpha = 0$. In that case

$$\Psi = \Psi_c + \beta V_2. \tag{8.39}$$

where $\Psi_c = -C_2^T(C_2C_2^T)^{-1}J$. So substituting the above equation to the work of the fingers, W is represented in terms of β:

$$W = \frac{1}{2}\Psi^T C_1 \Psi^T = a_0 + a_1\beta + a_2\beta^2, \tag{8.40}$$

where

$$\begin{aligned} a_0 &= \Psi_c^T C_1 \Psi_c, \\ a_1 &= 2\Psi_c^T C_1 V_2, \\ a_2 &= V_2^T C_1 V_2. \end{aligned}$$

Note that $a_0, a_2 > 0$ given C_1 is positive semi definite and neither V_2 nor Ψ_c lies in C_1's null space.

Substitute (8.39) into (8.36) and (8.37), we obtain the constraint for the target displacement:

$$a_{i1}\beta^2 + a_{i2}\beta + a_{i3} \geq 0, \tag{8.41}$$

$$a_{i4}\beta^4 + a_{i5}\beta^3 + a_{i6}\beta^2 + a_{i7}\beta + a_{i8} \geq 0, \tag{8.42}$$

where

$$\begin{aligned} a_{i1} &= v_2^T v_1, \\ a_{i2} &= v_2^T v_5 + v_1^T v_4, \\ a_{i3} &= v_4^T v_5, \\ a_{i4} &= (v_2^T v_1)^2 - \frac{v_1^T v_1 v_2^T v_2}{1+(\kappa\mu)^2}, \\ a_{i5} &= 2v_1^T v_2(v_2^T v_5 + v_1^T v_4) - 2\frac{v_4^T v_2 v_1^T v_1 + v_2^T v_2 v_1^T v_5}{1+(\kappa\mu)^2}, \\ a_{i6} &= 2v_1^T v_2 v_4^T v_5 + (v_2^T v_5 + v_1^T v_4)^2, \\ &- \frac{v_2^T v_2 v_5^T v_5 + 4v_2^T v_4 v_1^T v_5 + v_1^T v_1 v_4^T v_4}{1+(\kappa\mu)^2}, \\ a_{i7} &= 2v_4^T v_5(v_5^T v_2 + v_4^T v_1) - 2\frac{v_2^T v_4 v_5^T v_5 + v_1^T v_5 v_4^T v_4}{1+(\kappa\mu)^2}, \\ a_{i8} &= (v_4^T v_5)^2 - \frac{v_4^T v_4 v_5^T v_5}{1+(\kappa\mu)^2}, \\ v_1 &= L_i V_2, \\ v_2 &= R_i V_2, \end{aligned}$$

$$\begin{aligned} v_3 &= L_i \Psi_c, \\ v_4 &= R_i \Psi_c, \\ v_5 &= \mathbf{n}_i + L_i \Psi_c. \end{aligned}$$

Let β^* minimizes (8.40) subject to (8.41) and (8.42), Ψ^* given by (8.39) with $\beta = \beta^*$ is the target state.

8.2.5 Planning

The grasp planning problem now is reduced to a classical path planning problem: start from 0, find a path to Ψ^* in the four dimensional configuration space described by (8.36) and (8.37). We use the Rapid Reaching Tree (RRT) algorithm[20] to solve the path planning problem.

In the RRT algorithm, we grow two Rapid Reaching Trees from both the start and the target iteratively. At each iteration, a point $\alpha(i)$ is randomly generated in the configuration space. The algorithm finds the nearest point from one of the existing tree q_n to it. Connect $q_n\alpha(i)$ by a line segment. Such line segment may be interrupted by obstacles. In this case, the algorithm finds q_s, the point on the segment nearest to $\alpha(i)$ that is connected to q_n. Add q_s and $q_n\bar{}q_s$ to the tree. Then the algorithm tries to connect q_s to the other tree as well. If the second line segment is not interrupted, we have find a path and return the solution. We iterate the above procedure for a given amount of times before returning failure. The algorithm gives a sequence P of road points $\Psi_0, \Psi_1, \cdots, \Psi^*$, following which an object is lifted from the supporting plane.

Note that, as stated earlier, the vector $(1, 0, 1, 0)^T$ is a null vector of C_1, C_2 and thus R_1 and R_2. In other words, any component of Ψ in this direction will not affect the deformation, the support and grasp force, or the energy. Thus eliminating such vector can simply the problem to be three dimensional.

The RRT algorithm may generate more road points than necessary. In our scenario, every road point on the path indicates a turn for the fingers. Thus the number of road points is an indicator of the complexity of execution which we want to minimize. We perform an algorithm on the path given by RRT to simplify it.

Given a sequence P of road points $\Psi_1, \Psi_2, \cdots, \Psi_m$, where $m \geq 2$, we sequentially check whether a node Ψ_i, $2 \leq i \leq m-1$ can be omitted. That is, we check whether Ψ_{i-1} and Ψ_{i+1} can be connected by a collision free line. To check the segment $\overline{\Psi_a \Psi_b}$ is collision-free, we linearly interpolate the two points, such that any point Ψ_p on the line segment can be written as $\Psi_p = t\Psi_a + (1-t)\Psi_b$ for some $t \in [0,1]$. Substituting it into (8.36) and (8.37) will give us two, up to fourth order, polynomial constraints on t, which can be solved in constant time. The line is collision free if and only if the solution interval contains the entire interval $[0,1]$. We keep or remove such node according to the test result and move on to the next one, until all nodes are tested. It is easy to see such algorithm takes time linear to the size of the path.

The algorithm works well in practice. The path given directly by RRT usually has about 10 points, while most of the processed path has only 3 points including starting and target point.

Given any point Ψ, the force and normal at each contact point form an angle

$$\theta_i = \arcsin \frac{\mathbf{f}_i \times \mathbf{N}_i}{|\mathbf{f}_i|},$$

for $i = 1, 2$. The half friction cone angle is

$$\theta_m = \arctan \mu.$$

The ratio θ_i/θ_m is an indication of how much the force deviates from the normal of the contact, and thus an indication of how unstable the grasp is at this contact. Therefore we can use the tuple $D = (\theta_1, \theta_2)/\theta_m$ to indicate the vulnerableness of the grasp at the point Δ. To increase the overall stability of the planned path, we control the generation of the random nodes. Instead of generating Δ uniformly at random, we do so in the way that D follows a truncated 2-variate normal distribution. i.e.

$$\mathrm{g}(D = (x,y)) = \begin{cases} \frac{f(x,\sigma_1,e_1)f(y,\sigma_2,e_2)}{p}, \text{ if } (x,y) \in [0,1]^2, \\ 0, \text{ otherwise}, \end{cases} \tag{8.43}$$

where $f(x, \sigma, e)$ is the normal probability density at x, taking σ as the standard deviation, and e as the expectation, $i = 1, 2$. And p is the normalization coefficient. Usually the parameter is chosen as $\sigma_1 = \sigma_2 = 0.5$ and $e_1 = e_2 = 0$.

8.3 Grasp Planning for Solid Objects

After studying grasp planning on hollow object, we extend the same strategy to solid objects. Such objects can be seen as a generalized cylinder which is swept out by translating a region bounded by a 2D curve in the xy-plane along the z-direction by a short distance. The strain energy is given as

$$U = \frac{hE}{4(1+\nu)} \iint_S \left(\frac{2}{1-\nu}(\epsilon_x^2 + 2\nu\epsilon_x\epsilon_y + \epsilon_y^2) + \gamma_{xy}^2\right) dxdy,$$

where ϵ_x and ϵ_y are elongation in x- and y-direction respectively, and γ_{xy} is shearing. When applied FEM, such object is modeled by a triangular mesh. The displacement of any point within a triangle is interpolated by that of the triangle's vertices. The equations (8.3) to (8.7) carry over.

A solid object deforms under gravity. Two round fingers are making frictional contact with the object. Due to Flamant effect[39], the fingers make area contact once they squeeze the object. As a result of changing contact configuration, the system is no longer linear. Thus, the clean formulations in Section 8.2 do not hold. In stead, we will make estimations to solve the problem.

8.3.1 Configuration Space

Suppose at the start of grasp, the fingers' positions are $\mathbf{q}_1 = (q_{1x}, q_{1y})^T$ and $\mathbf{q}_2 = (q_{2x}, q_{2y})^T$. The 4D displacement vector of fingers Ψ can always be spanned by four basis vectors

$$\begin{aligned}
\mathbf{t} &= (\sqrt{2}/2, 0, \sqrt{2}/2, 0)^T, \\
\mathbf{l} &= (0, \sqrt{2}/2, 0, \sqrt{2}/2)^T, \\
\mathbf{s} &= \frac{(q_{2x} - q_{1x}, q_{2y} - q_{1y}, q_{1x} - q_{2x}, q_{1y} - q_{2y})^T}{\sqrt{2[(q_{2x} - q_{1x})^2 + (q_{2y} - q_{1y})^2]}}, \\
\mathbf{r} &= \frac{(-q_{1y}, q_{1x}, -q_{2y}, q_{2x})^T}{\sqrt{q_{1x}^2 + q_{1y}^2 + q_{2x}^2 + q_{2y}^2}},
\end{aligned}$$

which correspond to translation in x-direction, translation in y-direction (lifting), squeezing, and rotation. As mentioned before, the translation in x-direction does not introduce any deformation or contact force, and thus is irrelevant to the grasping. We also do not consider the rotation vector because of two reasons: 1. the linear elasticity model cannot handle large rotations; 2. during the grasp process, the object should be kept in roughly the same orientation. Thus we only consider Ψ from the two dimensional space spanned by $\mathbf{l}$ and $\mathbf{s}$. Let $\Phi = (s, l)^T$ be a state in the space, then

$$\Psi = (\mathbf{s}, \mathbf{l})\Phi. \tag{8.44}$$

Any state inside the state space should correspond to a configuration where the object is constraint, that is, the object should not slip off of any fingers. Given any state Φ, we can calculate the contact configuration at both fingers using the event driven algorithm described in [13], with which in hand, we check if the force at every contact region is out of the friction cone. Whether the forces are inside the friction cone determines whether Φ is in the state space.

8.3.2 Goal State

At the moment the object is lifted up, the contact force between the lowest point of the object and the supporting plane becomes 0. Let the contact set with the fingers be T and $m = |T|$. Such a state can be interpreted from another perspective: if the object is only in contact with the fingers with contact set T, then $\tilde{\mathbf{p}}_{\text{low},y} = 0$, where $\tilde{\mathbf{p}}_{\text{low}}$ is the location of the lowest point of the object at the current deformation.

Suppose the contact set stays the same, the displaced location of the lowest point is linearly determined by the contact points' displacements

$$\tilde{\mathbf{p}}_{\text{low},y} = D\bar{\Delta} + b + \mathbf{p}_{\text{low},y}, \tag{8.45}$$

where $\bar{\Delta}$ is the displacement vector of the contact nodes, $D = \partial\delta_{low,y}/\partial\bar{\Delta}$ is a m dimensional row vector, b is a constant value indicating the deformation due to gravity. D and b are constant matrix and vector respectively, and can be obtained from equation (8.4). When sliding is neglected, $\bar{\Delta}$ is linearly determined by Ψ, which in turn is linearly determined by Φ. So denote

$$\bar{\Delta} = A\Phi + B, \tag{8.46}$$

where A is a $2m \times 2$ constant matrix, and B is a constant $2m$ dimensional column vector. From equation (8.45) and (8.46), at lifting up

$$DA\Phi + DB + b + \mathbf{p}_{\text{low},y} = 0.$$

Thus all Φ at lifting up satisfy

$$\Phi = V_0 t - V_1, \tag{8.47}$$

where $t \in \mathcal{R}$ is a free variable, $V_0 = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} DA$ and $V_1 = \begin{pmatrix} 0 \\ (DB + b + \mathbf{p}_{\text{low},y})/b \end{pmatrix}$.

The contact force is linear to the contact displacement

$$\bar{F} = C\bar{\Delta} + F_c, \tag{8.48}$$

where C is the reduced stiffness matrix, and F_c is some constant due to gravity and compliance to finger shape. Substitute equation (8.47) and (8.46) into (8.48),

$$\bar{F} = V_2 t + V_3, \tag{8.49}$$

where $V_2 = CAV_0$ and $V_3 = CAV_1 + CB + F_c$. Now every component of $\bar{F}$ is linearly represented by t. To estimate the range of t that prevent the fingers from slipping, we calculate

$$F_a = \sum_{i \in T_j} R_i F_i,$$

where T_j, $j = 1, 2$ is the contact set for finger j. The rotation matrix $R_i = \begin{pmatrix} N_{iy} & -N_{ix} \\ N_{ix} & N_{iy} \end{pmatrix}$ rotates the unit normal N_i at point i to $(0,1)^T$. F_a stays inside the friction cone if

$$\begin{aligned} |F_{ax}| &\leq |F_{ay}|, \\ F_{ay} &\geq 0. \end{aligned}$$

Write $F_a = (f_{a1}t + f_{a2}, f_{a3}t + f_{a4})^T$, the above constraint becomes

$$(f_{a1}^2 - f_{a3}^2)t^2 + 2(f_{a1}f_{a2} - f_{a3}f_{a4})t + f_{a2}^2 - f_{a4}^2 \leq 0, \tag{8.50}$$

$$f_{a3}t + f_{a4} \geq 0, \tag{8.51}$$

Solving the constraint given by the above equations gives us a set of intervals of estimated valid t value.

The work done by the fingers is the estimated as the change of potential energy. Since before the grasp, the system's potential energy is constant, we only need to look at the energy at lifting up

$$E = \frac{1}{2}\Delta^T K \Delta + \mathbf{G}^T \Delta. \tag{8.52}$$

Conveniently write

$$\Delta = L\bar{\Delta} + Q, \tag{8.53}$$

where L and Q are matrices that can be derived from equation (8.4) to (8.7). Substitute equation (8.53) , (8.46) and (8.47) into (8.52)

$$E = e_2 t^2 + e_1 t + e_0,$$

where $e_2 = \frac{1}{2} V_0^T A^T L^T K L A V_0$, $e_1 = V_0^T A^T L K(LB + Q - LAV_1) + G^T LALAV_0$ and $e_0 = \frac{1}{2}(LB + Q - LAV_1)^2 - G^T LAV_1$. We find the t^* that minimizes E over the valid interval. And the corresponding Φ^* is the target state. With target state known, we can again apply RRT to plan the path.

8.3.3 Planning Algorithm

In planning the grasp, we first estimate the contact region by squeezing the object to a degree that it is liftable. Then using this contact region, we estimate the target location, and perform RRT to navigate to the target from the starting position. If the contact region is different from initial estimation, we replace the initial estimation with current one and iterate the procedure again until the region stays the same between two estimates.

8.4 Simulation

In this section, some simulation results are presented. Unless otherwise stated, the metric system is used. As is shown in Figure 8.1(a), an elliptical hollow object sit on the table with two points supporting it. The ellipse's long and short axis was 0.164 and 0.118 respectively. Its

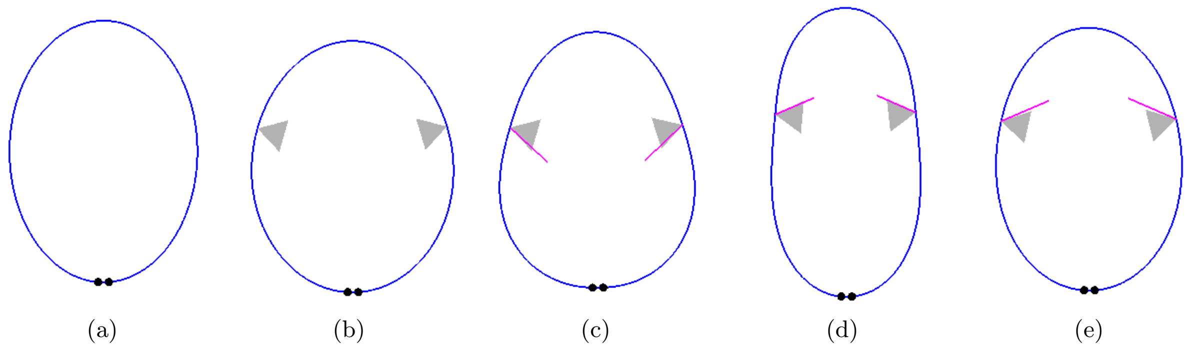


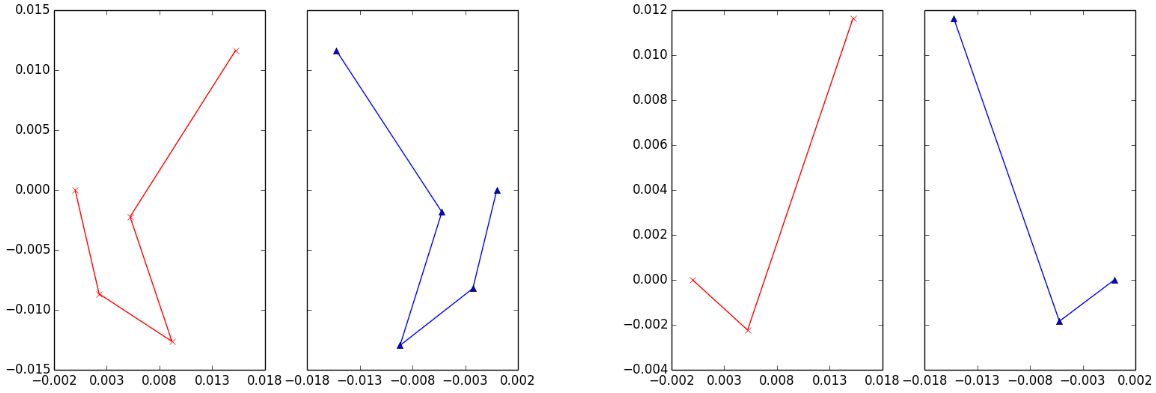
Figure 8.1: Grasping an ellipse: (a) the object is at its original shape; (b) it deforms under gravity; (c) it is under grasp at intermediate state as in Figure 8.2(b); (d) it is lifted; (e) the friction constraint is violated when the fingers travel straight to the goal state.

thickness was 1.8×10^{-3}, width 0.012, density 1.0×10^{3} and Young's Modulus 5×10^{7}. Under gravity, it deformed to be Figure 8.1(b). Two fingers were then placed at the position with triangles, which represent the friction cones.

We then ran the planning algorithm on the described shape. A path was planned by RRT from $(0, 0, 0, 0)^T$ to $(1.521, 1.162, 1.521, 1.164)^T \times 10^{-2}$. The path planned in the configuration space was mapped to the 2D plane to display. The initial path planned by RRT is shown in Figure 8.2(a), where the left (right) subplot corresponds to the trajectory of the left (right) finger. The path was post-processed to reduce the number of road points, as is shown in Figure 8.2(b). Note only the third intermediate road point of the path in Figure 8.2(a) remained.

We execute the planned path in Figure 8.2(b). At the starting point $(0, 0, 0, 0)^T$, the shape of the object is shown in Figure 8.1(b). At the lowest point $(-5.21, -1.84, 5.21, -2.23)^T \times 10^{-3}$, the object's shape is shown in Figure 8.1(c), where the pink short lines indicate the contact force at that point. At the final goal point, the shape is shown in Figure 8.1(d).

In Figure 8.1(e), the fingers traveled towards the goal point straightly from the starting point. At point $(4.56, 3.49, 4.56, 3.49)^T \times 10^{-3}$, which is 0.3 multiplying the goal state, the force was out of the friction cone, and thus result [13] in a failed grasp.



(a)

(b)

Figure 8.2: Planned paths: (a) planned by RRT, and (b) post-processed.

8.5 Discussion

In this chapter, a strategy to plan the trajectories of fingers to grasp deformable objects was introduced. After finding the goal state which minimizes the work done by fingers, the problem is reduced to a standard path planning problem, and can be solved using many path planning methods including RRT. This idea can be generalized to both 2D solid objects and 3D objects.

The finger work minimization usually results in a goal state that has the grasping force near or on the friction cone, and is thus a trade off with the stableness. The reduction of the number of road points may also reduce stableness because certain piece of straight paths may scrape the edge of configuration space with reduced flexibility. The idea of concentrate generated random points to "safer" area usually increases the number of random points needed for a successful planning, because the points near to a crowded area are less useful in terms of exploring unknown space.

CHAPTER 9. CONCLUSION

This dissertation studies grasping deformable objects using two fingers. The key idea is to specify desired finger displacements rather than forces, and to use them as constraints over an object so that its deformed shape can be computed. Specification of finger displacements over finger forces not only makes the strategy close to a real grasping scenario, but also helps stabilize the grasp. If constant forces are specified, grasping would act like an inverted pendulum and have no resistance to disturbance intended to cause rotation. If finger displacements are specified, however, disturbances up to certain magnitude can be resisted by friction at two contacts.

The strategy was applied on a variety of deformable objects, including hollow and solid 2D objects, and 3D ones. Except 2D hollow objects, fingertips usually make area contact with the object. With more than one point in contact, the contact configuration analysis is necessary. We calculate the configuration by tracking four events: stick, slip, contact establish and contact break. The configuration computation is mixed with deformation computation so that a grasp is computed in an iterative way.

In order to eliminate the influence of gravity on the shape of an object, we devised an algorithm to compute the original shape without gravity. The algorithm is an application of the fixed-point iteration with the hypothesized deformation as the iteration variable. The damping coefficient was introduced to widen the range of convergence.

The displacement-based strategy was generalized from involving only direct squeeze, to that the trajectories of the fingertips are curves. Such generalization not only broadens the graspable region since the initial frictional constraint is easier to be satisfied, but also allows multiple optimalities to be achieved. For example, the work done by the fingers, the ease of control, and the stability.

For future work, grasping deformable objects using more than two fingers is interesting. It is also interesting to try to predict the grasp without using precise modeling of the object. The manipulation of deformable objects may require extensive coordination of fingers.

APPENDIX A. MATCHING TWO POINTS CLOUDS WITH ROTATION, SCALING, AND TRANSLATION

An object was first drawn on the computer screen, and based on a printout of the drawing (with some enlargement), cut from a foam board. When we compared its deformations from simulation and experiment[1], the two sets of data points differed by scale, orientation, and translation. A homogeneous transformation needed to be applied to one set.

More formally, given two sets of points $P = \{\mathbf{p}_1, \cdots, \mathbf{p}_n\}$ and $Q = \{\mathbf{q}_1, \cdots, \mathbf{q}_n\}$ in the plane, where the point $\mathbf{p}_i$ corresponds to the point $\mathbf{q}_i$, for $i = 1, 2, \cdots, n$. We determine the scale s, rotation θ, and translation $(t_x, t_y)^T$ applied to Q that minimize the least-squares difference between two sets of points:

$$e = \sum_{i}^{n} (\mathbf{p}_i - M\mathbf{q}_i)^T (\mathbf{p}_i - M\mathbf{q}_i), \tag{A.1}$$

where M is the transformation matrix given by

$$M = \begin{pmatrix} s\cos\theta & -s\sin\theta & t_x \\ s\sin\theta & s\cos\theta & t_y \\ 0 & 0 & 1 \end{pmatrix}. \tag{A.2}$$

Least-squares matching of two sets of 2D or 3D points with known correspondeces under rigid motions was solved by Faugeras and Hebert (1986) and by Horn (1987) using quaternions, and by Schwartz and Sharir (1987) using a matrix-based method. Matching a set of points against a 3-D model up to rotation and translation (also with unknown point correspondences) could be effectively conducted by the iterative closest point algorithm (Besl and Mckay, 1992).

[1] The shape data in the experiment was obtained through image information.

In this algorithm, we also handle scaling since the data come in different metrics. We first obtain the four partial derivatives of e:

$$\begin{aligned}
\frac{\partial E}{\partial \theta} &= 2s\sum_{i}^{n}(\mathbf{p}_i - M\mathbf{q}_i)^T \begin{pmatrix} \sin\theta & -\cos\theta & 0 \\ \cos\theta & -\sin\theta & 0 \\ 0 & 0 & 0 \end{pmatrix} \mathbf{q}_i, \\
\frac{\partial E}{\partial s} &= -2\sum_{i}^{n}(\mathbf{p}_i - M\mathbf{q}_i)^T \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 \end{pmatrix} \mathbf{q}_i, \\
\frac{\partial E}{\partial t_x} &= -2\sum_{i}^{n}(\mathbf{p}_{ix} - s\cos\theta\mathbf{q}_{ix} + s\sin\theta\mathbf{q}_{iy} - t_x), \\
\frac{\partial E}{\partial t_y} &= -2\sum_{i}^{n}(\mathbf{p}_{iy} - s\sin\theta\mathbf{q}_{ix} - s\cos\theta\mathbf{q}_{iy} - t_y).
\end{aligned}$$

Write

$$(c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9, c_{10})^T = \frac{1}{n}\sum_{i}^{n}(\mathbf{p}_{ix}, \mathbf{p}_{iy}, \mathbf{q}_{ix}, \mathbf{q}_{iy}, \mathbf{p}_{ix}\mathbf{q}_{ix}, \mathbf{p}_{ix}\mathbf{q}_{iy}, \mathbf{p}_{iy}\mathbf{q}_{ix}, \mathbf{p}_{iy}\mathbf{q}_{iy}, \mathbf{q}_{ix}^2, \mathbf{q}_{iy}^2)^T.$$

Vanishing of the above four partial derivatives of e yields the following equations:

$$\begin{aligned}
c_3 s\cos\theta - c_4 s\sin\theta + t_x - c_1 &= 0, \quad (A.3) \\
c_3 s\sin\theta + c_4 s\cos\theta + t_y - c_2 &= 0, \quad (A.4) \\
c_3 t_x\sin\theta + c_4 t_y\sin\theta + c_4 t_x\cos\theta - c_3 t_y\cos\theta + (c_7 - c_6)\cos\theta - (c_8 + c_5)\sin\theta &= 0, \quad (A.5) \\
c_4 t_x\sin\theta - c_3 t_y\sin\theta - c_3 t_x\cos\theta - c_4 t_y\cos\theta + (c_8 + c_5)\cos\theta + (c_7 - c_6)\sin\theta - (c_9 + c_{10})s &= 0. \quad (A.6)
\end{aligned}$$

Assume $c_3^2 + c_4^2 \neq 0$. Multiply c_3 with (A.3) and c_4 with (A.4), and add them together, yielding

$$s\cos\theta = \frac{c_1c_3 + c_2c_4 - c_3t_x - c_4t_y}{c_3^2 + c_4^2}.$$

We also have, from $c_3 \times (A.4) - c_4 \times (A.3)$,

$$s\sin\theta = \frac{c_2c_3 - c_1c_4 + c_4t_x - c_3t_y}{c_3^2 + c_4^2};$$

from $\cos\theta \times (A.5) + \sin\theta \times (A.6)$,

$$s\sin\theta = \frac{c_4t_x - c_3t_y + c_7 - c_6}{c_9 + c_{10}}.$$

and lastly, by multiplying $\sin\theta$ on equation (A.5) and subtracting equation (A.6) multiplied by $\cos\theta$, we get

$$s\cos\theta = -\frac{c_3t_x + c_4t_y - c_8 - c_5}{c_9 + c_{10}}.$$

From the above four equations, we solve:

$$\begin{aligned}
t_x &= \frac{c_4(c_7 - c_6) - c_3(c_8 + c_5) + c_1(c_9 + c_{10})}{c_9 + c_{10} - (c_3^2 + c_4^2)} && \text{(A.7)}\\
t_y &= \frac{c_2(c_9 + c_{10}) - c_4(c_8 + c_5) - c_3(c_7 - c_6)}{c_9 + c_{10} - (c_3^2 + c_4^2)} && \text{(A.8)}\\
s &= \frac{\sqrt{[c_1c_3 + c_2c_4 - (c_8 + c_5)]^2 + [c_1c_4 - c_2c_3 + (c_7 - c_6)]^2}}{c_9 + c_{10} - (c_3^2 + c_4^2)} && \text{(A.9)}\\
\theta &= \text{atan2}(c_7 - c_6 + c_1c_4 - c_2c_3, c_8 + c_5 - (c_1c_3 + c_2c_4)). && \text{(A.10)}
\end{aligned}$$

Note that $c_9 + c_{10} > c_3^2 + c_4^2$ for $n \geq 2$.

In the special case $c = d = 0$, by solving (A.3) - (A.6),we can see that solutions (A.7) to (A.10) carries over, and is simplified to

$$\begin{aligned}
t_x &= a\\
t_y &= b\\
s &= \frac{\sqrt{(g - f)^2 + (h + e)^2}}{i + j}\\
\theta &= \text{atan2}(g - f, h + e).
\end{aligned}$$

APPENDIX B. MEASUREMENT OF THE PHYSICAL PARAMETERS

The Young's Modulus and Poisson's ratio of the rubber foam material are measured by pressing several rectangular samples of the material, and measuring relative physical factors, such as the pressing force and expanded width.

Multiple cuboids with the longest side between 5 to 10 cm were cut from the material. Its side lengths were recorded as h, a and b respectively in decreasing order. Weight neglected, each cuboid was placed on the surface of the finger phalange with the side h perpendicular to the surface. The top end of the cuboid was pressed with equally distributed force. The exerted force F was measured by the strain gauge sensor mounted in the joint of the finger tip. The changes in three dimensions of the object were measured as Δh, Δa and Δb respectively.

Assuming Δa and Δb are small compared with a and b, the Young's Modulus

$$E = \frac{Fh}{ab\Delta h}$$

Our results showed that $E \approx 5 \times 10^4$Pa.

With respect to poisson' ratio, let

$$\begin{aligned} p_1 &= -\log_{1+\frac{\Delta h}{h}}(1 - \frac{\Delta a}{a}) \\ p_2 &= -\log_{1+\frac{\Delta h}{h}}(1 - \frac{\Delta b}{b}). \end{aligned}$$

For isotropic materials, $p_1 = p_2$ and either one serves as Poisson ratio. The rubber foam material we used has Poisson's ratio 0.3.

APPENDIX C. FINGER KINEMATICS WITH MOUNTED TIPS

The Barret Hand was mounted with rounded fingertips to form a round shape for grasping the object. The kinematics is derived from the Barrett hand kinematics specified in the Barret hand user manual(section 9.5, pp.68 - 72). With the finger tips mounted, the center of the cylindrical tip is 62 mm away from the axis Z_{k3}(Figure 30, pp. 70). So override A_3 as 62 mm. Let the distance between two centers of the tips be d, and let the direction of X_{k3}(also the direction of the center line of the finger tip) be θ. When the hand's "spread" is set to 1575 (set to 1610 if the hand is mounted vertically, in order to balance the offset introduced by its own fingers' weight; the get button on the barrett hand control can tell you what parameter is really in effect), the $F1$ and $F2$ are in line with each other. Under such configuration, with the kinematics shown in the manual, we have:

$$\begin{aligned} d &= 2(A_w + A_1 + A_2\cos(\Phi_2 + \frac{P}{125}) + A_3\cos(\Phi_2 + \Phi_3 + \frac{4P}{375})) && \text{(C.1)} \\ \theta &= \Phi_2 + \Phi_3 + \frac{4P}{375}. && \text{(C.2)} \end{aligned}$$

where, P is the parameter of the joint 1 and 2 motor, and

$$\begin{aligned} A_w &= 25\text{mm} && \text{(C.3)} \\ A_1 &= 70\text{mm} && \text{(C.4)} \\ A_2 &= 70\text{mm} && \text{(C.5)} \\ A_3 &= 62\text{mm} && \text{(C.6)} \\ \Phi_2 &= 2.46^\circ && \text{(C.7)} \\ \Phi_3 &= 50^\circ && \text{(C.8)} \end{aligned}$$

Usually, $P \in (5000, 12000)$, the $d - \theta$ relationship is shown as in Fig. C.1. We can see they are almost linear to each other. So given d, we can use newton's method to look for θ.

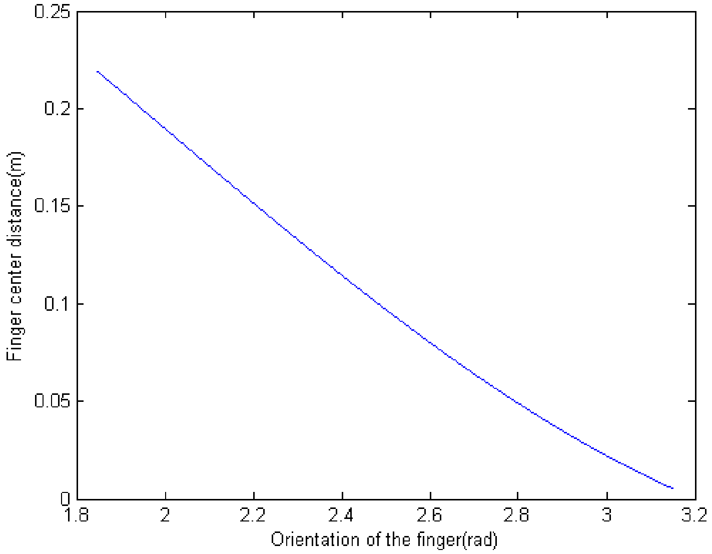


Figure C.1: distance - orientation relationship under normal range [5000, 12000]

However, if the d is given out of normal range, newton's method may not converge as in general the curve is highly non-linear.

BIBLIOGRAPHY

[1] Allgower, E. L. and Georg, K. (1997). Numerical path following. *Handbook of numerical analysis*, 5(3):207.

[2] Bicchi, A. and Kumar, V. (2000). Robotic grasping and contact: A review. In *ICRA*, pages 348–353. Citeseer.

[3] Boyd, S. P. and Wegbreit, B. (2007). Fast computation of optimal contact forces. *Robotics, IEEE Transactions on*, 23(6):1117–1132.

[4] Brost, R. C. and Goldberg, K. Y. (1994). A complete algorithm for synthesizing modular fixtures for polygonal parts. In *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pages 535–542. IEEE.

[5] Buss, M., Faybusovich, L., and Moore, J. B. (1998). Dikin-type algorithms for dextrous grasping force optimization. *The International Journal of Robotics Research*, 17(8):831–839.

[6] Choset, H. (2001). Coverage for robotics–a survey of recent results. *Annals of mathematics and artificial intelligence*, 31(1-4):113–126.

[7] Crandall, S. H., Lardner, T. J., Archer, R. R., Cook, N. H., and Dahl, N. C. (1978). An introduction to the mechanics of solids.

[8] Eisenberg, M. (1980). *Introduction to the mechanics of solids.* Addison-Wesley.

[9] Francavilla, A. and Zienkiewicz, O. (1975). A note on numerical computation of elastic contact problems. *International Journal for Numerical Methods in Engineering*, 9(4):913–924.

[10] Fung, Y. and Tong, P. (2001). Classical and computational solid mechan (advanced series in engineering science).

[11] Gallagher, R. H. (1975). Finite element analysis: fundamentals. *Prentice-Hall Civil Engineering and Engineering Mechanics Series, Englewood Cliffs: Prentice-Hall*, 1.

[12] Gopalakrishnan, K. and Goldberg, K. (2004). D-space and deform closure: a framework for holding deformable parts. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 1, pages 345–350. IEEE.

[13] Guo, F., Lin, H., and Jia, Y.-B. (2013). Squeeze grasping of deformable planar objects with segment contacts and stick/slip transitions. In *Robotics and Automation (ICRA), IEEE International Conference on*, pages 3736–3741. IEEE.

[14] Hirai, S., Tsuboi, T., and Wada, T. (2001). Robust grasping manipulation of deformable objects. In *Proc. IEEE Symposium on Assembly and Task Planning*, pages 411–416.

[15] Jia, Y.-B., Guo, F., and Lin, H. (2014). Grasping deformable planar objects: Squeeze, stick/slip analysis, and energy-based optimalities. *The International Journal of Robotics Research*, 33(6):866–897.

[16] Jia, Y.-B., Guo, F., and Tian, J. (2011). On two-finger grasping of deformable planar objects. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 5261–5266. IEEE.

[17] Jia, Y.-B., Lin, H., and Guo, F. (2013). Optimal two-finger squeezing of deformable objects. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 3514–3519. IEEE.

[18] Kerr, J. and Roth, B. (1986). Analysis of multifingered hands. *The International Journal of Robotics Research*, 4(4):3–17.

[19] Ladd, A. M. and Kavraki, L. E. (2004). Using motion planning for knot untangling. *The International Journal of Robotics Research*, 23(7-8):797–808.

[20] LaValle, S. M. (2006). *Planning algorithms.* Cambridge university press.

[21] Li, J.-W., Liu, H., and Cai, H.-G. (2003). On computing three-finger force-closure grasps of 2-d and 3-d objects. *Robotics and Automation, IEEE Transactions on*, 19(1):155–161.

[22] Li, Z. and Sastry, S. S. (1988). Task-oriented optimal grasping by multifingered robot hands. *Robotics and Automation, IEEE Journal of*, 4(1):32–44.

[23] Lin, H., Guo, F., Wang, F., and Jia, Y.-B. (2014). Picking up soft 3d objects by feeling when without tactile or force sensing.

[24] Luo, Q. and Xiao, J. (2006). Geometric properties of contacts involving a deformable object. In *Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2006 14th Symposium on*, pages 533–538. IEEE.

[25] Markcnscoff, X. and Yapadimitriou, C. H. (1987). Optimum grip of a polygon. *Dept. Comput. Sci., Stanford Univ., CA, Rep. STAN-CS-87-1153.*

[26] Markenscoff, X., Ni, L., and Papadimitriou, C. H. (1990). The geometry of grasping. *The International Journal of Robotics Research*, 9(1):61–74.

[27] Matsuno, T. and Fukuda, T. (2006). Manipulation of flexible rope using topological model based on sensor information. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 2638–2643. IEEE.

[28] Mirtich, B. and Canny, J. (1994). Easily computable optimum grasps in 2-d and 3-d. In *Robotics and Automation, IEEE International Conference on*, pages 739–747. IEEE.

[29] Mishra, B. (1995). Grasp metrics: Optimality and complexity. In *Proceedings of the workshop on Algorithmic foundations of robotics*, pages 137–165. AK Peters, Ltd.

[30] Mishra, B., Schwartz, J. T., and Sharir, M. (1987). On the existence and synthesis of multifinger positive grips. *Algorithmica*, 2(1-4):541–558.

[31] Moll, M. and Kavraki, E. (2006). Path planning for deformable linear objects. *Robotics, IEEE Transactions on*, 22(4):625–636.

[32] Nguyen, V.-D. (1988). Constructing force-closure grasps. *The International Journal of Robotics Research*, 7(3):3–16.

[33] Okamoto, N. and Nakazawa, M. (1979). Finite element incremental contact analysis with various frictional conditions. *International Journal for Numerical Methods in Engineering*, 14(3):337–357.

[34] Ponce, J., Stam, D., and Faverjon, B. (1993). On computing two-finger force-closure grasps of curved 2d objects. *The International Journal of Robotics Research*, 12(3):263–273.

[35] Ponce, J., Sullivan, S., Sudsang, A., Boissonnat, J.-D., and Merlet, J.-P. (1997). On computing four-finger equilibrium and force-closure grasps of polyhedral objects. *The International Journal of Robotics Research*, 16(1):11–35.

[36] Remde, A. and Henrich, D. (1999). Picking-up deformable linear objects with industrial robots.

[37] Reuleaux, F. (1963). *The kinematics of machinery: outlines of a theory of machines.* Dover.

[38] Ryaben'kii, V. S. and Tsynkov, S. V. (2006). *A theoretical introduction to numerical analysis.* CRC Press.

[39] Sadd, M. H. (2014). *Elasticity: theory, applications, and numerics.* Academic Press.

[40] Saha, M. and Isto, P. (2006). Motion planning for robotic manipulation of deformable linear objects. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2478–2484. IEEE.

[41] Sinha, P. R. and Abel, J. (1992). A contact stress model for multifingered grasps of rough objects. *Robotics and Automation, IEEE Transactions on*, 8(1):7–22.

[42] Strang, G. (2003). Introduction to linear algebra. *Cambridge Publication.*

[43] Tian, J. and Jia, Y.-B. (2010). Modeling deformations of general parametric shells grasped by a robot hand. *Robotics, IEEE Transactions on*, 26(5):837–852.

[44] van der Stappen, A. F., Wentink, C., and Overmars, M. H. (2000). Computing immobilizing grasps of polygonal parts. *The International Journal of Robotics Research*, 19(5):467–479.

[45] Wakamatsu, H., Arai, E., and Hirai, S. (2006). Knotting/unknotting manipulation of deformable linear objects. *The International Journal of Robotics Research*, 25(4):371–395.

[46] Wakamatsu, H. and Hirai, S. (2004). Static modeling of linear object deformation based on differential geometry. *The International Journal of Robotics Research*, 23(3):293–311.

[47] Wakamatsu, H., Hirai, S., and Iwata, K. (1996). Static analysis of deformable object grasping based on bounded force closure. In *Robotics and Automation, IEEE International Conference on*, volume 4, pages 3324–3329. IEEE.