IOWA STATE UNIVERSITY
Digital Repository

2014

Grasping of deformable 3D objects under gravity

Feifei Wang
Iowa State University

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

Part of the Computer Sciences Commons, and the Robotics Commons

Grasping of deformable 3D objects under gravity

by

Feifei Wang

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Computer Science

Program of Study Committee:

Yan-Bin Jia, Major Professor

Sourabh Bhattacharya

Guang Song

Iowa State University

Ames, Iowa

2014

DEDICATION

I would like to dedicate this thesis to my parents Jinggang Wang, Chunjuan Zhang and to my boyfriend Benjamin Kollasch 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 those who helped me with various aspects of conducting research and the writing of this thesis. First and foremost, Dr. Yan-Bin Jia for his guidance, patience and support throughout this research and the writing of this thesis. His insights and words of encouragement have often inspired me and renewed my hopes for completing my graduate education. I would like to thank my lab mates Feng Guo and Huan Lin as well. They gave me a lot of help during my thesis work. I would also like to thank my committee members for their efforts and contributions to this work: Dr. Sourabh Bhattacharya and Dr. Guang Song. Additionally, I would like to thank Dr. Hansen for his guidance from Mathematics department at Iowa State University. Finally, I would like to thank my friends and family for their loving guidance and financial assistance during the writing of this work.

ABSTRACT

The research area of robotic grasping of soft objects is difficult for multiple reasons: high cost computation of deformation, and the changes of the wrench space, contact areas, as well as point wise contact modes inside the areas.

This thesis describes modeling of grasped soft objects and recovering their gravity-free shapes. Chapter 1 will introduce the background and related work. Chapter 2 will focus on grasping and picking up soft objects. In the beginning, some results will be shortly described on resisting a third force applied on a 2D rigid body . The results will show that one of the forces must lie on an edge of the friction cone in order to achieve the optimized total normal force. Studying grasping of soft 2D objects will pave the way for picking up 3D objects. Even though there are some similar methods used in both situations, such as the four events of contact establishment, contact break, stick-to-slip and slip-to-stick, which can happen in the finger's squeezing process, there is still some significant difference between the 2D and 3D cases. This difference goes beyond just adding one more dimension, because the gravity effect has to be considered in 3D cases.

In Chapter 3, the focus will be on recovering the shape of a 3D object. Since we use its shape under gravity to construct the stiffness matrix and compute deformations, errors are observed in the experiment. The reason behind this, is that the stiffness matrix in conventional FEM practice already encodes the effect of gravity, which is considered again in the constitutive equation used for computing deformation. Then a numerical iteration method will be introduced to recover the gravity free shape.

Chapter1.   INTRODUCTION

Picking up of objects by robot fingers is inspired by the human behaviour. Some researchers have studied the theories and strategies on robot fingers grasping objects. Today, there are still many interesting topics in this area. Grasping a rigid body has been very well researched and understood, while there has been less research on the deformable object. However, in some aspects the study of the deformable object will be harder then that of rigid body. Finite element method and linear elasticity will be the most effective tools used to study the deformable objects. Let's first review the work that has been done in related research areas.

1.1   Related work on rigid body grasping

Rigid body grasping has been studied by many people in the late 20th century. B. Mishra, J. T. Schwartz, and M. Sharir (1987) have contributed on the definition of form closure. They gave sufficient and necessary conditions on the bounds on the number of contact points for 2D and 3D objects to have a form closure. R. C. Brost and K. Y. Goldberg (1994), A. F. Van der Stappen, C. Wentink , and M. H. Overmars (2000) have come up with the algorithms to compute all of the form closure grasps for polygons. E. Rimon and A. Blake (1999) have studied on caging the object with frictionless contact such that the object cannot escape. These study will be very helpful even in the deformable object research areas. For example, the forces we apply on the object still have to satisfy the force closure requirement to keep the object not having any rigid body movement. In 2D case, V-D. Nguyen (1998) efficiently computed the force-closure grasps for polygons and piecewise smooth shapes. J. C. Trinkle (1992) has formulated the force-closure test as a linear program. J. Kerr and B. Roth (1986) put up some measure standards for the quality of a grasp. It was to select grasping forces that were furthest

from violating any constraint.

1.2   Related work on deformable grasping

Compared to the rigid body grasping, there was not much work done on grasping deformable objects. H. Wakamatsu and S. Hirai (2004) have studied manipulation of flexible linear objects. This has been a very active area. Applications can be related to knotting and unknotting, pick up objects and path planning for a robot.

P. R. Sinha and J. M. Abel (1992) came up with a model for deformation, in order to predict contact forces without considering global deformation or grasp computation. Q. Luo and J. Xiao (2006) has studied from the geometric properties to improve simulation accuracy and efficiency. J. Tian and Y-B. Jia (2010) investigated point contacts grasping deformable modeling of shell-like objects. F. Guo, H. Lin, Y-B. Jia (2013) have done the work on squeeze grasping of deformable planar objects by specify the displacements rather than forces. Optimal strategies for resisting a third force by two fingers squeezing deformable objects was studied by Y-B. Jia, F. Guo, H. Lin (2013).

Our recent work investigated the strategies of picking up soft 3D objects with two fingers. In order to use more accurate model for stiffness matrix, recovering the gravity free shape of soft objects has also been studied.

Chapter2.   PICKING UP SOFT OBJECTS

The rationale for studying how to pick up soft objects is inspired by the human behaviour. In our daily life, when we grasp an object, we always try to figure out the comfortable direction and try to save some effort by our tactile feelings. Therefore, it is an interesting topic for us to perform some theoretical analysis. Later, we can apply the result to a robot hand in grasping.

2.1   Introduction

In this chapter, some simple strategies of picking up soft objects will be introduced. At the beginning, the discussion is the simplest case: grasping the 2D planar rigid body. Next, the focus will be on picking up the 2D soft objects. The 2D soft object uses line segment contact which makes it more complicated than the rigid body's simple point contact. Next, discussed is the different events which happened during the picking up process. The problem focused on at the end will be picking up soft 3D objects with two fingers. It is not simply one more dimension added on to the previous problem, but there are more technical issues coming up.

2.2   Grasping 2D objects

Most of the objects seen in daily life are 3D objects. However, it would be convenient to study the simple 2D object first. In order to simplify to 2D, the effect of gravity and also the third dimension of the object were ignored. In other words, those objects we are studying here have no height, and they are all planar objects.

2.2.1 Grasping while resisting a third force

For a rigid body, point contact is the only situation studied. The location of the two fingers and also the location of the third force being applied is known. The problem is to discern which direction and how much force should be applied on the two fingers such that their summations of the normal forces will achieve the minimum.

This is a typical optimization problem. To start looking at this problem is by setting up the equations from it. By simple knowledge from physics, it is known that these three forces must satisfy both the force balance law and moment balance law if the two fingers are to resist the third force successfully. In other words, the whole system has to be in the balance mode. Also, to achieve the force closure, the contact forces have to lie inside the corresponding friction cones. Suppose that the positions of the two fingers will be at $\boldsymbol{p}_1 = (p_{1x}, p_{1y})$ and $\boldsymbol{p}_2 = (p_{2x}, p_{2y})$. The position of the third disturbing force is at $\boldsymbol{p}_3 = (p_{3x}, p_{3y})$. Here, since we are studying the planar objects, the locations of the points on them will be represented by the 2D coordinates on a plane. The first component means the amount in x-direction. Correspondingly, the second component in the coordinates relates to the y-direction. Remember we have already specified the third force. Without lost of generality, we can always assume that the direction of the force is horizontally from right to the left, which is following the direction of the unit vector $(-1, 0)$. Then the third force would be $\boldsymbol{f}_3 = (f_{3x}, f_{3y}) = (f_{3x}, 0), f_{3x} < 0$, as well as the assumption that the normal vectors of the object at the place where the two robot fingers apply forces would be $\boldsymbol{n}_1 = (n_{1x}, n_{1y})$ and $\boldsymbol{n}_2 = (n_{2x}, n_{2y})$. The normal force of the two fingers would therefore be $\boldsymbol{f}_{1n} = \boldsymbol{f}_1 \cdot \boldsymbol{n}_1 = f_{1x}n_{1x} + f_{1y}n_{1y}$, and the same idea we can get $\boldsymbol{f}_{2n} = \boldsymbol{f}_2 \cdot \boldsymbol{n}_2 = f_{2x}n_{2x} + f_{2y}n_{2y}$. By the force balance law, we will have the following set of equations:

$$f_{1x} + f_{2x} + f_{3x} = 0 \tag{2.1}$$

$$f_{1y} + f_{2y} + f_{3y} = 0 \tag{2.2}$$

Since the total momentum should be zero, we get:

$$(f_{3x}p_{3y} - f_{3y}p_{3x}) + (f_{1x}p_{1y} - f_{1y}p_{1x}) + (f_{2x}p_{2y} - f_{2y}p_{2x}) = 0 \tag{2.3}$$

Then, from the first two equations, f_{2x} and f_{2y} can be written as the linear combinations of f_{1x} and f_{1y}. Thus, f_{1n} and f_{2n} can also be written as the linear combinations of f_{1x} and f_{1y}. Plug in to the third equation, then obtained is every unknown variables to be represented in terms of the variable f_{1x}. Apparently this is true, since originally there are four variables f_{1x}, f_{2x}, f_{1y} and f_{2y}, and in total three equations, so it's clear that we can do operations to simplify them until every variable is represented only by f_{1x}. For example, we can get

$$f_{1y} = \frac{f_{1x}(p_{1y} - p_{2y}) + f_{3y}(p_{2x} - p_{3x}) + f_{3x}(p_{3y} - p_{2y})}{p_{1x} - p_{2x}} \tag{2.4}$$

$$f_{1n} = \frac{f_{1x}((p_{1x} - p_{2x})n_{1x} + (p_{1y} - p_{2y})n_{1y}) + f_{3y}(p_{2x} - p_{3x}) + f_{3x}(p_{3y} - p_{2y})}{p_{1x} - p_{2x}} \tag{2.5}$$

Now that it is understood not only they can be represented by f_{1x}, but also in linear term of f_{1x}. Remember the goal is to come up with the best way to resist a third disturbing force. Since it's all rigid objects, the work will be zero. So the superior way is to minimize the magnitude of total normal forces. Therefore, our problem can be summarized and formulated as an optimization problem:

$$\text{Minimize} \qquad f_{1n}^2 + f_{2n}^2 \tag{2.6}$$

$$\text{Subject to :} \ (\frac{\boldsymbol{f}_1 \cdot \boldsymbol{n}_1}{\|\boldsymbol{f}_1\|\|\boldsymbol{n}_1\|})^2 \ \geq \ (\cos\tilde{\theta})^2 \tag{2.7}$$

$$(\frac{\boldsymbol{f}_2 \cdot \boldsymbol{n}_2}{\|\boldsymbol{f}_2\|\|\boldsymbol{n}_2\|})^2 \ \geq \ (\cos\tilde{\theta})^2 \tag{2.8}$$

$$f_{1n} \ > \ 0 \tag{2.9}$$

$$f_{2n} \ > \ 0 \tag{2.10}$$

Here, the last two constraints are applied because we don't discuss the the outward force since non of the fingers have an adhesive. Also we don't discuss the extreme situation, such like the friction coefficient is infinity big, because we don't consider the situation when at least one of the normal forces is equal to zero. $\tilde{\theta}$ is the angle of half of the friction. Meanwhile, if the friction coefficient of the material is μ, then they have the relation that $\mu = \arctan\tilde{\theta}$. Now we formulate our problem to be a quadratic minimization problem. Since the variable here is f_{1x},

and the other terms are in the linear form, therefore it is a quadratic optimization problem. Before any computing is done, a theorem was found after a great deal of experiments.

Theorem 1 *To achieve the minimum value, at least one of the finger forces should lie on an edge of the corresponding cone of friction, which also means at least one of the first two inequality constraints has to be an equality.*

Proof By contradiction. Consider the easiest case first when the object is symmetric and the normal direction is the same as y direction. Suppose when we achieve the minimum value in terms of f_{1n}, both of the inequality constraints are taking the strictly greater than sign. Thus we can pick an ϵ small enough, such that when we change f_{1n} to be $f_{1n} - \epsilon$, all the constraints are still satisfied. Due to the linearity, f_{2n} is changed to be $f_{2n} - \epsilon$. We can always achieve this by rotating the forces a little to keep the magnitude of the force, but the force on the normal direction becomes smaller. However, in this case, the objective function value changes from $f_{1n}^2 + f_{2n}^2$ to be $(f_{1n} - \epsilon)^2 + (f_{2n} - \epsilon)^2$, which is a smaller value and contradicts to the fact that the previous achieves the optimum. □

The simulation result also support our proof. In this thesis we use the metric system for every unit. For example, meters, kilograms, Newton and so on. Here we consider a simple example of a disc shape object with radius $r = 2$ and friction coefficient 0.8. The positions of the two fingers and the third force are at $\boldsymbol{p}1 = (-1, -\sqrt{3})$ and $\boldsymbol{p}_2 = (0, 2)$ and $\boldsymbol{p}_3 = (2, 0)$. The third force is $\boldsymbol{f}_3 = (-2, 0)$. The normals where the fingers are placed are $\boldsymbol{n}_1 = (1/2, \sqrt{3}/2)$ and $\boldsymbol{n}_2 = (0, -1)$. After using the Wolfram Mathematica 8.0 function NMinimize, the result that $f_{1x} = 1.30468$ is attained and the minimized magnitude of the total normal forces is 2.72956. Figure 2.1 illustrates this result.

The black arrow presents the forces of the two fingers and also the third disturbing force. From the figure we can see it is true that at least one of the finger force must lie on the friction cone.
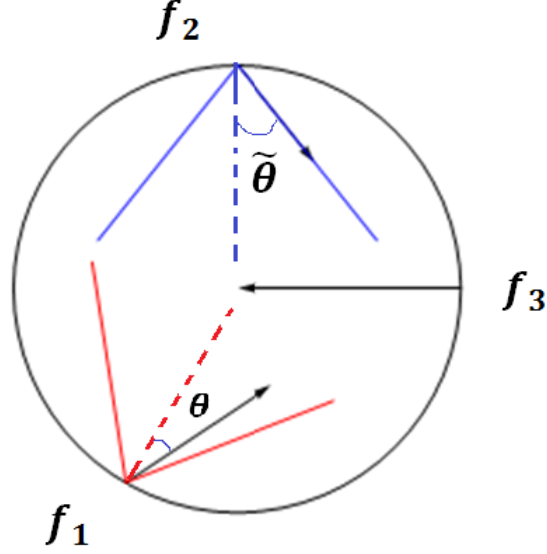
Figure 2.1 Optimal two fingers normal forces to an external disturbance (horizontal black arrow). The blue and red lines are the edges of the friction cones, and the other two black arrows represent the forces applied by the two fingers.

2.2.2 Grasping soft 2D objects

Grasping 2D soft objects is more challenging than grasping the rigid object, due to the changes of wrench space under deformation. With the contact area growing, the contact points contained in the contact line segment will switch their status between slip and stick. In this part, we will introduce two standards of squeeze grasping soft 2D objects by two fingers, which are pure squeeze grasp and stable squeeze grasp. Then we will discuss the four different events which happened during the process of grasping the objects. The paper F. Guo, H. Lin, Y-B. Jia (2013) also introduced simulations and experiments to support the assumptions.

To start with, we can consider the displacement field $\boldsymbol{\delta} = (u(x,y), v(x,y))^T$. Under it, every point $(x, y)^T$ inside the 2D planar object moves to $(x + u, y + v)^T$. The strain energy density is given as

$$U_0 = \frac{h}{2} \iint_S \left(\frac{E}{1 - \nu^2} (\epsilon_x^2 + 2\nu\epsilon_x\epsilon_y + \epsilon_y^2) + \frac{E}{2(1 + \nu)\gamma_{xy}^2} \right) dx\, dy. \tag{2.11}$$

where

$$\epsilon_x = \partial u/\partial x$$

$$\epsilon_y = \partial v/\partial y$$

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

are strains, and h is the very small thickness of the object. The figure 2.2 shows it. Denote
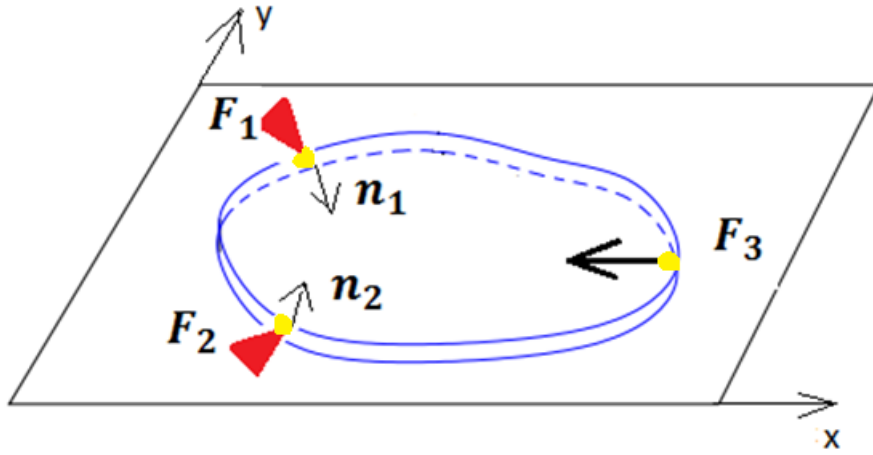
Figure 2.2 Example of thin object

the force at point (x, y) to be $\boldsymbol{f}(x, y)$. Then the total potential energy of the system is

$$\Pi = U_0 - \sum_{(x,y)^T} \boldsymbol{\delta}(x, y)^T \boldsymbol{f}(x, y). \tag{2.12}$$

Next, I will state several important theorems that we will need in the future.

Theorem 2 *Under linear elasticity, any displacement field* $\boldsymbol{\delta} = (u(x, y), v(x, y))^T$ *that yields zero strain energy is linearly spanned by the following three fields:* $\boldsymbol{t}_x = (1, 0)^T$, $\boldsymbol{t}_y = (0, 1)^T$ *and* $\boldsymbol{r} = (-y, x)^T$.

The proof can be found in Y-B. Jia, F. Guo, H. Lin (2013). Remove $\boldsymbol{\delta}$ from its projections onto $\boldsymbol{t}_x$, $\boldsymbol{t}_y$ and $\boldsymbol{r}$ will give us

$$\boldsymbol{\delta}_\perp = \boldsymbol{\delta} - \frac{\langle \boldsymbol{\delta}, \boldsymbol{t}_x \rangle}{\langle \boldsymbol{t}_x, \boldsymbol{t}_x \rangle} \boldsymbol{t}_x - \frac{\langle \boldsymbol{\delta}, \boldsymbol{t}_y \rangle}{\langle \boldsymbol{t}_y, \boldsymbol{t}_y \rangle} \boldsymbol{t}_y - \frac{\langle \boldsymbol{\delta}, \boldsymbol{r} \rangle}{\langle \boldsymbol{r}, \boldsymbol{r} \rangle} \boldsymbol{r}. \tag{2.13}$$

Definition 1 *A pure deformation field $\boldsymbol{\delta}_\perp$ is orthogonal to $\boldsymbol{t}_x$, $\boldsymbol{t}_y$ and $\boldsymbol{r}$ and contains no rigid body movement. If a squeeze generate pure deformation of the object, we call it is a pure squeeze.*

Definition 2 *If a squeeze minimizes the potential energy of the system among all squeezes of the same magnitude, we call it is a stable squeeze.*

During the squeeze of the two fingers, every contact points can be detected by the definition of the following four different events. (1) Event A: new contact. This is the case when we detect a new node on the boundary comes in to contact with one of the fingers. Add this new node to stick set T or the slip set P according to the result small squeeze test of if it will stay inside the friction cone(stick) or else (slip). (2) Event B: contact break. If the magnitude of the force of a point is zero, which is $\|\boldsymbol{f}_k\| = 0$, then this point will no longer be a contact point. In this case, the contact is broken for this point and we remove it from either the stick set T or the slip set P. (3) Event C: stick to slip. If the contact force is rotating out of the friction cone, then the point starts to slip. We remove this point from set T and add it to set P. (4) Event D: slip to stick. If the slip point changes its direction of slip, we remove it from P and add it to T.

These events will also be very helpful on analyzing the grasping of object on the 3D case, which we will focus more in this chapter.

2.3   Picking up soft 3D objects with two fingers

In this section, we will discuss a simple strategy of picking up a 3D soft object by two fingers. The idea is inspired by human being behaviour. In our daily life, when we try to pick up an object, we usually squeeze a little bit, and feel if we can lift the object up. If the squeeze depth is not big enough, we may need to squeeze a little bit more until it is enough to hold the weight and finally lift up the object. The paper written by H. Lin, F. Guo, F. Wang, Y-B. Jia (2014) described the details about the topic.

2.3.1   Theoretical background

In fact, the 2D case and 3D case have similar aspects in analysis with each other. However, there also exists a huge difference between them. To study the 3D case, it is not simply to add one more dimension. Due to the third dimension, we have to consider the effect of the gravity on the shape of the object, which was ignored in the 2D planar object case. Here is a picture 2.3 [1] simulates picking up a soft object.   We will begin with a brief review of linear elasticity
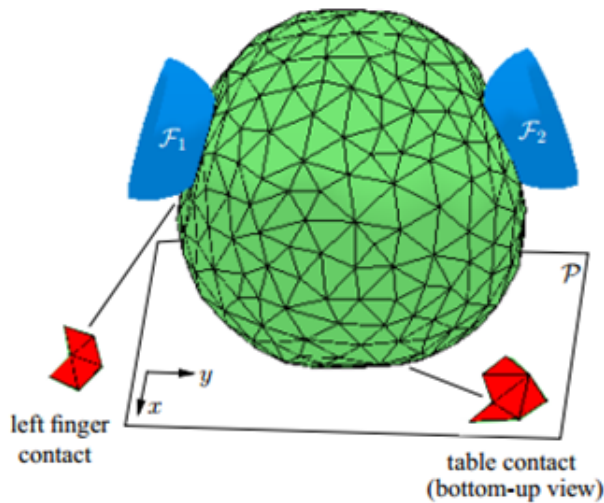

Figure 2.3   Lifting up a 3D object

applied in 3D situation. In the previous section, I have presented the terminologies in 2D case. Consider ever point in the 3D case moves from $(x, y, z)^T$ to $(x + u, y + v, z + w)^T$. Similarly,

---

[1]Courtesy of Huan Lin.

we have the normal strains and the shear strains denoted as follows:

$$\epsilon_x = \partial u / \partial x$$

$$\epsilon_y = \partial v / \partial y$$

$$\epsilon_z = \partial w / \partial z$$

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

$$\gamma_{yz} = \partial v / \partial z + \partial w / \partial y$$

$$\gamma_{zx} = \partial w / \partial x + \partial u / \partial z$$

The strain energy in the 3D case is

$$U_0 = \frac{E}{2(1+\nu)} \int_V \left[\frac{\nu}{1-2\nu} (\epsilon_x + \epsilon_y + \epsilon_z)^2 + (\epsilon_x^2 + \epsilon_y^2 + \epsilon_z^2) + \frac{1}{2}(\gamma_{xy}^2 + \gamma_{yz}^2 + \gamma_{zx}^2) \right] dV. \quad (2.14)$$

Compared to the 2D case, we have a similar theorem:

Theorem 3 *Under linear elasticity, any displacement field $\boldsymbol{\delta} = (u(x,y,z), v(x,y,z), w(x,y,z))^T$ that yields zero strain energy is linearly spanned by the following three fields: $(1,0,0)^T$, $(0,1,0)^T$, $(0,0,1)^T$, $(0,-z,y)^T$, $(z,0,x)^T$, and $(-y,x,0)^T$.*

The potential energy in 3D case is

$$\Pi = U - W = \frac{1}{2}\boldsymbol{\Delta}^T K \boldsymbol{\Delta} - \boldsymbol{\Delta}^T (\boldsymbol{F} + \boldsymbol{G}). \quad (2.15)$$

It reaches the minimum value with the constitutive equation

$$K\boldsymbol{\Delta} = \boldsymbol{F} + \boldsymbol{G}. \quad (2.16)$$

After the preview of the math and physics background, let now move back to focus on the strategy of picking up the object. The idea is not hard to understand. First first compute the object's initial configuration of lying on the table by iterative procedure. Then during the process of squeezing the object by two fingers, we will do the lift test simultaneously. If the object finally past the lift test, then it means we can successfully lift it up. Otherwise, if the squeeze depth is to big such that it is over the range of the linear elasticity, then the lift test failed.

Let us start by constructing the initial configuration of the soft object lying on a plane using algorithm 1. Here, we denote the indices of the nodes that are in contact with the two fingers and the plane to be $\mathbb{I}$, $\mathbb{J}$, and $\mathbb{K}$. Denote $\mathbb{P}$ to be the set that collects the indices of sliding nodes.

Algorithm 1 Compute initial resting configuration.

1: $\mathbb{P} \to \emptyset$.
2: Compute the change displacement vector $\boldsymbol{\Delta}'$ from the displacement vector $\boldsymbol{\Delta}$. Correspondingly compute the change force $\boldsymbol{F}'$ from the contact force $\boldsymbol{F}$.
3: $\boldsymbol{\Delta} \to \boldsymbol{\Delta} + \boldsymbol{\Delta}'$ and $\boldsymbol{F} \to \boldsymbol{F} + \boldsymbol{F}'$.
4: If there is no new contact node and $\boldsymbol{f}_k$ lies in side the friction cone for every $k \in \mathbb{K}$, then STOP.
5: Otherwise deal with the new contact.
6: Find sliding nodes and add the corresponding indices into $\mathbb{P}$.
7: Recompute the sliding nodes' displacements.

Briefly, the algorithm can be summarized as first evaluate the displacement without considering sliding. If there is some new contact node, add the furthermost displacement below plane and scale down displacements, and identify sliding nodes and recompute displacements, go back to the iterations. If there is no new contact points and no sliding point, that means we have successfully computes the initial configuration and we can stop.

After we finish computing the initial resting configuration, the two fingers will be translating in a constant direction step by step. During the process, we will do the lift test to see if the forces are big enough to lift the object. The lift test is about the liftable weight w during the process of lifting. Liftable weight is the weight that the finger fources can hold until one of the fingers almost starts to slide. Originally we suppose the liftable weight is zero. As the squeeze step increases, the liftable weight also goes up. Once the liftable weight reaches the object's weight, the lift test is passed. The algorithm is as follows.

There will also be four events happen during the process. The contact sets $\mathbb{I}$, $\mathbb{J}$, and $\mathbb{K}$ will not change until the next event happen. The definition of these four events are exactly the same as previous section: (A) contact establishment, (B) contact break, (C) stick to slip and (D) slip to stick.

During the process, it is important for us to track the movements of all the sliding nodes.

Algorithm 2 Pick up 3D deformable object by two fingers.

1: Input tetrahedron mesh of the object, initial table contact triangle with vertices $\boldsymbol{p}_q, \boldsymbol{p}_r, \boldsymbol{p}_s$. Finger contacts $\boldsymbol{p}_i, \boldsymbol{p}_j$, squeeze $(\boldsymbol{d}_1, \boldsymbol{d}_2)$.
2: Check if $\boldsymbol{p}_i, \boldsymbol{p}_j$ with the initial triangle forms a force closure. If not, return FAIL. Otherwise, continue to the next step.
3: Use algorithm 1 to compute the initial resting configuration.
4: Squeeze the object by translating the two fingers with the forces $\boldsymbol{F}_1, \boldsymbol{F}_2$.
5: Do lift test during the squeezing process.
6: If so, stop squeezing object and change to lifting the object. Record the squeeze depth.
7: If enough squeeze depth has been applied but still cannot be picked up, return FAIL. Else go back to step 4.

Remember their indices are collected in the set $\mathbb{P}$. The sliding can happen in the contact plane or the semi spherical fingers. Thus, we will have two different equations to solve the sliding variable d_k.

First, for the sliding points happen in the contact plane, the force $\boldsymbol{f}_k$, $k \in \mathbb{P}$ has to lie on the edge of the friction cone. What's more, each $\boldsymbol{f}_k$ is a linear function of all the variables d_l, $l \in \mathbb{P}$. Suppose the friction coefficient between the object and the table is μ_p, then we will have the system of $|\mathbb{P}|$ quadratic equations:

$$(1 + \mu_p^2)(\boldsymbol{f}_k \cdot \boldsymbol{z})^2 = \boldsymbol{f}_k \cdot \boldsymbol{f}_k. \tag{2.17}$$

Here the $\boldsymbol{z}$ is the normal of the plane, which is $(0, 0, 1)$. Then suppose the friction coefficient between the finger and the object is μ_F. Assume $\tilde{\boldsymbol{p}}_k$ is the position before slip, and reach some point $\boldsymbol{q}_k$. The contact force has the tangential component as

$$\boldsymbol{f}_{i\perp} = \boldsymbol{f}_k - (\boldsymbol{f}_k \cdot \boldsymbol{n}_k)\boldsymbol{n}_k. \tag{2.18}$$

The sliding will be along the opposite direction of $\boldsymbol{f}_{k\perp}$. Denote the sliding angle along the great circle of the finger to be θ_k, and $c_k = \cos\theta_k$, $s_k = \sin\theta_k$. The equations will be:

$$(1 + \mu_F^2)(\boldsymbol{f}_k \cdot \boldsymbol{n}_k)^2 = \boldsymbol{f}_k \cdot \boldsymbol{f}_k, \tag{2.19}$$

$$c_k^2 + s_k^2 = 1. \tag{2.20}$$

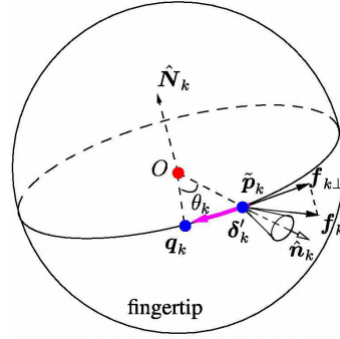Figure 2.4 helps us understanding the situation happens on a finger.

Figure 2.4   Sliding of a node on the hemispherical finger

After every $\boldsymbol{f}_k$ is written as linear combinations of the variables d_l, a large system of quadratic equations is attained. To store those quadratic equations, pull out the coefficients to be a coefficient matrix. Since it is a very complicated system, we can only solve it by numerical methods. Here, the homotopy continuation method is used to solve it. This method is also used in the function "RootFinding" in a mathematics software called "Maple". Homotopy method is used to find the isolated root of the system of polynomial equations with the same number of variables in each of the equations.This method is a continuous map from a set of constructed known solutions to the solutions of the input system. The map function is constructed as

$$\boldsymbol{H}(\boldsymbol{x},t) = \{(1-(1-t)^2)p_1(x_1,\ldots,x_n)-(1-t)^2q_1(x_1),\ldots,(1-(1-t)^2)p_n(x_1,\ldots,x_n)-(1-t)^2q_n(x_n)\}.$$

$$(2.21)$$

Here, $q_1(x_1),\ldots,q_n(x_n)$ are a sequence of generated random polynomials, and they are called the generated starting system. It is not hard for us to notice that when $t = 0$, the solution of $\boldsymbol{H}(\boldsymbol{x},0) = 0$ will be the solution of the generated starting system. When $t = 1$, $\boldsymbol{H}(\boldsymbol{x},1) = 0$ gives us the solution of the input system. We start from the following way.

Here, we need further explanation on step 4. Since $\boldsymbol{H}(\boldsymbol{x},t) = 0$, we can do the partial differentiation with it and get:

$$\frac{\partial \boldsymbol{H}}{\partial \boldsymbol{x}} \cdot \frac{d\boldsymbol{x}}{dt} + \frac{d\boldsymbol{H}}{dt} = 0. \tag{2.22}$$

Then we can get:

$$\frac{d\boldsymbol{x}}{dt} = -(\frac{\partial \boldsymbol{H}}{\partial \boldsymbol{x}})^{-1}\frac{d\boldsymbol{H}}{dt}. \tag{2.23}$$

Algorithm 3 Homotopy method.

1: Input the system of polynomials $p_1(x_1, x_2, \ldots, x_n), \ldots, p_n(x_1, x_2, \ldots, x_n)$.
2: Generate a start system $q_1(x_1), \ldots, q_n(x_n)$.
3: Construct the map function as equation (2.21).
4: Differentiate the system and turn it into an ODE problem. Time t changes from 0 to 1. The path will be followed by discrete steps.
5: At each step, using an ODE solution method as a predictor, then use the Newton's numerical method to polish the root to ensure the computed root stay on the path.
6: iterate enough steps and output the result.

Then change from $\boldsymbol{x}(0) \to \boldsymbol{x}(h)$, we can use the following to compute:

$$\boldsymbol{x}(h) = \boldsymbol{x}(0) + h\frac{d\boldsymbol{x}}{dt}(0). \tag{2.24}$$

The general iterations will be,

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + h\frac{d\boldsymbol{x}}{dt}(\boldsymbol{x}_n). \tag{2.25}$$

After each iteration, polishing of the values is needed by using Newton's method,

$$\boldsymbol{x}_{n+1}^* = \boldsymbol{x}_{n+1} - (\frac{\partial \boldsymbol{H}}{\partial \boldsymbol{x}}(\boldsymbol{x}_{n+1}))^{-1}\boldsymbol{H}(\boldsymbol{x}_{n+1}). \tag{2.26}$$

We choose homotopy method to solve the roots instead of Newton's reason. The reason is that homotopy method is a global convergent method. However, Newton's method only converge when you star with a very close initial guess to the real solution.

2.3.2 Simulations and experiments

We have done the simulations and experiments on four different objects: tomato, orange, steam bun and toy football. Their data is listed as below in Table(2.1).

The simulations and experiments figures is in Figure (2.5).

It would be an interesting topic for us to study about how the softness of an object affects the grasping and lifting up of this object. Young's modulus affects the softness of an object. If the number is bigger, it means the object's material is harder. When Young's modulus goes to infinity, that means the object is close to a rigid body.

To figure out how Young's modulus affect the grasping and lifting, I have conducted several simulation examples. I took down the data of the total work the finger did during the process

Table 2.1   Information of the objects

	tomato	orange	steam bum	football
vertices	490	600	658	875
surface facets	498	564	646	782
tetrahedra	2129	2692	2941	4058
weight w	1.246	1.868	0.467	0.489
Young's modulus E	1.0×10^5	1.4×10^5	2.0×10^3	6.0×10^3
Poisson's ratio ν	0.4	0.4	0.2	0.3
friction coefficient μ	0.32	0.34	0.4	0.5

of squeezing until the object is liftable. The study object is in the same shape as the football. There are in total 332 grid points on the surface of the football. The Poisson's ratios is 0.3, density of the object is 2×10^2. Here, we fix all the other parameters, and let the Young's modulus change from 1.0×10^3 to 6.0×10^4, part of the data of total work is shown in Table(2.2).

Table 2.2   Young's modulus with total work

E	total work
1.0×10^3	0.000862824
4.0×10^3	0.00150976
9.0×10^3	0.00342252
3.0×10^4	0.00475993
6.0×10^4	0.00770573

The plot data of the relationship between Young's modulus and fingers total work is in Figure(2.6).

From the data, we can preliminary get the result that to pick up a harder objects seems to cost more work. However, this topic still needs further understanding since we don't have the theoretical result currently. The little discontinuous jump in the picture is caused by the discretization of the object, and also the squeeze depth.

2.4   Future work

In the future, the goal will be to get theoretical results and further understandings of the relationship between the softness of the object and also the total work need to do in order to pick up the object, also the study on optimizing the finger placements and the squeezing
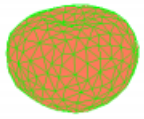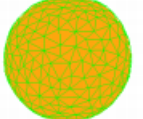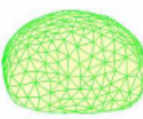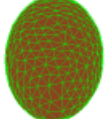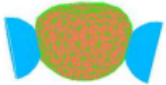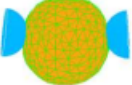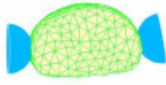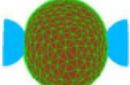
(11, 0.018, 1.86)	(15, 0.025, 2.75)	(8, 0.010, 2.41)	(10, 0.011, 3.12)

Figure 2.5   Picking up objects successfully. An entry in the last row lists the number of contact nodes, the squeeze depth, and the running time

directions.
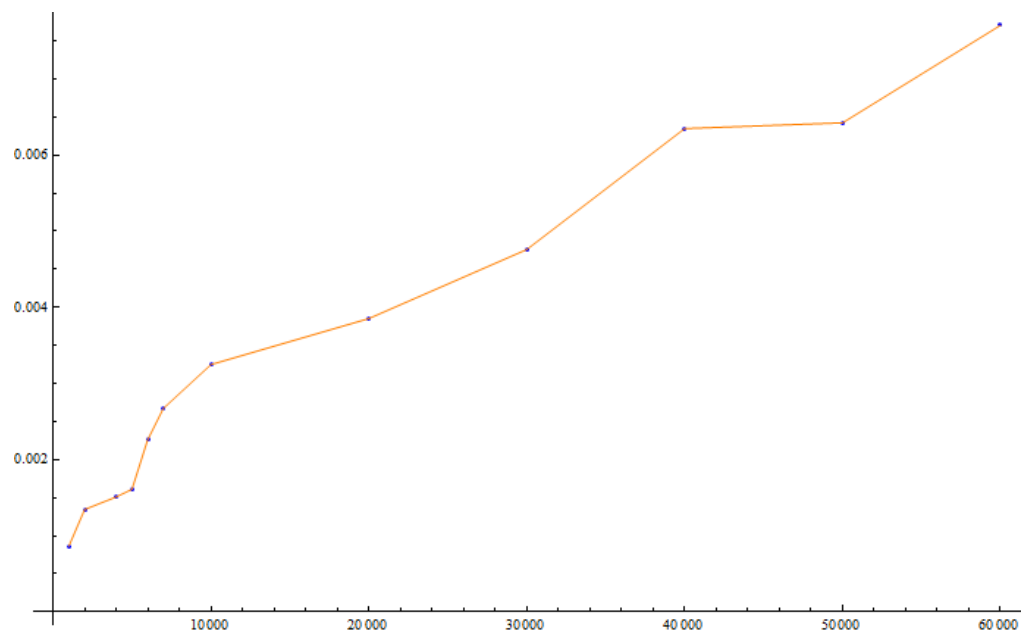
Figure 2.6   Relations between the Young's modulus and total work

Chapter3.   RECOVERING THE GRAVITY-FREE SHAPE OF A 3D DEFORMABLE OBJECT

The stiffness matrix of an object is based on the shape of it, which helps describe the displacement under certain forces and study the deformation of the object. Basically, a stiffness matrix reflects the geometric and material property of an object. When we construct the stiffness matrix based on the finite element method, we already neglect the fact that the object is already under gravity force and has some deformation based on the gravitational force. However, to get the accurate model, we require the exact shape of the object in a gravity free situation. Hence, it is interesting for us in gravity-free situation to explore and know the information of the original shape of an object.

3.1   Introduction

Everywhere people's daily life is affected by gravity. For example, when you place an animal organ on a plate, it will change the shape due to the different positions and faces you put it in. The same thing will happen to a tomato. When you put it upside down, it looks very different from when you put it face up. However strictly speaking, a solid deformable object setting on a plane will change shapes due to gravity, this is not equivalent to any kind of rigid body transformation. To pick up a soft solid object, not only does the force have to balance the gravity, but also the change of geometry and contact areas have to be considered.

The gravitational force acts throughout the solid relative to its volume. To obtain the deformed shape we need to get the minimized potential energy, which equals the body's strain energy minus the external forces related work. The finite element method gives us instruction to discretize the object body into meshes. Then based on the corresponding forces applied on

the body mesh to construct the stiffness matrix, displacement vector and the force vector can be set up as a linear system of equations. This solves the unknown displacement variables. Here, the equation is called the constitutive equation, which is a very classical result in mechanics and physics. It has the form of the product of stiffness matrix and displacement vector on the left, and the force vector on the right side of the equation.

Usually, when we construct the stiffness based on FEM theory, the shape we use is based on the observed shape, which is under gravity. However, the gravity affects the elements inside the matrix in nonlinear sense, it's not simply a coefficient multiplied by the matrix. To treat the gravity force, it's in a similar way to treat the body force that is applied on an object. It's not appropriate to simply exempt the gravitational force on both sides of the constitutive equation since the property of stiffness matrix is nonlinear to the gravity term. But in many situations, people use the shape under gravity to construct the stiffness matrix; this won't give them too much error in the situations like light object or perhaps there is so much body force that gravity force's effect can be ignored.

To get a more precise model when the gravity effect can't be ignored, the idea is to restore the original shape of the object, then construct the stiffness matrix based on a more accurate shape. With the gravity-free model, we can continue to calculate more body force effect applied on the object and obtain a more precise result. In the next several sections, we will review the method of FEM applying on computing the deformation of solid object sitting on a plane. Then describe the fixed point iteration method which we used to solve for the model in single tetrahedron and normal objects separately. Also, I will provide the related experiment about applying the gravity free model to calculate the deformation of an object and compare with the old method, which is using the under gravity model.

Metric system is applied everywhere in the discussion, e.g., for length we use meter, for mass we use kilogram, for force we use Newton, and for pressure we use Pascal and so on. We will not mention the units again during the discussion. When a vector is mentioned, we usually mean it is a column vector. Some other math notations, for example, a vector $\boldsymbol{v} = (v_1, v_2, \ldots, v_n)^T$ is a row vector, to calculate the derivative of the scalar u with respect to vector $\boldsymbol{v}$ will be $\partial u / \partial \boldsymbol{v} = (\frac{\partial u}{\partial v_1}, \ldots, \frac{\partial u}{\partial v_n})$. The derivative of a vector $\boldsymbol{u} = (u_1, u_2, \ldots, u_m)^T$ with respect to

another vector $\boldsymbol{v} = (v_1, v_2, \ldots, v_n)^T$ will be a matrix:

$$\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{v}} = \begin{pmatrix} \partial u_1/\partial v_1 & \ldots & \partial u_1/\partial v_n \\ \ldots & \ldots & \ldots \\ \partial u_m/\partial v_1 & \ldots & \partial u_m/\partial v_n \end{pmatrix}, \tag{3.1}$$

3.2 FEM applied in computing the deformation

3.2.1 Introductions about constitutive equation and reduced stiffness matrix

For a soft solid object sitting on a plane like a table, it deforms under gravity force and also the supporting force from the plane. The gravity force is usually ignored based on the traditional FEM application models. By using the 3D scanner, we can generate the triangulated 3D mesh of the object under gravity. We call those points on the mesh to be $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n$, and the corresponding displacement vectors with respect to those points will be $\boldsymbol{\delta}_1, \ldots, \boldsymbol{\delta}_n$. The stiffness matrix constructed over the current under gravity mesh will be called K'. The constitutive equation we have mentioned before for computing the deformation of the object is:

$$K'(\boldsymbol{\delta}_1^T, \ldots, \delta_n^T)^T = \boldsymbol{F} \tag{3.2}$$

Although the total gravitational force and supporting force will be balanced, it's easy to be seen that the $\boldsymbol{F}$ in the equation is not a simple zero, since there are still internal forces to balance each other. There seems to exist a contradiction at first glance of the equation, since the object is already balanced by the forces, where should the deformations come from. This is result by the inaccurate model that FEM usually used, because it doesn't concern the deformation caused by the gravitational force. However, in the real situation, it happened that we can't ignore the result by gravity when there is not a negligible amount of effect on the shape of the object. When the mass of the object is small, usually the gravity won't cause much deformation, thus using the inaccurate model won't give us much error. However, if we still use the old model, when the mass of the object is large enough so that the gravity effect can't be ignored, there can be some non negligible error evolved.

To fix this issue, we will compute the stiffness matrix of the object under gravity free shape, then apply this stiffness matrix to calculate further deformations of the object with other body forces. Here are some notations: we denote $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n$ to be the gravity free locations of the points on the mesh, and denote $\tilde{\boldsymbol{p}}_1, \ldots, \tilde{\boldsymbol{p}}_n$ to be the locations of the points under gravity (also observed shape). Each of them is a point with x, y, z coordinates. For $1 \leq i \leq n$, $\boldsymbol{p}_i = (x_i, y_i, z_i)^T$ and $\tilde{\boldsymbol{p}}_i = (\tilde{x}_i, \tilde{y}_i, \tilde{z}_i)^T$. Define the nodal displacement $\delta_i = \tilde{\boldsymbol{p}}_i - \boldsymbol{p}_i = (u_i, v_i, w_i)^T$. We assume that the gravitational force is uniformly applied on the object. Thus, for each of the triangulated tetrahedron in the mesh, each of the vertices takes one fourth of the the corresponding tetrahedron's gravity force. During the process of deformation under gravity, we assume that all of the vertices that contact the plane will still stick to the plane due to the large enough friction. Let those sticking points' index be from $n - s + 1, \ldots, n$, which are $\boldsymbol{p}_{n-s+1}, \ldots, \boldsymbol{p}_n$. Then correspondingly, since they are sticking to the plane, there is no changing position for them, so $\boldsymbol{\delta}_{n-s+1} = \ldots = \boldsymbol{\delta}_n = \boldsymbol{0}$. After which, to completely describe the problem, we will need the vector $\boldsymbol{P} = (\boldsymbol{p}_1^T, \ldots, \boldsymbol{p}_{n-s}^T)^T$, the vector $\tilde{\boldsymbol{P}} = (\tilde{\boldsymbol{p}}_1^T, \ldots, \tilde{\boldsymbol{p}}_{n-s}^T)^T$, and $\boldsymbol{\Delta} = (\boldsymbol{\delta}_1^T, \ldots, \boldsymbol{\delta}_{n-s}^T)^T$. We still have $\boldsymbol{P} = \tilde{\boldsymbol{P}} - \boldsymbol{\Delta}$.

For the gravitational force exerted on the points $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_{n-s}$, call it $\boldsymbol{G}$, which is a m dimensional vector, $m = 3(n - s)$. Due to the reason that the last s positions of points on the mesh are fixed, we will only be interested in the reduced stiffness matrix K, which is eliminated the last $3s$ rows and columns. The reduced stiffness matrix will be symmetric and positive definite. It will depends on $\boldsymbol{P}$, which is essentially depends on the solution corresponds to $\boldsymbol{\Delta}$ and $\tilde{\boldsymbol{P}}$. So we can rewrite the constitutive equation as:

$$K(\boldsymbol{\Delta})\boldsymbol{\Delta} = \boldsymbol{G} + \boldsymbol{F} \tag{3.3}$$

3.3 Stiffness matrix construction

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

$$U_0 = \frac{1}{2}(\sigma_x \epsilon_x + \sigma_y \epsilon_y + \sigma_z \epsilon_z + \tau_{xy} \gamma_{xy} + \tau_{yz} \gamma_{yz} + \tau_{zx} \gamma_{zx}) \tag{3.4}$$

where

$$\epsilon_x = \partial u/\partial x$$

$$\epsilon_y = \partial v/\partial y$$

$$\epsilon_z = \partial w/\partial z$$

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

$$\gamma_{yz} = \partial v/\partial z + \partial w/\partial y$$

$$\gamma_{zx} = \partial w/\partial x + \partial u/\partial z$$

are strains, and

$$\sigma_x = \frac{E}{(1+\nu)(1-2\nu)}[(1-\nu)\epsilon_x + \nu\epsilon_y + \nu\epsilon_z] \tag{3.5}$$

$$\sigma_y = \frac{E}{(1+\nu)(1-2\nu)}[\nu\epsilon_x + (1-\nu)\epsilon_y + \nu\epsilon_z] \tag{3.6}$$

$$\sigma_z = \frac{E}{(1+\nu)(1-2\nu)}[\nu\epsilon_x + \nu\epsilon_y + (1-\nu)\epsilon_z] \tag{3.7}$$

$$\tau_{xy} = \frac{E}{2(1+\nu)}\gamma_{xy} \tag{3.8}$$

$$\tau_{yz} = \frac{E}{2(1+\nu)}\gamma_{yz} \tag{3.9}$$

$$\tau_{zx} = \frac{E}{2(1+\nu)}\gamma_{zx} \tag{3.10}$$

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

Substitute Equations (3.5) to (3.10) in Equation (3.4),

$$U_0 = \frac{E}{4(1+\nu)}\int_V \left[\frac{2(1-\nu)}{1-2\nu}(\epsilon_x^2 + \epsilon_y^2 + \epsilon_z^2) + \frac{4\nu}{1-2\nu}(\epsilon_x\epsilon_y + \epsilon_y\epsilon_z + \epsilon_z\epsilon_x) + (\gamma_{xy}^2 + \gamma_{yz}^2 + \gamma_{zx}^2)\right] dV. \tag{3.11}$$

Rewrite the inside integrals to be perfect squares, we get:

$$U_0 = \frac{E}{2(1+\nu)}\int_V \left[\frac{\nu}{1-2\nu}(\epsilon_x + \epsilon_y + \epsilon_z)^2 + (\epsilon_x^2 + \epsilon_y^2 + \epsilon_z^2) + \frac{1}{2}(\gamma_{xy}^2 + \gamma_{yz}^2 + \gamma_{zx}^2)\right] dV. \tag{3.12}$$

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:

$$P = \sum_{i=1}^{4} c_i(x_i, y_i, z_i)^T, \tag{3.13}$$

$$d = \sum_{i=1}^{4} c_i(u_i, v_i, w_i)^T \tag{3.14}$$

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 (3.13) and (3.14):

$$P = (x, y, z)^T = (x_4, y_4, z_4)^T + \sum_{i=1}^{3} c_i(x_i - x_4, y_i - y_4, z_i - z_4)^T, \tag{3.15}$$

$$d = (u, v, w)^T = (u_4, v_4, w_4)^T + \sum_{i=1}^{3} c_i(u_i - u_4, v_i - v_4, w_i - w_4)^T. \tag{3.16}$$

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

$$\begin{bmatrix} \frac{\partial x}{\partial c_1} & \frac{\partial x}{\partial c_2} & \frac{\partial x}{\partial c_3} \\ \frac{\partial y}{\partial c_1} & \frac{\partial y}{\partial c_2} & \frac{\partial y}{\partial c_3} \\ \frac{\partial z}{\partial c_1} & \frac{\partial z}{\partial c_2} & \frac{\partial z}{\partial c_3} \end{bmatrix} = \begin{bmatrix} x_1 - x_4 & x_2 - x_4 & x_3 - x_4 \\ y_1 - y_4 & y_2 - y_4 & y_3 - y_4 \\ z_1 - z_4 & z_2 - z_4 & z_3 - z_4 \end{bmatrix} \tag{3.17}$$

$$\begin{bmatrix} \frac{\partial u}{\partial c_1} & \frac{\partial u}{\partial c_2} & \frac{\partial u}{\partial c_3} \\ \frac{\partial v}{\partial c_1} & \frac{\partial v}{\partial c_2} & \frac{\partial v}{\partial c_3} \\ \frac{\partial w}{\partial c_1} & \frac{\partial w}{\partial c_2} & \frac{\partial w}{\partial c_3} \end{bmatrix} = \begin{bmatrix} u_1 - u_4 & u_2 - u_4 & u_3 - u_4 \\ v_1 - v_4 & v_2 - v_4 & v_3 - v_4 \\ w_1 - w_4 & w_2 - w_4 & w_3 - w_4 \end{bmatrix}. \tag{3.18}$$

The absolute value of the determinant of the matrix in the right hand side of Equation (3.17) is 6 times the volume of the tetrahedron. Since the tetrahedrons have positive volume in this case, the matrix is fully ranked, and its inverse exists. From Equation (3.17), we obtain

$$\begin{bmatrix} \frac{\partial c_1}{\partial x} & \frac{\partial c_1}{\partial y} & \frac{\partial c_1}{\partial z} \\ \frac{\partial c_2}{\partial x} & \frac{\partial c_2}{\partial y} & \frac{\partial c_2}{\partial z} \\ \frac{\partial c_3}{\partial x} & \frac{\partial c_3}{\partial y} & \frac{\partial c_3}{\partial z} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial c_1} & \frac{\partial x}{\partial c_2} & \frac{\partial x}{\partial c_3} \\ \frac{\partial y}{\partial c_1} & \frac{\partial y}{\partial c_2} & \frac{\partial y}{\partial c_3} \\ \frac{\partial z}{\partial c_1} & \frac{\partial z}{\partial c_2} & \frac{\partial z}{\partial c_3} \end{bmatrix}^{-1} = \begin{bmatrix} x_1 - x_4 & x_2 - x_4 & x_3 - x_4 \\ y_1 - y_4 & y_2 - y_4 & y_3 - y_4 \\ z_1 - z_4 & z_2 - z_4 & z_3 - z_4 \end{bmatrix}^{-1} \tag{3.19}$$

Let us go back to the strains,

$$\epsilon_x = \frac{\partial u}{\partial x} = \sum_{i=1}^{3} \frac{\partial u}{\partial c_i} \frac{\partial c_i}{\partial x} = \sum_{i=1}^{3} \frac{\partial c_i}{\partial x}(u_i - u_4) = \sum_{i=1}^{3} \frac{\partial c_i}{\partial x} u_i - \sum_{i=1}^{3} \frac{\partial c_i}{\partial x} u_4. \tag{3.20}$$

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

$$
\begin{bmatrix}
\frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} & \frac{\partial w}{\partial x} \\[2mm]
\frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} & \frac{\partial w}{\partial y} \\[2mm]
\frac{\partial u}{\partial z} & \frac{\partial v}{\partial z} & \frac{\partial w}{\partial z}
\end{bmatrix}
=
\begin{bmatrix}
\frac{\partial c_1}{\partial x} & \frac{\partial c_2}{\partial x} & \frac{\partial c_3}{\partial x} & -\sum_{i=1}^3 \frac{\partial c_i}{\partial x} \\[2mm]
\frac{\partial c_1}{\partial y} & \frac{\partial c_2}{\partial y} & \frac{\partial c_3}{\partial y} & -\sum_{i=1}^3 \frac{\partial c_i}{\partial y} \\[2mm]
\frac{\partial c_1}{\partial z} & \frac{\partial c_2}{\partial z} & \frac{\partial c_3}{\partial z} & -\sum_{i=1}^3 \frac{\partial c_i}{\partial z}
\end{bmatrix}
\begin{bmatrix}
u_1 & v_1 & w_1 \\[2mm]
u_2 & v_2 & w_2 \\[2mm]
u_3 & v_3 & w_3 \\[2mm]
u_4 & v_4 & w_4
\end{bmatrix}.
\tag{3.21}
$$

With Equation (3.21), we can assemble the stiffness matrix.

Denote Q the first matrix on the right-hand side of (3.21), 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

$$
\epsilon_x^2 = (u_1, u_2, u_3, u_4)
\begin{pmatrix}
Q_{11}^2 & Q_{11}Q_{12} & Q_{11}Q_{13} & Q_{11}Q_{14} \\
Q_{12}Q_{11} & Q_{12}^2 & Q_{12}Q_{13} & Q_{12}Q_{14} \\
Q_{13}Q_{11} & Q_{13}Q_{12} & Q_{13}^2 & Q_{13}Q_{14} \\
Q_{14}Q_{11} & Q_{14}Q_{12} & Q_{14}Q_{13} & Q_{14}^2
\end{pmatrix}
\begin{pmatrix}
u_1 \\ u_2 \\ u_3 \\ u_4
\end{pmatrix},
\tag{3.22}
$$

$$
\epsilon_y^2 = (v_1, v_2, v_3, v_4)
\begin{pmatrix}
Q_{21}^2 & Q_{21}Q_{22} & Q_{21}Q_{23} & Q_{21}Q_{24} \\
Q_{22}Q_{21} & Q_{22}^2 & Q_{22}Q_{23} & Q_{22}Q_{24} \\
Q_{23}Q_{21} & Q_{23}Q_{22} & Q_{23}^2 & Q_{23}Q_{24} \\
Q_{24}Q_{21} & Q_{24}Q_{22} & Q_{24}Q_{23} & Q_{24}^2
\end{pmatrix}
\begin{pmatrix}
v_1 \\ v_2 \\ v_3 \\ v_4
\end{pmatrix},
\tag{3.23}
$$

$$
\epsilon_z^2 = (w_1, w_2, w_3, w_4)
\begin{pmatrix}
Q_{31}^2 & Q_{31}Q_{32} & Q_{31}Q_{33} & Q_{31}Q_{34} \\
Q_{32}Q_{31} & Q_{32}^2 & Q_{32}Q_{33} & Q_{32}Q_{34} \\
Q_{33}Q_{31} & Q_{33}Q_{32} & Q_{33}^2 & Q_{33}Q_{34} \\
Q_{34}Q_{31} & Q_{34}Q_{32} & Q_{34}Q_{33} & Q_{34}^2
\end{pmatrix}
\begin{pmatrix}
w_1 \\ w_2 \\ w_3 \\ w_4
\end{pmatrix}.
\tag{3.24}
$$

On the other hand,

$$\gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} = \sum_{i=1}^{4} (Q_{1i}v_i + Q_{2i}u_i),$$

$$\gamma_{yz} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} = \sum_{i=1}^{4} (Q_{3i}v_i + Q_{2i}w_i),$$

$$\gamma_{zx} = \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} = \sum_{i=1}^{4} (Q_{3i}u_i + Q_{1i}w_i).$$

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

$$\gamma_{xy}^2 = (\sum_{i=1}^{4} Q_{1i}v_i)^2 + (\sum_{i=1}^{4} Q_{2i}u_i)^2 + 2(\sum_{i=1}^{4} Q_{1i}v_i \cdot \sum_{i=1}^{4} Q_{2i}u_i) \qquad (3.25)$$

After we expend every term in equation 3.12, we also know that

$$U_0 = \frac{1}{2} d_t^T K d_t. \qquad (3.26)$$

Then we compare those two expansions to get the form of stiffness matrix K.

3.4   The case of one tetrahedron

Given a single tetrahedron with four vertices $(\tilde{x}_i, \tilde{y}_i, \tilde{z}_i)$, $i = 1, 2, \ldots, 4$, lying on a table under gravity, we want to recover its shape in the situation without gravity. Suppose the coordinates of the vertices in the original shape are (x_i, y_i, z_i), $i = 1, 2, \ldots, 4$. When we apply the gravity to the object, the displacement of each of the vertices will be $(u_i, v_i, w_i) = (\tilde{x}_i, \tilde{y}_i, \tilde{z}_i) - (x_i, y_i, z_i)$, $i = 1, 2, \ldots, 4$. In our model, we fix the bottom three points of the tetrahedron. Hence, $(u_i, v_i, w_i) = (0, 0, 0)$, $i = 2, 3, 4$. In order to solve for the displacement vector of $\mathbf{\Delta} = (u_1, v_1, w_1, 0, \ldots, 0)^T$, we will start from the constitutive equation:

$$K\mathbf{\Delta} = \mathbf{F} + \mathbf{G}, \qquad (3.27)$$

where K is the 12×12 stiffness matrix of the original shape, which is constructed based on the displacement variables u_i, v_i, w_i and data of the current shape $\tilde{x}_i, \tilde{y}_i, \tilde{z}_i$, and

$$\mathbf{F} = (0, 0, 0, f_{2x}, f_{2y}, f_{2z}, \ldots, f_{4x}, f_{4y}, f_{4z})^T, \qquad (3.28)$$

$$\boldsymbol{G} = (0, 0, -G/4, \ldots, 0, 0, -G/4)^T. \tag{3.29}$$

Here, G is the gravity of the tetrahedron. We apply the gravity force equally on the four vertices, which means $G/4$ along $-z$ direction for each.

Below, Figure 1 shows us an example of a single tetrahedron. Without loss of generality, we can always pick one surface of the tetrahedron to lie on the x-y plane, two of its vertices P_2 and P_4 along x-axis with one of them P_4 at the origin, and the other with coordinates $(\tilde{x}_2, 0, 0)$.
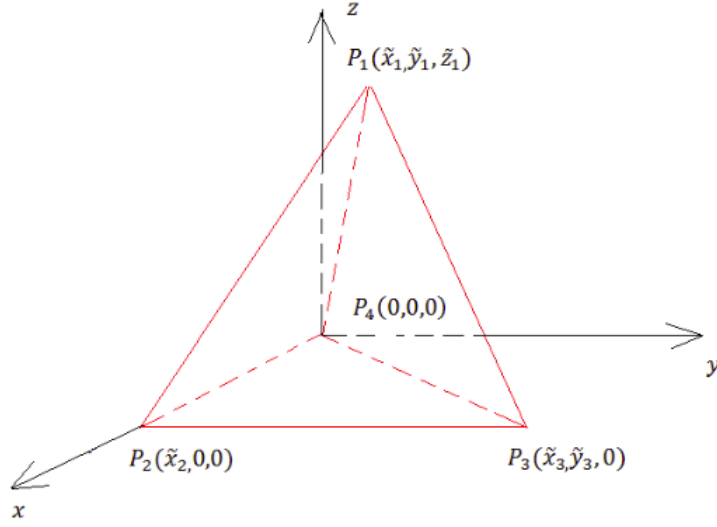
Figure 3.1   A tetrahedron

Nine constraints are imposed from fixing the vertices P_2, P_3, and P_4. We are left with only three variables, which are u_1, v_1, and w_1. Therefore, we can reduce the twelve nonlinear equations in equation(3.27) to just three equations, which are the first three rows of each of the component. The reduced system is shown as below:

$$\begin{pmatrix} \frac{EV}{4(1+\nu)(w_1-\tilde{z}_1)^2} & 0 & 0 \\ 0 & \frac{EV}{4(1+\nu)(w_1-\tilde{z}_1)^2} & 0 \\ 0 & 0 & \frac{EV(\nu-1)}{2(1+\nu)(2\nu-1)(w_1-\tilde{z}_1)^2} \end{pmatrix} \begin{pmatrix} u_1 \\ v_1 \\ w_1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -\frac{G}{4} \end{pmatrix}, \tag{3.30}$$

where E is the Young's modulus, ν is Poisson's ratio, and V is the volume of the original tetrahedron. We denote the above equation as:

$$\bar{K}\bar{\boldsymbol{\Delta}} = \bar{\boldsymbol{G}}, \tag{3.31}$$

Since $\bar{K}$ is a diagonal matrix, u_1 and v_1 will be 0. This implies that the displacement of the top point only happens in z direction. Recall that our goal is to solve $\bar{\boldsymbol{\Delta}}$. Since $\bar{K}$ is a full rank matrix, we can simply solve the system by:

$$\bar{\boldsymbol{\Delta}} = \bar{K}^{-1}\bar{\boldsymbol{G}}. \tag{3.32}$$

For one tetrahedron, it seems to be very easy to solve it algebraically, but for other complex models, we should consider using numerical iteration methods. Here, we introduce a damping factor $r \in (0, 1]$ to form the following iteration equation based on equation (3.32):

$$\bar{\boldsymbol{\Delta}}^{(k+1)} = (1 - r)\bar{\boldsymbol{\Delta}}^{(k)} + r\bar{K}(\bar{\boldsymbol{\Delta}}^{(k)})^{-1}\bar{\boldsymbol{G}}. \tag{3.33}$$

We may rewrite the method as:

$$\boldsymbol{\alpha}(\bar{\boldsymbol{\Delta}}) = (1 - r)\bar{\boldsymbol{\Delta}} + r\bar{K}(\bar{\boldsymbol{\Delta}})^{-1}\bar{\boldsymbol{G}}. \tag{3.34}$$

Then rewrite the major iteration part as another function:

$$\boldsymbol{\beta}(\bar{\boldsymbol{\Delta}}) = \bar{K}(\bar{\boldsymbol{\Delta}})^{-1}\bar{\boldsymbol{G}}. \tag{3.35}$$

the algorithm is:

Algorithm 4 iteration method for recovering the shape of the object without gravity

---

1: Compute the initial stiffness matrix of the object under gravity, and $\bar{\boldsymbol{\Delta}}^{(0)} = \boldsymbol{0}$.
2: At step k, k=1,2,..., apply an extra gravity force $\boldsymbol{G}$ and compute the reduced displacement vector $\bar{\boldsymbol{\Delta}}^{(k)}$ by using equation (3.32), which is based on the stiffness matrix $\bar{K}(\bar{\boldsymbol{\Delta}}^{(k-1)})$. Then, apply $-\bar{\boldsymbol{\Delta}}^{(k)}$ to the current shape and compute the new reduced stiffness matrix $\bar{K}(\bar{\boldsymbol{\Delta}}^{(k)})$.
3: $k = k + 1$.
4: Repeat steps 2 and 3 until the difference of the shapes in two consecutive steps are close enough.

The iteration method comes down to solving a fixed point problem:

$$\bar{\boldsymbol{\Delta}} = h(\bar{\boldsymbol{\Delta}}), \tag{3.36}$$

where,

$$
\begin{aligned}
h(\bar{\boldsymbol{\Delta}}) &= (1-r)\bar{\boldsymbol{\Delta}} + r\bar{K}(\bar{\boldsymbol{\Delta}})^{-1}\bar{\boldsymbol{G}} \\
&= (1-r)\begin{pmatrix} u_1 \\ v_1 \\ w_1 \end{pmatrix} + r\begin{pmatrix} \frac{4(1+\nu)(w_1-\tilde{z}_1)^2}{EV} & 0 & 0 \\ 0 & \frac{4(1+\nu)(w_1-\tilde{z}_1)^2}{EV} & 0 \\ 0 & 0 & \frac{2(1+\nu)(2\nu-1)(w_1-\tilde{z}_1)^2}{EV(\nu-1)} \end{pmatrix}\begin{pmatrix} 0 \\ 0 \\ -\frac{G}{4} \end{pmatrix} \\
&= \begin{pmatrix} (1-r)u_1 \\ (1-r)v_1 \\ (1-r)w_1 - \frac{r(1+\nu)(2\nu-1)G(w_1-\tilde{z}_1)^2}{2EV(\nu-1)} \end{pmatrix}.
\end{aligned}
$$

In the above, V is the volume of the original shape:

$$
V = \frac{1}{3} \times \frac{1}{2} \times x_2 \times y_3 \times z_1 = \frac{\tilde{x}_2\tilde{y}_3(\tilde{z}_1 - w_1)}{6}. \tag{3.37}
$$

Then,

$$
h(\bar{\boldsymbol{\Delta}}) = \begin{pmatrix} (1-r)u_1 \\ (1-r)v_1 \\ (1-r)w_1 + \frac{3r(1+\nu)(2\nu-1)G(w_1-\tilde{z}_1)}{E(\nu-1)\tilde{x}_2\tilde{y}_3} \end{pmatrix}. \tag{3.38}
$$

To analyze the convergence of the iteration method, we should consider two conditions: a) the two-norm of the Jacobian matrix of function h in (3.38) with respect to $\bar{\boldsymbol{\Delta}}$ is less than a constant which is less than one; b) h maps a space M to itself.

Now, we consider about the first condition. Compute the Jacobian matrix of function h from equation (3.38) with respect to $\bar{\boldsymbol{\Delta}}$:

$$
\frac{\partial h(\bar{\boldsymbol{\Delta}})}{\partial \bar{\boldsymbol{\Delta}}} = \begin{pmatrix} 1-r & 0 & 0 \\ 0 & 1-r & 0 \\ 0 & 0 & 1-r + \frac{3r(1+\nu)(2\nu-1)G}{E(\nu-1)\tilde{x}_2\tilde{y}_3} \end{pmatrix}, \tag{3.39}
$$

By definition, the two-norm of a matrix is the square root of the largest singular value. In this case, since the Jacobian is a diagonal matrix, the two-norm will be the absolute value of the largest diagonal entry. Now, since $\nu < 0.5$, the biggest entry in the Jacobian matrix is the last entry and it is positive. In order to converge, we derive:

$$
1 - r + \frac{3r(1+\nu)(2\nu-1)G}{E(\nu-1)\tilde{x}_2\tilde{y}_3} < 1. \tag{3.40}
$$

After simplifying it, we obtain:

$$\frac{3(1+\nu)(2\nu-1)}{E(\nu-1)\tilde{x}_2\tilde{y}_3}G < 1. \tag{3.41}$$

Since $G = mg$, where $g = 9.8N/kg$, and $\tilde{V} = \tilde{y}_3\tilde{z}_1/6$, we can transform the above inequality to obtain a relation between density $\tilde{\rho} = m/\tilde{V}$ and variable $\tilde{z}_1$:

$$\frac{3(1+\nu)(2\nu-1)}{E(\nu-1)\tilde{x}_2\tilde{y}_3}mg < 1. \tag{3.42}$$

Subsequently,

$$\frac{3(1+\nu)(2\nu-1)mg\tilde{z}_1/6}{E(\nu-1)\tilde{x}_2\tilde{y}_3\tilde{z}_1/6} < 1. \tag{3.43}$$

Substituting $\tilde{V}$ into the above inequality,

$$\frac{(1+\nu)(2\nu-1)mg\tilde{z}_1}{2E(\nu-1)\tilde{V}} < 1. \tag{3.44}$$

Plug in $\tilde{\rho} = m/\tilde{V}$,

$$\frac{(1+\nu)(2\nu-1)g\tilde{z}_1\tilde{\rho}}{2E(\nu-1)} < 1, \tag{3.45}$$

After simplification, we obtain:

$$\tilde{\rho} < \frac{2E(\nu-1)}{(1+\nu)(2\nu-1)g\tilde{z}_1}. \tag{3.46}$$

Thus, if the density of a single tetrahedron is within which the above inequality holds, then the numerical method will satisfy convergence condition a).

Finally, we consider the convergence condition b). In our algorithm, since we apply an extra gravity force at the beginning, the first step displacement happens only in the z direction, which means $u_1^{(0)}$ and $v_1^{(0)}$ will be 0. From equation (3.38), we know that in any step k, $u_1^{(k)}$ and $v_1^{(k)}$ will always be 0. Therefore the displacement happens only in z direction, which is w_1.

In other words, we only analyze the third element in equation (3.38), and see if we will find a interval of w_1 that maps to the same range interval of the third element. We rewrite this element as a line equation with variable w_1:

$$\begin{aligned}
h(\bar{\boldsymbol{\Delta}})(3) &= (1-r)w_1 + \frac{3r(1+\nu)(2\nu-1)G(w_1-\tilde{z}_1)}{E(\nu-1)\tilde{x}_2\tilde{y}_3} \\
&= (1-r+\frac{3r(1+\nu)(2\nu-1)G}{E(\nu-1)\tilde{x}_2\tilde{y}_3})w_1 - \frac{3r(1+\nu)(2\nu-1)G}{E(\nu-1)\tilde{x}_2\tilde{y}_3}z_1 \tag{3.47}
\end{aligned}$$

We can treat $\gamma = \alpha(w_1)$ as a function, which is a line in the w_1-γ plane. It's clear that the y-intercept is negative, and the slope is positive. Therefore, only when the slope is less than one, after several iterations, the interval of w_1 can be mapped to a smaller interval, which satisfies condition b):

$$1 - r + \frac{3r(1+\nu)(2\nu - 1)G}{E(\nu - 1)\tilde{x}_2\tilde{y}_3} < 1 \tag{3.48}$$

Hence, we notice that the above inequality is exactly the same as inequality (3.40). The rest of analysis here is the same as inequalities (3.41) to (3.46).

Also, since our goal is to find the interval that the function maps it to itself, we can consider about the line $\gamma = w_1$ intersect with the line $\gamma = \alpha(w_1)$. Then for any $d \leq w_1^*$, the region $D = [d, 0]$ will satisfy our requirement (b). Figure (3.2) illustrate it:
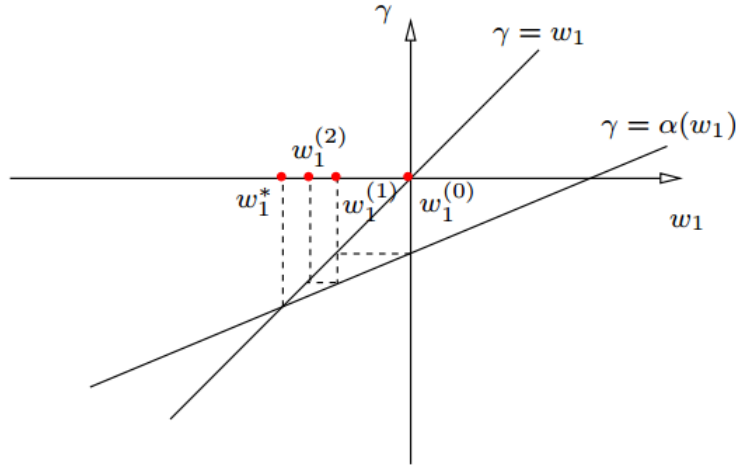

Figure 3.2   Convergence analysis of single tetrahedron

Therefore, combine condition a) and b), the sufficient condition for the iteration method to converge is:

$$\tilde{\rho} < \frac{2E(\nu - 1)}{(1 + \nu)(2\nu - 1)g\tilde{z}_1}. \tag{3.49}$$

Let's see the numerical simulation result of the iteration method applied on a single tetrahedron. Here, we chose Young's modulus $E = 5 \times 10^3$, mass 0.0765, density 10^3, and Poisson's ratio $\nu = 0.3$. The vertices of the tetrahedron are: $\tilde{\boldsymbol{p}}_1 = (\sqrt{3}, 1, 2\sqrt{2})^T/40$, $\tilde{\boldsymbol{p}}_2 = (\sqrt{3}/20, 0, 0)^T$, $\tilde{\boldsymbol{p}}_3 = (\sqrt{3}, 3, 0)^T/40$, $\tilde{\boldsymbol{p}}_4 = (0, 0, 0)$,

Table 3.1   Convergence condition (b) analysis

	$r = 1$	$r = 0.5$
$\tilde{\rho}$	# iters	# iters
10^2	4	10
5×10^2	4	14
10^4	6	17
2×10^3	11	27

The figure 3.3 illustrates several more examples and comparisons between iteration steps and parameter r.
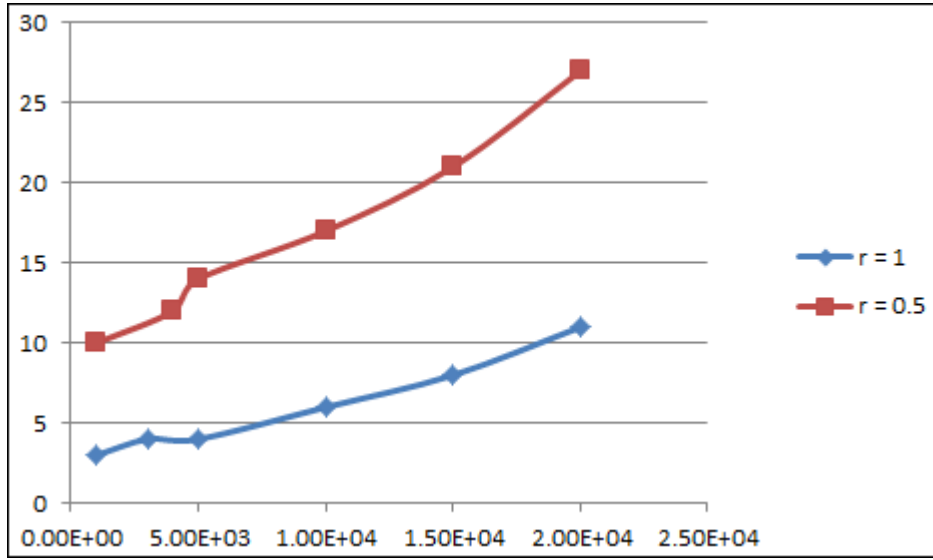
Figure 3.3   Iteration comparisons

The result we got is $w_1 \approx -0.00435467$. It tells us the gravity free shape of the tetrahedron would be the top vertex moves up for $-w1$. We can also get the result of the bottom three inner forces of each of the vertex. The result will be listed after figure 3.3.

$$
\begin{aligned}
(f_{2x}, f_{2y}, f_{2z})^T &\approx (-0.0694, 0.04, 0.25)^T \\
(f_{3x}, f_{3y}, f_{3z})^T &\approx (0, -0.08, 0.25)^T \\
(f_{4x}, f_{4y}, f_{4z})^T &\approx (0.0694, 0.04, 0.25)^T
\end{aligned}
\tag{3.50}
$$

Then the ratio between the magnitudes of the tangential and normal forces at each of

p_2, p_3, p_4 will be:

$$\text{ratio 2}: \frac{\sqrt{f_{2x}^2 + f_{2y}^2}}{f_{2z}} \approx \frac{(-0.0694)^2 + (0.04)^2}{0.25} = 0.320494$$

$$\text{ratio 3}: \frac{\sqrt{f_{3x}^2 + f_{3y}^2}}{f_{3z}} \approx \frac{(0)^2 + (-0.08)^2}{0.25} = 0.320494$$

$$\text{ratio 4}: \frac{\sqrt{f_{4x}^2 + f_{4y}^2}}{f_{4z}} \approx \frac{(0.0694)^2 + (0.04)^2}{0.25} = 0.320494 \tag{3.51}$$

Therefore, the result of the ratios is consistent with what we thought. They are the same number due to the symmetry of the shape of the tetrahedron. Some frictional forces are needed to keep the tetrahedron to stay on the table.

3.5   General shape

3.5.1   Jacobian under 2-norm

The iteration function is

$$\boldsymbol{\alpha}(\boldsymbol{\Delta}) = \boldsymbol{\Delta}' = (1 - r)\boldsymbol{\Delta} + r\bar{K}^{-1}(\boldsymbol{\Delta})\bar{\boldsymbol{G}}, \tag{3.52}$$

which is based on equation 3.33, where $r \in (0, 1]$ is the parameter, $\bar{K}$ is the reduced stiffness matrix, and $\bar{\boldsymbol{G}}$ is the reduced gravity vector. Clearly the gravity on each node is constant due to our way of distributing the gravity, it is proportional to the density ρ, let us denote

$$\bar{\boldsymbol{G}} = \breve{\boldsymbol{G}}\rho, \tag{3.53}$$

and $\breve{\boldsymbol{G}}$ is a vector with the same size as $\bar{\boldsymbol{G}}$ Also, given that the stiffness matrix is scaled by Young's Modulus E, denote

$$\bar{K} = E\breve{K}, \tag{3.54}$$

we will get:

$$\bar{K}^{-1} = \frac{1}{E}\breve{K}^{-1}. \tag{3.55}$$

Let $s = \rho/E$,

$$\boldsymbol{\Delta}' = (1-r)\boldsymbol{\Delta} + rs\check{K}^{-1}\check{\boldsymbol{G}}. \tag{3.56}$$

It would be interesting for us to know about the convergence of the iteration method for general model. In the next part, we will discuss about the condition that the two norm of the Jacobian being less than one, which is one of the sufficient conditions for the iteration method to converge. The Jacobian of $\boldsymbol{\Delta}'$ is

$$J[\boldsymbol{\Delta}'] = (1-r)I + rsJ[\check{K}^{-1}\check{\boldsymbol{G}}]. \tag{3.57}$$

Denote $J_\beta = J[\check{K}^{-1}\check{\boldsymbol{G}}]$. The 2-norm of the Jacobian is

$$
\begin{aligned}
\|J[\boldsymbol{\Delta}']\|_2 &= \|(1-r)I + rsJ_\beta\|_2 \\
&= \max_{\boldsymbol{x}} \|[(1-r)I + rsJ_\beta]\boldsymbol{x}\|_2 \\
&= \max_{\boldsymbol{x}} \|(1-r)\boldsymbol{x} + rsJ_\beta\boldsymbol{x}\|_2,
\end{aligned} \tag{3.58}
$$

where $\boldsymbol{x}$ is a unit vector.[1] Suppose the eigenvectors of J_β are $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{3n-9}$, and the corresponding eigenvalues are $\lambda_1, \lambda_2, \ldots, \lambda_{3n-9}$. So

$$\boldsymbol{x} = \sum_{i=1}^{3n-9} (\boldsymbol{v}_i^T \boldsymbol{x}) v_i. \tag{3.59}$$

Set $p_i = \boldsymbol{v}_i^T \boldsymbol{x}$, and plug in the Equation 3.59,

$$
\begin{aligned}
\|J[\boldsymbol{\Delta}']\|_2 &= \max_{\boldsymbol{x}} \|(1-r)\sum_{i=1}^{3n-9} p_i\boldsymbol{v}_i + rs\sum_{i=1}^{3n-9} \lambda_i p_i\boldsymbol{v}_i\|_2 \\
&= \max_{\boldsymbol{x}} \|\sum_{i=1}^{3n-9} (1-r+rs\lambda_i)p_i\boldsymbol{v}_i\|_2 \\
&= \max_{\boldsymbol{x}} \sqrt{\sum_{i=1}^{3n-9} (1-r+rs\lambda_i)^2 p_i^2} \\
&= \max_i |1-r+rs\lambda_i|.
\end{aligned} \tag{3.60}
$$

Since $\sum_{i=1}^{3n-9} p_i^2 = 1$, we can get to the last step.

---

[1]Proof and analysis was provided by Feng Guo.

Let $\lambda_{\max}$ be the maximum eigenvalue and $\lambda_{\min}$ be the minimum eigenvalue. Fix s, and define a group of lines with respect to r:

$$f_i(r) = 1 + (s\lambda_i - 1)r \tag{3.61}$$

this group of line segments will be bounded by the following two because of the restrictions limit on $\lambda_{\max}$ and $\lambda_{\min}$

$$
\begin{aligned}
f_{\max}(r) &= 1 + (s\lambda_{\max} - 1)r \\
f_{\min}(r) &= 1 + (s\lambda_{\min} - 1)r
\end{aligned}
\tag{3.62}
$$

i.e. $f_{\min} \leq f_i \leq f_{\max}, \forall r \in (0, 1]$. This suggests that

$$\|J[\boldsymbol{\Delta}']\|_2(r) = max(|f_{\min}|, |f_{\max}|). \tag{3.63}$$

The following discussions will be dependent on the values of $\lambda_{\max}$ and $\lambda_{\min}$.

1. When $\lambda_{\max} \geq \frac{1}{s}$, $\|J[\boldsymbol{\Delta}']\|_2 \geq f_{\max} \geq 1, \forall r \in (0, 1]$.

2. When $0 \leq \lambda_{\max} < \frac{1}{s}$ and $\lambda_{\min} \geq -\lambda_{\max}$, $\forall r \in (0, 1], \|J[\boldsymbol{\Delta}']\|_2 = f_{\max} < 1$. The optimal value of $\|J[\boldsymbol{\Delta}']\|_2^* = s\lambda_{\max}$ will be reached at $r^* = 1$.

   Proof   When $\lambda_{\min} \geq 0$, $\forall r \in (0, 1], 0 \leq f_{\min} \leq f_{\max}$. The minimum of $\|J[\boldsymbol{\Delta}']\|_2$ is reached when $r = 1$.

   When $-\lambda_{\max} \leq \lambda_{\min} < 0$,

   $$|f_{\max}(1)| = s\lambda_{\max} \geq -s\lambda_{\min} = |f_{\min}(1)|$$

   So $\forall r \in (0, 1], f_{\max} \geq f_{\min}$. $\qquad\qquad\square$

3. When $\lambda_{\max} < \frac{1}{s}$ and $\lambda_{\min} < -\lambda_{\max}$,

   $$r^* = \frac{2}{2 - s(\lambda_{\min} + \lambda_{\max})},$$

   that's when the minimum is achieved.

   $$\|J[\boldsymbol{\Delta}']\|_2^* = \frac{s(\lambda_{\max} - \lambda_{\min})}{2 - s(\lambda_{\max} + \lambda_{\min})}$$

   For the other situations,

(a) $\lambda_{\min} > -\frac{1}{s}$, r can be any value in $(0, 1]$;

(b) $\lambda_{\min} \leq -\frac{1}{s}$, r can only be values in $(0, \frac{2}{1-s\lambda_{\min}})$.

The proof will be similar to the previous one.

In summation, when $\lambda_{\max} \leq 0$, given any s, there always exists some r so that the iteration can converge.

$$
\begin{aligned}
\|J[\mathbf{\Delta}']\|_2^* &= \frac{\lambda_{\max} - \lambda_{\min}}{2/s - (\lambda_{\max} + \lambda_{\min})} \\
&< -\frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \\
&= 1 - \frac{2\lambda_{\max}}{\lambda_{\max} + \lambda_{\min}} \\
&\leq 1.
\end{aligned}
\tag{3.64}
$$

3.5.2 Analysis of the Jacobian with 1-norm or ∞-norm

The computing time of the eigenvalues of J_β will be long. Therefore, we may take a look at the other norms to see if there can be any difference. For example, to compute the 1-norm or ∞-norm, we only have to go through all the elements in J_k, which may help us save a copious amount of time in computing.

By definition, the 1-norm (∞-norm) of a matrix is the maximum of the matrix's absolute column (row) sum. So the 1-norm is:

$$
\kappa_i = \sum_{j \neq i} |A_{ji}|
\tag{3.65}
$$

the ∞-norm be:

$$
\kappa_i = \sum_{j \neq i} |A_{ij}|
\tag{3.66}
$$

where A_{ij} is the element of A on the i-th row and j-th column. To simplify the notation, in the following analysis we denote $A = J_\beta$.

Then

$$
\begin{aligned}
\|J[\mathbf{\Delta}']\| &= \|(1 - r)I + rsA\| \\
&= \max_i \||1 - r + rsA_{ii}| + rs\kappa_i\|
\end{aligned}
\tag{3.67}
$$

Let

$$g_i = |1 - r + rsA_{ii}| + rs\kappa_i,$$

then if $A_{ii} \geq 0$,

$$g_i = r[s(A_{ii} + \kappa_i) - 1] + 1, \tag{3.68}$$

and if $A_{ii} < 0$,

$$g_i = \begin{cases} r[s(\kappa_i + A_{ii}) - 1] + 1, & r \in (0, \frac{1}{1-sA_{ii}}] \\ r[s(\kappa_i - A_{ii}) + 1] - 1, & r \in (\frac{1}{1-sA_{ii}}, 1] \end{cases} \tag{3.69}$$

Clearly that for all $i = 1, \cdots, 3n - 9$, $g_i \geq 0$.

Let $a = \max_i(\kappa_i + A_{ii})$, and $b = \max_j(\kappa_j - A_{jj})$, where $j \in \{x \in Z^+ | x \leq 3n - 9 \text{ and } A_{xx} < 0\}$, then it's not hard to see that $\|J[\mathbf{\Delta}']\| = \max\{r(sa - 1) + 1, r(sb + 1) - 1\}$.

Depending on the values of a and b, there are several cases:

1. When $a \geq \frac{1}{s}$, $\|J[\mathbf{\Delta}']\| \geq 1$.

2. When $a < \frac{1}{s}, b \leq a$, $\|J[\mathbf{\Delta}']\| = r(sa - 1) + 1 < 1$. With $r^* = 1$, and $\|J[\mathbf{\Delta}']^*\| = sa$.

3. When $a < \frac{1}{s}, b > a$,

$$\|J[\mathbf{\Delta}']\| = \begin{cases} r(sa - 1) + 1, & r \in (0, \frac{2}{s(b-a)+2}] \\ r(sb + 1) - 1, & r \in (\frac{2}{s(b-a)+2}, 1] \end{cases} \tag{3.70}$$

Here $r^* = \frac{2}{s(b-a)+2}$, where $\|J[\mathbf{\Delta}']\|^* = \frac{s(a+b)}{s(b-a)+2}$. The range of possible r could be

(a) $(0, 1]$, when $b < \frac{1}{s}$

(b) $(0, \frac{2}{sb+1})$, when $b \geq \frac{1}{s}$

Suppose $a = \kappa_h + A_{hh}$, and $b = \kappa_l - A_{ll}$. Note that

$$\begin{aligned} r^* &= \frac{2}{s(b-a)+2} \\ &= \frac{2}{s(\kappa_l - A_{ll} - \kappa_h - A_{hh}) + 2} \\ &= \frac{2}{s(\kappa_l + A_{ll} - \kappa_h - A_{hh} - 2A_{ll}) + 2} \\ &\geq \frac{2}{-2sA_{ll} + 2} \\ &= \frac{1}{1 - sA_{ll}}. \end{aligned}$$

So the r^* is always in the valid range of $[\frac{1}{1-sA_{ii}}, 1]$.

3.6 Simulation, experiment and results

Based on the result from previous section, to do a little bit transformation,we can get the Jacobian matrix in general norm with the following, here $\boldsymbol{x}$ is a very small vector:

$$\|J_\alpha\| = \|J\| \approx \max \frac{\|(1-r)\boldsymbol{x} + r(\boldsymbol{\beta} - K^{-1}K(\boldsymbol{\Delta} + \boldsymbol{x})\boldsymbol{\beta})\|}{\|\boldsymbol{x}\|}, \tag{3.71}$$

where

$$\boldsymbol{\beta}(\boldsymbol{\Delta}) = K^{-1}(\boldsymbol{\Delta})\boldsymbol{G}. \tag{3.72}$$

We did the numerical simulation on a ball with radius 0.05, density 700, Young's modulus 2.5×10^5, and Poisson's ratio 0.3. The figure 3.4 shows us the ball deformed under gravity sitting on the table with one triangle facet contacting with the table. The three vertices were fixed.There are in total 367 vertices, 1144 tetrahedrons and 2613 triangular facets. The result of the computed $\boldsymbol{\Delta}$ was 0.239769 in 2-norm.
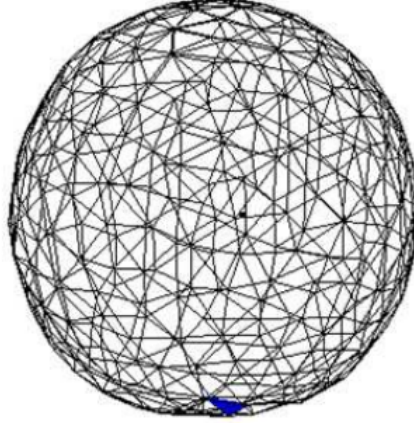

Figure 3.4   A ball sitting on a table under gravity

The next figure shows us during the process, we compare the approximated Jacobian norm $\frac{\|J_\alpha \boldsymbol{x}\|}{\boldsymbol{x}}$ with respect to iteration steps between different r values. Each of the curve below represents different executions. Here, we take the $\boldsymbol{x} = \boldsymbol{\Delta}^{(l)} - \boldsymbol{\Delta}^{(l-1)}$, and also evaluate the value of Jacobian approximately at $\boldsymbol{\Delta}^{(l-1)}$.

$$\frac{\|J_\alpha(\Delta^{(l-1)})(\Delta^{(l)}-\Delta^{(l-1)})\|}{\|\Delta^{(l)}-\Delta^{(l-1)}\|}$$
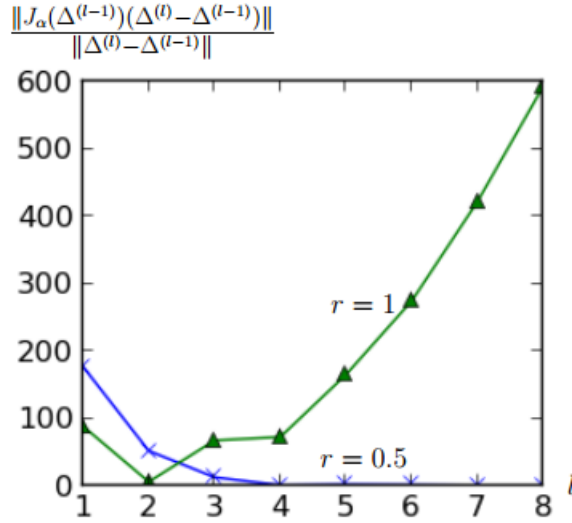

Figure 3.5   Curves from the simulation

From the figure 3.5 we can see that the iteration method succeed with the parameter $r = 0.5$ but failed with the parameter $r = 1$. The plot of the curve with respect to $r = 1$ goes dramatically up at the end. But the iteration method parameter $r = 0.5$ curve goes to zero at the end, which means the norm of the Jacobian matrix should be within a small range, and thus the convergence of the iteration method in this situation would be convergent.

In order to prove our gravity free recover helps the accuracy of computing deformations of objects, we have conducted several experiments. In the experiment, the gelatin with name brand "Jell-O" was used. We use the 3D scanner to get the tetrahedron mesh of the gelatin, then use the iteration numerical method to recover the shape under gravity free situation. Finally we wanted to flip the jelly, and predict the shape under gravity by using the old FEM stiffness matrix model and also our recovered shape stiffness matrix model. The original picture of the jelly will be shown below.

The gelatin is in a bowl shape as shown in figure 3.6. We put it bottom up on a platform, the bigger disk surface is in contacting with the plane. The 3D scanner we used was from NextEngine, Inc., and to simplify the mesh using MeshLab (http://meshlab.sourceforge.net/). The measured Young's modulus of this gelatin was 3×10^3, the Poisson's ratio was 0.4, the density was 9.6×10^2. The picture of the gelatin under gravity with the mesh output from the

Figure 3.6   Gelatin pudding

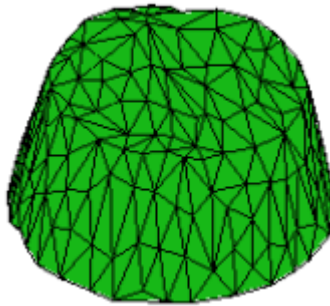scanner, and then our predicted gelatin shape in the mesh are:

Figure 3.7   Mesh representation of the pudding

The mesh we got from the 3D scanner was then transformed using the code from Computational Geometry Algorithm Library (http://www.cgal.org/) into the tetrahedron mesh we needed with 1119 vertices and 1012 facets. In total there were 5152 tetrahedra inside the body of the model.

Using the same assumption that the vertices on the face that contacting the platform were fixed, we restored the shape of the gelatin by using the iteration numerical method we have introduced. The iteration parameter r was set to be 1, so the major part of the iteration function will be $\boldsymbol{\beta}$ from equation (3.35). The total number of iteration steps is 8 by applying the criteria that $||\boldsymbol{\Delta}^{(k+1)} - \boldsymbol{\Delta}^{(k)}|| < 10^{-6}$. The detailed table of the data is in table 3.2:

The figures of the flipped gelatin in real, triangulated mesh, predicted mesh by old stiffness matrix model and the predicted shape by using recovering gravity free shape model are shown in figure 3.9.
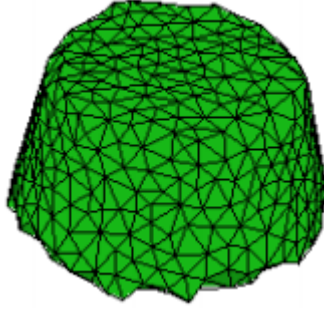
Figure 3.8   Gelatin with recovered shape mesh

Table 3.2   Information of measurements

	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	7.8×10^{-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}

There are more figures [2] shown below for us to analyze the data. We plot the 2-norm of the vector $\boldsymbol{\Delta}$ during the iteration. This measures the difference between the current gelatin shape with the observed gelatin. At the end of the iteration, we can use the observed gelatin shape add the vector $\boldsymbol{\Delta}$ to get the recovered gravity-free shape of the gelatin. These figures are shown in figure 3.10 and in figure 3.11

3.7   Future work

Our model now computes the gravity free shape. In the future, more understanding on the convergence condition of the iteration method is needed. Also, the relation involves the object geometric information, density and the control parameter r are needed for further study as well as the need to get a better understanding of the physical intuition of the iteration method.

After which, to conduct more experiment on different objects, such as fruit, human organs and so on to prove the recover of the gravity free shape is necessary. In addition, the involvement

---

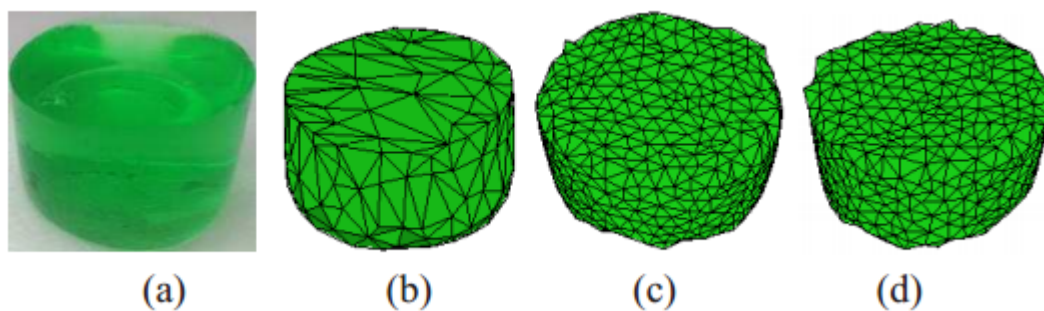[2]The simulation and experiment figures were provided by Huan Lin.

Figure 3.9   Flipped gelatin in (a) original. (b) original shape triangulation. (c) predicted shape
using old stiffness matrix model. (d) predicted shape using gravity-free model
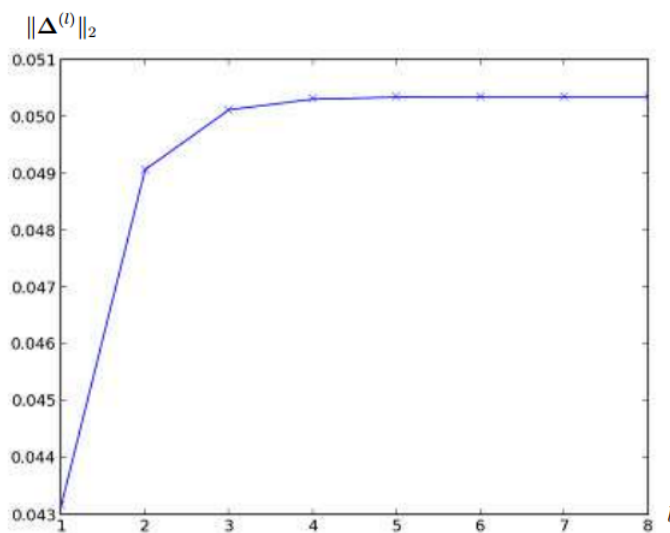

Figure 3.10   2-norm of the vector $\mathbf{\Delta}$ during the iterations at lth step

of robotic manipulation, such as the use of a robot hand grasping the deformable object in the
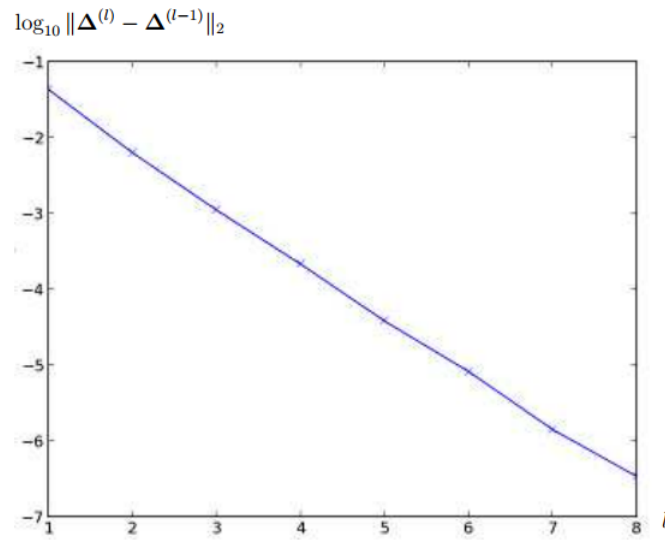experiment, rather than only allowing the object rest on the table in different positions.

Figure 3.11   2-norm of the logarithm of difference between the vector $\boldsymbol{\Delta}$ in two consecutive steps during the iterations

Chapter4.   SUMMARY AND DISCUSSION

The outline of this thesis paper starts with picking up objects, including both 2D objects and 3D objects. In the 2D case, first introduced was the the problem of optimizing the two fingers total normal forces, during the process of resisting a third force . The result shows that at least one of the fingers force has to be along the direction of its friction cone edge. Next discussed, was the algorithm in grasping 2D soft objects. After which segued in to two new concepts, stable squeeze and pure squeeze. Then explained, were the four events of the contact mode during the process of squeezing. For grasping and lifting a 3D object, gravity is an important factor that needs to be considered. Finally explored, was the strategy of computing initial configuration and then applying the two finger squeeze with the lift test in the process.

Since most of the stiffness matrix we have constructed are based on the observed deformed shape of an object under gravity, it is not an accurate model. The idea of recovering the gravity free shape of the object was implemented by using the numerical iteration method.

There is also prospective work that can be considered in the future: including but not limited to, a more in-depth understanding of the convergence of the numerical method being applied on a general model, and how the softness of the object affects the result of picking up strategies.

BIBLIOGRAPHY

A. Bicchi and V. Kumar (2000). Robotics grasping and contact: a review. *Proceedings of the IEEE International Conference on Robotics and Automation. pp. 348-353*

F. Guo ,H. Lin, Y-B. Jia (2013). Squeeze Grasping of Deformable Planar Objects with Segment Contacts and Stick/Slip Transitions. *IEEE intl. Conf. Robot. Autom.*

Y-B. Jia, F. Guo, H. Lin (2013). Grasping Deformable Planar Objects: Squeeze, Stick/Slip Analysis, and Energy-Based Optimalities. *International Journal of Robotics Research, 2013.*

H. Lin, F. Guo, F. Wang, Y-B. Jia (2014). Picking up Soft 3D Objects with Two Fingers. *IEEE International Conference on Robotics and Automation, 2014.*

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

R. C. Brost and K. Y. Goldberg (1994). A complete algorithm for synthesizing modular fixtures for polygonal parts. *Proceedings of the IEEE International Conference on Robotics and Automation, pp. 535-542.*

A. F. Van der Stappen, C. Wentink , and M. H. Overmars (2000). Computing immobilizing grasps of polygonal parts. *International Journal of Robotics Research, 19(5): 467-479.*

E. Rimon and A. Blake (1999). Caging planar bodies by one-parameter two-fingered gripping systems. *International Journal of Robotics Research, 18: 299-318.*

V-D. Nguyen (1998). Constructing force-closure grasps. *International Journal of Robotics Research, 7(3): 3-16.*

J. C. Trinkle (1992). On the stablity and instantaneous velocity of grasped frictionless objects. *IEEE Transactions on Robotics and Automation, 8(5):560-572.*

J. Kerr and B. Roth (1986). Analysis of multifingered hands. *International Journal of Robotics Research, 4(4):3-17.*

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

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

Q. Luo and J. Xiao (2006). Geometric properties of contacts involving a deformable object. *In proceedings of the IEEE Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, pp:533-538.*

J. Tian and Y-B. Jia (2010). Modeling deformations of general parametric shells grasped by a robot hand. *IEEE Transactions on Robotics and Automation, 8(5):560-572.*